**Contacting BMC Software**

You can access the BMC Software website at [http://www.bmc.com](http://www.bmc.com). From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

**United States and Canada**

**Address**  
BMC SOFTWARE INC  
2103 CITYWEST BLVD  
HOUSTON TX  
77042-2827  
USA

**Telephone**  
▪ 713 918 8800  
▪ 800 841 2031

**Fax**  
713 918 8000

**Outside United States and Canada**

**Telephone**  
▪ (01) 713 918 8800

**Fax**  
▪ (01) 713 918 8000

© Copyright 1999-2017 BMC Software, Inc.

Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of BMC Software, Inc.

BMC, BMC Software, the BMC logo, the BMC Software logo, and other BMC marks, and the tagline “Bring IT to Life” are the exclusive properties of BMC Software, Inc., or its affiliates or subsidiaries and are registered or may be registered with the U.S. Patent and Trademark Office and in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

For BMC Control-M Products that are licensed on the “per CPU – Full Capacity” unit of measurement and installed in an Amazon Web Services (“AWS”) or Microsoft Azure (“Azure”) cloud environment, a license is required for the total number of CPUs in each AWS or Azure instance upon which the Product is installed or which the Product manages, either remotely or locally. For AWS, one CPU is equivalent to one vCPU, as defined by AWS. For Azure, one CPU is equivalent to up to four Virtual Cores (as defined by Azure), rounded up to the closest multiple of four.

**Server Endpoint Licensing**

All machines upon which any Control-M component is installed or upon which Control-M managed workload runs must be licensed. This includes Control-M Agent/Agentless platforms onto which one or more application plug-ins are installed but also includes Control-M Agent/Agentless platforms where no application plug-ins are installed. Control-M Agent/Agentless platforms on which jobs are ordered are counted regardless of whether those jobs execute or not. All Server Endpoints are counted, including development, staging, QA, pre-production, test, and production environments.
The licensing guidelines for application plug-ins are as follows:

- **ERP and BI/DI:** The application server(s) upon which Control-M managed processes are executed should be licensed in addition to the Control-M Agent machine(s) (in some cases, this may be the same machine).

- **Databases:** Each database server upon which Control-M managed database related processes are being executed should be counted in addition to the Control-M Agent machine(s).

- **AFT and MFT:** Only the machine(s) upon which the AFT or MFT plug-in is installed should be licensed.

- **Web Services, Java & Messaging:** only the Control-M Agent machine(s) upon which the plug-in is installed should be licensed.

- **Backup:** The Control-M Agent machine(s) where the Backup plug-in is installed and also the hosts which are running the backup server software should be licensed (note that this excludes the client machines for which the Backup Server software is managing actual backup processes except where a backup takes place of the backup server machine itself).

- **Cloud:** only the Control-M Agent machine(s) upon which the plug-in is installed should be licensed.

- **Hadoop:** All machines in each managed Hadoop Cluster should be licensed.

IBM® Tivoli® Business Service Manager, IBM Tivoli Workload Scheduler, IBM Cognos, IBM InfoSphere DataStage, IBM iSeries, IBM Websphere, and AIX® are the trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

UNIX® is the registered trademark of The Open Group in the US and other countries.

Linux is the registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

SAP® R/2 and SAP R/3, SAP Business Objects, and SAP NetWeaver are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

**Restricted rights legend**

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Field 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.

**Customer support**

You can obtain technical support by using the BMC Software Customer Support website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, see “Before contacting BMC.”

**Support website**

You can obtain technical support from BMC 24 hours a day, 7 days a week at (http://www.bmc.com/support). From this website, you can:
▪ Read overviews about support services and programs that BMC offers
▪ Find the most current information about BMC products
▪ Search a database for issues similar to yours and possible solutions
▪ Order or download product documentation
▪ Download products and maintenance
▪ Report an issue or ask a question
▪ Subscribe to receive proactive e-mail alerts when new product notices are released
▪ Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter **SupID:<yourSupportContractID>**, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC

Have the following information available so that Customer Support can begin working on your issue immediately:

▪ **Product information**
  • Product name
  • Product version (release number)
  • License number and password (trial or permanent)
▪ **Operating system and environment information**
  • Machine type
  • Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  • System hardware configuration
  • Serial numbers
  • Related software (database, application, and communication) including type, version, and service pack or maintenance level
▪ **Sequence of events leading to the issue**
▪ **Commands and options that you used**
▪ **Messages received (and the time and date that you received them)**
  • Product error messages
  • Messages from the operating system, such as `file system full`
  • Messages from related software
License key and password information

If you have questions about your license key or password, contact BMC as follows:

▪ (USA or Canada) Contact the Order Services Password Team at 800 841 2031, or send an e-mail message to ContractsPasswordAdministration@bmc.com.

▪ (Europe, the Middle East, and Africa) Fax your questions to EMEA Contracts Administration at +31 20 354 8702, or send an e-mail message to password@bmc.com.

▪ (Asia-Pacific) Contact your BMC sales representative or your local BMC office.

Third party Software

For the provisions described in the BMC License Agreement and Order related to third party products or technologies included in the BMC Product, see https://docs.bmc.com/docs/display/workloadautomation/Control-M+Workload+Automation+Documentation and click Third-party software (TPS).
## Contents

Introduction to Control-M Workload Change Manager......................................................... 7

Control-M Workload Change Manager installation .............................................................. 8
Installing Control-M Workload Change Manager on Windows ............................................. 8
Installing Control-M Workload Change Manager on UNIX .................................................. 9

Control-M Workload Change Manager setup in CCM ......................................................... 10
Folder authorizations ........................................................................................................... 10
Defining Control-M Workload Change Manager system parameters ................................. 13
Control-M Workload Change Manager system parameters ............................................... 14
Configuring secure communication between Control-M Workload Change Manager and a REST Service .......................................................... 15
Control-M Workload Change Manager advanced system parameters ............................... 16
Workload Change Manager integration with a change management system ....................... 17
Activating an external application for a Control-M Workload Change Manager request ........ 24

Control-M Workload Change Manager setup in the Control-M client ............................... 25
Site customizations management ....................................................................................... 25
Site standards management ............................................................................................... 27
Assigning a Site standard to a folder/folders ................................................................... 38
Exporting/importing Site Standards .................................................................................. 39
Introduction to Control-M Workload Change Manager

Control-M Workload Change Manager is a Control-M add-on, which enables you to do the following:

- In Control-M Workload Change Manager web application, application developers/analysts or the web users, can make changes to business job flows and submit them as requests to a Control-M scheduler or check them in to the Control-M Database. These change requests are related to your Control-M definitions in Control-M.

- In Control-M, a Control-M Administrator can create standards to assist schedulers and web users to define folders/job(s) according to your organization's standards.

A Control-M Administrator creates site standards, as described in Site standards management (on page 27), and then assigns the standards to folders, as described in Assigning a Site standard to a folder/folders (on page 38). This ensures that when web users, and schedulers define jobs, the job definitions comply with your organization's standards.

Using these standards, web users can create business job flows using a web interface. The web user can either submit the job flow as a request, or check it in to be part of the Control-M Database.

If the web user submits the job flow as a request, the Control-M scheduler receives these requests through Control-M. The requests appear in the Planning-Home page along with his other Workspaces and workflow. For more information on the request workspace, see Request Workspace. The Control-M scheduler works together with the web user, easily communicating through Notes, to review and approve business workflow requests. After the request is approved, the Control-M scheduler checks-in the request to be part of the Control-M database.

As a Control-M Administrator, you can do the following to set up Control-M Workload Change Manager:

- Install Control-M, as described in Control-M Workload Change Manager installation (on page 7).

- Configure Control-M Workload Change Manager system parameters and authorizations, as described in Control-M Workload Change Manager setup in CCM (on page 10).

- Define settings and standards, as described in Control-M Workload Change Manager setup in the Control-M client (on page 25).
Control-M Workload Change Manager installation

The following procedures describe how to install Control-M Workload Change Manager on a Control-M/EM server on UNIX and Windows, which enables you to define Control-M Workload Change Manager system parameters, define rules for web application users, and create requests and job flows from the Control-M Workload Change Manager site:

- Installing Control-M Workload Change Manager on Windows (on page 8)
- Installing Control-M Workload Change Manager on UNIX (on page 9)

Control-M Workload Change Manager is a Control-M/EM component, which utilizes Control-M/EM memory and CPU resources. Therefore, Control-M Workload Change Manager requires additional memory and CPU resources, depending on the number of concurrent Control-M Workload Change Manager users and the number of requests. In most cases after installation, Control-M Workload Change Manager requires the same amount of memory that is consumed by the current Control-M/EM GUI server.

Installing Control-M Workload Change Manager on Windows

This procedure describes how to install Control-M Workload Change Manager on a Control-M/EM server on Windows, which enables you to start working with Control-M Workload Change Manager.

Before you begin

- Ensure that the database server that contains the Control-M/EM database is up and running.
- Verify that the default web server port 18080, 8443 for HTTPS, or another configured port that you want to use is open and free for use.
- Download the Control-M Workload Change Manager installation files (see Product Distribution in the Control-M Release Notes).

➢ To install Control-M Workload Change Manager on Windows:

1. From the Control-M Workload Change Manager activation CD, double-click the setup.exe file.
   The Control-M Workload Change Manager Installation Wizard appears.
2. Do one of the following:
   - Interactive install: Follow the on-screen instructions until the installation is complete.
   - Automatic install: Create a parameter file and then run the automatic install in a non-interactive mode, as follows:
     a. Follow the on-screen instructions until the Summary window.
     b. Click Generate and select the location to create the XML parameter file.
Installing Control-M Workload Change Manager on UNIX

This procedure describes how to install Control-M Workload Change Manager on a Control-M/EM server on UNIX, which enables you to start working with Control-M Workload Change Manager.

Before you begin

- Ensure that the database server that contains the Control-M/EM database is up and running.
- Verify that the default web server port 18080, 8443 for HTTPS, or another configured port that you want to use is open and free for use.
- Download the Control-M Workload Change Manager installation files (see Product Distribution in the Control-M Release Notes).

To install Control-M Workload Change Manager on UNIX:

1. Mount the installation CD from the root user and log off the root user before continuing with the installation.
2. Log in to the Control-M/EM account.
   
   **NOTE:** If you are using the GUI, set the environment variable `DISPLAY` to value `<HostName>:0.0`. It is not necessary to set `DISPLAY` for console mode or silent mode. (For more information, see Setting environment variables in UNIX.)
3. Type the following command:
   
   `<cdPath>/setup.sh`
4. Follow the on-screen instructions until the installation is complete.
Control-M Workload Change Manager setup in CCM

In Control-M Configuration Manager you can do the following to setup Control-M Workload Change Manager:

- **Define folder authorizations** (on page 10): Enables you to grant users access to specific folders in Control-M and an authorization level for each folder. In addition, you can define different authorizations per job within a specific folder based on Application and Sub Application criteria.

- **Define Control-M Workload Change Manager system parameters** (on page 13): Enables you to set Control-M Workload Change Manager configurations.

- **Activate an external application for a Control-M Workload Change Manager request** (on page 24): Enables you to activate an external application when a Control-M Workload Change Manager request is submitted or rejected using a script.

**Folder authorizations**

Folder authorizations grant users access to specific folders in the Planning domain and an authorization level for each folder. In addition, you can define different authorizations per job within a specific folder based on Application and Sub Application criteria.

**EXAMPLE:** If you want to allow the user to update jobs that belong to Application **App1** but restrict this user from changing the Folder Properties, such as User Daily, Site Standards, and Business field values, define the fields, as follows:

- **Access Level=** **Browse**
- **Control-M, Library, and Folder fields=** *
- **Job Access Level=** **Update**
- **Application=** **App1**
## Field | Description
--- | ---
Access Level | Determines one of the following access levels for each user and group:
- **Browse**: Enables the user to view and refresh objects in Control-M
- **Update**: Enables the user to add and edit objects in Control-M
- **Full**: Enables the user to add, edit, and delete objects in Control-M

Control-M Server | Defines the name of the Control-M/Server (or Control-M for z/OS) that processes the job.

Library | Defines the name of the library that contains the job’s folder.

Folder | Defines the name of the folder. In the Properties pane, this parameter indicates the folder where the job belongs.

Job Level Authorization | Determines whether to enable authorizations on jobs in a specific folder based on Application and Sub Application criteria.

**NOTE:** You can only define job level authorizations if Control-M Workload Change Manager was installed (see Control-M Workload Change Manager installation (on page 7)). If Control-M Workload Change Manager was uninstalled, you can still define job level authorizations.

Job Access Level | Determines one of the following access levels for each user and group:
- **Browse**: Enables the user to view and refresh objects in Control-M
- **Update**: Enables the user to add and edit objects in Control-M
- **Full**: Enables the user to add, edit, and delete objects in Control-M

Application | Provides a logical name for sorting groups of jobs. This parameter is used to supply a common descriptive name to a set of related job groups.

Sub Application | Indicates the name of the Sub Application where the job belongs logically. It is a sub-category of the Application parameter. For example, the Application is Finances, and the Sub Application is Payroll.
The following procedures describe how to assign, edit, and delete folder authorizations:

- **Assigning a Folder authorization** (on page 12)
- **Editing a Folder authorization** (on page 12)
- **Deleting a Folder authorization** (on page 13)

### Assigning a Folder authorization

This procedure describes how to assign a Folder authorization for a Control-M/EM user or group, which enables you to grant users access to specific folders in Control-M and an authorization level for each folder.

➢ **To assign a Folder authorization:**

1. From the **Security** tab, in the **Security** group, click **Authorizations**. The Control-M/EM Authorizations window appears.

2. Do one of the following:
   - If you want to define a Folder authorization for a Control-M/EM user, select the **Users** tab and double-click the user that you want to apply an authorization.
   - If you want to define a Folder authorization for a Control-M/EM group, select the **Group** tab and double-click the group that you want to apply an authorization.

3. Select the **Folders** tab and click ![Folder Symbol].

   The **User Authorizations: Folders** dialog box appears.

4. For each field, type or select the required value, as described in **Folder authorizations** (on page 10) and then click **OK**.

   **NOTE:** You can use pattern matching strings and an * to denote all values, as described in **Pattern matching strings**.

   The folder authorization appears in the **User Authorization: <Control-M/EM user>** window.

### Editing a Folder authorization

This procedure describes how to edit a Folder authorization for a Control-M/EM user or group, which enables you to grant users access to specific folders in Control-M and an authorization level for each folder.

➢ **To edit a Folder authorization:**

1. From the **Security** tab, in the **Security** group, click **Authorizations**. The Control-M/EM Authorizations window appears.

2. Do one of the following:
   - If you want to edit a Folder authorization for a Control-M/EM user, select the **Users** tab and double-click the user that you want to edit.
   - If you want to edit a Folder authorization for a Control-M/EM group, select the **Group** tab and double-click the group that you want to edit.
3. Select the **Folders** tab and then select the Folder authorization that you want to edit.

4. Click ✂.
   
   The **User Authorizations: Folders** dialog box appears.

5. Edit the required fields, as described in **Folder authorizations** (on page 10) and then click **OK**.

   **NOTE:** You can use pattern matching strings and an * to denote all values, as described in Pattern matching strings.

   The updated Folder authorization appears in the **User Authorization: <Control-M/EM user>** window.

### Deleting a Folder authorization

This procedure describes how to delete a Folder authorization for a Control-M/EM user or group.

➢ **To delete a Folder authorization:**

1. From the **Security** tab, in the **Security** group, click **Authorizations**.
   
   The **Control-M/EM Authorizations** window appears.

2. Do one of the following:
   
   - If you want to delete a Folder authorization for a Control-M/EM user, select the **Users** tab and double-click the user contains the authorization that you want to delete.
   
   - If you want to delete a Folder authorization for a Control-M/EM group, select the **Group** tab and double-click the group contains the authorization that you want to delete.

3. Select the **Folders** tab and then select the Folder authorization that you want to delete.

4. Click ✗.
   
   A confirmation message appears.

5. Click **Yes**.

### Defining Control-M Workload Change Manager system parameters

This procedure describes how to define Control-M Workload Change Manager system parameters.

➢ **To define Control-M Workload Change Manager system parameters:**

1. From the **Components Tree** pane, select the **Control-M/EM** component and from the **Home** tab, in the **Definitions** group, click **System Parameters**.
   
   The **Control-M/EM System Parameters** dialog box appears.

2. In the left pane, click Control-M Workload Change Manager.

3. For each field, type the required value, as described in **Control-M Workload Change Manager system parameters** (on page 14) and Control-M Web parameters.

4. Click **Activate Changes**.
Control-M Workload Change Manager system parameters

The following table describes Control-M Workload Change Manager system parameters. To define these parameters, see Defining Control-M Workload Change Manager system parameters (on page 13).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mail Sender</td>
<td>Defines the sender email address that is used to send notifications to Control-M Workload Change Manager end users.</td>
</tr>
<tr>
<td>New Folder Default Strictness Level:</td>
<td>Determines whether to enforce validation on a new folder or not.</td>
</tr>
<tr>
<td>Enable Site Standard Admin Mode</td>
<td>Determines whether a Control-M Administrator can set the Enforce Validations checkbox for a Site Standard in Folders. If this parameter is selected, the checkbox is disabled for other users. You must also set the New Folder Default Strictness Level parameter to Strict.</td>
</tr>
<tr>
<td>Enable Change Management Integration</td>
<td>Enables the integration between Control-M Workload Change Manager and a third party change management system. If disabled, it does not affect the other Workload Change Manager system parameters.</td>
</tr>
<tr>
<td>REST Service URL</td>
<td>Defines a URL for the third party change management system REST service that is used to be integrated with Workload Change Manager. For more information see Workload Change Manager integration with a change management system (on page 17).</td>
</tr>
<tr>
<td></td>
<td>If you want to use an HTTPS URL, see Configuring secure communication between Control-M Workload Change Manager and a REST Service (on page 15).</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> To use an HTTPS URL the web server must be configured to work with HTTPS.</td>
</tr>
<tr>
<td>REST Service Timeout</td>
<td>Determines the time, in seconds, for the REST Service invocation to be considered not responsive when connecting Control-M Workload Change Manager to the REST service.</td>
</tr>
<tr>
<td>Change Management Status Button</td>
<td>Determines whether the Change Management Status button is visible or hidden in the Control-M application. If enabled, users can click the button to view the status of a request in the change management system.</td>
</tr>
<tr>
<td>Change Ticket Field Editable</td>
<td>Determines whether or not the Change Ticket field in Control-M Workload Change Manager and in Control-M can be edited.</td>
</tr>
</tbody>
</table>
Configuring secure communication between Control-M Workload Change Manager and a REST Service

This procedure describes how to configure a secure communication between Control-M Workload Change Manager and a change management system, which enables you to use an HTTPS URL when connecting through the REST Service.

➢ To configure an HTTPS REST Service URL to work with Control-M Workload Change Manager:

1. Run the following command to check if the certificate of your web server is already trusted by BMC Software:
   - **Windows**: keytool-list-keystore..\lib\security\cacerts
   - **UNIX**: keytool-list -keystore../lib/security/cacerts

2. If your web server certificate is not trusted by BMC Software, you need to import your own certificate from a trusted CA into the default keystore:
   - **Windows**: C:\Program Files\BMC Software\Control-M EM 9.0.00\Default\JRE\binkeytool-import-alias myca-keystore..\lib\security\cacerts-file C:\someca.cer
   - **UNIX**: cd ctm_em/JRE_64/binkeytool-import-alias myca-keystore../lib/security/cacerts-file someca.cer
     The default password is Changeit

3. If you are using BMC_JAVA_HOME you need to import the certificate using your own password.
Control-M Workload Change Manager advanced system parameters

The following table describes Control-M Workload Change Manager system parameters from the Control-M/EM system parameters Advanced window. For a description of all other Control-M Workload Change Manager system parameters, see Defining Control-M Workload Change Manager system parameters (on page 13).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumDaysKeepApprovedRequests</td>
<td>Determines the number of days to keep approved requests data in the Control-M/EM database.</td>
</tr>
<tr>
<td></td>
<td>Default: 14 days</td>
</tr>
<tr>
<td></td>
<td>Valid values: 0-28 days</td>
</tr>
<tr>
<td>SiteBackgroundColor</td>
<td>Determines the site background color.</td>
</tr>
<tr>
<td></td>
<td>Default: SolidBlue</td>
</tr>
<tr>
<td>SiteBackgroundColorValues</td>
<td>Determines the site background color values.</td>
</tr>
<tr>
<td></td>
<td>Default: SolidBlue, SolidGray, SolidBrown, SolidOrange, PatternBlue, PatternGray, PatternBrown, PatternOrange</td>
</tr>
<tr>
<td>WCMEmailSubjectTemplate</td>
<td>Defines the template for the e-mail’s subject field that is sent to the End users after the request state changes to one of the following:</td>
</tr>
<tr>
<td></td>
<td>▪ Returned</td>
</tr>
<tr>
<td></td>
<td>▪ Approved</td>
</tr>
<tr>
<td></td>
<td>Default: Your request &lt;REQUEST_NAME&gt; for Ticket ID &lt;CHANGE_ID&gt; has been &lt;ACTION&gt;</td>
</tr>
<tr>
<td>WCMCallbackResponseTimeout</td>
<td>Determines the maximum number of seconds for a job flow to check-in or synchronize before connection timeout. The following error message is displayed:</td>
</tr>
<tr>
<td></td>
<td>Connection timeout, please check your request status in the Home page.</td>
</tr>
<tr>
<td></td>
<td>Default: 60 seconds</td>
</tr>
</tbody>
</table>
Workload Change Manager integration with a change management system

This feature enables you to integrate Control-M Workload Change Manager with a REST Service that connects to a third party change management system. This allows you to do the following:

- Verify the change ticket status when a Control-M Workload Change Manager request is submitted, approved, returned, rejected or deleted.
- Create a Change Ticket ID automatically through the change management system upon a Workload Change Manager web application request submission, which is then assigned to the submitted request. If the request Change Ticket ID is empty, a new Change Ticket ID is created. If a request Change Ticket ID is defined, the REST Service verifies the Change Ticket ID in the change management system.

To enable this feature, you must define the required system parameters, as described in Control-M Workload Change Manager system parameters (on page 14).

You can create a REST Service that contains a set of REST API parameters that are applied when a request status is changed and/or when a request is submitted. You need to create your own REST Service that is used to integrate Control-M Workload Change Manager with your change management system. The REST Service API has the following calls:

- The two phases of the Request status update for all statuses of a Request. The REST Service automatically calls these two phases during any change of the Request status. The REST Service checks approval to proceed with the status change, and if the status change is not approved by the change management system a return message is displayed to the user.
  - validateChangeState
  - stateChanged

- You can ignore certain request states and just return HTTP RESPONSE 200.
  - getChangeStatus

For more information on the REST Service API parameters see, REST Service API parameters (on page 18). For REST Service API examples, see REST Service API examples (on page 20).

BMC Software provides an example REST Service for integrating Control-M Workload Change Manager with Remedy. For more information, see Integrating Remedy with Control-M Workload Change Manager (on page 22).

**Note:** The Remedy example provides two actions; create and verify a Change Ticket ID and verifying approval of a request when it is submitted for approval. You can add more properties for the different Control-M Workload Change Manager request statuses; such as returned, rejected, etc. You can modify and add properties that suit your organizations’ needs. You can also use the Remedy example as a basis to create your own Rest Service for a different change management system.

The REST Service validates the change ticket status, and if approved the change ticket status is changed to the desired state. If the request was not approved, a scheduler/web user will not be able to continue with the action. By implementing the REST Service API, you are able to verify the change ticket status automatically, prior to request approval, and get notified if the change ticket is not in the desired state. A Control-M Scheduler cannot check in the request if it is not approved by the change management system.
Messages are displayed to the scheduler/web user indicating the returned status of the request in the change management system. If enabled, a **Change Management Status** button is displayed in the ribbon that can be used to check the status of the request in the change management system. If the action was not approved by the change management system, you need to take appropriate corrective actions to resolve the issue by contacting the relevant individual in your organization.

**REST Service API parameters**

The following table describes the REST Service API parameters that you can use and define when you set up your own REST Service to integrate Control-M Workload Change Manager with a change management system:

**Request parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeID</td>
<td>Determines the Change Ticket ID that is assigned to the request. If a Request does not contain a Change Ticket ID, the REST Service retrieves one from the change management system and assigns it to the request.</td>
</tr>
<tr>
<td>ctmRequestID</td>
<td>Determines the Control-M Workload Change Manager Request ID.</td>
</tr>
<tr>
<td>Name</td>
<td>Determines the name of the Request.</td>
</tr>
<tr>
<td>newState</td>
<td>Specifies the new status of the Request.</td>
</tr>
<tr>
<td>OldState</td>
<td>Specifies the old status of the Request.</td>
</tr>
<tr>
<td>creationTime</td>
<td>Determines the creation time of the Request. UTC time in the following format: YYYYMMDDhhmmssUTC</td>
</tr>
<tr>
<td>lastTransferSideTimestamp</td>
<td>Determines the last transfer side timestamp. UTC time in the following format: YYYYMMDDhhmmssUTC</td>
</tr>
<tr>
<td>endUser</td>
<td>Specifies the associated end user of the Request.</td>
</tr>
<tr>
<td>scheduler</td>
<td>Specifies the associated scheduler of the Request.</td>
</tr>
</tbody>
</table>
Response parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>Defines the returned message that is displayed to the user.</td>
</tr>
<tr>
<td>changeID</td>
<td>Determines the Change Ticket ID that is updated in Control-M.</td>
</tr>
</tbody>
</table>

REST Service API endpoints:

<table>
<thead>
<tr>
<th>URL</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST /WCMRemedyIntegration/validat eChangeState</td>
<td>updateStateInformationStatus</td>
</tr>
</tbody>
</table>

For examples, see REST Service API examples (on page 20).
REST Service API examples

The following are examples of a REST Service API that is used to integrate Control-M Workload Change Manager with a change management system:

**Example 1**: Control-M calls the REST Service before updating the status. The change management system checks the update details and if it approves the status change, the content of request is then updated, such as the Change Ticket ID. If the change status is rejected a message is displayed to the user:

<table>
<thead>
<tr>
<th>Request Body examples</th>
<th>HTTP Code</th>
<th>Request Response examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;changeID&quot;: &quot;REM1&quot; &quot;ctmRequestID&quot;: &quot;22&quot; &quot;name&quot;: &quot;Testing request&quot; &quot;newState&quot;: &quot;Approved&quot; &quot;oldState&quot;: &quot;SchedulerWork&quot; &quot;creationTime&quot;: &quot;20151027123500&quot; &quot;endUser&quot;: &quot;user1&quot; &quot;scheduler&quot;: &quot;scheduler1&quot;</td>
<td>400</td>
<td>&quot;message&quot;: &quot;REM1 change management is not APPROVED&quot;</td>
</tr>
<tr>
<td>&quot;name&quot;: &quot;Testing request&quot; &quot;ctmRequestID&quot;: &quot;22&quot; &quot;newState&quot;: &quot;Submitted&quot; &quot;creationTime&quot;: &quot;20151027123500&quot; &quot;endUser&quot;: &quot;user1&quot; &quot;scheduler&quot;: &quot;scheduler1&quot;</td>
<td>200</td>
<td>&quot;changeID&quot;: &quot;REM222&quot;</td>
</tr>
<tr>
<td>&quot;changeID&quot;: &quot;REM1&quot; &quot;ctmRequestID&quot;: &quot;22&quot; &quot;name&quot;: &quot;Testing request&quot; &quot;newState&quot;: &quot;Submitted creationTime&quot;: &quot;20151027123500&quot; &quot;endUser&quot;: &quot;user1&quot; &quot;scheduler&quot;: &quot;scheduler1&quot;</td>
<td>200</td>
<td>&quot;changeID&quot;: &quot;REM222&quot;</td>
</tr>
</tbody>
</table>
Example 2: Control-M calls this REST Service after the actual status update so that the change management system commits/aborts the update log message. Then it returns a status and a message for the commit/abort action:

<table>
<thead>
<tr>
<th>Request Body examples</th>
<th>HTTP Code</th>
<th>Request Response examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;changeID&quot;: &quot;REM1&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;ctmRequestID&quot;: &quot;22&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;name&quot;: &quot;Testing request&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;newState&quot;: &quot;Approved&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;oldState&quot;: &quot;SchedulerWork&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;creationTime&quot;: &quot;20151027123500&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;endUser&quot;: &quot;user2&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;schedule&quot;: &quot;scheduler2&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>

Example 3: Control-M calls this REST Service to obtain the change status from the change management system:

<table>
<thead>
<tr>
<th>Request Body examples</th>
<th>HTTP Code</th>
<th>Request Response examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;ChangeID&quot;: &quot;REM1&quot;</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>&quot;CtmRequestID&quot;: &quot;em1\22&quot;</td>
<td></td>
<td>&quot;message&quot;: &quot;The change is approved&quot;</td>
</tr>
<tr>
<td></td>
<td>400</td>
<td>&quot;message&quot;: &quot;The ticket is not approved&quot;</td>
</tr>
<tr>
<td>&quot;ChangeID&quot;: &quot;REM1&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;CtmRequestID&quot;: &quot;em1\22&quot;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Integrating Remedy with Control-M Workload Change Manager

This procedure describes how to integrate Remedy with Control-M Workload Change Manager through a REST Service. This procedure contains example code which can be modified to fit your specific criteria.

Before you begin:

- Ensure that you have the following installed:
  - Java JDK 1.8
  - Apache MAEVEN build utility

➢ To integrate Remedy with Control-M Workload Change Manager:

1. Navigate to the following directory:
   - Windows: `<em_InstallFolder>\WCMRemedyIntegration\src\main\webapp\WEB-INF\classes`
   - UNIX: `<em_InstallFolder>/WCMRemedyIntegration/src/main/webapp/WEB-INF/classes`

2. Open the `RemedyIntegration.properties` file to configure the settings of your Remedy properties, as described in Remedy integration file properties (on page 23).

3. In the same directory, open the `NewTicketFields.properties` file to configure the field ID values that are used to create a Change Ticket ID. You can modify the values to reflect your Remedy environment values.

   Note: The values of the fields can contain values for the REST Service request such as: Control-M Request `${name}` by `${endUser}` performed by `${scheduler}`.

4. Open a command line and navigate to:
   `<em_InstallFolder>/WCMRemedyIntegration`

5. Enter the following commands to create a WAR package:
   - Windows:
     a. `SET JAVA_HOME=`
     b. `SET M2_HOME=`
     c. `SET PATH=%JAVA_HOME%\bin;%M2_HOME%\bin;%PATH%`
     d. `mvn package`
   - UNIX:
     e. `setenv JAVA_HOME`
     f. `setenv M2_HOME`
     g. `set path = ($JAVA_HOME/bin $M2_HOME/bin $PATH)`
     h. `mvn package`

   The MAEVEN build utility creates a WAR and JAR file in the target directory.

6. Deploy the WAR file to a Web Server. You can use a Control-M/EM Web Server by copying the WAR file into the following directory:
- **Windows**: `<em_installFolder>\emweb\tomcat\webapps`
- **UNIX**: `<em_installFolder>/etc/emweb/tomcat/webapps`

Remedy integration file properties

The following table describes the RemedyIntegration.properties file definitions that are used to configure your Remedy environment settings in order to integrate it with Control-M Workload Change Manager:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServerName</td>
<td>Defines the name of your Remedy server.</td>
</tr>
<tr>
<td>UserName</td>
<td>Defines the user name in Remedy.</td>
</tr>
<tr>
<td>IsUserPasswordEncrypted</td>
<td>Determines whether the user password is encrypted or not. If you want to encrypt the user password, do the following:</td>
</tr>
<tr>
<td></td>
<td>1. Complete Integrating Remedy with Control-M Workload Change Manager (on page 22) and build the MAEVEN package.</td>
</tr>
<tr>
<td></td>
<td>2. Run the following command:</td>
</tr>
<tr>
<td></td>
<td>* java -cp target\WCMRemedyIntegration-9.jar \com.bmc.ctmem.wcminteg.changemanagement.AESCrypt \abcd1234*</td>
</tr>
<tr>
<td></td>
<td>3. Update the encrypted password in the UserPassword property</td>
</tr>
<tr>
<td></td>
<td>4. Build the MAEVEN package again.</td>
</tr>
<tr>
<td>Valid Values:</td>
<td></td>
</tr>
<tr>
<td>▪ yes</td>
<td></td>
</tr>
<tr>
<td>▪ no</td>
<td></td>
</tr>
<tr>
<td>Default: no</td>
<td></td>
</tr>
<tr>
<td>UserPassword</td>
<td>Defines the password of the Remedy user. If you have created an encrypted password, you can define the password in this property.</td>
</tr>
<tr>
<td>CreateForm</td>
<td>Creates a new change management entry in the Remedy form.</td>
</tr>
<tr>
<td>ModifyForm</td>
<td>Determines an existing change management entry to modify in the Remedy form.</td>
</tr>
<tr>
<td>ChangeIdFieldId</td>
<td>Specifies the field ID number that contains the Change Ticket ID in Remedy. If the Change Ticket ID is not defined in the submitted request, it is created automatically and is then assigned to the request. If a Change Ticket ID is defined, then it is verified in Remedy.</td>
</tr>
</tbody>
</table>
### Property Table

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TicketApprovedFieldId</td>
<td>Specifies the field ID number that contains the status of the request.</td>
</tr>
<tr>
<td>TicketApprovedFieldValue</td>
<td>Specifies the field value which determines if the status is approved.</td>
</tr>
</tbody>
</table>

**Note:** The Remedy example provides two actions; create and verify a Change Ticket ID and verifying approval of a request when it is submitted for approval. You can add more properties for the different Control-M Workload Change Manager request statuses; such as returned, rejected, etc. You can modify and add properties that suit your organizations’ needs.

### Activating an external application for a Control-M Workload Change Manager request

This procedure describes how to activate an external application when a Control-M Workload Change Manager request is submitted, approved, rejected or deleted. This enables administrators to define a script that runs when a request changes states.

- **To activate an external application for a Control-M Workload Change Manager request:**

1. Prepare a script using the following input parameters, as follows:
   - Request name
   - Request description
   - New request state
   - Old request state
   - Creation Time (UTC time in the following format: YYYYMMDDhhmmssUTC)
   - LastTransferSideTimestamp (UTC time in the following format: YYYYMMDDhhmmssUTC)
   - Change ticket ID
   - Associated End user
   - Associated Scheduler

2. From the CCM, define the full path to the script in the `SendRequestToScript` system parameter, as described in Defining Control-M/EM system parameters.
Control-M Workload Change Manager setup in the Control-M client

In Control-M, you can do the following to setup Control-M Workload Change Manager:

- **Create Site customizations** (on page 25): Enables you to simplify job properties and restrict creation of job types for Control-M Workload Change Manager web users.
- **Create Site standards** (on page 27): Enables you to define settings that ensure Control-M Workload Change Manager web users and Control-M schedulers follow your organization's standards.
- **Assign a Site standard to a folder/folders** (on page 38): Enables you to assign a site standard to new or existing folder or multiple folders.

Site customizations management

The Site customizations tool enables you to do the following:

- **Simplify job properties**: Define which fields, a Control-M Workload Change Manager web user, can access when creating or editing jobs. You can hide or show entire tabs of the job properties pane such as Scheduling, Prerequisites, and Actions. You can also select specific options in these tabs to hide or show.
- **Restrict creation of job types**: Restrict which job types a web user can create in a job flow. For example, if you do not want web users to modify job definitions for job type File Transfer, you can disable it in Site customizations. The web user cannot create this form of job.

You can create several Site customizations, however only one can be set as a default Site customization that is used by all web users. If you choose to have several Site Customizations, for different web users, you will need to provide the web user with the Site customization that is relevant to them.

The following procedures describe how to create, edit, duplicate, and delete a Site customization:

- **Creating a Site customization** (on page 26)
- **Editing a Site customization** (on page 26)
- **Duplicating a Site customization** (on page 27)
- **Deleting a Site customization** (on page 27)

For more information on job properties and job types, see Job definition.

You must have full privileges to create Site customizations, as described in Privileges.
Creating a Site customization

This procedure describes how to create a Site customization, which enables you to simplify job properties, and restrict creation of job types for a Control-M Workload Change Manager web user.

➢ To create a Site customization:

1. From the **Tools** domain, in the **Planning** area, select **Site customizations**.
   
   The Site customizations tab appears.

2. From the **Site customizations** group in the ribbon, click ![New](image)
   
   The Site customization pane appears on the right.

3. In the **Name** field, type a name for the new Site customization.

4. If you want to set the Site customization as the default, click ![Default](image)

5. Do one or both of the following:
   
   - To display the list of options and tabs, click **Job properties**, and uncheck the options you want to hide.
   
   - To display the list of job types, click **Job types**, and uncheck the job types you want to prevent web users from creating.

6. Click ![Save](image)

   The Site customization is created.

Editing a Site customization

This procedure describes how to edit a Site customization, which enables you to modify the details of the Site customization. There might be active job flows using the site customization you want to edit. The changes take effect after a web user logs into Control-M Workload Change Manager.

➢ To edit a Site customization:

1. From the **Tools** domain, in the **Planning** area, select **Site customizations**.
   
   The Site customizations tab appears.

2. Select the Site customization you want to edit.
   
   The Site customization pane appears on the right.

3. Edit the details of the site customization according to your needs. For more information, see Creating a Site customization (on page 26).

4. Click ![Save](image)

   The Site customization is modified with the new changes.

26
Duplicating a Site customization

This procedure describes how to duplicate a Site customization, which enables you to make a copy of an existing Site customization and edit its contents and save it as a new Site customization.

➢ To duplicate a Site customization:

1. From the Tools domain, in the Planning area, select Site customizations.
   The Site customizations tab appears.

2. Select the Site customization you want to duplicate.

3. From the Site customizations group, click Duplicate.
   The duplicated Site customization is created, and appears on the right pane. You can change the name and definitions of the Site customization, as described in Creating a Site customization (on page 26).

Deleting a Site customization

This procedure describes how to delete a Site customization. There might be active job flows using the Site customization you want to delete.

➢ To delete a Site customization:

1. From the Tools domain, in the Planning area, select Site customizations.
   The Site customizations tab appears.

2. Select the Site customization you want to delete.

3. From the Site customizations group, click Delete.
   The Site customization is deleted.

Site standards management

The Site standard tool enables you to define settings that ensure Control-M Workload Change Manager web users and Control-M schedulers follow your organization's standards.

These standards are applied at the folder level, to new and existing folders, as described in Folder management. All jobs, new or old, have to comply with the Site standard rules. You can create multiple Site standards, and assign different Site standards to different folders.

You can set rules such as character length for a specific field, as described in Site standards rules editor parameters (on page 34).

The Site standard tool consists of the following parts:
Control-M Workload Change Manager Setup Guide

- **Folder/job parameters** (on page 28): Defines rules and restrictions for specific Control-M folder/job parameters.
- **Business parameters** (on page 30): Defines parameters specific to your business, that carry a special business meaning.
- **Internal rules** (on page 31): Defines rules that can be reused as rules in Control-M folder/job parameters and business parameters.
- **Condition format**: Defines a default condition name format that is used by the Workload Change Manager web user, and the Control-M scheduler.

The Site standard rules are validated, and errors and warnings appear to the web user upon submission of the job flow as request/checking in the job flow, and to the scheduler upon check-in. The handling of these warnings/errors depends on the enforcement policy set at the folder level. For more information, see Enforcing validations.

The following procedures describe how to create, edit, duplicate, and delete a Site standard:

- **Creating a Site standard** (on page 32)
- **Editing a Site standard** (on page 37)
- **Duplicating a Site standard** (on page 38)
- **Deleting a Site standard** (on page 38)

You must have full privileges to create Site standards, as described in Privileges.

**Folder/job parameters**

In the Site standard tool, you can specify restrictions and rules for Control-M folder/job parameters. The rules you define override the Control-M rules. However, you can only choose to alter the rules supported by Control-M, but not add new ones. For example, if a parameter can have from 1 - 64 characters, you can choose to set the length between 1 - 64, but not more than 64. This means, you can set the length to be only 50 characters, but not 65.

You can set a default value, or possible values, or allow certain characters, as described in **Site standards rules editor parameters** (on page 34).

You can also use **Business parameters** (on page 30) and **Internal rules** (on page 31) as pattern parts of the Control-M folder/job parameter.
EXAMPLE: Folder Name in your organization can only be from 1-10 characters, cannot contain /!*!, and only applies to OS job type.

Name: Job Name
Applied On: OS
Default Value:

Possible Values/Pattern Parts:
Possible Values

Length:
Minimum Length: 1
Maximum Length: 10

Characters:
Any character is allowed
Except for these:

After this rule is defined in the Site standard, and the Site standard is assigned to the folder, the Folder Name parameter rule is applied. The Control-M Workload Change Manager web user and the Control-M scheduler modifying the folder with that Site standard will have to comply with the Folder Name rule you defined. If they do not comply with the rule, a warning/error appears. They can ignore or fix the warning/error depending on the enforcement policy you set, as described in Enforcing validations.

For more information on how to define Control-M folder/job parameters, see Creating a Site standard (on page 32).
Control-M Workload Change Manager Setup Guide

Business parameters

In addition to the Control-M folder/job parameters rules and restrictions, you can define additional parameters that carry a specific meaning to your organization.

EXAMPLE: In your organization, you define your Job Name with a prefix of department ID. Your Job Name consists of business parameter [DepID] followed by free text.

You can create a Business parameter named Department ID and apply the desired rules and restrictions, like possible values, pattern parts, length, etc, as described in Site standards rules editor parameters (on page 34). After creating the business parameter, you can use it in a Control-M folder/job parameter, in this case in Job Name. You can also reuse other business parameters you defined, or Internal rules (on page 31).

After this rule is defined in the Site standard, and the Site standard is assigned to the folder, the Job Name parameter rule is applied. The Control-M Workload Change Manager web user and the Control-M scheduler modifying the folder with that Site standard will have to comply with the Job Name rule you defined. If they do not comply with the rule, a warning/error appears. They can ignore or fix the warning/error depending on the enforcement policy you set, as described in Enforcing validations. For more information on how to define Business parameters, see Creating a Site standard (on page 32).
Internal rules

In the Site standard tool, you can create internal rules that can be used in Folder/job parameters (on page 28) and Business parameters (on page 30). Internal rules are recurrent rules that you use in Control-M folder/job parameters and Business parameters.

Instead of creating this rule every time you create a folder/job parameter rule, you can create an internal rule and specify the invalid characters, as described in Site standards rules editor parameters (on page 34). Now, you can reuse the internal rule in each folder/job parameter rule you create.

**EXAMPLE:** All folder/job parameters cannot contain the characters !*/.

- Job name cannot contain characters !*/.
- Application cannot contain !*/.
- File name cannot contain !*/.

![Internal rule example](image)

After this rule is defined in the Site standard, and the Site standard is assigned to the folder, the rule is applied. The Control-M Workload Change Manager web user and the Control-M scheduler modifying the folder with that Site standard will have to comply with the rule you defined. If they do not comply with the rule, a warning/error appears. They can ignore or fix the warning/error depending on the enforcement policy you set, as described in Enforcing validations.

For more information on how to define internal rules, see Creating a Site standard (on page 32).
Creating a Site standard

This procedure describes how to create a Site standard, which enables you to set rules and regulations to Control-M folder/job parameters, Business parameters, and internal rules.

➢ To create a Site standard:
1. From the Tools domain, in the Planning area, select Site standards.
   The Site standards tab appears
2. From the Edit group in the ribbon, click .
   The Site standard appears.
3. In the Name field, type a name for the new Site standard.
4. Do one or all of the following:
   • Create Folder/job parameters (on page 28) rules, as described in Creating folder/job parameter rules (on page 32).
   • Create Business parameters (on page 30), as described in Creating business parameters (on page 33).
   • Create Internal rules (on page 31), as described in Creating internal rules (on page 33).
5. Click .
   The Site standard is created.

Creating folder/job parameter rules

This procedure describes how to create a folder/job parameter rule, which enables you set rules and restrictions to specific Control-M folder/job parameters.

Before you begin

Ensure that you have completed Creating a Site standard (on page 32).

➢ To create folder/job parameter rules:
1. In the Site standard, in the Folder/job parameters section, click .
   The rules editor pane appears on the right.
2. From the Name drop-down list, select the Control-M parameter.
3. Define the rules, as described in Site standards rules editor parameters (on page 34).
4. Click .
   The folder/job parameter rules are created.
Creating business parameters

This procedure describes how to create a Business parameter rule.

For more information on Business parameters functionality, see Business parameters (on page 30).

Before you begin

Ensure that you have completed Creating a Site standard (on page 32).

➢ To create business parameters:

1. In the Site standard, click Advanced.

2. In the Business parameters section, click +.
   The rules editor pane appears on the right.

3. In the Name field, type a name for the Business parameter. Select a name that has a business value to your organization.

4. Define the rules, as described in Site standards rules editor parameters (on page 34).

5. Click .
   The business parameter rules are created.

Creating internal rules

This procedure describes how to create internal rules.

For more information on internal rules functionality, see Internal rules (on page 31).

Before you begin

Ensure that you have completed Creating a Site standard (on page 32).

➢ To create internal rules:

1. In the Site standard, click Advanced.

2. In the Internal rules section, click +.
   The rules editor pane appears on the right.

3. In the Name field, type a name for the internal rule.

4. Define the rules, as described in Site standards rules editor parameters (on page 34).

5. Click .
   The internal rule is created.
## Site standards rules editor parameters

The following table lists the Site standards rules editor parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Applied to</strong></td>
<td>Applies job parameter rules to specific job types in the same Site Standard.</td>
</tr>
<tr>
<td><strong>EXAMPLE:</strong></td>
<td>Job Name applied to OS job type</td>
</tr>
<tr>
<td><strong>Default Value</strong></td>
<td>Defines a default value for a folder/job parameter.</td>
</tr>
<tr>
<td><strong>Required</strong></td>
<td>Indicates that the parameter is mandatory.</td>
</tr>
<tr>
<td><strong>Possible values</strong></td>
<td>Defines more than one value that a Control-M Workload Change Manager web user and Control-M scheduler can use. Any one of these options is valid. You cannot define pattern parts if you define possible values. If you define other rules like length, or character restrictions, the possible values have to comply with these rules.</td>
</tr>
<tr>
<td><strong>Pattern Parts</strong></td>
<td>Indicates that a parameter is composed of a pattern.</td>
</tr>
<tr>
<td><strong>Business parameters (on page 30)</strong></td>
<td>Indicates that a business parameter can be used as a pattern part of a Control-M folder/job parameter.</td>
</tr>
<tr>
<td><strong>EXAMPLE:</strong></td>
<td>Job Name: [Business Parameter][*]</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Folder/job parameters</td>
<td>Indicates that a folder/job parameter rule can be used as a pattern part of another folder/job parameter rule.</td>
</tr>
<tr>
<td></td>
<td><strong>EXAMPLE:</strong> Job Name:[Application]</td>
</tr>
<tr>
<td>Internal rules</td>
<td>Indicates that an internal rule can be used as a pattern part of folder/job parameter and a business parameter.</td>
</tr>
<tr>
<td></td>
<td><strong>EXAMPLE:</strong> Job Name:[Internal Rule][1-9 characters][*]</td>
</tr>
<tr>
<td>Fixed text</td>
<td>Indicates that a fixed value is used in the parameter as a constant string.</td>
</tr>
<tr>
<td>Free text</td>
<td>Indicates that any value can be used in the parameter.</td>
</tr>
<tr>
<td><strong>Parameter</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Length</td>
<td>Determines the minimum and maximum length of characters in a parameter.</td>
</tr>
<tr>
<td>Characters</td>
<td>Determines character restrictions, such as which characters are allowed, which characters are excluded, the use of uppercase and lowercase letters, and digits in a parameter.</td>
</tr>
<tr>
<td></td>
<td><strong>EXAMPLE:</strong> Job Name: [1-9 characters][#%4&amp;]</td>
</tr>
<tr>
<td>Patterns</td>
<td>Determines patterns/values to be excluded in the site standard rule, which prohibits users from using them in a job definition.</td>
</tr>
<tr>
<td></td>
<td><strong>EXAMPLE:</strong> If you do not want a root user to be defined in the Run As parameter, you can do the following:</td>
</tr>
<tr>
<td></td>
<td>Folder/Job attribute: Run as</td>
</tr>
<tr>
<td></td>
<td>Pattern Parts: Free text</td>
</tr>
<tr>
<td></td>
<td>Patterns: Root</td>
</tr>
<tr>
<td></td>
<td>You can use more than one pattern/value to exclude, which you separate by &quot;,&quot;.</td>
</tr>
<tr>
<td></td>
<td>You can also use &quot;*&quot; as a wildcard to exclude a string.</td>
</tr>
<tr>
<td></td>
<td><strong>EXAMPLE:</strong> If you do not want users to define an AIX host in the Host/Host Group parameter and your organization uses strings such as aix1-d1d2, aix2-d2d3, etc. as a naming convention for all AIX hosts, you can do the following:</td>
</tr>
<tr>
<td></td>
<td>Folder/Job attribute: Host/Host Group</td>
</tr>
<tr>
<td></td>
<td>Pattern Parts: Free text</td>
</tr>
<tr>
<td></td>
<td>Patterns: aix*</td>
</tr>
<tr>
<td>Validate according to topology</td>
<td>Verifies that the defined Host/Host Group exists as an agent, remote host, or host group in the Control-M Configuration manager. This prevents the user from defining hosts that do not exist on the Control-M on which the job runs.</td>
</tr>
<tr>
<td></td>
<td>This checkbox appears when you select the <strong>Host/Host group</strong> Control-M parameter from the <strong>Folder/Job Attribute</strong> drop-down list.</td>
</tr>
</tbody>
</table>
### Parameter | Description
---|---
**Destination** | Enables you to define specific rules for Notifications before and after jobs completion/On-Do Notify messages for a specific destination. The two options in the Folder/Job Attribute drop-down list appear as Notification Message and Do Notify Message. The rule is applied when a user defines a notification message on the destination you set in the Site Standard rule.  
**EXAMPLE:** For a Do Notify Message, destination Alerts Window, users can only use fixed text Smart followed by free text; Smart*.  
Folder/Job attribute: Do Notify Message  
Pattern Parts: Fixed Text (SMART)  
Pattern Parts: Free text

**Rerun Limitation** | Enables you to limit the minimum and maximum rerun interval in minutes, hours, and days. The rule is applied when a user defines a cyclic job in the Scheduling tab of the Planning domain in Workload Change Manager and Control-M. You can also set a default value. Take note, the default value has to comply with the limitation rule you set.  
The new option appears in the Folder/Job Attribute drop-down list as Rerun Interval.

---

### Editing a Site standard

This procedure describes how to edit a Site standard, which enables you to modify the details of the Site standard. There might be folders assigned to the Site standard you want to edit.

➢ **To edit a Site standard:**

1. From the **Tools** domain, in the **Planning** area, select **Site standards**.  
The Site standards tab appears.

2. Select the Site standard you want to edit.  
The Site standard opens

3. Edit the details of the Site standard according to your needs. For more information, see [Creating a Site standard](#) (on page 32).

4. Click .  
The Site standard is modified with the new changes.
Duplicating a Site standard

This procedure describes how to duplicate a Site standard, which enables you to make a copy of an existing Site standard and edit its contents and save it as a new Site standard.

➢ To duplicate a Site standard:

1. From the Tools domain, in the Planning area, select Site standards.
   The Site standards tab appears.
2. Select the Site standard you want to duplicate.
3. From the Edit group in the ribbon, click Duplicate.
   The duplicated Site standard is created, and appears. You can change the name and definitions of the Site standard, as described in Creating a Site standard (on page 32).

Deleting a Site standard

This procedure describes how to delete a Site standard. There might be active job flows using the Site standard you want to delete.

➢ To delete a Site standard:

1. From the Tools domain, in the Planning area, select Site standards.
   The Site standards tab appears.
2. Select the Site standard you want to delete.
3. From the Edit group, click Delete.
   The Site standard is deleted.

Assigning a Site standard to a folder/folders

This procedure describes how to assign a Site standard to a folder/folders from the Folder tool.

For more information on Site standards, see Site standards management (on page 27).

➢ To assign a Site standard to a folder/folders:

1. From the Tools domain, select Folders.
   The Folders tab appears.
2. Do one of the following:
   - If you want to assign a Site standard to one folder, select the folder, and click Set.
   - If you want to assign a Site standard to multiple folders, select all folders, and click Set.
The Set Site standard window appears.

3. Do one of the following:
   - If you selected one folder, do the following:
     a. From the Site standard drop-down list, select the Site standard.
     b. Select the Business parameters, if any, and assign the value.
   - If you selected more than one folder, do the following:
     c. Select the Site standards and Business Fields area check-box, and then select the Site standard from the drop-down list, and set the value for the Business parameter.

The Site standard is set.

Exporting/importing Site Standards

Exporting/Importing Site Standards enables you to do the following, by running the exportsitestandards and importsitestandards utilities:

**NOTE:** It is recommended that you back up your Workload Change Manager entities before performing any changes, by running the util utility (-wcm type).

- Perform a mass update on multiple Site Standards, where you can add/update a Site Standard rule in a large number of existing Site Standards. When adding a new rule, it is best to add the rule to an existing Site Standard in the Tools section, as described in Site standards management (on page 27). This serves as a template in which you export using the utility.

- Export one or multiple Site Standards from one environment to another, between two environments on different Control-M/Enterprise Managers. In this case, you may want to update some rules before importing them into the destination environment. For example, some values might change between a test environment and a production environment.

After you export the Site Standards, you can modify the output file that is created, as described in Site Standards output file (on page 40). You can then import the Site Standards, as described in the importsitestandards utility. Once you have successfully imported your Site Standards, it is best to validate the changes and ensure that there are no errors.
Site Standards output file

After you run the exportsitestandards, an output file is created in the destination you have specified in the utility command. The output file is a Protocol Buffers file, that contains the extracted Site Standards data. This file has a specific structure that must be adhered to, or else import fails.

There are a few important notes to consider when making changes to the output file:

- When creating a new rule that you want to apply to multiple standards, it is recommended that you add the rule to an existing Site Standard in the Tools section, as described in Site standards management (on page 27). This serves as a template in which you export using the utility. You can then copy the rule to the Site Standards output file which you have exported. This ensures that you have the right structure of the rule, and prevents errors.

- If you are exporting and then importing Site Standards between two environments, you must change the Site Standard’s name in the output file if the Site Standard with the same name exists. The name cannot exceed the limit of 100 characters.

- The description must not exceed the limit of 4000 characters.

- If you are changing a value in a rule, you must change the "comment" as well.

```plaintext
rule {
  id: 10003
  comment: "FixedText"
  comment: "prefix"
  result {
    message_formatter: "SiteStandardMessageFormatter"
    parameter: "MissingTextPatternPart"
  }
  compare {
    value: "prefix"
  }
}
```

- When making changes to multiple Site Standards, you can copy the rules manually to the output file, or run an update script that you have created.

The following are examples of Site Standard rules and how they appear in the output files:

- Applied on job type sample (on page 41)
- Possible values in output file sample (on page 45)
- Pattern parts in output file sample (on page 48)
- Character rules in output files (on page 57)
- Length rules in output files sample (on page 61)
Applied on job type sample

If you have a Site Standard that contains more than one rule for the same field, you must create sub-rules under the main rule in the output file.

In the sample below, a Site Standard contains one rule for Application that is applied on job type OS only, with the **Required** checkbox checked.
In the sample below, it appears as follows in the output file. The rule contains the **Require** value, and the **When** value which indicates which job type the rule is applied on, is directly under the rule.

```json
site_standard {
    name: "sample"
    description: ""
    created_time: "20170103140135UTC"
    updated_time: "20170103140318UTC"
    user_name: "emuser"
    enabled: true
    data {
        catalog_file_data {
            catalog {
                name: "SiteStandardCatalog"
                data_rule {
                    type: "FolderData"
                }
            }
            data_rule {
                type: "JobData"
                rule {
                    id: 10001
                    comment: "ControlMField"
                    comment: "AppliedOnJobType=OS"
                    name: "ControlMField:application"
                    context: "application"
                    if: "job || job-in-smart && >=800"
                    require {
                        value: Nonempty
                    }
                    choice {
                        named_value {
                            value: "a"
                            name: ""
                        }
                        named_value {
                        value: "b"
                        name: ""
                    }
                        named_value {
                        value: "c"
                        name: ""
                    }
                    when {
                        context: "{external Job.EntityType}"
                        compare {
                            value: "OS"
                        }
                    }
                }
            }
        }
    }
}
```
In the sample below, a Site standard contains multiple rules for field Application that applies to OS job type, z/OS job type and all job types:

In the output file, there is one main rule and subsequent sub-rules for the OS, z/OS, and all job types. As shown below, the **When** values (highlighted in yellow below) are in each sub-rule. The **Require** value (highlighted in yellow below) in the main rule, overrides the required values in the sub-rules.
Possible values in output file sample

In the following sample, you could view how a Site Standard with possible values defined for Folder/Job rules, Business Parameters, and Internal rules will appear in the output file.

- A Site Standard with possible values defined for a job rule. In this case we have a rule for the field Description, that contains three possible values a, b, c. The three values also contain descriptions a (value a), b (value b), c (value c).

In the output file, the possible values for the job rule appear under data rule type "JobData", with the choice of the three values a, b, c. The description of each value is in the "name" tag.

```json

data_rule {
  type: "JobData"
  rule {
    id: 10001
    comment: "ControlMField"
    name: "ControlMField: description"
    context: "description"
    choice {
      named_value {
        value: "a"
        name: "value a"
      }
      named_value {
        value: "b"
        name: "value b"
      }
      named_value {
        value: "c"
        name: "value c"
      }
    }
  }
}
```
A Site Standard with possible values defined for a Business Parameter that contains three possible values business1, business2, and business3. The three values also contain descriptions business1 (value 1), business2 (value 2), business3 (value 3).

In the output file, the possible values for the Business Parameter appear under data rule type "FolderData", with the choice of the three values business1, business2, and business3. The description of each value is in the "name" tag.

```json
data_rule {
  type: "FolderData"
  rule {
    id: 10002
    comment: "BusinessField"
    name: "BusinessField:SampleBusinessField"
    context: "BusinessField:SampleBusinessField"
    choice {
      named_value {
        value: "business1"
        name: "value1"
      }
      named_value {
        value: "business2"
        name: "value2"
      }
      named_value {
        value: "business3"
        name: "value3"
      }
    }
  }
}```
A Site Standard with possible values defined for an Internal Rule that contains three possible values rule1, rule2, and rule3.
In the output file, the possible values for the Internal Rule appear under data rule type "InternalField", with the choice of the three values rule1, rule2, and rule3. The description of each value is in the "name" tag, in this sample, it is empty.

```json
data_rule {
    type: "InternalField"
    rule {
        id: 10003
        comment: "InternalField"
        name: "InternalField:SampleInternalRule"
        context: "InternalField:SampleInternalRule"
        choice {
            named_value {
                value: "rule1"
                name: ""
            }
            named_value {
                value: "rule2"
                name: ""
            }
            named_value {
                value: "rule3"
                name: ""
            }
        }
    }
}
```

**Pattern parts in output file sample**

In the following sample, you could view how a Site Standard with rules that contain Pattern Parts appear in the output file. Since you can have a rule made up of a combination of Pattern Parts, it is important to know how they are structured in the output file.
A Site Standard with a Folder/Job rule that contains a combination of fixed text and free text Pattern Parts. In this case we have a rule for the field Description, that contains three pattern parts, fixed text (value=prefix), free text, and fixed text (value=suffix).
In the output file, the Pattern Parts combination appear under "Slide". In this case, since we have three Pattern Parts, there are three rules in the "Slide".

```json
data_rule {
  type: "JobData"
  #example for fixed and free text
  rule {
    id: 10002
    comment: "ControlMField"
    name: "ControlMField:description"
    context: "description"
    require {
      value: Nonempty
    }
  }

  slide {
    rule {
      id: 10003
      comment: "FixedText"
      comment: "prefix"
      result {
        message_formater: "SiteStandardMessageFormatter"
        parameter: "MissingTextPatternPart"
      }
      compare {
        value: "prefix"
      }
    }

    rule {
      id: 10004
      comment: "FreeText"
      comment: "*"
      length {
        min: 0
      }
    }

    rule {
      id: 10005
      comment: "FixedText"
      comment: "sufix"
      result {
        message_formater: "SiteStandardMessageFormatter"
        parameter: "MissingTextPatternPart"
      }
      compare {
        value: "sufix"
      }
    }
  }

  use_backtracking: true
  result {
    message_formater: "SiteStandardMessageFormatter"
    parameter: "InvalidPatternPart"
  }

```
A Site Standard with a Folder/Job rule that contains a Folder/Job Attribute as a pattern part. In this case we have a rule for the field Application, that contains Job Name attribute as a pattern part.
In the output file, the Pattern Parts rule for Application with pattern part Job Name appears under "Slide".

```json
rule {
  id: 10006
  comment: "ControlMField"
  name: "ControlMField:application"
  context: "application"
}

slide {
  rule {
    id: 10007
    comment: "ControlMField"
    name: "job_name"
    result {
      message_formatter: "SiteStandardMessageFormatter"
      parameter: "MissingControlMFieldPatternPart"
    }
    compare {
      value: "{external Job.job_name}"}
  }
  use_backtracking: true
  result {
    message_formatter: "SiteStandardMessageFormatter"
    parameter: "InvalidPatternPart"
  }
```
A Site Standard with a Folder/Job rule that contains a combination of Business Parameter and free text Pattern Parts. In this case we have a rule for the field Command Line, that contains two pattern parts, Business Parameter (value=SampleBusinessField), and followed by free text.
In the output file, the Pattern Parts combination appear under "Slide". In this case, since we have two Pattern Parts, there are two rules in the "Slide".

```
rule {
    id: 10009
    comment: "ControlMField"
    name: "ControlMField:cmd_line"
    context: "cmd_line"

    Slide {
        rule {
            id: 10010
            comment: "BusinessField"
            comment: "SampleBusinessField"
            result {
                message_formatter: "SiteStandardMessageFormatter"
                parameter: "MissingBusinessFieldPatternPart"
            }
            compare {
                value: "{external SampleBusinessField__BusinessFieldValue}"
            }
        }
        rule {
            id: 10011
            comment: "FreeText"
            comment: "="
            length {
                min: 0
            }
        }
        use_backtracking: true
        result {
            message_formatter: "SiteStandardMessageFormatter"
            parameter: "InvalidPatternPart"
        }
    }
```
A Site Standard with a Folder/Job rule that contains a combination of an Internal Rule and free text Pattern Parts. In this case we have a rule for the field Job Name, that contains two pattern parts, Internal Rule (value=SampleInternalRule), and followed by free text. The Internal Rule contains three possible choices, rule1, rule2, and rule3.

In the output file, the Pattern Parts combination appear under "Slide". In this case, since we have a pattern part Internal Rule that contains three Possible Values, we also see a rule within the Internal Rule with the three choices, followed by the free text rule.
```json
rule {
    id: 10016
    comment: "ControlMField"
    name: "ControlMField:job_name"
    context: "job_name"
}
rule {
    id: 10012
    comment: "ControlMField"
    name: "ControlMField:job_name"
    if: "!any-folder"
}
slide {
    id: 10013
    comment: "InternalField"
    comment: "SampleInternalRule"
    result {
        message_formatter: "SiteStandardMessageFormatter"
        parameter: "MissingInternalFieldPatternPart"
    }
}
rule {
    id: 10014
    choice {
        named_value {
            value: "rule1"
            name: ""
        }
        named_value {
            value: "rule2"
            name: ""
        }
        named_value {
            value: "rule3"
            name: ""
        }
    }
}
rule {
    id: 10015
    comment: "FreeText"
    comment: "="
    length {
        min: 0
    }
}
use_backtracking: true
result {
    message_formatter: "SiteStandardMessageFormatter"
    parameter: "InvalidPatternPart"
}
```
Character rules in output files

In the following sample, you could view how a Site Standard with character rules appears in the output files.

- A Site Standard with a Folder/Job rule that contains a character rule that allows any character to be used except for specific characters. In this case we have a rule for the field Application, and any character is allowed except for ABC.
In the output file, the character rule appears in "charset" as part of the Control-M field rule. The exception appears as follows:

```plaintext
rule {
    id: 10001
    comment: "ControlMField"
    name: "ControlMField:application"
    context: "application"
    require {
        value: Nonempty
    }

    charset {
        chars: "ABC"
        opposite: true
    }

    result {
        message_formatter: "SiteStandardMessageFormatter"
        parameter: "DisallowedCharactersRestriction"
    }
}
```
A Site Standard with a Folder/Job rule that contains a character rule that allows specific characters to be used, such as upper case letters, lower case letters, digits, and/or specific characters. In this case we have a rule for the field Description, and upper case, lower case, digits, and *()& are allowed.
In the output file, the character rule appears in "charset" as part of the Control-M field rule. In this case, since we selected upper case, lower case, and digits you can see in the output file that in comment: "ULD" stands for upper case, lower case, and digits. The additional allowed characters appear in comment: "*()&". All of the specific allowed characters are listed in chars.

```json
rule {
    id: 10002
    comment: "ControlMField"
    name: "ControlMField:description"
    context: "description"
    charset {
        comment: "*()&"
        comment: "ULD"
        chars: "*()&ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789"
        result {
            message_formatter: "SiteStandardMessageFormatter"
            parameter: "AllowedCharactersOnlyRestriction"
        }
    }
}
```
Length rules in output files sample

In the following sample, you can view how length rules in a Site Standard rule appear in the output files.

- A Site Standard with a Folder/Job rule that contains a length rule that determines the minimum and maximum length of characters. In this case we have a rule for the field Application, and the length is of minimum five characters and maximum ten.
In the output file, the length rule appears as follows:

```json
data_rule {
    type: "JobData"
    rule {
        id: 10001
        comment: "ControlMField"
        name: "ControlMField:application"
        context: "application"
        require {
            value: Nonempty
        }
        length {
            min: 5
            max: 10
        }
    }
}
```