

# MPF Workflow Manager REST API

Version 0.9 (build 20161213)

## NOTICE

*This software (or technical data) was produced for the U.S. Government under contract, and is subject to the Rights in Data-General Clause 52.227-14, Alt. IV (DEC 2007).*

*Copyright 2016 The MITRE Corporation. All Rights Reserved.*

# Introduction

REST-based web services for the MPF Workflow Manager

**Host:** hostname:8080

**Base Path:** /workflow-manager

## Table of contents

### 1. Definitions

- 1.1. EnvironmentVariableModel
- 1.2. SystemMessage
- 1.3. DeployedNodeManagerModel
- 1.4. JobCreationRequest
- 1.5. JobCreationMediaData
- 1.6. DeployedServiceModel
- 1.7. JobCreationResponse
- 1.8. ServiceModel
- 1.9. NodeManagerModel
- 1.10. SingleJobInfo
- 1.11. InfoModel
- 1.12. MpfResponse
- 1.13. AggregatePipelineStatsModel
- 1.14. AllJobsStatisticsModel

### 2. Paths

- 2.1. GET /rest/info
- 2.2. GET /rest/jobs/stats
- 2.3. POST /rest/jobs
- 2.4. GET /rest/jobs/{id}
- 2.5. POST /rest/jobs/{id}/cancel
- 2.6. GET /rest/jobs/{id}/output/detection
- 2.8. POST /rest/jobs/{id}/resubmit
- 2.9. GET /rest/nodes/config
- 2.9. POST /rest/nodes/config
- 2.10. GET /rest/nodes/hosts
- 2.11. GET /rest/nodes/info
- 2.12. GET /rest/nodes/services
- 2.13. POST /rest/nodes/services/{serviceName}/start
- 2.14. POST /rest/nodes/services/{serviceName}/stop

2.15. GET /rest/pipelines

2.16. GET /rest/system-message

2.17. POST /rest/system-message

2.18. GET /rest/system-message/type/{typeFilter}

2.19. DELETE /rest/system-message/{id}

## 1. Definitions

### 1.1. EnvironmentVariableModel

Name	Type	Description	Required
name	string	The name of the environment variable	Yes
value	string	The value of the environment variable	Yes
sep	string	The separator for values if wanting to append to an existing environment variable, for example, ":"	No

### 1.2. SystemMessage

Name	Type	Description	Required
datePosted	string	The message timestamp, include only if you want to change the timestamp	No
msg	string	The message	Yes
msgEnum	string	The enumerated name of predefined message	No
msgType	string	The type of message (intended target)	No
removeStrategy	string	The strategy for removing this message	No
severity	string	The severity of the message	No

### 1.3. DeployedNodeManagerModel (Response only)

Name	Type	Description
nodeModels	array	See <b>DeployedServiceModel</b> in the <b>Definitions</b> section.

### 1.4. JobCreationRequest

Name	Type	Description	Required
buildOutput	boolean	True to build the output objects; otherwise false (true by default)	No
externalId	string	External id	No
jobProperties	object	A key-value pair using strings which may be used to override the parameters associated with the pipeline.	No
media	array	See <b>JobCreationMediaData</b> in the <b>Definitions</b> section.	Yes
pipelineName	string	The name of the pipeline used to process the media URIs obtained from 'rest/pipelines' - example URIs: 'http://localhost/images/image.png' and 'file:///home/user/images/image.jpg'	Yes
priority	integer	Priority of the job, integer values 0 - 9 (0 being the lowest) are valid (default value is set in mpf.properties)	No

Name	Type	Description	Required
callbackURL	string	When the job completes, the callback will perform a GET or POST to the callbackURL with the parameters 'jobId' and 'externalId' of the completed job. For example, on a GET to a callbackURL: /api.example.com/foo?jobId=1&externalId=someid. Another example: /api.example.com/foo?someparam=something&jobId=1&externalId=someid. Example when no externalId is provided: /api.example.com/foo?jobId=1.	No
callbackMethod	string	A value of 'GET' or 'POST' to perform a HTTP callback	No

### 1.5. JobCreationMediaData

Name	Type	Description	Required
mediaUri	string	A valid (properly encoded) URI to a single media source for the job.	Yes
properties	object	A map of key-value pairs using strings which may be used to override the parameters associated with the pipeline or job properties for this medium.	No

### 1.6. DeployedServiceModel (Response only)

Name	Type	Description
kind	string	Simple (binary) or generic (jar)
lastKnownState	string	A string that will be one of the following values: Unknown, Configured (configured, but no other indication of presence), Launching (asked to start), Running, ShuttingDown (requested to shut down), ShuttingDownNoRestart, Inactive (inactive after a regular shutdown request), InactiveNoStart, Delete, DeleteInactive (inactive and removed from the config)
name	string	Name of the service
rank	integer	Service index
restartCount	integer	Number of times this service has restarted
serviceCount	integer	The amount of services
unlaunchable	boolean	True if the service has reached a condition where it is not launch able; otherwise false

### 1.7. JobCreationResponse (Response only)

Name	Type	Description
jobId	integer	The job id of the created job
mpfResponse		See <b>MpfResponse</b> in the <b>Definitions</b> section.

### 1.8. ServiceModel

Name	Type	Description	Required
args	array	Arugments for the process	No
cmd	string	The process command	Yes
environmentVariables	array	See <b>EnvironmentVariableModel</b> in the <b>Definitions</b> section.	No

Name	Type	Description	Required
serviceCount	integer	The number of services to start	Yes
serviceDescription	string	Description of the service	Yes
serviceLauncher	string	'simple' (binary) or 'generic' (jar)	Yes
serviceName	string	Name of the service	Yes
workingDirectory	string	Working directory for the service	Yes

### 1.9. NodeManagerModel

Name	Type	Description	Required
host	string	The hostname of the server running the node manager	Yes
services	array	See <b>ServiceModel</b> in the <b>Definitions</b> section.	Yes

### 1.10. SingleJobInfo (Response only)

Name	Type	Description
endDate	string	The date of job completion
jobId	integer	The id of the job
jobPriority	integer	The priority of the job, an integer value between 0 and 9 with 0 being the lowest priority
jobProgress	number	A float value from 0 to 100 representing the progress of the job
jobStatus	string	A string that will be one of the following values: UNKNOWN, INITIALIZED (the job has been initialized but not started), JOB_CREATION_ERROR (indicates that a job was received, but a job could not be created from the contents of the request), IN_PROGRESS, BUILDING_OUTPUT_OBJECT, COMPLETE, CANCELLING, CANCELLED, ERROR
outputObjectPath	string	The server file path to the detection output object
pipelineName	string	A pipeline string from /rest/pipelines/available
startDate	string	The date the job started
terminal	boolean	True if jobStatus is JOB_CREATION_ERROR, COMPLETE, CANCELLED, or ERROR; otherwise false

### 1.11. InfoModel (Response only)

Name	Type	Description
app	string	Application name (Media Processing Framework)
appFamily	string	Application family (Workflow Manager)
buildNum	string	Build number
gitBranch	string	git branch
gitHash	string	git hash
version	string	Version number ("x.x.x")

### 1.12. MpfResponse (Response only)

Name	Type	Description
message	string	Error message if responseCode is not 0

Name	Type	Description
responseCode	integer	0 if success, 1 if error

### 1.13. AggregatePipelineStatsModel (Response only)

Name	Type	Description
count	integer	The amount of jobs, in any job status, processed with the pipeline
maxTime	integer	The longest amount of time in milliseconds taken to complete a job used to build this model
minTime	integer	The shortest amount of time in milliseconds taken to complete a job used to build this model
states	object	A key-value pair of (string, integer) where the key is the job status and the value is the amount of jobs currently meeting that status
totalTime	integer	The total amount of time taken in milliseconds to complete processing of all jobs used to compile this model
validCount	integer	The amount of jobs with a job status of COMPLETE, which is the amount of total jobs used to compile the statistics in this model

### 1.14. AllJobsStatisticsModel (Response only)

Name	Type	Description
elapsedTimeMs	integer	Time in milliseconds it took to retrieve this model
aggregatePipelineStatsMap	object	A key-value pair of (string, AggregatePipelineStatsModel) where the key is a pipeline name
jobTypes	integer	Amount of different elements in the aggregatePipelineStatsMap, which is equal to the amount of different pipelines used by all of the processed jobs
totalJobs	integer	The total amount of jobs used to compile the statistics

## 2. Paths

### 2.1 GET /rest/info

Summary	Returns metadata about the Workflow Manager, such as version and build number	
Description	Note that some of this information is set only during official builds; if you get a '0' or 'unknown', that indicates that this was not an official build.	
Operation Id		
Produces	application/json	
Parameters	no parameters	
Tags	meta	
Responses	<b>code</b>	<b>description</b>
	200	Successful response See <b>InfoModel</b> in the <b>Definitions</b> section.
	401	Bad credentials

### 2.2 GET /rest/jobs/stats

Summary	Compiles the AllJobsStatisticsModel using all of the submitted jobs.	
Description		
Operation Id		
Produces	application/json	
Parameters	no parameters	
Tags	statistics	
Responses	<b>code</b>	<b>description</b>
	200	Successful response See <b>AllJobsStatisticsModel</b> in the <b>Definitions</b> section.
	401	Bad credentials

### 2.3 POST /rest/jobs

Summary	Creates and submits a job using a JSON JobCreationRequest object as the request body.	
Description	<p>The pipelineName should be one of the values in 'rest/pipelines'. The media array should contain objects with a valid mediaUri. For example: http://localhost/images/image.png. Another example: file:///home/user/images/image.jpg. A callbackURL (optional) and callbackMethod (GET or POST) may be added. When the job completes, the callback will perform a GET or POST to the callbackURL with the parameters 'jobId' and 'externalId' of the completed job. For example, on a GET to a callbackURL: /api.example.com/foo?jobId=1&amp;externalId=1. Another example: /api.example.com/foo?someparam=something&amp;jobId=1&amp;externalId=1. Example when no externalId is provided: /api.example.com/foo?jobId=1. The body of a POST callback will always include the 'jobId' and 'externalId', even if the latter is 'null'. An optional jobProperties object contains String key-value pairs which override the pipeline's job properties for this job.</p> <p>The body of a POST callback will be:</p> <pre>{   "jobId":1,   "externalId":"someid" }</pre>	
Operation Id		
Produces	application/json	



Consumes application/json

## Parameters

Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
jobCreationRequest	body	See <b>JobCreationRequest</b> in the <b>Definitions</b> section.	Yes						

Tags jobs

## Responses

code	description
201	Job created
	See <code>JobCreationResponse</code> in the <b>Definitions</b> section.
401	Bad credentials

## 2.4 GET /rest/jobs/{id}

<b>Summary</b>	Gets a SingleJobInfo model for the job with the id provided as a path variable.
----------------	---------------------------------------------------------------------------------

### Description

Operation Id

Produces	application/json
----------	------------------

### Parameters

Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
id	path	Job Id	Yes	long	int64				

Tags jobs

## Responses

code	description
200	Successful response See <b>SingleJobInfo</b> in the <b>Definitions</b> section.
400	Invalid job id, the response body will contain no content
401	Bad credentials

## 2.5 POST /rest/jobs/{id}/cancel

<b>Summary</b>	Cancels the job with the supplied job id.
----------------	-------------------------------------------

### Description

Operation Id

Produces	application/json
----------	------------------

### Parameters

Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
id	path	Job id	Yes	integer	int64				

Tags jobs

## Responses

code	description
200	Successful cancellation attempt See <b>MpfResponse</b> in the <b>Definitions</b> section.
400	Invalid job id, the response will still be an <b>MpfResponse</b> with more information
401	Bad credentials

## 2.6 GET /rest/jobs/{id}/output/detection

**Summary** Gets the JSON detection output object of a specific job using the job id as a required path variable.

Description										
Operation Id										
Produces	application/json									
Parameters	Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
	id	path	Job id	Yes	long	int64				
Tags	job									
Responses	code	description								
	200	Successful response JsonOutputObject example.								

```

{
  "externalJobId": "string",
  "jobId": 0,
  "media": [
    {
      "detectionProcessingErrors": {
        },
      "length": 0,
      "markupResult": {
        "id": 0,
        "message": "string",
        "path": "string",
        "status": "string"
      },
      "mediaId": 0,
      "message": "string",
      "mimeType": "string",
      "path": "string",
      "sha256": "string",
      "status": "string",
      "tracks": {} (DESCRIBED BELOW)
    }
  ],
  "objectId": "string",
  "pipeline": {
    "description": "string",
    "name": "string",
    "stages": [
      {
        "actionType": "string",
        "actions": [
          {
            "algorithm": "string",
            "description": "string",
            "name": "string",
            "properties": {} (A key-value pair using strings which may be used to override
              the default parameters associated with the underlying action)
          }
        ],
        "description": "string",
        "name": "string"
      }
    ]
  },
  "priority": 0,
  "siteId": "string",
  "status": "string",
  "timeStart": "string",
  "timeStop": "string"
}

```

#### Tracks example

When using a pipeline with face detection and markup actions

```

{
  "+#FACE DETECTION ACTION": [{
    "id": "e1b810ab07a9fc1459d088867356cf5b63f544d170f78181b86f97b8b78a1919",
    "startOffset": 0,
    "stopOffset": 1,
    "type": "FACE",
    "source": "+#FACE DETECTION ACTION",
    "exemplar": {
      "x": 114,
      "y": 208,
      "width": 285,
      "height": 285,
      "confidence": 7.9901013,
      "metadata": "",
      "offset": 0,
      "artifactExtractionStatus": "COMPLETED",
      "artifactPath": ""
    },
    "detections": [{
      "x": 114,
      "y": 208,
      "width": 285,
      "height": 285,
      "confidence": 7.9901013,
      "metadata": "",
      "offset": 0,
      "artifactExtractionStatus": "COMPLETED",
      "artifactPath": ""
    }]
  }],
  "+#FACE DETECTION ACTION#GENERIC MARKUP ACTION": []
},

```

400 Invalid job id, the response body will contain no content

401 Bad credentials

## 2.7 POST /rest/jobs/{id}/resubmit

**Summary** Resubmits the job with the provided job id. If the job priority parameter is not set the default value will be used.

**Description**

**Operation Id**

**Produces** application/json

**Parameters**

Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
id	path	Job id	Yes	long	int64				
jobPriority	query	Job priority (0-9 with 0 being the lowest) - OPTIONAL	No	integer	int32				

**Tags** jobs

**Responses**

code	description
200	Successful resubmission request. See <b>JobCreationResponse</b> in the <b>Definitions</b> section.
400	Invalid job id, the response will still be a <b>JobCreationResponse</b> with more information
401	Bad credentials

## 2.8 GET /rest/nodes/config

<b>Summary</b>	Retrieves the configuration information of all NodeManagers, including all services configured to run on each NodeManager		
<b>Description</b>			
<b>Operation Id</b>			
<b>Produces</b>	application/json		
<b>Parameters</b>	no parameters		
<b>Tags</b>	nodes		
<b>Responses</b>	<b>code</b>	<b>description</b>	
	200	Successful response	
		<b>Schema type</b>	array
		See <b>NodeManagerModel</b> in the <b>Definitions</b> section.	
	401	Bad credentials	

## 2.9 POST /rest/nodes/config

<b>Summary</b>	Save the NodeManager configuration using a JSON array of NodeManagerModel objects.									
<b>Description</b>	Each node manager in the cluster will be notified of the change(s) and attempt to conform to those changes.									
<b>Operation Id</b>										
<b>Produces</b>	application/json									
<b>Consumes</b>	application/json									
<b>Parameters</b>	<b>Name</b>	<b>In</b>	<b>Description</b>	<b>Required</b>	<b>Type</b>	<b>Format</b>	<b>Collection Format</b>	<b>Default</b>	<b>Min</b>	<b>Max</b>
	nodeManagerModels	body	all NodeManagerModel objects as a JSON array specifying all the configurations for all the nodes in the cluster	Yes						
<b>Tags</b>	nodes									
<b>Responses</b>	<b>code</b>	<b>description</b>								
	201	Config saved. See <b>MpfResponse</b> in the <b>Definitions</b> section.								
	401	Bad credentials								

## 2.10 GET /rest/nodes/hosts

<b>Summary</b>	Returns a collection of key-value pairs containing a hostname and its node configuration status. If the configuration status is 'true' that hostname is configured to run a node manager.		
<b>Description</b>	The response is a set of JSON key-value pairs, where the key is the hostname and the value is the configuration status.		
<b>Operation Id</b>			
<b>Produces</b>	application/json		
<b>Parameters</b>	no parameters		
<b>Tags</b>	nodes		
<b>Responses</b>	<b>code</b>	<b>description</b>	
	200	Successful response	
		<b>Schema type</b>	object

{ "host1": true, "host2": true, ..., "hostn": true }

401

Bad credentials

## 2.11 GET /rest/nodes/info

**Summary** Retrieves information about the currently configured node managers and the statuses of the services within each configured node manager.

**Description**

**Operation Id**

**Produces** application/json

**Parameters** no parameters

**Tags** nodes

Responses	code	description
	200	Successful response. See <b>DeployedNodeManagerModel</b> in the <b>Definitions</b> section.
	401	Bad credentials

## 2.12 GET /rest/nodes/services

**Summary** Retrieves services that are currently available for deployment to a node.

**Description** In the response, the key is the serviceName and the value is a ServiceModel object.

**Operation Id**

**Produces** application/json

**Parameters** no parameters

**Tags** nodes

Responses	code	description
	200	Successful response  <b>Schema type</b> array  See <b>ServiceModel</b> in the <b>Definitions</b> section.
	401	Bad credentials

## 2.13 POST /rest/nodes/services/{serviceName}/start

**Summary** Starts the service named serviceName

**Description** This method returns a NodeServiceStatusChangeResult and a HTTP 200 status code on success.

**Operation Id**

**Produces** application/json

**Parameters**

Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
serviceName	path	The fully qualified name of the service to be shutdown, (e.g., localhost.localdomain:markup:1)	Yes	string					

**Tags** nodes

Responses	code	description
	200	Successful response. See <b>MpfResponse</b> in the <b>Definitions</b> section.
	401	Bad credentials

## 2.14 POST /rest/nodes/services/{serviceName}/stop

**Summary** Shuts down the service named serviceName

**Description** This method returns a NodeServiceStatusChangeResult and a HTTP 200 status code on success.

**Operation Id**

**Produces** application/json

Parameters	Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
	serviceName	path	The fully qualified name of the service to be shutdown, (e.g., localhost.localdomain:markup:1)	Yes	string					

**Tags** nodes

Responses	code	description
	200	Successful response. See <b>MpfResponse</b> in the <b>Definitions</b> section.
	401	Bad credentials

## 2.15 GET /rest/pipelines

**Summary** Retrieves list of available pipelines.

**Description** The response will be a JSON array of strings with each item being a pipeline.

**Operation Id**

**Produces** application/json

**Parameters** no parameters

**Tags** pipelines

Responses	code	description
	200	Successful response <b>Schema type</b> array [ "pipeline1", "pipeline2", ..., "pipelineN" ]
	401	Bad credentials

## 2.16 GET /rest/system-message

**Summary** Retrieves all system messages.

**Description** Returns a JSON array of all System Messages.

**Operation Id**

**Produces** application/json

**Parameters** no parameters

**Tags** system-message

Responses	code	description
	200	Successful response <b>Schema type</b> array See <b>SystemMessage</b> in the <b>Definitions</b> section.
	401	Bad credentials

## 2.17 POST /rest/system-message

Summary	Adds a system message.									
Description	Adds a system message.									
Operation Id										
Produces	application/json									
Consumes	application/json									
Parameters	Name	In	Description	Required	Type	Format	Collection Format	Default	Min	Max
	msg	body	The message	Yes	string					
	msgType	query	The message type refers to the audience it is intended for (defaults to 'all'); valid values are: 'all', 'admin', and 'login'	No	string					
	removeStrategy	query	Strategy governing how a message is to be removed from the queue (defaultst to 'manual'); valid values are 'atServerStartup', and 'manual'	No	string					
	severity	query	Severity of the message (defaults to 'info'); valid values are 'info', 'warning' ,and 'error'	No	string					
Tags	system-message									
Responses	code	description								
	201	Successfully added. See <b>MpfResponse</b> in the <b>Definitions</b> section.								
	401	Bad credentials								

Responses	code	description
	200	Successfully deleted See <b>MpfResponse</b> in the <b>Definitions</b> section.
	401	Bad credentials