

# Configuring On-Premise Integration Servers for webMethods Cloud

Version 10.5

October 2019

This document applies to webMethods Integration Server 10.5 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2007-2023 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <https://softwareag.com/licenses/>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <https://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <https://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

**Document ID: IS-OP-105-20231222**

# Table of Contents

<b>About this Guide</b> .....	<b>5</b>
Document Conventions.....	6
Online Information and Support.....	6
Data Protection.....	7
<b>1 About Configuring an On-Premise Integration Server</b> .....	<b>9</b>
What is an On-Premise Integration Server?.....	10
About Accounts.....	10
About Applications.....	10
Deploying Assets.....	11
Steps to Configure an On-Premise Integration Server.....	11
<b>2 Configuring Settings</b> .....	<b>13</b>
About Integration Cloud Settings.....	14
Specifying Settings.....	14
Editing and Deleting Settings.....	14
Setting Maximum Thread Pool Usage for Cloud Requests.....	15
<b>3 Configuring Accounts for Integration Cloud</b> .....	<b>17</b>
Overview.....	18
Viewing the Accounts that Reside on the Server.....	18
Creating Accounts on an On-Premise Integration Server.....	19
Uploading Accounts to Integration Cloud.....	22
Testing Accounts.....	23
Enabling and Disabling Accounts.....	23
Deleting On-Premise Accounts.....	23
Editing Accounts.....	24
<b>4 Managing Applications</b> .....	<b>25</b>
About Sharing Metadata on an On-Premise Integration Server.....	26
Viewing the Applications that Reside on Your Server.....	27
Defining Applications.....	28
Uploading Applications.....	29
Deleting Applications.....	30
Editing Applications.....	30
<b>5 Managing Docker Services</b> .....	<b>33</b>
Overview.....	34
Specifying Services to Expose to Consumers in Integration Cloud.....	34
<b>6 Connecting to webMethods Integration Cloud through a Proxy</b> .....	<b>37</b>
About Configuring Integration Server.....	38

Creating a Proxy Server Alias.....	38
Updating JVM Configuration Settings for Proxies.....	38
Configuring the Internet Proxy.....	39

# About this Guide

■ Document Conventions .....	6
■ Online Information and Support .....	6
■ Data Protection .....	7

---

This guide describes how to configure Integration Server as an on-premise server for use with webMethods Integration Cloud. It contains information for administrators who configure and manage on-premise Integration Servers and for application developers who want to create services that will be accessed by integrations through Integration Cloud.

To use this guide effectively, you should understand the basic concepts described in the *webMethods Integration Server Administrator's Guide* and *webMethods Integration Cloud Help*. You should also be familiar with all the services you want to share with Integration Cloud.

## Document Conventions

---

Convention	Description
<b>Bold</b>	Identifies elements on a screen.
Narrowfont	Identifies service names and locations in the format <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies:  Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies:  Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the   symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

## Online Information and Support

---

### Software AG Documentation Website

You can find documentation on the Software AG Documentation website at <http://documentation.softwareag.com>. The site requires credentials for Software AG's Product Support

---

site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

## Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to [empower@softwareag.com](mailto:empower@softwareag.com) with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at <https://empower.softwareag.com/>.

You can find product information on the Software AG Empower Product Support website at <https://empower.softwareag.com>.

To submit feature/enhancement requests, get information about product availability, and download products, go to [Products](#).

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the [Knowledge Center](#).

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at [https://empower.softwareag.com/public\\_directory.asp](https://empower.softwareag.com/public_directory.asp) and give us a call.

## Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at <http://techcommunity.softwareag.com>. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

## Data Protection

---

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.



# 1 About Configuring an On-Premise Integration Server

---

■ What is an On-Premise Integration Server? .....	10
■ About Accounts .....	10
■ About Applications .....	10
■ Deploying Assets .....	11
■ Steps to Configure an On-Premise Integration Server .....	11

## What is an On-Premise Integration Server?

---

An *on-premise* Integration Server is any Integration Server that is configured to share service metadata with and enable for the execution of those services by integrations defined in webMethods Integration Cloud. The service metadata uploaded from an on-premise Integration Server provides the accounts and applications required to create integrations on Integration Cloud.

## About Accounts

---

An *account* on the on-premise Integration Server is a connection that you create to enable the communication of data from the on-premise Integration Server to Integration Cloud and vice versa.

When the on-premise Integration Server receives requests, it processes the requests using the account configured to process requests for a specific stage. A *stage* is an environment configured on the Integration Cloud that represents a specific point in the development cycle of an integration service. When you configure a connection to an Integration Cloud server, the on-premise Integration Server fetches the stages you defined on the Integration Cloud server, and when you configure the account, you define the stage on which the on-premise Integration Server should listen for requests. By configuring accounts for specific stages, you limit the requests that each account processes. For more information about configuring a connection to an Integration Cloud server, see [“Configuring Settings” on page 13](#).

For example, the Integration Cloud server might have one environment for development (Default), one for testing (Test), and another for production (Live). You would configure each on-premise account to listen for requests for a specific stage. So, if you develop an integration on the Default stage and invoke a service on the on-premise Integration Server, the account listening for requests on the Default stage receives and processes the request. For more information about defining stages in Integration Cloud, see *webMethods Integration Cloud Help*.

## About Applications

---

A group of services that you share with Integration Cloud is called an *application*. Applications are created on the on-premise Integration Server and uploaded to the Integration Cloud. Integration Cloud can execute any service hosted by an on-premise Integration Server. When you share services in an application, you are sharing the *metadata* for the service. For Integration Server services, the metadata you share is the service name, service signature, display name, and service comments. Users of Integration Cloud can then create integrations that invoke services defined in the applications.

When Integration Cloud executes a service, the on-premise Integration Server returns all the results to the Integration Cloud. You can batch the results to limit the number of results returned to the Integration Cloud. For more information, see the **Batch Data** field in [“Defining Applications” on page 28](#).

After you create applications, you upload them to Integration Cloud. If the application changes on the on-premise Integration Server, you must upload the application again for the changes to be replicated to Integration Cloud.

When you upload applications to the Integration Cloud server, you associate one or more accounts that the application can use to access services on the on-premise Integration Server. If the account associated with an application changes, you can upload the account to the Integration Cloud server without having to upload the application. For more information about uploading accounts, see [“Uploading Accounts to Integration Cloud ” on page 22](#).

## Deploying Assets

---

You can deploy settings, accounts, and applications as assets using webMethods Deployer. For more information, see *webMethods Deployer User’s Guide*.

## Steps to Configure an On-Premise Integration Server

---

The following table describes the process for using Integration Server Administrator to configure an on-premise Integration Server:

Step	Description
Step 1	Create a user account on the Integration Cloud server. For more information, see <i>webMethods Integration Cloud Help</i> .
Step 2	Specify the settings for the Integration Cloud server. See <a href="#">“Configuring Settings” on page 13</a> .
Step 3	Create the accounts by which Integration Cloud can connect to the on-premise Integration Server. See <a href="#">“Configuring Accounts for Integration Cloud ” on page 17</a> .
Step 4	Define and upload the applications to share through Integration Cloud. See <a href="#">“Managing Applications” on page 25</a> .
Step 5	Define on-premise Integration Server services that can be invoked in webMethods Integration Cloud running the on-premise Integration Server inside a Docker container. See <a href="#">“Managing Docker Services” on page 33</a> .



# 2 Configuring Settings

---

- About Integration Cloud Settings ..... 14
- Specifying Settings ..... 14
- Editing and Deleting Settings ..... 14
- Setting Maximum Thread Pool Usage for Cloud Requests ..... 15

## About Integration Cloud Settings

The **Settings** screen allows you to specify the location and login details of Integration Cloud for which you will define accounts and applications.

## Specifying Settings

Complete the following steps to specify the settings for Integration Cloud.

**Note:**

For each on-premise Integration Server, you can specify settings for only one Integration Cloud server.

### > To specify the account settings for Integration Cloud

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the Navigation panel, click **Settings**.
3. Under **Settings**, complete the fields as follows:

Field	Description
<b>User Name</b>	User name for an account on Integration Cloud.
<b>Password</b>	Password identified in the user account for <b>User Name</b> .
<b>webMethods Cloud URL</b>	The URL of Integration Cloud with which to share accounts and applications created on the on-premise Integration Server. For example, the URL would be of the following format:  https://<sub-domain>.<domain-name>  , for example, https://softwareag-education.webmethodscloud.com  where softwareag-education is the sub-domain name and webmethodscloud.com is the domain-name.

4. Click **Update Settings**.

Integration Server connects to Integration Cloud specified in the **webMethods Cloud URL** and downloads the configuration information that is required to receive any incoming requests.

## Editing and Deleting Settings

You can edit the account settings details to specify a different Integration Cloud server.

**Note:**

To delete the settings, delete the information from the fields on the **Settings** screen.

➤ **To edit the account settings**

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Settings**.
3. Update the settings.
4. Click **Update Settings**.

## Setting Maximum Thread Pool Usage for Cloud Requests

When an application developed on Integration Cloud invokes a service on an on-premise Integration Server, the on-premise Integration Server uses a thread from the server thread pool to process the service request. The processing thread invokes the service, posts the response, and then terminates which frees up the thread to process other requests. The on-premise Integration Server can process multiple service requests from Integration Cloud simultaneously.

The `watt.server.threadPool.cloudRequests` specifies the maximum percentage of the server thread pool that can be used for processing Integration Cloud requests. The default is 75.

**Note:**

After application of a fix that includes PIE-60686 (IS\_10.5\_WmCloud\_Fix1 and higher), the default is 5 which means that if the maximum size of the server thread pool is 1000, up to 50 server threads at a time can be used to process cloud requests.

At start up, Integration Server uses the percentage to calculate the number of threads that can be used and logs a message with that information to the `server.log`. At run-time, when Integration Server reaches the maximum number of threads that can be used to process requests from Integration Cloud, Integration Server writes the following message and blocks new server requests from Integration Cloud until a thread becomes available.

```
[ISS.0021.8040W] Server thread pool is using the maximum number of threads allowed for processing requests from the webMethods Cloud. Requests from webMethods Cloud will block until a thread becomes available.
```

**Note:**

If `watt.server.threadPool.cloudRequests` is set to 0, the on-premise Integration Server processes requests from webMethods Cloud one at a time. The on-premise Integration Server blocks processing of further requests from webMethods Cloud until the current service execution completes. This delays processing of requests from webMethods Cloud.



# 3 Configuring Accounts for Integration Cloud

---

- Overview ..... 18
- Viewing the Accounts that Reside on the Server ..... 18
- Creating Accounts on an On-Premise Integration Server ..... 19
- Uploading Accounts to Integration Cloud ..... 22
- Testing Accounts ..... 23
- Enabling and Disabling Accounts ..... 23
- Deleting On-Premise Accounts ..... 23
- Editing Accounts ..... 24

## Overview

---

When you create an account on the on-premise Integration Server, you specify the connection parameters that the on-premise Integration Server uses to access Integration Cloud for which you defined settings as described in [“Specifying Settings” on page 14](#). Each account listens for requests from a specific stage on Integration Cloud. You can define as many accounts for each stage as is necessary for your needs and upload them to make them available to Integration Cloud.

You enable accounts on the on-premise Integration Server to allow them to serve any requests that originate from Integration Cloud. If an account is disabled on the on-premise Integration Server, any requests sent from Integration Cloud will time out depending on the amount of time specified in the **Request Timeout** field. For more information about the **Request Timeout** field, see [“Creating Accounts on an On-Premise Integration Server” on page 19](#).

You can upload accounts in two ways: as part of an application or individually, separate from applications. In order to upload an account as part of an application, you must first create the account and then associate it to the application when you upload it to Integration Cloud as described in [“Uploading Applications” on page 29](#).

You can upload accounts separately from applications in the event that you need to override the settings of a previously uploaded account. You might do this if you associate an account while uploading an application to Integration Cloud and later change the account details. This way, you can change the account details without having to upload the entire application.

## Viewing the Accounts that Reside on the Server

---

The **Accounts** screen lists all of the Integration Cloud accounts that reside on your on-premise Integration Server. It also displays whether each account is enabled.

### ➤ To view the accounts that reside on the on-premise Integration Server

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.

The **Accounts** screen displays the list of accounts that were created on the on-premise Integration Server. Integration Server Administrator displays the following information for each account:

Column	Description
<b>Alias</b>	The name of the account.
<b>Description</b>	Description of the account.
<b>Stage</b>	The Integration Cloud stage from which the on-premise Integration Server receives requests.

Column	Description						
<b>Last Uploaded Time</b>	The last time the account was uploaded to Integration Cloud.						
<b>Upload</b>	Displays one of the following icons to allow you to upload the account: <table border="1" data-bbox="479 445 1333 674"> <thead> <tr> <th>This icon...</th> <th>Indicates that the account is...</th> </tr> </thead> <tbody> <tr> <td></td> <td>New or has been edited since the last time it was uploaded to Integration Cloud.</td> </tr> <tr> <td></td> <td>Is synchronized with the account that has already been uploaded to Integration Cloud.</td> </tr> </tbody> </table>	This icon...	Indicates that the account is...		New or has been edited since the last time it was uploaded to Integration Cloud.		Is synchronized with the account that has already been uploaded to Integration Cloud.
This icon...	Indicates that the account is...						
	New or has been edited since the last time it was uploaded to Integration Cloud.						
	Is synchronized with the account that has already been uploaded to Integration Cloud.						
<b>Test</b>	Displays the  icon to allow you to test the account.						
<b>Enabled</b>	Indicates whether the account is enabled.						
<b>Delete</b>	Displays the  icon to allow you to delete the account from the on-premise Integration Server and Integration Cloud.						

## Creating Accounts on an On-Premise Integration Server

Perform the following procedure to create an account on an on-premise Integration Server.

### > To create an account

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Click **Create On-Premise Account**.
4. Under **General Settings**, complete the fields as follows:

Field	Description
<b>Enable</b>	Enables or disables an Integration Cloud account. Valid values: <ul style="list-style-type: none"> <li>■ <b>Yes</b> - Enables an Integration Cloud account.</li> <li>■ <b>No</b> - Disables an Integration Cloud account.</li> </ul>
<b>Alias Name</b>	A unique name for the account.
<b>Description</b>	Description of the account.

Field	Description
<b>Stage</b>	The Integration Cloud stage from which the on-premise Integration Server receives requests. The list is populated by the stages defined on Integration Cloud.

**Note:**

The typical life cycle of an integration development involves creating integrations, testing them, and making them production worthy. Each of these activities can be termed as different stages of an integration life cycle development.

5. Under **Account Settings**, complete the fields as follows:

Field	Description
<b>Maximum Reconnection Attempts</b>	Specify the maximum number of reconnection attempts that Integration Server should make if the connection to Integration Cloud fails. When the connection between webMethods Cloud and on-premise Integration Server is down, Integration Server attempts to reconnect to webMethods Cloud with a random interval between 100 milliseconds to 10 seconds for the first two minutes and then waits for a period of 30 to 60 seconds before attempting to retry the connection. The on-premise Integration Server continues in this manner until connectivity to webMethods Cloud is restored.

**Request Timeout** Maximum amount of time (in milliseconds) that Integration Cloud waits for the on-premise Integration Server to process a request. If the on-premise Integration Server is not listening for a request or if it takes longer to process the request than the specified time, Integration Cloud issues an error and stops listening for a response. The default is 60000 milliseconds (1 minute). "

When the request time expires, Integration Cloud shows this exception "[ISS.0021.8042E] Error occurred while executing service <on-prem-servicename> with request ID <requestID>. The on-premise Integration Server did not respond within the configured request timeout of <requestTimeoutValue> milliseconds. Check on-premise logs.

If the invoked on-premise service takes more time to execute than the **Request Timeout** value, Integration Server writes the following error message to the server log: [ISS.0021.8041E] Error occurred while executing service <serviceName> with request ID <requestID>. Service execution took <x> milliseconds, which is more than the configured request timeout of <y> milliseconds.

**Note:**

Field	Description
	<p>The error message ISS.0021.8041E was added for PIE-65251 which is part of IS_10.5_WmCloud_Fix2 and higher.</p> <p>When responding to a request from Integration Cloud, the on-premise Integration Server adds a time-to-live (TTL) property to the response. The on-premise Integration Server uses the <b>Request Timeout</b> value set for the Integration Cloud account as the TTL value for the response. If the on-premise Integration Server sends the response after the timeout value for the request elapses on Integration Cloud, the response remains on Integration Cloud only until the response expires (that is, only until the TTL in the response elapses).</p> <p><b>Note:</b> The use of the <b>Request Timeout</b> value as a TTL for the response applies only after installation of a fix that includes issue PIE-60585 (IS_10.5_WmCloud_Fix1 and higher).</p>
<b>Time to Live</b>	<p>If you are batching the data from the on-premise Integration Server, the length of time in seconds that the execution results remain in the cache of the on-premise Integration Server. The value must be greater than 0. The default is 60 seconds.</p>
<b>Allowed On-Premise Hosts</b>	<p>(Optional.) The on-premise Integration Server might use multiple addresses, depending on which network or proxy it uses to access Integration Cloud. Specify a comma-separated list of IP addresses that can receive requests from Integration Cloud. Only those IP addresses specified can receive requests.</p> <p>If no value is specified, Integration Cloud derives the IP address of the on-premise Integration Server that uploads the account to Integration Cloud and allows only that IP address to receive requests from Integration Cloud.</p>
<b>Run As User</b>	<p>Specify the user name you want the on-premise Integration Server to use when running the service. Click  to search for a user and select the user. You can select users from the local or central directory.</p> <p>The on-premise Integration Server runs the service as if the user you specify is the authenticated user who invoked the service. If the service is governed by an ACL, ensure that you specify a user who is allowed to invoke the service.</p>

6. If you want to test the account, click **Test Account Settings**.

Integration Server Administrator displays a status line that indicates whether the account is successful or not. The status line is displayed at the top of the screen.

7. Click **Save Changes**.

## Uploading Accounts to Integration Cloud

After an account is created, you upload it to Integration Cloud so that it can be used to execute services on the on-premise Integration Server. If you change the account after it is uploaded to Integration Cloud, you must upload it again for the changes to become effective.

### Note:

If you want to upload the account as part of an application, use the procedure described in [“Uploading Applications” on page 29](#).

### ➤ To upload accounts to Integration Cloud

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Click one of the following in the **Upload** column for the account you want to upload.

#### Click this icon... If the account on the on-premise Integration Server is...



New or has been edited since the last time it was uploaded to Integration Cloud.

- For a new account, this icon indicates that the account has not been uploaded to Integration Cloud.
- For an account that already exists on Integration Cloud, this icon indicates that the account on the on-premise Integration Server is not synchronized with the one on Integration Cloud.



Synchronized with the account that has already been uploaded to Integration Cloud.

When you upload the account, Integration Server Administrator:

- Displays a status line that indicates whether the account has been uploaded successfully. The status line is displayed at the top of the screen.
- Updates the **Last Uploaded Time** field to indicate the time that the account was uploaded and displays the  icon to indicate that the account on Integration Cloud is synchronized with the one on the on-premise Integration Server.

### Note:

If an Integration Cloud account is disabled during an upload, Integration Server uploads the account successfully. However, you will be notified to enable the account to serve any requests originating from Integration Cloud.

## Testing Accounts

After you add an account, you can test the account settings to ensure that the account is valid. Use the following procedure to test the account.

### > To test the account

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Click the ▶ icon in the **Test** column for the account you want to test.

Integration Server Administrator displays a status line that indicates whether the account is valid. The status line is displayed at the top of the screen.

## Enabling and Disabling Accounts

When accounts are disabled, Integration Cloud cannot execute services on the on-premise Integration Server. After the account is enabled, Integration Server automatically establishes connectivity with Integration Cloud at startup and is ready to serve any requests originating from Integration Cloud. If you have not enabled an account while creating it, you can also enable the account using the following procedure:

### Note:

When an account is disabled, requests sent to the on-premise Integration Server remain in the queue until they are read or expire.

### > To enable or disable an account

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Under the **Enabled** column of the **On-Premise Accounts** table, select one of the following:

Click...	To...
<b>No</b>	Enable an Integration Cloud account.
<b>Yes</b>	Disable an Integration Cloud account.

## Deleting On-Premise Accounts

When you no longer need an account, you can delete it. Deleting an account from the on-premise Integration Server also deletes the account from Integration Cloud.

**Note:**

If the account is in use by any of the integration flows in Integration Cloud, the delete operation will fail.

➤ **To delete an on-premise account**

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Click the **✗** icon in the row that corresponds to the account you want to delete in the **Delete** column of the **On-Premise Accounts** table.
4. When Integration Server asks you to confirm that you want to delete the account, click **OK**.

## Editing Accounts

---

After creating an account, you can edit the details. If you edit an account, you must upload the account for the changes to take effect on Integration Cloud.

➤ **To edit an account**

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Accounts**.
3. Locate the account you want to edit and click on the name in the **Alias** column.
4. Update the information for the account.
5. Click **Save Changes**.
6. To upload the account, follow the procedure described in [“Uploading Accounts to Integration Cloud”](#) on page 22.

# 4 Managing Applications

---

■ About Sharing Metadata on an On-Premise Integration Server .....	26
■ Viewing the Applications that Reside on Your Server .....	27
■ Defining Applications .....	28
■ Uploading Applications .....	29
■ Deleting Applications .....	30
■ Editing Applications .....	30

## About Sharing Metadata on an On-Premise Integration Server

---

You create applications on the on-premise Integration Server to share services with Integration Cloud. Keep the following points in mind while sharing services through an application:

- You can share only services running on the on-premise Integration Server configured to create applications on Integration Cloud.
- You can share only services contained in custom packages.
- You can share services from different packages in the same application. For example, if service A is located in package A, and service B is located in package B, you can add both service A and service B to the same application.
- You can share only those services that have an input and/or output signature specified.
- You can share only those services whose signatures are of the following data types:
  - String
  - String List
  - Document
  - Document Reference
  - Document List
  - Document Reference List
  - Object
  - Object List
- Any Object or Object List placed into the service pipeline, and therefore any Object or Object List in a service signature, must be a serializable data type. If it cannot be serialized, the pipeline cannot be exchanged with a remote Integration Server. Further, the class names for the data type must be included in the Integration Server whitelist classes file or a package whitelist classes file for the pipeline to be deserialized at the destination Integration Server. Classes defined by Integration Server are already included in the whitelist class file. This requirement applies to all Object and Object List variables, including those contained in a Document or Document List. For more information about the Integration Server whitelist, see the *webMethods Integration Server Administrator's Guide*.
- You can set the on-premise Integration Server to send service results to Integration Cloud in batches.
- You cannot share service signatures that include:
  - Cyclical dependencies of document references.
  - Fields of type String Table, including fields of type String Table in a Document.
  - An empty Document or Document List.

- You must configure one or more accounts to associate with the application before you can upload the application to Integration Cloud. For more information about configuring accounts, see [“Configuring Accounts for Integration Cloud ” on page 17.](#)
- You must upload the application for the updates to be shared with Integration Cloud if you edit:
  - The application.
  - The signature or referenced Document of a service shared by the application.
- When you upload an application, it replaces the application and operations available on Integration Cloud with the one that you upload.

## Viewing the Applications that Reside on Your Server

The Applications screen lists all of the applications on the on-premise Integration Server that you can share with Integration Cloud.

➤ **To view the applications you can share with Integration Cloud from the on-premise Integration Server**

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Applications**.

The Applications screen displays the list of applications that were created on the on-premise Integration Server. Integration Server Administrator displays the following information for each application:

Column	Description
<b>Name</b>	Name of the application.
<b>Description</b>	Description of the application.
<b>Last Uploaded Time</b>	The last time the application was uploaded to Integration Cloud.
<b>Upload</b>	Displays one of the following icons to allow you to upload the application:
<b>This icon...</b>	<b>Indicates that the application is...</b>
	New or has been edited since the last time it was uploaded to Integration Cloud.
	Synchronized with the application that has already been uploaded to Integration Cloud.

Column	Description
<b>Delete</b>	Displays the  icon to allow you to delete the application from the list and from Integration Cloud.

## Defining Applications

---

Perform the following procedure to define applications to share with Integration Cloud.

### ➤ To define an application to share with Integration Cloud

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Applications**.
3. Click **Define webMethods Cloud Application**.
4. Under **webMethods Cloud Application**, complete the following fields:

Field	Description
<b>Name</b>	A unique name for the application. The application name cannot exceed 32 characters, contain reserved words and characters that are used in Java or C/C++ (such as <i>for</i> , <i>while</i> , and <i>if</i> ), or the following illegal characters:  "#-&@^!%*:\$./\ \ ` ; , ~ + = ) (   } { [ ] > < "
<b>Description</b>	Description of the application.

5. In the **Assign Services to Application** area, specify the services to expose to Integration Cloud as follows:
  - a. Under **Package/Services**, click  to expand the services available in the package.
  - b. Select each service you want to expose to Integration Cloud.
  - c. If you want the service to return results to Integration Cloud in batches, rather than returning the results all at once, click **Batch Data**.

If the top level output signature of the service contains only one field, and the field is a document list or document reference list (both of which are IData arrays), **Batch Data** is selected by default. Otherwise, it will be cleared.

By selecting this option, Integration Server uses `pub.flow.iterator` to batch service results. For more information about the `pub.flow.iterator` service, see *webMethods Integration Server Built-In Services Reference*.

- d. Under **Display Name**, specify the name of the service as it should appear in Integration Cloud or accept the default.

The **Display Name** field cannot contain reserved words and characters that are used in Java or C/C++ (such as *for*, *while*, and *if*) or the following illegal characters:

```
"#-&@^!%*:.\/\ \ `;~+=)(|}{][><"
```

**Note:**

Applications are displayed in Integration Cloud as *operations*. Operations are named according to the name defined by the **Display Name** field. For more information about operations in Integration Cloud, see *webMethods Integration Cloud Help*.

6. Click **Save Changes**.

## Uploading Applications

After you define an application, you must upload it to Integration Cloud before using it in an integration.

### > To upload an application

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Applications**.
3. Click one of the following icons in the row that corresponds to the application you want to upload in the **Upload** column of the **webMethods Cloud Applications** table.

This icon...	Indicates that the application on the on-premise Integration Server is...
	<p>New or has been edited since the last time it was uploaded to Integration Cloud.</p> <ul style="list-style-type: none"> <li>■ For a new application, this icon indicates that the application has not been uploaded to Integration Cloud.</li> <li>■ For an application that already exists on Integration Cloud, this icon indicates that the application on the on-premise Integration Server is not synchronized with the one on Integration Cloud.</li> </ul>
	Synchronized with the application that has already been uploaded to Integration Cloud.

4. When the **Upload Application** screen appears, from the **Select** column of the **Select Accounts** area, select one or more accounts to associate with the application.

5. Click **Upload**.
6. When Integration Server asks you to confirm that you want to upload the application, click **OK**.

When you upload the application, the on-premise Integration Server:

- Uploads the application to Integration Cloud, replacing the existing application.
- Updates the **Last Uploaded Time** column of the **webMethods Cloud Applications** screen to indicate that the application on Integration Cloud is synchronized with the one on the on-premise Integration Server.
- Shares the service name, service signature, display name, and service comments with Integration Cloud.

**Note:**

If the application has been uploaded previously to Integration Cloud, uploading the application again overwrites the existing application.

**Note:**

If any account associated with an application is disabled during an upload, Integration Server uploads the application successfully. However, you will be notified to enable the account to serve any requests originating from Integration Cloud.

## Deleting Applications

---

When you no longer want to share applications with Integration Cloud, you can delete them. Deleting an application from the on-premise Integration Server also deletes the application and its corresponding operations from the Integration Cloud.

### > To delete an application

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Applications**.
3. Click the **X** icon in the row that corresponds to the application you want to delete in the **Delete** column of the **webMethods Cloud Applications** list.
4. When Integration Server asks you to confirm that you want to delete the application, click **OK**.

## Editing Applications

---

After creating an application, you can edit the application details. If you edit an application, you must upload the application for the changes to take effect on Integration Cloud.

**> To edit an application**

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Applications**.
3. Locate the application you want to edit and click on the name in the **Name** column.
4. Update the information for the application.
5. Click **Save Changes**.

**Note:**

The updated application is not available for use on Integration Cloud until you upload it to Integration Cloud. Until then, Integration Cloud uses the last version of the application that you uploaded. To upload the application, follow the procedure described in [“Uploading Applications” on page 29](#).



# 5 Managing Docker Services

---

■ Overview .....	34
■ Specifying Services to Expose to Consumers in Integration Cloud .....	34

## Overview

---

You can create a Docker image from an installed and configured Integration Server instance and then run the Docker image inside a Docker container. Integration Server provides a script to create Docker files, build Docker images, load or push the Docker image to a Docker registry hosted on-premise, and push the Docker image to a Docker registry hosted on Integration Cloud.

Docker support is available on Linux and UNIX systems for which Docker provides native support.

**Note:**

You must enable CSRF Guard in Integration Server before you create the Docker image and upload it to Integration Cloud. Enabling the CSRF security feature will prevent CSRF attacks. See the *webMethods Integration Server Administrator's Guide* on the Software AG Documentation website at <http://documentation.softwareag.com> for information on how to enable CSRF Guard.

## Specifying Services to Expose to Consumers in Integration Cloud

---

If you want to expose Integration Server services running inside a Docker container in Integration Cloud to consumers, you must specify the Integration Server services before creating the Docker image for Integration Server. The **webMethods Cloud Docker Services** screen lists all the services to expose to consumers from a Docker container.

### ➤ To specify the services to expose in Integration Cloud

1. Open the Integration Server Administrator if it is not already open.
2. In the **webMethods Cloud** menu of the navigation panel, click **Docker Services**.

The **webMethods Cloud Docker Services** screen is displayed. The **Package** column lists all packages in Integration Server.

3. In the **Docker Services** column, specify the services to expose to consumers from your custom packages. You can specify the services using one or more of the following ways:
  - Directive and fully qualified name of the service in the following format:  
*/directive/folder.subfolder:serviceName*  
For example: `/invoke/is.assets:getChecksums`. Use a comma to separate the services.
  - The URL for invoking the service. For the URL, specify a snippet that is unique enough for Integration Cloud to accurately map it to the service in the Docker container. Use a comma to separate the services.
  - A URL alias created using **Settings > URL Aliases > Create URL Alias**. Use a comma to separate the services.

4. Click **Save Changes**.

**Note:**

For information about how to create, run, and push Docker images to Integration Cloud, see the *webMethods Integration Server Administrator's Guide*. For information about managing Docker containers and repositories in Integration Cloud, see the *webMethods Integration Cloud Help*.



# 6 Connecting to webMethods Integration Cloud through a Proxy

---

■ About Configuring Integration Server .....	38
■ Creating a Proxy Server Alias .....	38
■ Updating JVM Configuration Settings for Proxies .....	38
■ Configuring the Internet Proxy .....	39

## About Configuring Integration Server

---

If your on-premise Integration Server is to connect to webMethods Integration Cloud through an Internet proxy, you may need to perform additional configuration which falls into the following categories:

- Creating a proxy server alias.
- Adding proxy-related Java system properties to the Java Virtual Machine in which the on-premise Integration Server runs.
- Configuring the Internet proxy for use with the on-premise Integration Server and webMethods Integration Cloud.

## Creating a Proxy Server Alias

---

If your company requires use of a proxy server for establishing outbound connections to applications located in the cloud, then you need to configure a proxy server alias for your on-premise Integration Server. A proxy server alias identifies the proxy server and the port on the proxy server through which you want to route requests and any credentials needed to access the proxy server.

When creating a proxy server alias for use for connection to webMethods Integration Cloud, you must do the following:

- Specify the proxy server alias as the default proxy server alias.
- Specify the proxy host and port number.
- Set the protocol to HTTPS.
- If the proxy server is configured for basic authentication, the proxy server alias must specify the user name and password required to access the proxy server.

You configure a proxy server alias using the **Settings > Proxy Servers** page in Integration Server Administrator. For detailed information about proxy servers and how to configure a proxy server, see the *webMethods Integration Server Administrator's Guide*.

## Updating JVM Configuration Settings for Proxies

---

Depending on your networking and security requirements, you may need to set Java system properties related to proxy servers. Because the connection between the on-premise Integration Server and webMethods Integration Cloud is over HTTPS, you need to set some or all of the following:

- `https.proxyHost`
- `https.proxyPort`
- `http.nonProxyHosts`
- `https.proxyUser`

- `https.proxyPassword`

Because the proxy property values need to be set for the Java Virtual Machine (JVM) in which Integration Server runs, you update the `custom_wrapper.conf` to ensure the proxy parameter values are supplied when the JVM launches. Specifically, you add a `wrapper.java.additional.n` property that specifies the property name and value that you want to pass to Integration Server, where *n* is a unique sequence number. The property name must be preceded by `-D`.

For example, the `wrapper.java.additional` properties in the `custom_wrapper.conf` file might look similar to the following:

```
wrapper.java.additional.204=-Dhttps.proxyHost=YOUR_PROXY_HOST
wrapper.java.additional.205=-Dhttps.proxyPort=YOUR_PROXY_PORT
wrapper.java.additional.206=-Dhttps.proxyUser=YOUR_PROXY_USER
wrapper.java.additional.207=-Dhttps.proxyPassword=YOUR_PROXY_PASSWORD
wrapper.java.additional.208=-Dhttp.nonProxyHosts=localhost|127.0.0.1
wrapper.java.additional.209=-DHTTP_PROXY=YOUR_PROXY_HOST:YOUR_PROXY_PORT
wrapper.java.additional.210=-DHTTP_AUTH=YOUR_PROXY_USER:YOUR_PROXY_PASSWORD
```

For instructions about passing Java system properties to Integration Server and the JVM used by Integration Server, see the *webMethods Integration Server Administrator's Guide*.

To pass system properties to Microservices Runtime, you must update the `JAVA_CUSTOM_OPTS` property in `Integration Server_directory/bin/server.bat(sh)`. For more information about passing system properties to Microservices Runtime, see *Developing Microservices with webMethods Microservices Runtime*.

**Note:**

If Integration Server uses Java SE Development Kit 8, Update 111 or later and the Internet proxy is configured to enforce HTTPS Basic authentication, you may need to add the following system properties to the `custom_wrapper.conf`:

```
jdk.http.auth.proxying.disabledSchemes
jdk.http.auth.tunneling.disabledSchemes
```

## Configuring the Internet Proxy

When using a proxy server for outbound requests from on-premise Integration Server to webMethods Integration Cloud, you may need to configure the following on the proxy server:

- **Whitelist entries.** If the proxy server employs a whitelist of allowed URLs, you need to add the URL for webMethods Integration Cloud. A URL for the webMethods Integration Cloud uses the following format:

```
tenantId.webmethodscloud.com
tenantId.webmethodscloud.eu
```

Where *tenantId* is the unique ID assigned to your organization's tenancy on the platform.

For more information about specifying whitelist entries for an Internet proxy, refer to the documentation for the Internet proxy used by your company.

- Tunneling with HTTP CONNECT. To ensure end-to-end encryption for TLS over HTTPS, configure the Internet proxy to accept the HTTP CONNECT method. If the Internet proxy does not accept the HTTP CONNECT, a connection from the on-premise Integration Server to webMethods Integration Cloud through the proxy server can still be established but end-to-end encryption with TLS will not be provided.
- Long running HTTP connections. Configure the proxy server to be friendly to long running HTTP 1.0 and 1.1 connections. This may improve performance and limit latencies.