



# CHAPTER 3

## REST Commands

---

**Revised: May 4, 2011**

This chapter describes the REST commands and includes the following sections:

- [Job Control Commands, page 3-1](#)
- [Timed Job Control Commands, page 3-31](#)
- [Node Status Commands, page 3-39](#)
- [System Setting Commands, page 3-47](#)
- [Profile Commands, page 3-50](#)
- [Translate and Submit Job Commands, page 3-59](#)
- [Miscellaneous Commands, page 3-63](#)

## Job Control Commands

The following commands are used to query job status and manage submitted jobs:

- [Get Job List, page 3-2](#)
- [Get Job List for User, page 3-5](#)
- [Get Job List Since, page 3-8](#)
- [Get Job Summary, page 3-11](#)
- [Get Job Detail, page 3-14](#)
- [Get Job Detail with Accounting, page 3-18](#)
- [Get Job XML, page 3-23](#)
- [Get Job Notifications, page 3-24](#)
- [Stop Job, page 3-26](#)
- [Delete Job, page 3-27](#)
- [Reschedule Job, page 3-28](#)
- [Retry Job, page 3-29](#)
- [Update Job Priority, page 3-30](#)

## Get Job List

Get list of non-reaped jobs submitted to the ECS. Use this request to get the initial timestamp that can be used as a seed for the Get Job List Since request.


**Note**

This request can potentially return a large amount of data if many jobs have been submitted to the ECS.

**URL Template**

http://localhost/mxerest/jobs

**HTTP Method**

GET

**XML Response Data Structure**

Field	Description
jobList	Root tag.
timestamp	Timestamp for job status. To be used as a seed for querying subsequent job status and returning only jobs that have changed since timestamp.
type	Job list type: <b>Full</b> - all non-reaped jobs are returned <b>Delta</b> - only jobs changed since the timestamp are returned
jobSummary	Job summary section id = job ID href = link to job detail
title	Job title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes

Field	Description
completionReason	Job completion reason. <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job stopped at request of operator <b>SystemStopped</b> - job stopped by system
submitTime	Time job was submitted: yyyy-mm-dd hh:mm:ss
completeTime	Time job was completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority
liveWorker	Live worker name (capture station) reserved by this job.
reservedDuration	Amount of time in seconds the live worker (capture station) is reserved.
timedJobId	The ID of the associated timed job. <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of a job XML, including any user-defined metadata.

### Parameters

None

### Expected Results

Returns <timestamp> and list of job summaries <jobSummary> with links to the detailed job status.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<jobList>
<type>full</type>
  <timestamp>1126807551</timestamp>
  <jobSummary id="2" href="http://localhost/mxerest/jobs/2/detail">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:06:46</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
    <timedJobId>0</timedJobId>
  </jobSummary>
  <jobSummary id="3" href="http://localhost/mxerest/jobs/3/detail">
    <title>---</title>
    <author>---</author>
```

```
<status>Pending</status>  
<completionReason>NotComplete</completionReason>  
<state>new</state>  
<submitTime>2005-09-14 15:09:23</submitTime>  
<completeTime />  
<priority>100</priority>  
<liveWorker></liveWorker>  
<reservedDuration>0</reservedDuration>  
<timedJobId>0</timedJobId>  
</jobSummary>  
</jobList>
```

**Limitations**

None

## Get Job List for User

Get list of jobs for user (based on user-defined-metadata parameter 'user-id').

### URL Template

http://localhost/mxerest/jobs/user/[user-id]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
jobList	Root tag.
timestamp	Timestamp for job status. To be used as a seed for querying subsequent job status and returning only jobs that have changed since timestamp.
type	Job list type: <b>Full</b> - all non-reaped jobs are returned <b>Delta</b> - only jobs changed since the timestamp are returned
jobSummary	Job summary section id = job ID href = link to job detail
title	Job title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes
completionReason	Job completion reason. <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job stopped at request of operator <b>SystemStopped</b> - job stopped by system
submitTime	Time job was submitted: yyyy-mm-dd hh:mm:ss

Field	Description
completeTime	Time job was completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority
liveWorker	Live worker name (capture station) reserved by this job.
reservedDuration	Amount of time in seconds the live worker (capture station) is reserved.
timedJobId	The ID of the associated timed job. <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of a job XML, including any user-defined metadata.

### Parameters

None

### Expected Results

Returns <timestamp> and list of job summaries <jobSummary> with links to the detailed job status.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<jobList>
<type>full</type>
  <timestamp>1126807551</timestamp>
  <jobSummary id="2" href="http://localhost/mxerest/jobs/2/detail">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:06:46</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
    <timedJobId>0</timedJobId>
  </jobSummary>
  <jobSummary id="3" href="http://localhost/mxerest/jobs/3/detail">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:09:23</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
    <timedJobId>0</timedJobId>
  </jobSummary>
</jobList>
```

**Limitations**

None

## Get Job List Since

Get list of jobs submitted to ECS that have changed since specified timestamp.

### URL Template

http://localhost/mxerest/jobs/since/[timestamp]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
timestamp	Timestamp for job status. To be used as a seed for querying subsequent job status and returning only jobs that have changed since timestamp.
type	Job list type: <b>Full</b> - all non-reaped jobs are returned <b>Delta</b> - only jobs changed since the timestamp are returned
jobSummary	Job summary section id = job ID href = link to job detail
title	Job title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes
completionReason	Job completion reason. <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job stopped at request of operator <b>SystemStopped</b> - job stopped by system
submitTime	Time job was submitted: yyyy-mm-dd hh:mm:ss



Field	Description
completeTime	Time job was completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority
liveWorker	Live worker name (capture station) reserved by this job.
reservedDuration	Amount of time in seconds the live worker (capture station) is reserved.
timedJobId	The ID of the associated timed job. <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of a job XML, including any user-defined metadata.

### Parameters

Parameter	Description	Expected Value	Status
timestamp	Timestamp value returned from previous get job list or get job list since call.	yyyy-mm-dd hh:mm:ss	NA

### Expected Results

Returns list of job summaries for any job changed, added, or deleted since specified timestamp

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<jobList>
  <type>delta</type>
  <timestamp>1126807551</timestamp>
  <jobSummary id="2" href="http://localhost/mxerest/jobs/2/detail">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:06:46</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
    <timedJobId>0</timedJobId>
  </jobSummary>
  <jobSummary id="3" href="http://localhost/mxerest/jobs/3/detail">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:09:23</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
  </jobSummary>
</jobList>
```

```
    <timedJobId>0</timedJobId>  
  </jobSummary>  
</jobList>
```

**Limitations**

None

## Get Job Summary

Returns job ID, status timestamp, title, author, status, state, submit time, complete time, and priority. Also includes link to job detail.

### URL Template

http://localhost/mxerest/jobs/[jobId]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
timestamp	Timestamp for job status. To be used as a seed for querying subsequent job status and returning only jobs that have changed since timestamp.
type	Job list type: <b>Full</b> <b>Delta</b>
jobSummary	Job summary section id = job ID href = link to job detail
title	Job title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes
completionReason	Job completion reason. <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job stopped at request of operator <b>SystemStopped</b> - job stopped by system
submitTime	Time job was submitted: yyyy-mm-dd hh:mm:ss

Field	Description
completeTime	Time job was completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority
liveWorker	Live worker name (capture station) reserved by this job.
reservedDuration	Amount of time in seconds the live worker (capture station) is reserved.
timedJobId	The ID of the associated timed job. <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of a job XML, including any user-defined metadata.

### Parameters

Parameter	Description	Expected Value	Status
jobId	Job ID		

### Expected Results

Summary of specified job.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<jobSummary id="14" href="http://localhost/mxerest/jobs/14/detail">
<title>Documentation Example</title>
<author>admin</author>
<status>Running</status>
<completionReason>NotComplete</completionReason>
<state>new</state>
<submitTime>2010-11-19 11:25:32</submitTime>
<completeTime/>
<priority>2</priority>
<liveWorker/>
<reservedDuration>0</reservedDuration>
<timedJobId>0</timedJobId>

<userData>
<submit-machine>johndoe-xpm</submit-machine>
<job-profile>Blackberry 81xx WMV 16x9</job-profile>
<source-clip>\\johndoe-xpm\media\Documentation Example.mov</source-clip>
<start-timecode>00:00:00:00</start-timecode>
<stop-timecode>00:00:00:00</stop-timecode>
<source-inpoint>00:00:00.000</source-inpoint>
<source-outpoint>00:00:00.000</source-outpoint>
<description/>
<keywords/>
<audience-rating>None</audience-rating>
<basename>Documentation Example</basename>
</userData>
</jobSummary>
```

**Limitations**

None

## Get Job Detail

Returns job summary information along with detail for each worker task.

### URL Template

`http://localhost/mxerest/jobs/[jobId]/detail`

### HTTP Method

GET

### XML Response Data Structure

Field	Description
jobDetail	Job detail section id = job ID
title	Job Title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled. <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state: <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes
completionReason	Job completion reason: <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job was stopped at request of operator <b>SystemStopped</b> - job was stopped by system
submitTime	Time the job was submitted: yyyy-mm-dd hh:mm:ss
completeTime	Time the job completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority
liveWorker	Live worker name (capture status) reserved by this job.

Field	Description
reservedDuration	The amount of time in seconds the live worker (capture station) is reserved.
timeJobId	The ID of the associated timed job <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of job XML including any user-defined metadata.
taskList	The section containing the worker task detail.
task	Worker task section: id = worker task ID
status	Task status: <b>Pending</b> - task is waiting to be scheduled <b>Dependent</b> - task is dependent on one or more task start/complete events <b>Provisioned</b> - request to start task has been sent to LCS <b>Running</b> - the task is executing <b>StopRequest</b> - request to stop task has been sent to LCS <b>Completed</b> - task completed successfully <b>Failed</b> - task failed
type	Task type/worker name (prefilter, fileman, flash, etc.)
startTime	Task start time: yyyy-mm-dd hh:mm:ss OR “-not started-”
stopTime	Task stop time: yyyy-mm-dd hh:mm:ss
percentComplete	Percentage of task completed
elapsedSeconds	Current elapsed time in seconds for the task.
failToken	list of error message tokens for the task.
failMessage	Concatenated list of error messages/warnings for the task.
userData	User data for the task, for example task display name and task profile name.

### Parameters

Parameter	Description	Expected Value	Status
jobId	Job ID		

### Expected Results

Job detail for specified job including detail for each task.

**XML Response**

```

<?xml version="1.0" encoding="utf-8"?>
  <jobDetail id="6">
    <title>---</title>
    <author>---</author>
    <status>Pending</status>
    <completionReason>NotComplete</completionReason>
    <state>new</state>
    <submitTime>2005-09-14 15:13:40</submitTime>
    <completeTime />
    <priority>100</priority>
    <liveWorker></liveWorker>
    <reservedDuration>0</reservedDuration>
    <timedJobId>0</timedJobId>

    <userData><submit-machine>VA01LTMGROSS</submit-machine><job-profile>package</job-profile><
source-clip>C:\DOCUME~1\mgross\LOCALS~1\Temp\temp_any_file.any</source-clip><start-timecode
e>00:00:00:00</start-timecode><stop-timecode>00:00:00:00</stop-timecode><source-inpoint>00
:00:00.000</source-inpoint><source-outpoint>00:00:00.000</source-outpoint><description></d
escription><keywords></keywords><audience-rating>none</audience-rating><basename></basenam
e></userData>
    <taskList>
      <task id="26">
        <status>Pending</status>
        <type>prefilter</type>
        <startTime> -- not started --</startTime>
        <stopTime />
        <percentComplete>0</percentComplete>
        <elapsedSeconds>0</elapsedSeconds>

        <userData><display-name>prefilter</display-name><task-profile>Precondition</task-profile><
/userData>
          <failToken />
          <failMessage />
        </task>
      <task id="27">
        <status>Pending</status>
        <type>prefilter</type>
        <startTime> -- not started --</startTime>
        <stopTime />
        <percentComplete>0</percentComplete>
        <elapsedSeconds>0</elapsedSeconds>

        <userData><display-name>prefilter</display-name><task-profile>Precondition</task-profile><
/userData>
          <failToken />
          <failMessage />
        </task>
      <task id="28">
        <status>Dependent</status>
        <type>prefilter</type>
        <startTime> -- not started --</startTime>
        <stopTime />
        <percentComplete>0</percentComplete>
        <elapsedSeconds>0</elapsedSeconds>

        <userData><display-name>prefilter</display-name><task-profile>package</task-profile></user
Data>
          <failToken />
          <failMessage />
        </task>
      <task id="29">
        <status>Dependent</status>

```



```
<type>microsoft</type>
<startTime> -- not started --</startTime>
<stopTime />
<percentComplete>0</percentComplete>
<elapsedSeconds>0</elapsedSeconds>

<userData><task-profile>proxy</task-profile><display-name>microsoft</display-name><clip-lo
cation>\\VA01LTMGROSS\media\wmv</clip-location></userData>
  <failToken />
  <failMessage />
</task>
<task id="30">
  <status>Depend</status>
  <type>fileman</type>
  <startTime> -- not started --</startTime>
  <stopTime />
  <percentComplete>0</percentComplete>
<elapsedSeconds>0</elapsedSeconds>
  <userData><display-name>cleanup</display-name></userData>
  <failToken />
  <failMessage />
</task>
</taskList>
</jobDetail>
```

### Limitations

None

## Get Job Detail with Accounting

Returns job summary information along with detail for each worker task including accounting data (e.g. source-fps). See MXE Accounting Data section, [Accounting Tags, page 7-4](#) for more detail on what accounting data is returned.

### URL Template

http://localhost/mxerest/jobs/[jobId]/accounting

### HTTP Method

GET

### XML Response Data Structure

Field	Description
jobDetail	Job detail section id = job ID
title	Job Title
author	Job author
status	Job status: <b>Pending</b> - job is waiting to be scheduled. <b>Running</b> - job is running <b>Completed</b> - job has completed successfully <b>Failed</b> - job has failed
state	Job state: <b>New</b> - job is new since last status request <b>Update</b> - job status has changed since last status request <b>Deleted</b> - job has completed/failed and will be reaped from memory within 60 minutes
completionReason	Job completion reason: <b>NotComplete</b> - job has not started or is running <b>Succeeded</b> - job completed successfully <b>Failed</b> - job failed <b>UserStopped</b> - job was stopped at request of operator <b>SystemStopped</b> - job was stopped by system
submitTime	Time the job was submitted: yyyy-mm-dd hh:mm:ss
completeTime	Time the job completed: yyyy-mm-dd hh:mm:ss <b>Note</b> This is blank if the job is still running.
priority	Job priority

Field	Description
liveWorker	Love worker name (capture status) reserved by this job.
reservedDuration	The amount of time in seconds the live worker (capture station) is reserved.
timeJobId	The ID of the associated timed job <b>Note</b> This is 0 if there is no associated timed job.
userData	XML data from the user-data-job section of job XML including any user-defined metadata.
taskList	The section containing the worker task detail.
task	Worker task section: id = worker task ID
status	Task status: <b>Pending</b> - task is waiting to be scheduled <b>Dependent</b> - task is dependent on one or more task start/complete events <b>Provisioned</b> - request to start task has been sent to LCS <b>Running</b> - the task is executing <b>StopRequest</b> - request to stop task has been sent to LCS <b>Completed</b> - task completed successfully <b>Failed</b> - task failed
type	Task type/worker name (prefilter, fileman, flash, etc.)
startTime	Task start time: yyyy-mm-dd hh:mm:ss OR “-not started-”
stopTime	Task stop time: yyyy-mm-dd hh:mm:ss
percentComplete	Percentage of task completed
elapsedSeconds	Current elapsed time in seconds for the task.
failToken	list of error message tokens for the task.
failMessage	Concatenated list of error messages/warnings for the task.
userData	User data for the task, for example task display name and task profile name.
<b>Accounting Tags</b>	
<b>Description</b>	
<b>General Tags</b>	
worker-type	<b>Note</b> Refer to “ <a href="#">Worker-Types</a> ” section on page 7-6 table for a list of values.
<b>Input-related Tags</b>	
input-type	Type of input. For example, tape, live, or file.
input-file-name	Source file name, including path.

Field	Description
start-timecode	Start time for file and tape jobs.
stop-timecode	Stop time for tape jobs only.
source-duration	Job duration in seconds.
decoder-type	Decompression pipeline used.
video-format	Video format of the source clip.
audio-format	Audio format of the source clip.
source-width	Width of the input clip.
source-height	Height of the input clip.
source-fps	Frame rate of the input clip.
source-sample-rate	Audio sample rate of the input clip.
source-num-audio-channels	Number of audio channels in the input clip.
capture-start-time	Time of day the capture started.
capture-stop-time	Time of day the capture stopped.
<b>Output-related Tags</b>	
generated-file	This is the name of the intermediate output file produced by the prefilter. This tag will only be present when the worker type is the prefilter.
max-fps	Maximum frames per second of the output file.
max-width	Maximum width of the output file.
max-height	Maximum height of the output file.
duration	Duration of the output file.
max-video-bitrate	Maximum bitrate of the output file.
max-audio-bitrate	Maximum audio bitrate of the output file.
video-codec	Video codec used in the output file.
audio-codec	Audio codec used in the output file.
image-codec	Miage codec used in the output file.
output-file-name	Primary output file(s)
output-file-size	Size of the output file.
output-stream-name	Name of the stream.
output-stream-port	Port of the stream.
user-data	Codec-specific data.

**Parameters**

Parameter	Description	Expected Value	Status
jobId	Job ID		

**Expected Results**

Job detail including task detail and accounting data.

**XML Response**

```

<?xml version="1.0" encoding="utf-8"?>
<jobDetail id="120">
  <title>accounting test</title>
  <author>admin</author>
  <status>Completed</status>
  <completionReason>Succeeded</completionReason>
  <state>new</state>
  <submitTime>2007-06-13 08:19:04</submitTime>
  <completeTime>2007-06-13 08:19:35</completeTime>
  <priority>2</priority>
  <liveWorker></liveWorker>
  <reservedDuration>0</reservedDuration>
  <timedJobId>0</timedJobId>

  <userData><submit-machine>VA01LTMGROSS</submit-machine><job-profile>conform</job-profile><
source-clip>C:\media\clips\Bridge.mov</source-clip><start-timecode>00:00:00:00</start-time
code><stop-timecode>00:00:00:00</stop-timecode><source-inpoint>00:00:00.000</source-inpoin
t><source-outpoint>00:00:00.000</source-outpoint><description></description><keywords></ke
ywords><audience-rating>none</audience-rating><basename>Bridge</basename></userData>
  <taskList>
    <task id="458">
      <status>Completed</status>
      <type>prefilter</type>
      <startTime>2007-06-13 08:19:04</startTime>
      <stopTime>2007-06-13 08:19:35</stopTime>
      <percentComplete>100</percentComplete>
      <elapsedSeconds>27</elapsedSeconds>

      <userData><display-name>prefilter</display-name><task-profile>conform</task-profile></user
Data>
      <failToken />
      <failMessage />
      <accountingData>
        <worker-type>prefilter both</worker-type>
        <input-file-name>c:\media\clips\Bridge.mov</input-file-name>
        <output-file-name>c:/TMP\Bridge437_pp.avi</output-file-name>
        <generated-file>c:/TMP\Bridge437_pp.audio</generated-file>
        <input-type>file</input-type>
        <decoder-type>quicktime</decoder-type>
        <source-duration>5.405</source-duration>
        <duration>5.405</duration>
        <start-timecode>00:00:00:00</start-timecode>
        <source-sample-rate>44100</source-sample-rate>
        <source-num-audio-channels>1</source-num-audio-channels>
        <source-width>720</source-width>
        <source-height>486</source-height>
        <source-fps>29.970</source-fps></accountingData>
      </task>
    <task id="459">
      <status>Completed</status>
      <type>mpeg</type>
      <startTime>2007-06-13 08:19:04</startTime>
      <stopTime>2007-06-13 08:19:35</stopTime>
      <percentComplete>100</percentComplete>
      <elapsedSeconds>30</elapsedSeconds>

      <userData><task-profile>conform</task-profile><display-name>mpeg</display-name><clip-locat
ion>\\VA01LTMGROSS\output\mpeg\Bridge.mpg</clip-location></userData>
      <failToken />
      <failMessage />
      <accountingData>
        <worker-type>MPEG both</worker-type>

```

```
<max-fps>29.970</max-fps>
<max-width>720</max-width>
<max-height>480</max-height>
<duration>5.405</duration>
<max-video-bitrate>50000000</max-video-bitrate>
<max-audio-bitrate>384000</max-audio-bitrate>
<video-codec>mpeg2</video-codec>
<audio-codec>Layer 2</audio-codec>
<output-file-name>C:\output\mpeg\Bridge.mpg</output-file-name>
<output-file-size>34422788</output-file-size>
<user-data></user-data>
  </accountingData>
</task>
</taskList>
</jobDetail>
```

**Limitations**

None

## Get Job XML

Get submitted job XML.

### URL Template

http://localhost/mxerest/jobs/[jobId]/jobxml

### HTTP Method

GET

### XML Response Data Structure

Data structure based on submitted job.

### Parameters

Parameter	Description	Expected Value	Status
jobId	Job ID		

### Expected Results

Expanded Job XML for specified job

### XML Response

```
<job xmlns:ecs="urn:Anystream.com:MXE:ecs">
    [JOB XML]
</job>
```

### Limitations

None

## Get Job Notifications

Get job task notification (error) messages. Returns a list of task notifications containing a message and time of message

### URL Template

http://localhost/mxerest/jobs/[jobId]/notifications

### HTTP Method

GET

### XML Response Data Structure

Field	Description
notificationList	Section containing worker task notifications.
notification	Task notification: <b>taskId</b> - task ID <b>taskType</b> - task type
time	Time of notification: yyyy-mm-dd hh:mm:ss
nodeHost	Hostname of node that executed task.
taskExecutionId	Task execution ID

### Parameters

Parameter	Description	Expected Value	Status
jobId	Job ID		

### Expected Results

List of task-specific errors if job failed.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<notificationList>
  <notification taskId="26" taskType="windowsmedia2">
    <message>Internal error, MBREncStart() failed.</message>
    <time>2009-05-21 10:56:02</time>
    <nodeHost>DEVTESTECS</nodeHost>
    <taskExecutionId>30</taskExecutionId>
  </notification>
  <notification taskId="26" taskType="windowsmedia2">
    <message>[EC_COMPLETED] Task Execution 30 is now complete. Reason = Failed.</message>
    <time>2009-05-21 10:56:02</time>
    <nodeHost>DEVTESTECS</nodeHost>
    <taskExecutionId>30</taskExecutionId>
  </notification>
  <notification taskId="27" taskType="cleanup">
    <message>[EC_DEPENDFAIL] Task Execution 31 will not run due to dependent Task
Execution 30. Reason = Failed.</message>
    <time>2009-05-21 10:56:02</time>
    <nodeHost>DEVTESTECS</nodeHost>
```



```
<taskExecutionId>31</taskExecutionId>  
</notification>  
</notificationList>
```

**Limitations**

None

## Stop Job

Stop a running job.

### URL Template

http://localhost/mxerest/stopjob

### HTTP Method

POST

### POST Data Structure

Field	Description
stopJob	Command.
id	ID of job to stop.

### XML Response Data Structure

Field	Description
status	success
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Job is stopped. Success response.

### POST Data

```
<stopJob id="[jobId]"></stopJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

Job must be running to be stopped.

## Delete Job

Delete a job.

### URL Template

http://localhost/mxerest/deletejob

### HTTP Method

POST

### POST Data Structure

Field	Description
deleteJob	Command.
id	ID of job to be deleted.

### XML Response Data Structure

Field	Description
status	success
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Job will be stopped and deleted from job status monitor. Success response

### POST Data

```
<deleteJob id="[jobId]"></deleteJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

None

## Reschedule Job

Reschedule/resubmit previously submitted job.

### URL Template

http://localhost/mxerest/reschedulejob

### HTTP Method

POST

### POST Data Structure

Field	Description
rescheduleJob	Command.
id	ID of job to be rescheduled.

### XML Response Data Structure

Field	Description
status	success
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Job is restarted and success response.

### POST Data

```
<rescheduleJob id="[jobId]"></rescheduleJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

Can only reschedule a job that has completed.

## Retry Job

Retry failed (and dependent) tasks for a completed job that had errors.

### URL Template

`http://localhost/mxerest/retryjob`

### HTTP Method

POST

### POST Data Structure

Field	Description
retryJob	Command.
id	ID of job to be retried.

### XML Response Data Structure

Field	Description
status	success
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Failed and dependent tasks are restarted. Success response.

### POST Data

```
<retryJob id="[jobId]"></retryJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

Can only retry a completed job.

## Update Job Priority

Update job priority (to integer value > 0). Jobs with higher priorities receive precedence over jobs with lower priorities.

### URL Template

http://localhost/mxerest/updatejob

### HTTP Method

POST

### POST Data Structure

Field	Description
updateJob	Command
jobSummary	Container for request
id	ID of updated job
priority	New priority for job

### XML Response Data Structure

Field	Description
status	success
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Job priority is reset to new value. Success response.

### POST Data

```
<updateJob>
  <jobSummary id="[jobId]">
    <priority>[jobPriority]</priority>
  </jobSummary>
</updateJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

Can only update priority of pending job.

# Timed Job Control Commands

Commands used to query and manage timed jobs.

- [Get Timed Job List, page 3-32](#)
- [Get Timed Job, page 3-34](#)
- [Pause Timed Job, page 3-36](#)
- [Resume Timed Job, page 3-37](#)
- [Delete Timed Job, page 3-38](#)

## Get Timed Job List

Get list of timed jobs.

### URL Template

http://localhost/mxerest/timedJobs

### HTTP Method

GET

### XML Response Data Structure

Field	Description
timedJobList	Section containing list of timed jobs
timedJob	Timed job id = timed job ID
title	Job title
author	Job author
status	Job status: <b>Active</b> <b>Inactive</b>
priority	Job priority
startTime	Start time of job: yyyy-mm-dd hh:mm:ss
lastAdded	Time job was last submitted: yyyy-mm-dd hh:mm:ss OR Never
period	Interval (seconds) at which job is submitted
reservedDuration	Amount of time in seconds live worker (capture station) is reserved.
liveWorker	Live worker name (capture station) reserved by this job.

### Parameters

Parameter	Description	Expected Value	Status
NONE			

### Expected Results

List of timed jobs.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<timedJobList>
<timedJob id="31">
  <title>timed job test</title>
  <author>admin</author>
  <status>Active</status>
```



```
<priority>2</priority>
<startTime>2006-09-13 00:00:00</startTime>
<lastAdded>never</lastAdded>
<period>86400</period>
<liveWorker></liveWorker>
<reservedDuration>0</reservedDuration>
</timedJob>
</timedJobList>
```

**Limitations**

None

## Get Timed Job

Get timed job summary.

### URL Template

http://localhost/mxerest/timedJobs/[jobId]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
timedJob	Timed job id = timed job ID
title	Job title
author	Job author
status	Job status: <b>Active</b> <b>Inactive</b>
priority	Job priority
startTime	Start time of job: yyyy-mm-dd hh:mm:ss
lastAdded	Time job was last submitted: yyyy-mm-dd hh:mm:ss OR Never
period	Interval (seconds) at which job is submitted
reservedDuration	Amount of time in seconds live worker (capture station) is reserved.
liveWorker	Live worker name (capture station) reserved by this job.

### Parameters

Parameter	Description	Expected Value	Status
jobId	Timed job ID.		

### Expected Results

Summary of timed job.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<timedJob id="31">
  <title>timed job test</title>
  <author>admin</author>
  <status>Active</status>
  <priority>2</priority>
```

```
<startTime>2006-09-13 00:00:00</startTime>  
<lastAdded>never</lastAdded>  
<period>86400</period>  
<liveWorker></liveWorker>  
<reservedDuration>0</reservedDuration>  
</timedJob>
```

**Limitations**

None

## Pause Timed Job

Pause a timed job.

### URL Template

`http://localhost/mxerest/PauseTimedJob`

### HTTP Method

POST

### POST Data Structure

Field	Description
pauseTimedJob	Command.
id	ID of job to be paused.

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Timed job is paused and success response.

### POST Data

```
<pauseTimedJob id="[jobId]"></pauseTimedJob>
```

### XML Response

```
<status>success</status>
```

### Limitations

None

## Resume Timed Job

Resume a paused timed job.

### URL Template

http://localhost/mxerest/ResumeTimedJob

### HTTP Method

POST

### POST Data Structure

Field	Description
resumeTimedJob	Command.
id	ID of paused job to be resumed.

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Timed job is submitted at next available period and success response.

### POST Data

```
<resumeTimedJob id="[jobId]"></resumeTimedJob>
```

### XML Response

```
<status>success</status>
```

### Limitations

Timed job must be paused first.

## Delete Timed Job

Delete a timed job.

### URL Template

http://localhost/mxerest/DeleteTimedJob

### HTTP Method

POST

### POST Data Structure

Field	Description
deleteTimedJob	Command.
id	ID of timed Job to be deleted.

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Timed job is stopped and deleted from status monitor. Success response.

### POST Data

```
<deleteTimedJob id="[jobId]"></deleteTimedJob>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

None

# Node Status Commands

Node Status commands are used to query the status of Cisco MXE 3500 nodes. A node is a host machine that comprises a Local Control System (LCS) and a set of licensed workers and optionally an ECS.

- [Get Node List, page 3-40](#)
- [Get Node Summary, page 3-42](#)
- [Get Node Detail, page 3-44](#)

## Get Node List

Get a list of configured host machines.

### URL Template

http://localhost/mxerest/nodes

### HTTP Method

GET

### XML Response Data Structure

Field	Description
nodeInfoList	Section containing list of node summaries
nodeInfo	Node summary section id - node id href - link to node detail
capacity	Node capacity. Indicates the relative processing capacity of the host, expressed as the maximum sum of all worker expenses that can be run on the host at any given time.
taskLoad	Sum of all task expenses. Expense indicates share of host's capacity that a single task for the given worker consumes.
load	Task load / capacity as a percentage
host	Host name of node
tempdir	Temporary directory for workers executing on node
status	Node status: enabled disabled

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of nodes configured in system.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<nodeInfoList>
<nodeInfo id="3" href="http://localhost/mxerest/nodes/3/detail">
  <host>localhost</host>
  <capacity>10</capacity>
  <taskLoad>0</taskLoad>
  <load>0</load>
  <tempdir>\\va-acrttest03\TMP</tempdir>
  <status>enabled</status>
</nodeInfo>
```



```
</nodeInfoList>
```

**Limitations**

None

## Get Node Summary

Get node summary for node ID (or hostname).

### URL Template

`http://localhost/mxerest/nodes/[nodeId]`

`http://localhost/mxerest/nodes/[nodeHostName]`

### HTTP Method

GET

### XML Response Data Structure

Field	Description
nodeInfoList	Section containing list of node summaries
nodeInfo	Node summary section id - node id href - link to node detail
capacity	Node capacity. Indicates the relative processing capacity of the host, expressed as the maximum sum of all worker expenses that can be run on the host at any given time.
taskLoad	Sum of all task expenses. Expense indicates share of host's capacity that a single task for the given worker consumes.
load	Task load / capacity as a percentage
host	Host name of node
tempdir	Temporary directory for workers executing on node
status	Node status: enabled disabled

### Parameters

Parameter	Description	Expected Value	Status
nodeId	Node ID		
nodeHostName	Node host name		

### Expected Results

Node summary.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<nodeInfoList>
<nodeInfo id="3" href="http://localhost/mxerest/nodes/3/detail">
  <host>localhost</host>
  <capacity>10</capacity>
  <taskLoad>0</taskLoad>
  <load>0</load>
```

```
<tempdir>\\va-acrtest03\TMP</tempdir>  
<status>enabled</status>  
</nodeInfo>  
</nodeInfoList>
```

**Limitations**

None

## Get Node Detail

Get node detail for node ID (or host name) including status for each worker associated with the node and task status for any task being executed by the worker.

### URL Template

http://localhost/mxerest/nodes/[nodeId]/detail

http://localhost/mxerest/nodes/[nodeHostName]/detail

### HTTP Method

GET

### XML Response Data Structure

Field	Description
nodeInfoList	Section containing list of node summaries
nodeInfo	Node summary section id - node id href - link to node detail
capacity	Node capacity. Indicates the relative processing capacity of the host, expressed as the maximum sum of all worker expenses that can be run on the host at any given time.
taskLoad	Sum of all task expenses. Expense indicates share of host's capacity that a single task for the given worker consumes.
load	Task load / capacity as a percentage
host	Host name of node
tempdir	Temporary directory for workers executing on node
status	Node status: enabled disabled
workerStatus	Worker status section
type	Type of worker (prefilter, quicktime, etc.)
status	Worker status (online/offline/paused)
taskCount	Number of tasks executed by worker
expense	Task expense indicating share of host's capacity that a single task can consume
runs	Number of successful tasks executed
fails	Number of failed tasks executed
abbrev	Worker type abbreviation. <b>Note</b> Defaults to <type> value
task	Worker task section: id - worker task ID <b>Note</b> Only present if task is executing on worker

Field	Description
status	Task status: <b>pending</b> <b>failed</b> <b>depend</b> <b>completed</b> <b>running</b> <b>unknown</b> <b>Note</b> A value of <b>depend</b> indicates that the task is pending and its execution depends on the successful completion of some other task.
type	Task type/worker name (prefilter, fileman, flash, etc.)
startTime	Task start time: yyyy-mm-dd hh:dd:ss OR "--not started--"
stopTime	Task stop time: yyyy-mm-dd hh:dd:ss
percentComplete	Percentage of task completed
elapsedSeconds	Duration in seconds since task began. <b>Note</b> A value of 0 indicates that the task has not yet started
userData	Client data submitted on original job task request
failToken	Error message identifier if task failed
failMessage	Error message if task failed

### Parameters

Parameter	Description	Expected Value	Status
nodeId	Node ID		
nodeHostName	Node host name.		

### Expected Results

Detail node report for requested nodeId or nodeHostName.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<nodeInfoList>
<nodeInfo id="3" href="http://localhost/mxerest/nodes/3/detail">
  <host>localhost</host>
  <capacity>10</capacity>
  <taskLoad>1</taskLoad>
  <load>10</load>
  <tempdir>\\va-acrttest03\TMP</tempdir>
  <status>enabled</status>
  <workerStatus id="795">
    <type>prefilter</type>
    <status>online</status>
    <taskCount>1</taskCount>
    <expense>1</expense>
    <runs>0</runs>
    <fails>0</fails>
  </workerStatus>
</nodeInfo>
</nodeInfoList>
```

```

<abbrev>PF</abbrev>
<task id="7138">
  <status>Running</status>
  <type>prefilter</type>
  <startTime>2006-09-29 08:49:03</startTime>
  <stopTime />
  <percentComplete>0</percentComplete>
  <elapsedSeconds>0</elapsedSeconds>
  <userData>prefilterLow Bandwidth</userData>
  <failToken />
  <failMessage />
</task>
</workerStatus>
<workerStatus id="796">
  <type>quicktime</type>
  <status>online</status>
  <taskCount>0</taskCount>
  <expense>1</expense>
  <runs>0</runs>
  <fails>0</fails>
  <abbrev>QT</abbrev>
</workerStatus>
<workerStatus id="797">
  <type>real</type>
  <status>online</status>
  <taskCount>0</taskCount>
  <expense>1</expense>
  <runs>0</runs>
  <fails>0</fails>
  <abbrev>RL</abbrev>
</workerStatus>
</nodeInfo>
</nodeInfoList>

```

**Limitations**

None

# System Setting Commands

Commands for retrieving system settings.

- [Get System Settings, page 3-48](#)
- [Get System Settings for Names, page 3-49](#)

## Get System Settings

Get list of all system settings.

### URL Template

<http://localhost/mxerest/settings>

### HTTP Method

GET

### XML Response Data Structure

Field	Description
settingList	Command.
setting	System setting name - setting name href - link to setting summary

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of configured system settings.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<settingList>
<setting name="banner-loc"
href="http://localhost/mxerest/settings/banner-loc">images\MXE.gif</setting>
<setting name="company-logo-loc"
href="http://localhost/mxerest/settings/company-logo-loc">images\Cisco.gif</setting>
</settingList>
```

### Limitations

None



## Get System Settings for Names

Get system settings matching specified setting names.

### URL Template

http://localhost/mxerest/settings/[name1]/[name2]/.../[nameN]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
settingList	Section containing list system settings
setting	System setting name - setting name href - link to setting summary

### Parameters

Parameter	Description	Expected Value	Status
name1	Setting name delimited by /		

### Expected Results

List of system settings matching name(s).

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<settingList>
  <setting name="banner-loc"
    href="http://localhost/mxerest/settings/banner-loc">images\MXE.gif</setting>
  <setting name="company-logo-loc"
    href="http://localhost/mxerest/settings/company-logo-loc">images\Cisco.gif</setting>
</settingList>
```

### Limitations

None

# Profile Commands

Commands for querying details for job profiles and profile types.

- [Get Profile Types, page 3-51](#)
- [Get All Profile Summaries, page 3-52](#)
- [Get Profile Summaries for Type, page 3-53](#)
- [Get Detailed Profile by Type and Name, page 3-54](#)
- [Get Profile Spaces, page 3-56](#)
- [Get Profile Summaries by Type and Profile Space, page 3-57](#)
- [Get Detailed Profile by Type, Name, and Profile Space, page 3-58](#)

## Get Profile Types

Get list of all job profile types. Returns a list of <profileType> with links to the profile summary by type.

### URL Template

http://localhost/mxerest/profiles

### HTTP Method

GET

### XML Response Data Structure

Field	Description
profileTypeList	Section containing list profile types
profileType	Profile type: type - profile type href - link to profile summary list for profile type

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of profile types.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<profileTypeList>
<profileType type="adi" href="http://localhost/mxerest/profiles/adi">VOD ADI</profileType>
<profileType type="adi-templates"
href="http://localhost/mxerest/profiles/adi-templates">VOD ADI Templates</profileType>
<profileType type="adi-transforms"
href="http://localhost/mxerest/profiles/adi-transforms">VOD ADI Transforms</profileType>
</profileTypeList>
```

### Limitations

None

## Get All Profile Summaries

Get list of all job profile summaries. Returns a list of profile summaries <profileSummary> with links to the detailed profile

### URL Template

http://localhost/mxerest/profiles/all

### HTTP Method

GET

### XML Response Data Structure

Field	Description
profileSummaryList	Section containing list profile summaries
profileSummary	Profile name without extension.

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of all profile summaries.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<profileSummaryList>
  <profileSummary type="pp" name="conform"
    href="http://localhost/mxerest/profiles/pp/conform" />
  <profileSummary type="pp" name="Digital Master.Digital Master"
    href="http://localhost/mxerest/profiles/pp/Digital+Master.Digital+Master" />
  <profileSummary type="pp" name="Digital Master_watermark.Digital Master_watermark"
    href="http://localhost/mxerest/profiles/pp/Digital+Master_watermark.
    Digital+Master_watermark" />
  <profileSummary type="pp" name="Full Frame"
    href="http://localhost/mxerest/profiles/pp/Full+Frame" />
  <profileSummary type="pp" name="High Bandwidth"
    href="http://localhost/mxerest/profiles/pp/High+Bandwidth" />
  <profileSummary type="pp" name="Low Bandwidth"
    href="http://localhost/mxerest/profiles/pp/Low+Bandwidth" />
  <profileSummary type="pp" name="package"
    href="http://localhost/mxerest/profiles/pp/package" />
</profileSummaryList>
```

### Limitations

None

## Get Profile Summaries for Type

Get list of profile summaries by specified type. Returns a list of profile summaries <profileSummary> with links to the detailed profile.

### URL Template

http://localhost/mxerest/profiles/[profileType]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
profileSummaryList	Section containing list profile summaries
profileSummary	Profile name without extension.

### Parameters

Parameter	Description	Expected Value	Status
profileType	Type of profile.	type - profile type (flash, job, etc.) href - link to profile detail	

### Expected Results

None

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<profileSummaryList>
<profileSummary type="pp" name="conform"
href="http://localhost/mxerest/profiles/pp/conform" />
<profileSummary type="pp" name="Digital Master.Digital Master"
href="http://localhost/mxerest/profiles/pp/Digital+Master.Digital+Master" />
<profileSummary type="pp" name="Digital Master_watermark.Digital Master_watermark"
href="http://localhost/mxerest/profiles/pp/Digital+Master_watermark.Digital+Master_watermark" />
<profileSummary type="pp" name="Full Frame"
href="http://localhost/mxerest/profiles/pp/Full+Frame" />
<profileSummary type="pp" name="High Bandwidth"
href="http://localhost/mxerest/profiles/pp/High+Bandwidth" />
<profileSummary type="pp" name="Low Bandwidth"
href="http://localhost/mxerest/profiles/pp/Low+Bandwidth" />
<profileSummary type="pp" name="package"
href="http://localhost/mxerest/profiles/pp/package" />
</profileSummaryList>
```

### Limitations

None

## Get Detailed Profile by Type and Name

Get detailed profile contents by specified type and name.

### URL Template

http://localhost/mxerest/profiles/[profileType]/[profileName]

### HTTP Method

GET

### XML Response Data Structure

Determined by Job XML.

### Parameters

Parameter	Description	Expected Value	Status
profileType	Type of profile.	flash, job, h264, etc.	
profileName	Profile name without extension.		

### Expected Results

XML contents of specified profile.

### XML Response

```
<profile type="pp" name="package" xmlns:axe="urn:
microsoft.com:ecs"><class>preprocessor</class><enabled>yes</enabled><export><encoder><fast
-start>no</fast-start><immediate>no</immediate></encoder><preprocessor><two-pass-capture>n
o</two-pass-capture></preprocessor></export><parameters><audio><channels><axe:lookup
name="audio-channels" /></channels><compressor><ratio>1</ratio><threshold>6</threshold></co
mpressor><enabled><axe:lookup
name="audio-enabled" /></enabled><fade-in>0.0</fade-in><fade-out>0.0</fade-out><high-pass>0
</high-pass><low-pass>0</low-pass><volume><adjust>50</adjust><normalize>50</normalize><typ
e>none</type></volume></audio><cc><over-video>false</over-video></cc><in-point>0.000</in-p
oint><out-point>0.000</out-point><prefilter>on</prefilter><video><black-point><point>0</po
int><transition>15</transition></black-point><brightness>100</brightness><bumper><input-di
rectory><axe:lookup
name="bumper-directory" /></input-directory><input-file></input-file><prefilter>off</prefil
ter></bumper><color-range-expansion>no</color-range-expansion><contrast>100</contrast><cro
p><bottom>0</bottom><left>0</left><right>0</right><top>0</top></crop><custom-blur>0.0</cus
tom-blur><destination><fps><axe:lookup name="video-max-fps" /></fps><height><axe:lookup
name="video-max-height" /></height><progressive><axe:lookup
name="video-progressive" /></progressive><width><axe:lookup
name="video-max-width" /></width></destination><enabled><axe:lookup
name="video-enabled" /></enabled><fade-in>0.0</fade-in><fade-out>0.0</fade-out><field-domin
ance>upper</field-dominance><gamma>1.0</gamma><hue>0</hue><inverse-telecine>off</inverse-t
elecine><noise-reduction>0.0</noise-reduction><saturation>100</saturation><single-field>of
f</single-field><temporal-smoothing>1</temporal-smoothing><trailer><input-directory><axe:l
ookup
name="bumper-directory" /></input-directory><input-file></input-file><prefilter>off</prefil
ter></trailer><unsharp-mask>no</unsharp-mask><unsharp-mask-radius>0.0</unsharp-mask-radius
><unsharp-mask-weight>100</unsharp-mask-weight><watermark><height>100</height><input-direc
tory><axe:lookup
name="watermark-directory" /></input-directory><input-file></input-file><mode>composite</mo
```

```
de><strength>100</strength><width>200</width><x>0</x><x-origin>right</x-origin><y>0</y><y-  
origin>bottom</y-origin></watermark><white-point><point>255</point><transition>15</transit  
ion></white-point></video></parameters><version>2.0</version></profile>
```

**Limitations**

None

## Get Profile Spaces

Get list of profile spaces.

### URL Template

http://localhost/mxerest/profilespaces

### HTTP Method

GET

### XML Response Data Structure

Field	Description
profileSpaceList	Section containing list profile spaces
profileSpace	Profile space section
name	Name of profile space
directory	Profile directory path

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of profile spaces.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<profileSpaceList>
  <profileSpace>
    <name>test profiles</name>
    <directory>c:\profiles</directory>
  </profileSpace>
</profileSpaceList>
```

### Limitations

None



## Get Profile Summaries by Type and Profile Space

For a given profile space, get list of profile summaries by specified type. Returns a list of profile summaries <profileSummary> with links to the detailed profile.

### URL Template

http://localhost/mxerest/profilespaces/[profileSpace]/[profileType]

### HTTP Method

GET

### XML Response Data Structure

Field	Description
profileSummaryList	Section containing list profile summaries
profileSummary	Profile summary type - profile type (flash, job, etc.) href - link to profile detail

### Parameters

Parameter	Description	Expected Value	Status
profileSpace	Profile space name.		
profileType	Profile type.	flash, job, h264, etc.	

### Expected Results

List of profile summaries for given profile space and profile type.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<profileSummaryList>
  <profileSummary type="job" name="notif_admin"
href="http://localhost/mxerest/profilespaces/test+profiles/job/notif_admin" />
  <profileSummary type="job" name="YouTube"
href="http://localhost/mxerest/profilespaces/test+profiles/job/YouTube" />
</profileSummaryList>
```

### Limitations

None

## Get Detailed Profile by Type, Name, and Profile Space

Get detailed profile contents by specified profile space, type, and name.

### URL Template

`http://localhost/mxerest/profilespaces/[profileSpace]/[profileType]/[profile name]`

### HTTP Method

GET

### XML Response Data Structure

XML contents of the specified profile.

### Parameters

Parameter	Description	Expected Value	Status
profileSpace	Profile space name.		
profileType	Profile type.	flash, job, h264, etc.	
profileName	Profile name without extension.		

### Expected Results

XML contents of selected profile.

### XML Response

```
<profile type="job" name="conform" xmlns:axe="urn: anystream.com:ecs">
  <class>job</class>
  <enabled>yes</enabled>
  <parameters>
    <mpeg>conform.mpeg.awp</mpeg>
    <output>conform.out.awp</output>
    <preprocessor>conform.pp.awp</preprocessor>
  </parameters>
  <version>2.0</version>
</profile>
```

### Limitations

None

# Translate and Submit Job Commands

Commands to apply XSLT transforms and planners and submit jobs to MXE platform.

- [Translate Job XML Using Single Pass, page 3-60](#)
- [Translate Job XML Using Multiple Passes, page 3-61](#)
- [Submit Job, page 3-62](#)

## Translate Job XML Using Single Pass

Translate input XML non-recursively; apply only first matching XSLT transform or planner.

### URL Template

http://localhost/mxerest/translate/single

### HTTP Method

POST

### POST Data Structure

Determined by Job XML or planner-submit.

### Parameters

NA

### Expected Results

Transform of plan XML after a single pass.

### POST Data

Plan XML to be translated. Can be an MXE plan XML format (such as job XML or planner-submit) or custom XML

### Example:

```
<jobPackage>
  <submitParameterList>
    <replace name='test1'>Test1Text</replace>
    <replace name='test2'>Test2Text</replace>
  </submitParameterList>
  <jobList>
    <job>@{test1}</job>
    <job>@{test2}</job>
    <job>@{profile-dir}</job>
  </jobList>
</jobPackage>
```

### XML Response

Plan XML with first transform or planner applied.

### Limitations

None

## Translate Job XML Using Multiple Passes

Apply matching XSLT transforms and planners recursively until input XML is in a format suitable for submission to the ECS (jobPackage or job XML).

**URL Template**

http://localhost/mxerest/translate

http://localhost/mxerest/translate/multi

**HTTP Method**

POST

**XML Response Data Structure**

Determined by job XML.

**POST Data Structure**

Determined by Job XML or planner-submit.

**Parameters**

NA

**Expected Results**

Fully expanded Job XML.

**Limitations**

None

## Submit Job

Recursively apply matching XSLT transforms and planners and then submit job to MXE platform (i.e. ECS).

### URL Template

http://localhost/mxerest/submit

### HTTP Method

POST

### XML Response Data Structure

Field	Description
jobSubmitStatusList	Section containing status for each submitted job.
jobSubmissionStatus	Section containing job submission status.
name	Unique name for each job of the "Job Package".
jobID	Job ID <b>Note</b> "-1" indicates an error has occurred.
exceptionText	This is an optional tag containing the exception error message if an error occurred.

### POST Data Structure

Determined by Job XML.

### Parameters

NA

### Expected Results

Job is submitted to MXE and job ID is returned.

### POST Data

Plan XML to be submitted. This can be any MXE plan XML format, for example Job XML, planner-submit, or custom XML .

### XML Response

Job submission status containing list of submitted job IDs.

```
<?xml version="1.0" encoding="utf-8"?>
<jobSubmitStatusList>
  <jobSubmissionStatus name="myJobName">
    <jobID>270</jobID>
  </jobSubmissionStatus>
</jobSubmitStatusList>
```

### Limitations

None

# Miscellaneous Commands

- [Reload Configuration, page 3-64](#)
- [Get Version, page 3-65](#)
- [Get License List, page 3-66](#)
- [Get User-defined Metadata, page 3-67](#)
- [Start Live Capture, page 3-69](#)
- [Stop Live Capture, page 3-70](#)
- [Get Active IP Capture List, page 3-71](#)

## Reload Configuration

Reload MXE API configuration including XSLT transforms and planners

### URL Template

http://localhost/mxerest/reloadConfiguration

### HTTP Method

POST

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

API configuration is reloaded and success response is returned.

### POST Data

None

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>  
<status>success</status>
```

### Limitations

None



## Get Version

Get MXE platform version.

### URL Template

http://localhost/mxerest/Version

### HTTP Method

GET

### XML Response Data Structure

Field	Description
version	MXE platform version number.

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

Platform version number.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<version>3.2</version>
```

### Limitations

None

## Get License List

Get list of MXE licenses.

### URL Template

http://localhost/mxerest/Licenses

### HTTP Method

GET

### XML Response Data Structure

Field	Description
licenseList	Section containing list of licenses
license	License section name - name of feature list count - license count (always 1)
component	License component (feature) name - license tag limit - concurrency limit inUseCount - number of currently used (checked out) licenses

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of license features with concurrency limit and in use counts.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<licenseList>
<license name="enterprise" count="1" />
<license name="floating" count="1">
  <component name="1770AudioNorm" limit="10" inUseCount="0" />
  <component name="ac351" limit="10" inUseCount="0" />

  <component name="fileman" limit="10" inUseCount="0" />
  <component name="filemanAdv" limit="10" inUseCount="0" />
  <component name="h264" limit="10" inUseCount="0" />

  <component name="messenger" limit="10" inUseCount="0" />
  <component name=" microsoft" limit="10" inUseCount="0" />
</license>
</licenseList>
```

### Limitations

None

## Get User-defined Metadata

Get list of user-defined metadata (UDM).

### URL Template

http://localhost/mxerest/Udm

### HTTP Method

GET

### XML Response Data Structure

Field	Description
userDefinedMeta dataList	Section containing list of user defined meta data (UDM)
userDefinedMeta Data	User defined meta data item
id	UDM value
name	UDM name/key
type	UDM data type (Integer, String, Decimal, Enum)
description	UDM description
enum	Enumeration section (if type is Enum)
id	Enumeration ID
name	Enumeration name
value	Enumeration value
isDefault	If true, indicates default value for enumeration

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

NA

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<userDefinedMeta dataList>
<userDefinedMeta Data>
  <id>1</id>
  <name>cgms-enabled</name>
  <type>Integer</type>
  <description>Enable CGMS-A Override</description>
</userDefinedMeta Data>
<userDefinedMeta Data>
  <id>3</id>
  <name>Organization ID</name>
  <type>Enum</type>
  <description>Org ID</description>
  <enum>
```

```
<id>2</id>
<name>Org A</name>
<value>100</value>
<isDefault>True</isDefault>
</enum>
<enum>
<id>3</id>
<name>Org B</name>
<value>200</value>
<isDefault>False</isDefault>
</enum>
</userDefinedMetaData>
</userDefinedMetaDataList>
```

**Limitations**

None

## Start Live Capture

Start live capture.

### URL Template

http://localhost/mxerest/startCapture

### HTTP Method

POST

### POST Data Structure

Field	Description
start-capture	Command
host	Host name of node running capture prefilter task.
port	Port of capture prefilter task.

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Live capture is started and success response is returned.

### POST Data

```
<?xml version="1.0" encoding="utf-8"?>
<start-capture>
  <host>localhost</host>
  <port>5008</port>
</start-capture>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

None

## Stop Live Capture

Stop live capture.

### URL Template

http://localhost/mxerest/stopCapture

### HTTP Method

POST

### POST Data Structure

Field	Description
stop-capture	Command
host	Host name of node running capture prefilter task.
port	Port of capture prefilter task.

### XML Response Data Structure

Field	Description
status	Success.
exception	message = error message stacktrace = stack trace where exception occurred

### Parameters

NA

### Expected Results

Live capture is stopped and success response is returned.

### POST Data

```
<?xml version="1.0" encoding="utf-8"?>
<stop-capture>
  <host>localhost</host>
  <port>5008</port>
</stop-capture>
```

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<status>success</status>
```

### Limitations

None

## Get Active IP Capture List

Returns list of running IP capture host names.

### URL Template

http://localhost/mxerest/ipcaptures

### HTTP Method

GET

### XML Response Data Structure

Field	Description
ipCaptureList	Section containing list of IP captures.
ipCapture	Section containing IP capture details.
name	The IP capture name.
host	Name of the host running the IP capture prefilter.

### Parameters

Parameter	Description	Expected Value	Status
None			

### Expected Results

List of running IP captures.

### XML Response

```
<?xml version="1.0" encoding="utf-8"?>
<ipCaptureList>
  <ipCapture>
    <name>NIGHTLY NEWS FEED1</name>
    <host>DEVTESTLCS1</host>
  </ipCapture>
</ipCaptureList>
```

### Limitations

None

