LIONBRIDGE

Lionbridge Connector for Episerver

Installation and Configuration Guide

Version 1.6.3

May 5th, 2021

Copyright

© Copyright 2021 Lionbridge Technologies, Inc. All rights reserved.

Lionbridge and the Lionbridge logotype are registered trademarks or trademarks of Lionbridge Technologies, Inc. in the United States and other countries. All other trademarks used herein are the property of their respective owners. The opinions of third-party contributors remain theirs alone and do not necessarily represent the views of Lionbridge Technologies.

Contents

1 Welcome to the Lionbridge Connector for Episerver ---- 5
1.1 Terminology ---- 5
1.2 About the Clay Tablet Platform ---- 7
1.3 How the Connector Works with Episerver ---- 7
1.4 Using this Guide ---- 8
1.5 How to Contact Lionbridge Connector Support ---- 8
1.5.1 Submitting a Support Ticket ---- 9
1.5.2 Information to Include in a Support Ticket ---- 9
1.5.3 Viewing and Updating Your Support Ticket in Zendesk ---- 10
1.5.4 Signing Up for a Zendesk Account for Lionbridge Connectors ---- 11
1.5.5 Getting a Zendesk Password if You Previously Emailed Lionbridge Connectors 13
2 Before You Install ---- 16
2.1 System Requirements ---- 16
2.2 Setting Your System Date, Time, and Time Zone Correctly ---- 16
2.3 Configuring Network Settings for a Firewall ---- 16
3 Installing the Lionbridge Connector ---- 18
3.1 Updating Your Connector Installation ---- 20
3.2 Uninstalling the Connector ---- 25
4 Configuring the Connector ---- 26
4.1 Adding the Lionbridge Connector Gadget ---- 26
4.2 Activating the Scheduled Job and Setting the Run Interval ---- 27
4.3 Activating the Translation Heartbeat Job ---- 28
4.4 Activating Background Jobs ---- 30
4.5 Configuring Target XML and Source XML ---- 31
4.6 Configuring Language Mapping and Asset Packaging ---- 33
4.7 Configuring Lionbridge Freeway Settings ---- 35
4.8 Configuring Logging when the Connector is Installed Locally ---- 37
4.9 Configuring Logging in DXC Environments ---- 39

4.10 Configuring Multiple Users and Translation Workflows ---- 41

5 Troubleshooting ---- 42

6 Appendix: Language Codes ---- 43

Index ---- 52

1 Welcome to the Lionbridge Connector for Episerver

Welcome to the Lionbridge Connector for Episerver (Connector). This is Lionbridge's connector between Episerver and the Clay Tablet Platform.

1.1 Terminology

Amazon AWS	Amazon Web Services. A suite of web application products developed and sold by Amazon.com. Clay Tablet uses various AWS offerings in order to leverage their infrastructure and build rich, dynamic solutions for its customers, specifically, the Clay Tablet Platform. For details, see http://aws.amazon.com.
Amazon S3	Amazon Simple Storage Service. For details, see: http://aws.amazon.com/s3/. The Connector and the Clay Tablet Platform use Amazon S3 to provide temporary storage services for the content sent to and from translation.
Amazon SQS	Amazon Simple Queue Service. For details, see: http://aws.amazon.com/sqs/. The Connector uses Amazon SQS to provide Message Queue Services.
Asset	Any content or document being sent for translation, including metadata. Assets are created by the Connector.
Clay Tablet (CTT)	Clay Tablet Technologies, a Lionbridge company, and the corporate entity that publishes the Connector and the Clay Tablet Platform.
Clay Tablet Platform	The hosted (IaaS) connectivity platform that receives and routes content from content systems, including content management systems (CMSs), to translation providers and back during implementation. The Lionbridge Connector Team configures the Platform based on the number and nature of systems involved in your system architecture.
Episerver	The Episerver CMS (Content Management System). For more information, see: http://www.episerver.com/
Freeway	The name of the Lionbridge translation portal for submitting content to and retrieving content from the Lionbridge translation provider.
FTP Server	File Transfer Protocol (FTP) is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. Translation providers may receive and send files for translation using an FTP server.
IaaS	Infrastructure as a Service. The Clay Tablet Platform is an IaaS, because it is a hosted platform.

Keys	The Connector uses keys to establish a secure, discrete connection between the Connector instance and the Platform. Very important: Do not copy the CMS address keys to multiple Episerver instances, because this is a violation of the License Agreement. Using the same CMS address keys on multiple Episerver instances will cause the Lionbridge App or Connector to behave unexpectedly, which can result in lost translation content, orphaned projects, and inaccurate translation status reports. The Lionbridge Connector team will support technical issues caused by duplicating or incorrectly installing CMS address keys only on a time and materials basis.
Lionbridge	The publisher of the connector, the Freeway translation portal and a translation provider. Users connect to the Freeway translation portal to submit content to and retrieve content from the Lionbridge translation provider.
Lionbridge Connector for Episerver (Connector)	The connector software that Lionbridge provides, which plugs into your Episerver installation to provide connectivity to the hosted Platform. In this document it is referred to as the Connector. This is the software you are installing and configuring as you work through this document.
MT	Machine translation. The translation provider can be a machine translation service, such as Google Translate.
NuGet	Open source package manager for the Microsoft development platform. For more information, see: http://www.nuget.org/
Producer	CMS or another content system that sends content or documents out for translation. In this case, this is Episerver.
Provider	A provider of translation services. The delivery of assets to the provider may be via an FTP server or a TMS connector.
Scheduled Job	Episerver functionality to run scheduled jobs. These can be run repeatedly with a specified interval or started manually.
Support Asset	Supporting documents and their metadata. Support assets are not translated by the translation provider, but they provide helpful context for the translator.
TMS	Translation management system that the translation provider uses.

1.2 About the Clay Tablet Platform

The Clay Tablet translation-connectivity platform is the easiest, most flexible way to integrate content systems, including content management systems (CMSs) and other content producers, with translation providers and translation technologies.

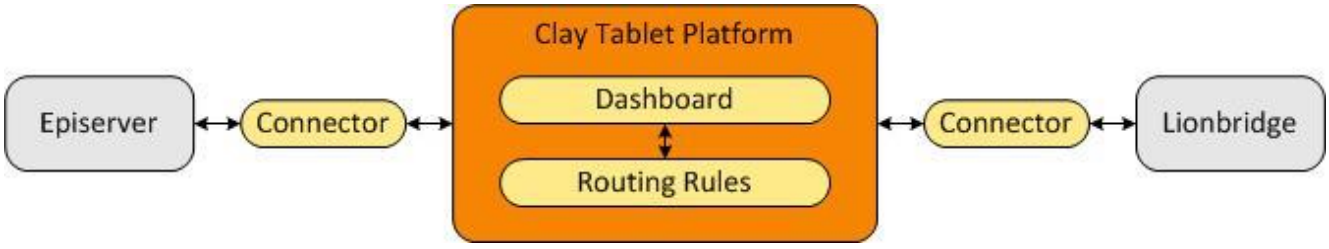
Clay Tablet Platform is the hosted (IaaS) connectivity platform that receives and routes content from content management systems to translation providers and back, including to Lionbridge via Freeway or onDemand. It is hosted on Amazon Web Services (AWS). During implementation, the Lionbridge Connector Team configures the Platform for your translation solution, based on the translation providers or systems you use. The Clay Tablet Platform uses the following services on AWS:

- S3 (Amazon Simple Storage Service), which provides storage services for the content sent to and from translation.
- SQS (Amazon Simple Queue Service), which provides message queue services.

1.3 How the Connector Works with Episerver

The Lionbridge Connector (Connector) is an important part of the Clay Tablet translation solution.

The Connector is installed on your system as an add-in to Episerver, through a NuGet installation package. Its functionality is displayed to the users as part of Episerver.



Your translation systems architecture might look like the configuration above. It may have additional content systems or translation providers, but the core concepts remain the same. During implementation, the Lionbridge Connectors team works with you and your translation providers to configure and test the other elements of your translation solution, which are the Clay Tablet Platform's connections to your translation providers' systems.

1.4 Using this Guide

Purpose of this guide

This guide describes everything you need to know to install and configure the Lionbridge Connector (Connector) for Episerver. It describes the delivery package contents, system requirements, installation instructions, and configuration procedures.
Recommendation: Review the user guide to fully understand the powerful features of the Connector.

Who should use this guide

This guide is intended for Episerver administrators and system integrators.

What you should already know

This document assumes that your company already has an installed instance of Episerver. It also assumes that Lionbridge is your company's translation provider, and the Clay Tablet Platform is already set up for your company. It assumes that you have a strong working knowledge of Episerver.

How to find out more about the Lionbridge Connector for Episerver

For information on using the Lionbridge Connector to send and receive content for translation from Episerver, read the *Lionbridge Connector for Episerver User Guide*.

Documentation conventions

This guide uses the following conventions:

Convention	Description
Bold	Highlights screen elements such as buttons, menu items, and fields.
`Courier`	Highlights input, file names, and paths.
Italics	Highlights terms for emphasis, variables, or document titles.
>	Indicates a menu choice. For example, "Select **Edit> Select All**."

1.5 How to Contact Lionbridge Connector Support

Email @: connectors@lionbridge.com

Telephone: +1-416-363-0888

You can submit a support ticket either:

- by email
- from the Lionbridge Connector Zendesk page, using your web browser

1.5.1 Submitting a Support Ticket

1. Do one of the following:
   - Email connectors@lionbridge.com, and cc (carbon copy) anyone to include in the ticket correspondence.

     Important: Include the information and attachments in your email that are listed in the sub-sections below.
   - Create a ticket in Zendesk:
     a. Open the Lionbridge Connector Zendesk page in your browser: https://connectors.zendesk.com.
     b. Sign in to Zendesk.

        Note: If you do not have sign-in credentials yet, then click either **Sign up** or **Get a password**, and follow the onscreen instructions.

        Important: Include the information and attachments that are listed in the sub-sections below.
     c. Click **Submit a request**.
     d. In the **CCs** field, add anyone to include in the ticket correspondence.

   Zendesk automatically creates a ticket and responds to everyone included in the cc field.
2. Everyone in the original cc receives updates unless they request to be removed from the ticket.

   Important: Check your email spam folder (especially first-time Zendesk users) as sometimes email notifications from Zendesk are marked as spam.

When the issue is resolved, Lionbridge closes the ticket.

1.5.2 Information to Include in a Support Ticket

- client name
- CMS or content system name and version
- Connector or App version installed
- name of job for which the issue occurs

- date of job submission
- detailed description of the issue
- any error text–copy and paste, if applicable

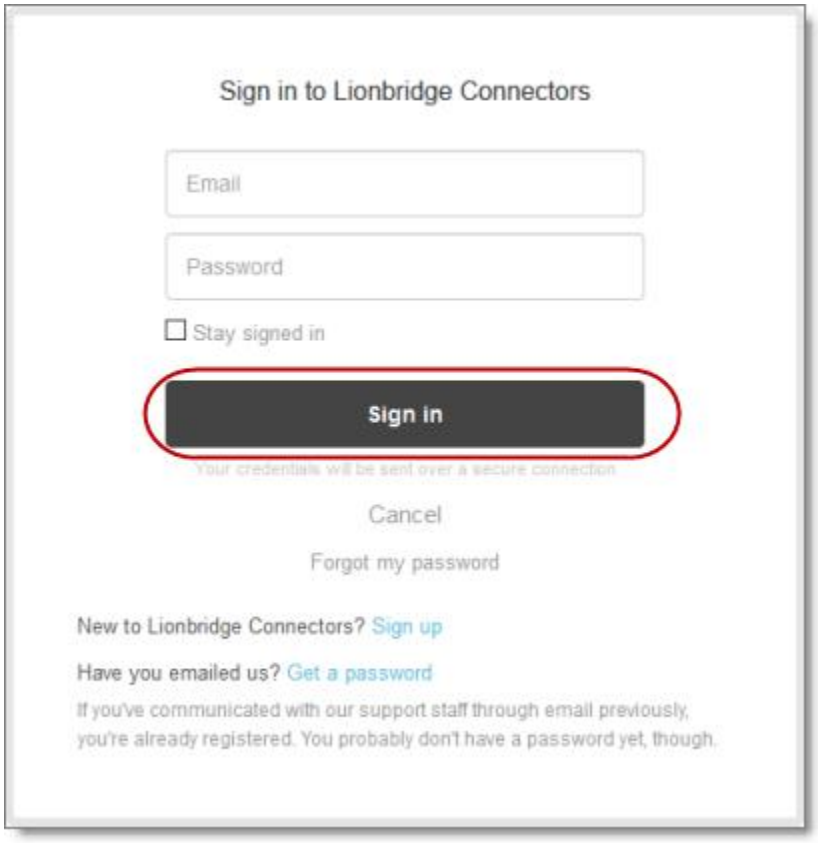
Files to attach to the support ticket:

- CMS log files for the date the issue occurred
- Connector or App log files for the date the issue
- occurred screen capture of the issue

1.5.3 Viewing and Updating Your Support Ticket in Zendesk

Important: You must log into Zendesk to view your support tickets there.

1. Open the Lionbridge Connector Zendesk page in your browser: https://connectors.zendesk.com.
2. Enter your credentials, and click **Sign in**.



Note: If you do not have sign-in credentials yet, then click either **Sign up** or **Get a password**, and follow the onscreen instructions.

3. After signing in, click **My activities** to view the tickets you opened or where you are cc'd.
4. To update tickets, you can reply or attach files.

For more information, refer to "Submitting and tracking support requests" in Zendesk's *Help Center guide for end-users*, at: https://support.zendesk.com/hc/en-us/articles/203664386-Help-Center-guide-for-agents-and-end-users.

Important: Zendesk refers to a *support ticket* as a *support request*. These terms are interchangeable.

1.5.4 Signing Up for a Zendesk Account for Lionbridge Connectors

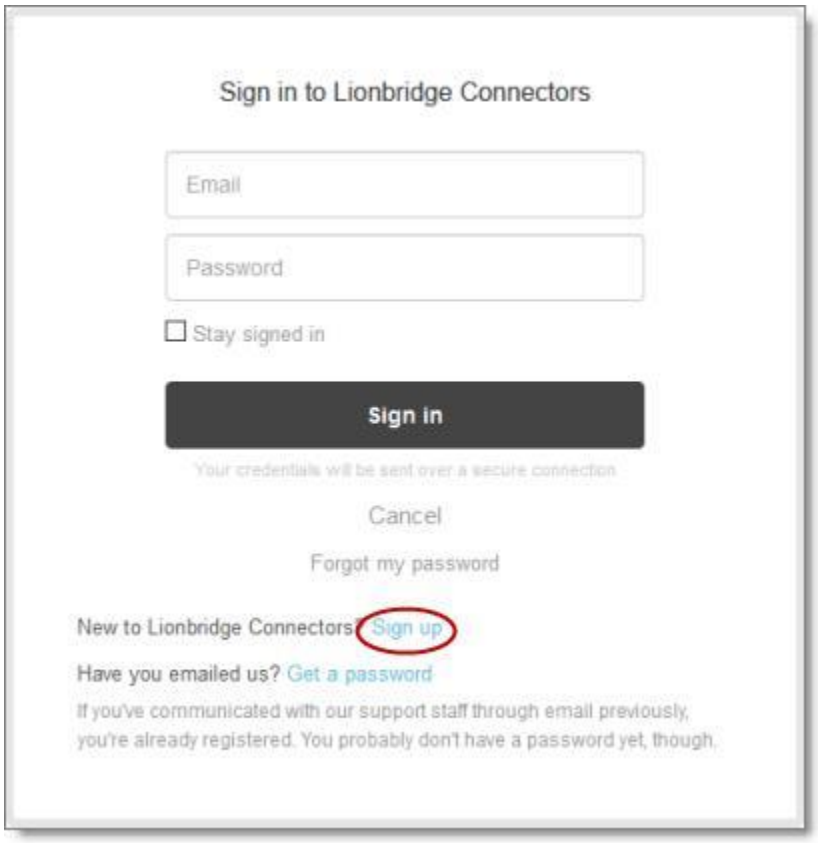
You can create a new Zendesk account for Lionbridge Connectors.

Note: If you have previously emailed Lionbridge Connectors Support at connectors@lionbridge.com to create a support ticket, you can get a password for your email account. For detailed instructions, see "How to Get a Password if You Have Previously Emailed Lionbridge Connectors."

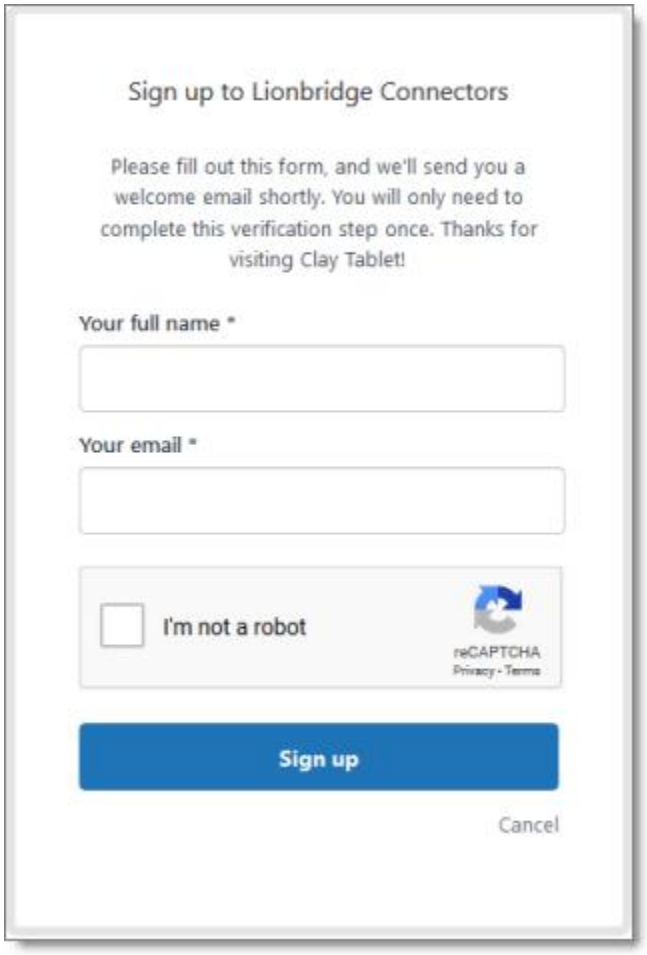
To sign up for a Zendesk account:

1. Open the Lionbridge Connector Zendesk page in your browser: https://connectors.zendesk.com.

2. Click **Sign up** link.



The **Sign up** page opens.



3. Enter your name and email address, and select the **I'm not a robot** check box.
4. Click **Sign up**.

   The **Sign-up complete** page opens. You will receive a verification email shortly with a verification link that enables you to sign in. If you do not receive an email within a few minutes, please check your junk or spam folder.

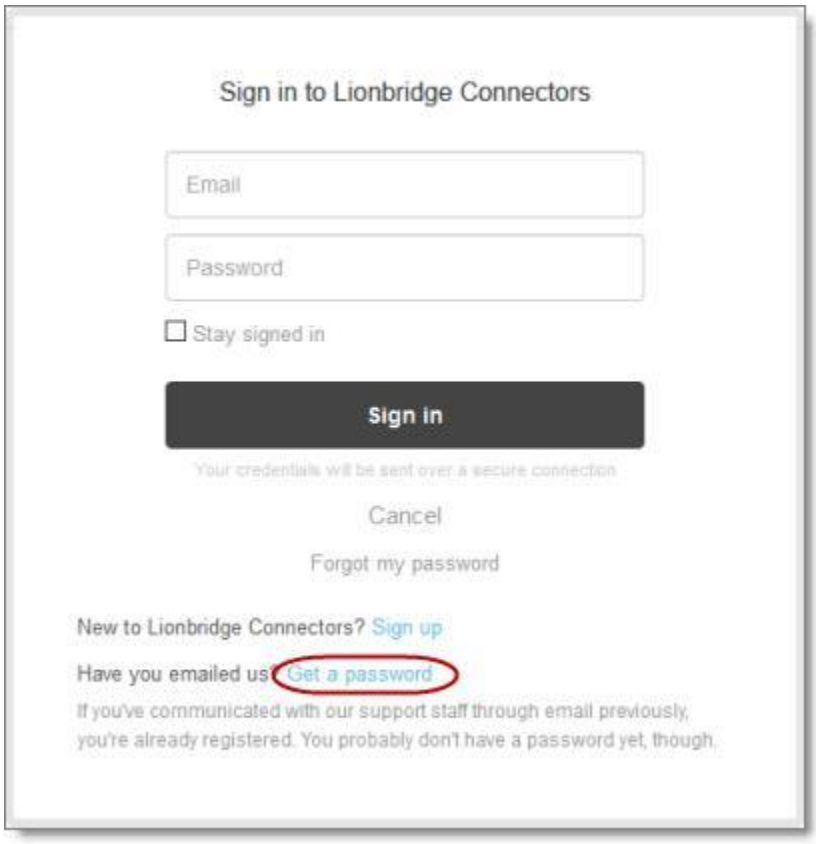
5. Click the link in the verification email to create a password and sign into Zendesk.

1.5.5 Getting a Zendesk Password if You Previously Emailed Lionbridge Connectors

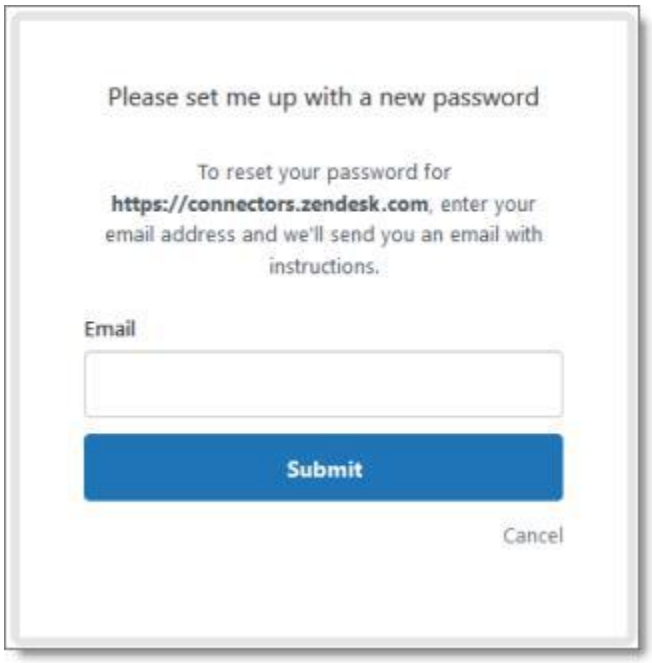
If you have previously emailed Lionbridge Connectors Support at connectors@lionbridge.com to create a support ticket, you can get a password for your email account.

To get a password:

1. Open the Lionbridge Connector Zendesk page in your browser: https://connectors.zendesk.com.
2. Click the **Get a password** link.



The **Please set me up with a new password** page opens.



3. Enter the email address from which you emailed Lionbridge Connectors Support (connectors@lionbridge.com).
4. Click **Submit**.

   Zendesk sends you an email with instructions for creating your password.
5. Follow the instructions in the email from Zendesk to create your password. You can now sign in to Zendesk to create, view and update your support tickets.

2 Before You Install

Before you begin to install the Lionbridge Connector (Connector) for Episerver, please review the system requirements, described below, and perform the following pre-installation procedures:

- Back up your Episerver database.
- If you received the Connector as a NuGet package (with file extension `.nupkg`) directly from Lionbridge, ensure that it is saved to a location that is accessible during the installation process. Alternatively, you can download the package directly from the Episerver NuGet feed, using the NuGet Package Manager Console. For details, see "Installing the Lionbridge Connector" on page 18.

2.1 System Requirements

The Lionbridge Connector for Episerver supports Episerver versions 10 and 11.

The Lionbridge Connector for Episerver has no additional hardware or software requirements beyond those of Episerver. For detailed requirements, refer to the Episerver documentation, available at: http://world.episerver.com/documentation.

2.2 Setting Your System Date, Time, and Time Zone Correctly

The Connector sends content to and receives content from the Clay Tablet Platform, which is hosted in the Amazon Web Services (AWS) environment. AWS requires any machines that connect to its applications to have the correct system time and date settings.

Important: Before proceeding, ensure that the system date, time, and time zone are set correctly on any systems that will run the Connector. If these settings are incorrect, the following error message is displayed: `Error. The difference between the request time and the current time is too large.`

2.3 Configuring Network Settings for a Firewall

Optional step. If you have a firewall, you must configure your ports so that the Connector can communicate with the Clay Tablet Platform. The Connector must be able to communicate with the Clay Tablet Platform by initiating the following outbound network connections:

Protocol	Port Number	Description	Location to Configure
HTTPS	Port 443	For secure access to the Clay Tablet License Server	https://api.clay-tablet.net/license/v1
HTTPS	Port 443	For secure access to Amazon's AWS S3 service	*.s3.amazonaws.com/*
HTTPS	Port 443	For secure access to Amazon's AWS SQS service	https://queue.amazonaws.com/*
HTTPS	Port 443	For secure access to Amazon's AWS SQS service	https://sqs.us-east-1.amazonaws.com/*

To send content for translation to Lionbridge Freeway, also configure:

Protocol	Port Number	Description	Location to Configure
HTTPS	Port 443	For secure access to Lionbridge Freeway	https://fwapi.lionbridge.com/obvibundles/freewayauth.asmx
HTTPS	Port 443	For secure access to Lionbridge Freeway	https://fwapi.lionbridge.com/obvibundles/service.asmx

3 Installing the Lionbridge Connector

Important: Before installing the Connector, back up the Episerver databases.

You use Visual Studio NuGet to install the Connector package. There are two ways to install the Connector, depending on whether you received the installation package directly from Lionbridge or you will download it from the Episerver NuGet feed.

Episerver version numbers in Lionbridge Connector packages

Important: There are different Connector packages for Episerver 10 and Episerver 11. Ensure that you download and install the correct package for your version of Episerver.

You can identify the Episerver version for the Lionbridge Connector from the last four digits of the NuGet package name, as described in the following table below.

Episerver Version	**Lionbridge Connector Versioning (MajorVersion.MinorVersion.Patch.Episerver Version)**	**Example**
Episerver 10	1.x.x.1000	1.5.0.1000
Episerver 11	1.x.x.1100	1.5.0.1100

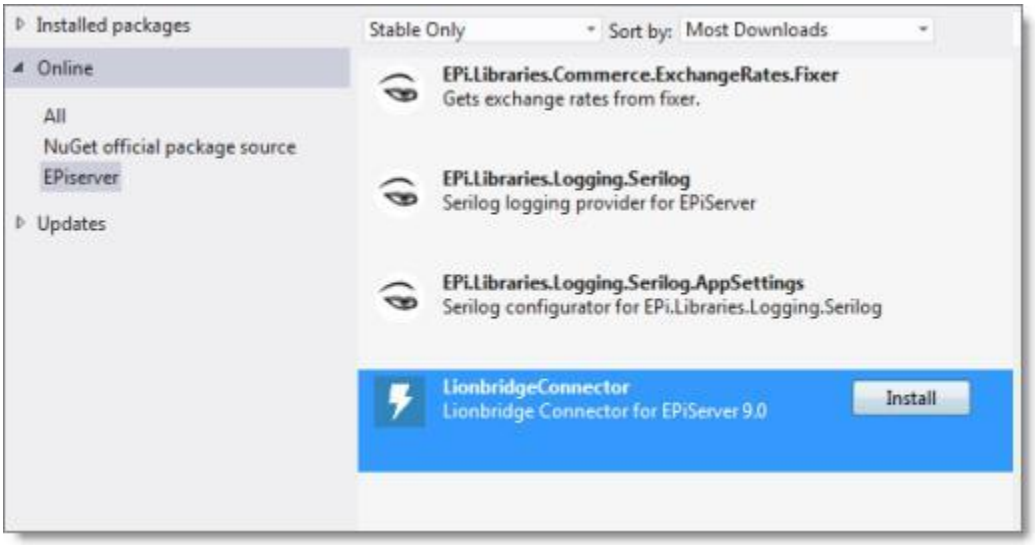
To download and install the Connector:

Important: Follow these instructions if you *did not* receive the Connector as a NuGet package (with file extension `.nupkg`) directly from Lionbridge.

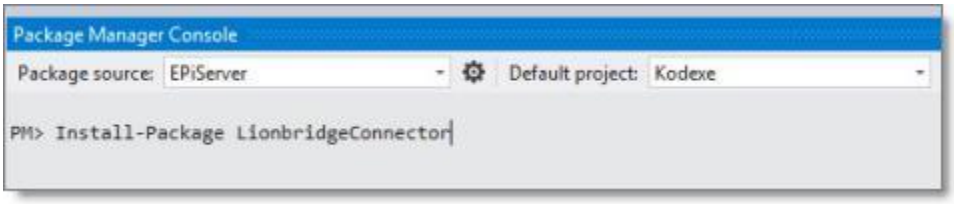
1. In Visual Studio, open the NuGet Package Manager Console.
2. Do one of the following:

3 Installing the Lionbridge Connector

- In the left pane, select **Online > Episerver**. Then select the **LionbridgeConnector** package, and click **Install**.



- In the command line, type: `Install-Package LionbridgeConnector`.



To install the Connector:

Important: Follow these instructions if you received the Connector as a NuGet package (with file extension `.nupkg`) directly from Lionbridge.

1. In Visual Studio, open the NuGet Package Manager Console.
2. Type the following command in the command line, ensuring that you specify the latest Connector version for your CMS version:
   `Install-Package LionbridgeConnector -source [.nupkg file path] -version.x.x.1100`
   For example, to install version 1.5.0 of the Connector on CMS version 11, type:
   `Install-Package LionbridgeConnector -source C:\LionbridgeConnector -version 1.5.0.1100`

Installation location

The Connector is installed in the `\modules\_protected` folder of the Episerver web root. If you use another folder for add-ons, you must manually move the Connector files to your modules folder.

3.1 Updating Your Connector Installation

If the Lionbridge Connector (Connector) is already installed, follow the steps below to update your version of the Connector. The method and commands for updating your Connector depend on:

- your current Connector version
- your Episerver version
- whether you are updating your Episerver version
- whether you are using the Nuget site or the Nuget package to perform the updates

Refer to the tables below, and use the appropriate procedure for your particular scenario.

Using the Nuget site:

1. In Visual Studio, open the NuGet Package Manager Console.
2. In the command line, type: `Update-Package LionbridgeConnector -version` *`MajorVersion.MinorVersion.Patch.EpiserverVersion`*.
   For example, type: `Update-Package LionbridgeConnector -version 1.5.0.1100`

For detailed instructions, refer to the table below. Always use the latest available version of the Connector.

Using the NuGet package (with file extension .nupkg) directly from Lionbridge:

1. In Visual Studio, open the NuGet Package Manager Console.
2. In the command line, type: `Update-Package LionbridgeConnector -source.` `[`*`.nupkg file path`*`] -version` *`MajorVersion.Minor Version.Patch.EpiserverVersion`*.
   For example, type: `Update-Package LionbridgeConnector -source. [c:\LionbridgeConnector] -version 1.5.0.1100`

For detailed information about version numbers, refer to the following table. Always use the latest version.

Episerver version numbers in Lionbridge Connector packages

You can identify the Episerver version for the Lionbridge Connector from the last four digits of the NuGet package name, as described in the following table below.

Episerver Version	Lionbridge Connector Versioning (MajorVersion.MinorVersion.Patch.Episerver Version)	Example
Episerver 10	1.x.x.1000	1.5.0.1000
Episerver 11	1.x.x.1100	1.5.0.1100

When updating the Connector from Episerver 9

Episerver 9 Scenarios	From	To	Instructions
Updating CMS and Connector from Episerver 9 to Episerver 10	1.2.4.0900	1.5.0.1000 or higher	1. Update Episerver CMS to version 10. 2. Update to the latest version of the Connector supporting Episerver 10. 3. Specify the exact name of the latest version of the Connector, for example: ▪ NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1000` ▪ NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\... \ - version 1.5.0.1000` 4. Run a `.sql` script that you obtain from the Lionbridge Connector Team.

Episerver 9 Scenarios	From	To	Instructions
Updating CMS and Connector from Episerver 9 to Episerver 11	1.2.4.0900	1.5.0.1100 or higher	1. Update Episerver CMS to version 11. 2. Update to the latest version of the Connector supporting Episerver 11. 3. Specify the exact name of the latest version of the Connector, for example: ▪ NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1100` ▪ NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\... \ - version 1.5.0.1100` 4. Run a `.sql` script that you obtain from the Lionbridge Connector Team.

When updating the Connector from Episerver 10

Episerver 10 Scenarios	From	To	Instructions
Updating Connector on Episerver 10	1.3.x.1000	1.5.0.1000 or higher	1. Update to the latest version of the Connector supporting Episerver 10. 2. Specify the exact name of the latest version of the Connector, for example: • NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1000` • NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\...\ - version 1.5.0.1000` 3. Run a `.sql` script that you obtain from the Lionbridge Connector Team.
Updating Connector on Episerver 10	1.3.9.1000	1.5.0.1000 or higher	1. Update to the latest version of the Connector supporting Episerver 10. 2. Specify the exact name of the latest version of the Connector, for example: • NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1000` • NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\... \ - version 1.5.0.1000`

Episerver 10 Scenarios	From	To	Instructions
Updating CMS and Connector from Episerver 10 to Episerver 11	1.3.x.1000	1.5.0.1100 or higher	1. Update the Episerver CMS packages to version 11. 2. Update to the latest version of the Connector supporting Episerver 11. 3. Specify the exact name of the latest version of the Connector, for example: ▪ NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1100` ▪ NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\...\ - version 1.5.0.1100` 4. Run a `.sql` script that you obtain from the Lionbridge Connector Team.
Updating CMS and Connector from Episerver 10 to Episerver 11	1.3.9.1000 or higher	1.5.0.1100 or higher	1. Update the Episerver CMS packages to version 11. 2. Update to the latest version of the Connector supporting Episerver 11. 3. Specify the exact name of the latest version of the Connector, for example: ▪ NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1100` ▪ NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\... \ - version 1.5.0.1100`

When updating the Connector from Episerver 11

Episerver 11 Scenarios	From	To	Instructions
Updating Connector on Episerver 11	1.4.2.1100	1.5.0.1100 or higher	1. Update the Connector. 2. Specify the exact name of the latest version of the Connector, for example: • NuGet Site Package: `PM> Update-Package LionbridgeConnector -version 1.5.0.1100` • NuGet file Package: `PM> Update-Package LionbridgeConnector -source C:\...\ - version 1.5.0.1100`

3.2 Uninstalling the Connector

Before uninstalling the Connector, ensure that the site is *stopped* in Visual Studio.

To uninstall the Connector:

1. In Visual Studio, open the NuGet Package Manager Console.
2. In the command line, type: `Remove-LionbridgeConnector`.

4 Configuring the Connector

4 Configuring the Connector

After installing the Connector, you configure the following settings:

1. "Adding the Lionbridge Connector Gadget" on page 26.
2. "Activating the Scheduled Job and Setting the Run Interval" on page 27.
3. "Activating the Translation Heartbeat Job" on page 28. Optional.
4. "Activating Background Jobs" on page 30.
5. "Configuring Target XML and Source XML" on page 31.
6. "Configuring Language Mapping and Asset Packaging" on page 33.
7. "Configuring Lionbridge Freeway Settings" on page 35.
8. Logging settings. See either:
   - "Configuring Logging when the Connector is Installed Locally" on page 37.
   - "Configuring Logging in DXC Environments" on page 39.
9. Optional. "Configuring Multiple Users and Translation Workflows" on page 41.

Important: While testing your installation, back up the Episerver database before sending out a project for translation. After confirming that your installation is stable, back up the database daily.

4.1 Adding the Lionbridge Connector Gadget

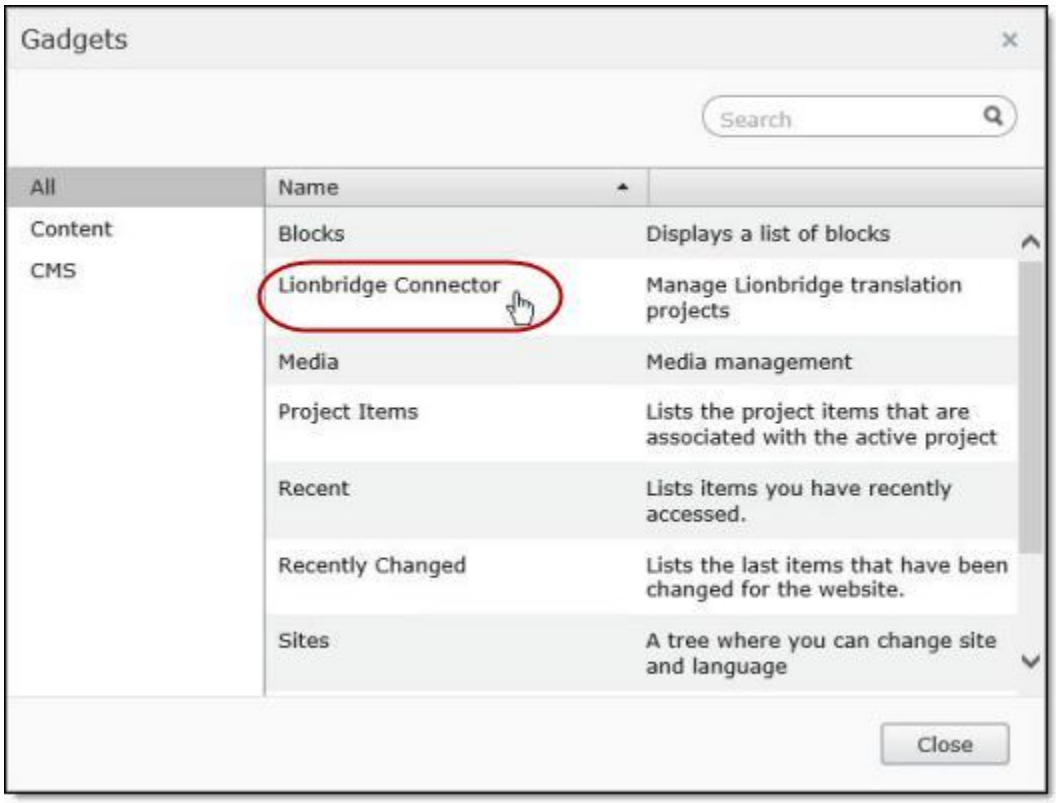
1. In Episerver **CMS**, click **Edit**.

2. Click the **Toggle assets pane** icon.

The **Assets** pane is displayed on the right.

3. In the **Assets** pane, click the Settings icon , and then select **Add Gadgets** from the context menu.

The **Gadgets** window opens.



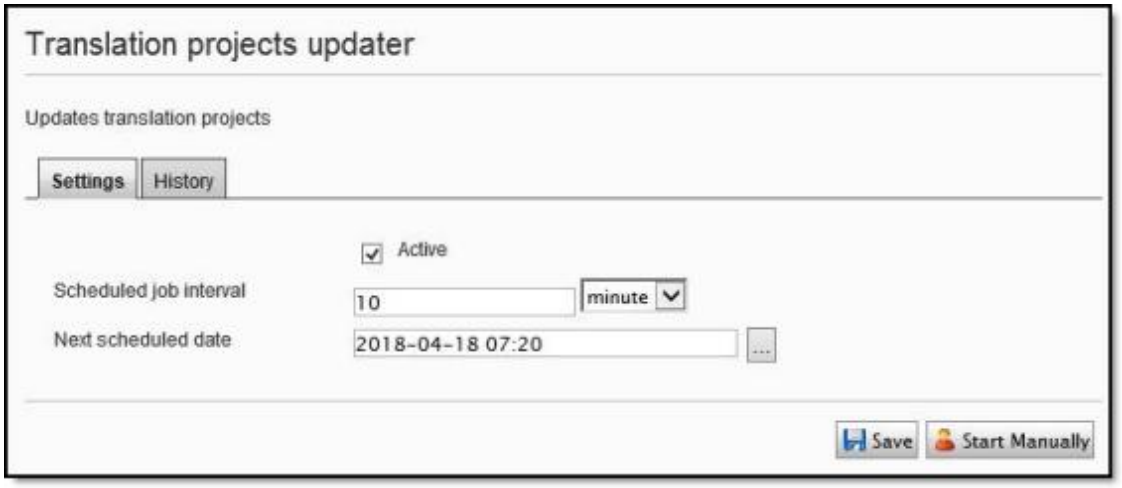
4. Click **Lionbridge Connector**.

4.2 Activating the Scheduled Job and Setting the Run Interval

1. In Episerver **CMS**, click **Admin**.

2. In the left pane, in the **Scheduled Jobs** section, click **Translation projects updater**.

The **Translation projects updater** page opens, displaying the Connector scheduled job.



3. Select the **Active** check box to activate the job.
4. Specify the **Scheduled job interval**.
5. Click **Save**.

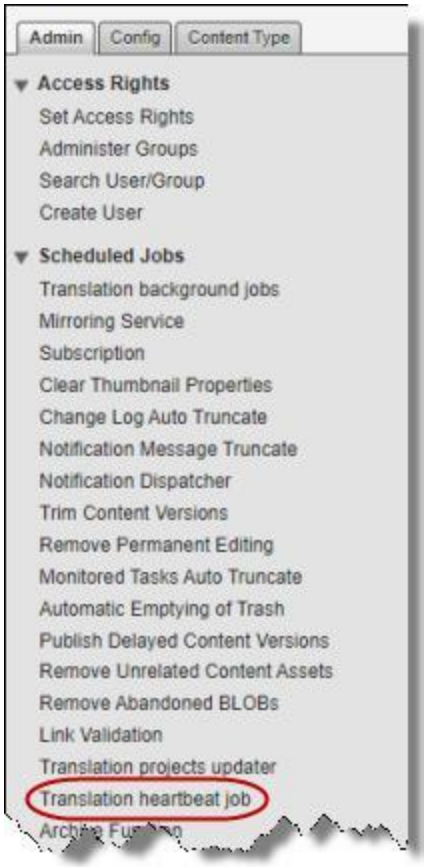
4.3 Activating the Translation Heartbeat Job

You can send a heartbeat message to the Clay Tablet Platform. This enables it to monitor the health of the Connector. To send this heartbeat message you activate and configure the **Translation heartbeat job**.

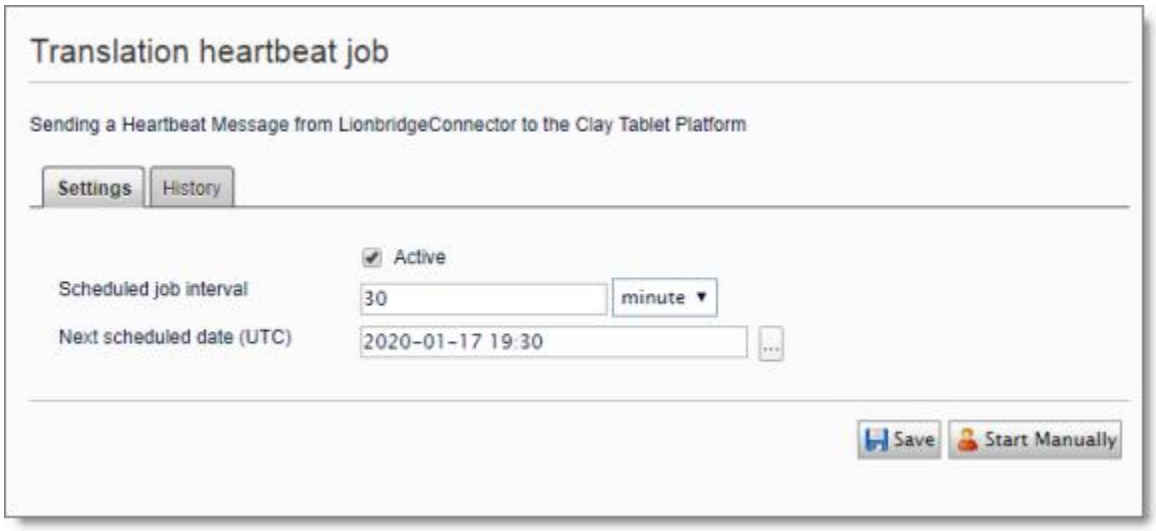
To activate the Translation heartbeat job:

1. In Episerver **CMS**, click **Admin**.

2. In the left pane, in the **Scheduled Jobs** section, click **Translation heartbeat job**.



The **Translation heartbeat job** page opens.



3. Select the **Active** check box to activate the job.
4. Specify the **Scheduled job interval**.
5. Click **Save**.

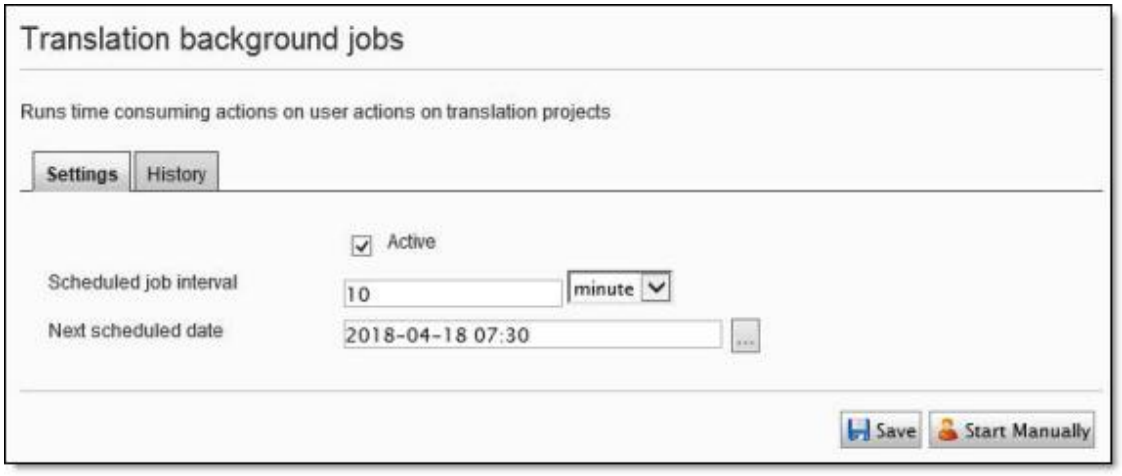
Tip: To view a list of heartbeat messages sent to the Clay Tablet Platform, including dates, times, and status, click the **History** tab.

4.4 Activating Background Jobs

1. In Episerver **CMS**, click **Admin**.

2. In the left pane, in the **Scheduled Jobs** section, click **Translation background jobs**.

The **Translation background jobs** page opens.



3. Select the **Active** check box.
4. Specify the **Scheduled job interval**, for example, 30 minutes.
5. Click **Save**.

4.5 Configuring Target XML and Source XML

Before you can use the Connector, you must configure `source.xml` and `target.xml`. These are available with your Lionbridge license. Ensure that you have this information before proceeding.

- The Connector uses the CMS address key, which is in `source.xml`, to move the files your company exports from Episerver to the Clay Tablet Platform. The Platform then forwards your exported files to your translation provider. If you have multiple Episerver installations, there must be one key (file) for each environment. Very important: Do not copy the CMS address keys to

multiple Episerver instances, because this is a violation of the License Agreement. Using the same CMS address keys on multiple Episerver instances will cause the Lionbridge App or Connector to behave unexpectedly, which can result in lost translation content, orphaned projects, and inaccurate translation status reports. The Lionbridge Connector team will support technical issues caused by duplicating or incorrectly installing CMS address keys only on a time and materials basis.

Note: The Connector always initiates calls to the Clay Tablet Platform. However, the CMS address keys enable establishing a secure, discrete connection between the Connector instance and the Platform. They also support the return of files from translation.

- Your company's platform key is in `target.xml`. This is your company's license for the Clay Tablet Platform.

To configure `source.xml` and `target.xml`:

1. In the bottom-right corner of the Lionbridge Connector gadget in the right pane, click the Settings icon , and then select **Settings** from the context menu.

2. In the **Workspace settings** tab, edit the **Target XML** and **Source XML** fields.

Episerver automatically saves your changes.

4.6 Configuring Language Mapping and Asset Packaging

In the **Workspace settings** tab:

- You map Episerver languages to Clay Tablet languages in the `language mapping.xml` file.
- You configure how the Connector packages multiple assets in a translation project.

To configure language mappings and grouped-asset handling:

1. In the bottom-right corner of the Lionbridge Connector gadget in the right pane, click the Settings icon , and then select **Settings** from the context menu.

2. In the **Workspace settings** tab, edit the **Language Mapping XML** field to map each Episerver language code to the corresponding Clay Tablet language code. For a list and description of Clay Tablet language codes, see "Appendix: Language Codes" on page 43.
   For example:

```
<Languages>
  <Language>
    <EPiServer>sv</EPiServer>
    <ClayTablet>sv-SE</ClayTablet>
  </Language>
  <Language>
    <EPiServer>da</EPiServer>
    <ClayTablet>da-DK</ClayTablet>
  </Language>
</Languages>
```

   Note: If you leave this section empty, the Connector will use the default values.

3. In the **Workspace settings** tab, you can configure how the Connector packages multiple assets in a translation project.

- If you *select* the **Use grouped asset handling** check box, then the Connector packages all pages and blocks selected for translation into a single `XML` file for each target language.

  Recommendation: Select this check box, especially if Lionbridge Freeway will translate terms for SEO.
- If you *clear* the **Use grouped asset handling** check box, then the Connector sends each page and block selected for translation as a separate `XML` file.

Episerver automatically saves your changes.

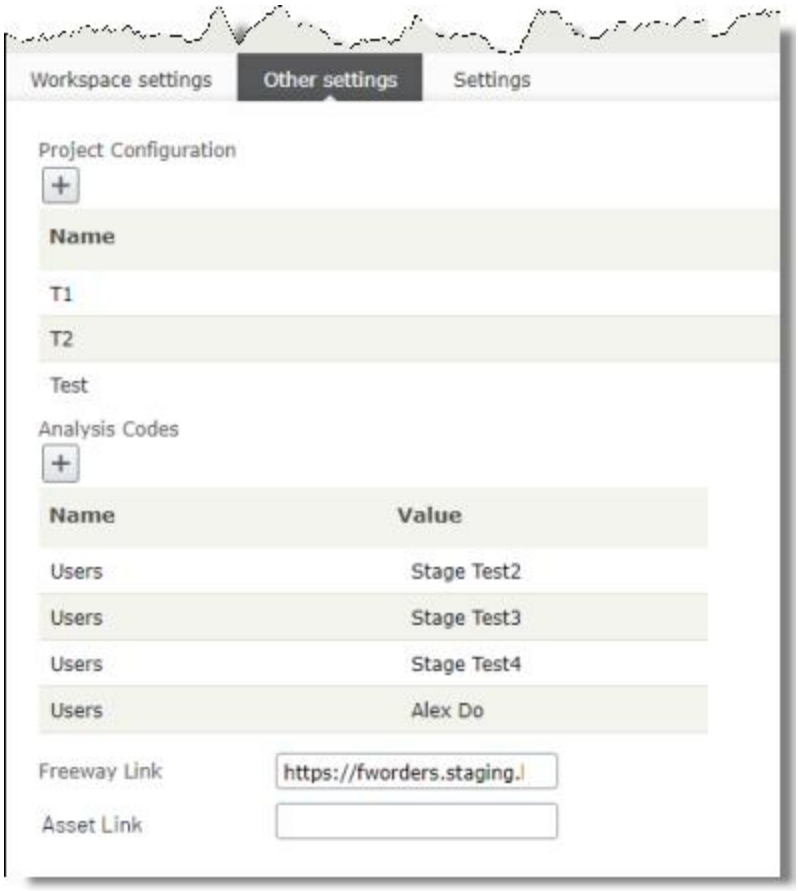
4.7 Configuring Lionbridge Freeway Settings

In the **Other settings** tab, you can configure Lionbridge Freeway-specific settings.

To configure settings for integration with Lionbridge Freeway:

1. In the bottom-right corner of the Lionbridge Connector gadget in the right pane, click the Settings icon , and then select **Settings** from the context menu.

2. Click the **Other settings** tab.



3. You can configure the following settings:

Setting	Description
Project Configuration	The name of a workflow configuration. There may be multiple workflow configurations for your translation provider (Lionbridge Freeway) that are set up by the Lionbridge Connector Team. For example, there may be one workflow for human translation and another configuration for machine translation. • Click the icon to open a dialog box where you can create the name of a workflow configuration. Enter the name of the configuration in the **Name** field, and then click **Add**. After setup by the Lionbridge Connector Team, this configuration will be available for selection in the **Project Configuration** dropdown list, which is displayed when creating or editing a translation project.
Analysis Codes	The name of a Lionbridge Freeway analysis code. • Click the icon to open a dialog box where you add an analysis code. Enter the name in the **Name** field and enter its value in the **Value** field, and then click **Add**. These analysis codes will be available for selection in the **Analysis Codes** section that is displayed when creating or editing a translation project. Important: You must contact your Lionbridge project manager to ensure that these analysis codes are in Freeway.
Freeway Link	The link to the Lionbridge Freeway bundle ID. This will be provided by the Lionbridge Connector Team. There will be different links for staging and production environments.
Asset Link	Optional. You can customize the links displayed in the **Link** column in the **Lionbridge Connector Dashboard** to display a non-default domain. Enter the non-default domain, for example: `http://example.com`. Recommendation: Configure this if in the **Admin** tab, on the **Manage Websites** page, the URL configured for the public website (front-end URL) is different than the domain of the environment (back-end URL). This may cause the Dashboard to display errors when loading, or prevent links from being displayed.

Episerver automatically saves your changes.

4.8 Configuring Logging when the Connector is Installed Locally

Note: If the Connector is installed in a DXC cloud environment, see page 39.

In an environment where the Connector is *installed locally*, you can activate full verbose logging for it. This is useful for identifying errors.

To configure logging:

1. Append the following to `EPiServerLog.config`:

```
<appender name="lionbridgeFileLogAppender"
type="log4net.Appender.RollingFileAppender" >
  <!-- Consider moving the log files to a location outside the
  web application -->
  <file value="App_Data\LionbridgeErrors.log" />
  <encoding value="utf-8" /> <staticLogFileName
  value="true"/> <datePattern
  value=".yyyyMMdd.'log'" /> <rollingStyle
  value="Date" /> <threshold value="Verbose" />

  <lockingModel type="log4net.Appender.FileAppender+MinimalLock"
  />
  <appendToFile value="true" />
  <layout type="log4net.Layout.PatternLayout">
    <conversionPattern value="%date [%thread] %level %logger:
    %message%n" />
  </layout>
</appender>
```

2. Add the following logger:

```
</logger>
  <logger name="Lionbridge" additivity="false">
  <level value="Verbose" />
  <appender-ref ref="lionbridgeFileLogAppender" />
</logger>
```

The generated log file is: `\App_Data\LionbridgeErrors.log`.

4.9 Configuring Logging in DXC Environments

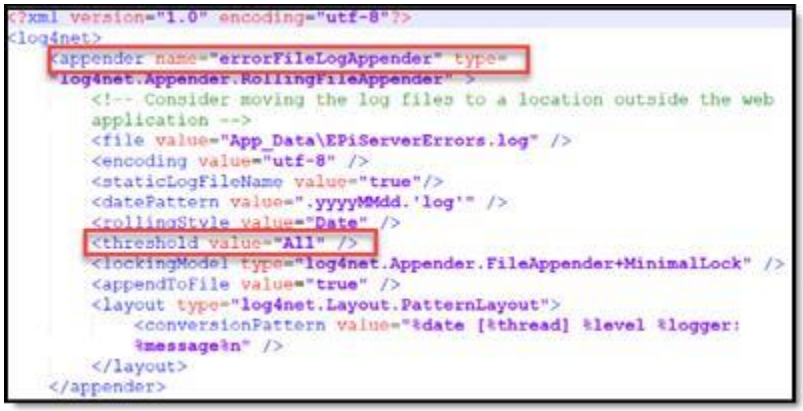
Note: If the Connector is *installed locally*, see page 37.

If the Connector is installed in a DXC cloud environment, you must make some changes to access your Connector log files.

To access Connector log files in a DXC environment:

1. Open `EPiServerLog.config` for editing.

   Under the `<appender name="errorFileLogAppender"` section, change the `threshold value` to `"All"`.



2. Under the `<!-- Lionbridge logger-->` section:
   - Change `level value` to `"All"`.
   - Change `appender-ref ref` to `"errorFileLogAppender"`.

3. Under the`<root>`section, change `level value` to `"Error"`.

This will retrieve all the Connector log messages, while retrieving only error-level log messages for Episerver. If you require a different level of Episerver log messages, then revert your changes after producing the Connector log files required for troubleshooting.

Tip: To obtain an `EPiServerLog.config` file with these changes already implemented, contact Lionbridge Connector Support.

Sample modified file

```
<?xml version="1.0" encoding="utf-8"?>
<log4net>
    <appender name="errorFileLogAppender" type=
    "log4net.Appender.RollingFileAppender" >
        <!-- Consider moving the log files to a location outside the web
        application -->
        <file value="App_Data\EPiServerErrors.log" />
        <encoding value="utf-8" />
        <staticLogFileName value="true"/>
        <datePattern value=".yyyyMMdd.'log'" />
        <rollingStyle value="Date" />
        <threshold value="All" />
        <lockingModel type="log4net.Appender.FileAppender+MinimalLock" />
        <appendToFile value="true" />
        <layout type="log4net.Layout.PatternLayout">
            <conversionPattern value="%date [%thread] %level %logger:
            %message%n" />
        </layout>
    </appender>
    <appender name="outputDebugStringAppender" type=
    "log4net.Appender.OutputDebugStringAppender" >
        <layout type="log4net.Layout.PatternLayout">
            <conversionPattern value="[%thread] %level %logger: %message%n" />
        </layout>
    </appender>

    <!-- Lionbridge appender-->
    <appender name="lionbridgeFileLogAppender" type=
    "log4net.Appender.RollingFileAppender" >
        <file value="App_Data\LionbridgeErrors.log" />
        <encoding value="utf-8" />
        <staticLogFileName value="true"/>
        <datePattern value=".yyyyMMdd.'log'" />
        <rollingStyle value="Date" />
        <threshold value="Verbose" />
        <lockingModel type="log4net.Appender.FileAppender+MinimalLock" />
        <appendToFile value="true" />
        <layout type="log4net.Layout.PatternLayout">
            <conversionPattern value="%date [%thread] %level %logger:
            %message%n" />
        </layout>
    </appender>

    <!-- Reduce amount of debug logging -->
    <logger name="EPiServer.Core.OptimisticCache" additivity="false">
        <level value="Error" />
    </logger>
    <logger name="EPiServer.Core.ContentProvider" additivity="false">
        <level value="Error" />
    </logger>
    <logger name="EPiServer.Data.Dynamic.Providers.DbDataStoreProvider"
    additivity="false">
        <level value="Error" />
    </logger>
    <logger name="EPiServer.Data.Providers.SqlDatabaseHandler" additivity=
    "false">
        <level value="Error" />
    </logger>
    <logger name="EPiServer.Data.Providers.ConnectionContext" additivity=
    "false">
        <level value="Error" />
    </logger>

    <!-- Lionbridge logger-->
    <logger name="Lionbridge" additivity="false">
        <level value="All" />
        <appender-ref ref="errorFileLogAppender" />
    </logger>
    <root>
        <!--Be careful, setting this value to All, Debug or Info will affect
        performance.-->
        <level value="Error" />

        <!--Enabled file logging-->
        <appender-ref ref="errorFileLogAppender" />

        <!-- Use DebugView (Microsoft/SysInternals) and enable Capture Win32
        + Capture Global Win32 (
        http://technet.microsoft.com/en-us/sysinternals/bb896647). -->
        <!--appender-ref ref="outputDebugStringAppender" /-->
    </root>
</log4net>
```

4.10 Configuring Multiple Users and Translation Workflows

The Connector supports multiple users. This enables configuring the Connector so that the creator of a translation project determines which Lionbridge Freeway account is assigned to that project. This requires setup in Freeway as well. To enable this configuration, contact the Lionbridge Connector Team and your Lionbridge project manager.

5 Troubleshooting

Problem	Explanation/Solution
Translation projects are stuck at 0% after sending them for translation.	Ensure that the scheduled job has been activated. For detailed instructions, see "Activating the Scheduled Job and Setting the Run Interval" on page 27.
The Connector does not send out custom properties for translation.	The Connector supports sending for translation only custom properties that inherit from the `ProperyLongString` and `PropertyString` classes.
Lionbridge Freeway analysis codes are specified, but they are not displayed or available for selection when sending out projects for translation.	The URLs where Lionbridge Freeway provides the authentication service and the analysis code service are not correctly configured. If you have a firewall, you must configure your ports so that the Connector can communicate with these services. If you have a firewall, you must configure your ports so that the Connector can communicate with these URLs. For detailed instructions, see "Configuring Network Settings for a Firewall" on page 16.

6 Appendix: Language Codes

6 Appendix: Language Codes

For detailed instructions on mapping Episerver language codes to Clay Tablet language codes for every language your company uses for translation, see "Configuring Language Mapping and Asset Packaging" on page 33.

The Connector has the following language codes:

Type	Language Identifier	Language Code
String	Afrikaans	"af-ZA"
String	Albanian	"sq-AL"
String	Amharic	"am-ET"
String	Arabic_Algeria	"ar-DZ"
String	Arabic_Bahrain	"ar-BH"
String	Arabic_Egypt	"ar-EG"
String	Arabic_Iraq	"ar-IQ"
String	Arabic_Jordan	"ar-JO"
String	Arabic_Kuwait	"ar-KW"
String	Arabic_Lebanon	"ar-LB"
String	Arabic_Libya	"ar-LY"
String	Arabic_MiddleEast	"ar-XR"
String	Arabic_Morocco	"ar-MA"
String	Arabic_Oman	"ar-OM"
String	Arabic_Qatar	"ar-QA"
String	Arabic_Saudi_Arabia	"ar-SA"
String	Arabic_Syria	"ar-SY"
String	Arabic_Tunisia	"ar-TM"
String	Arabic_UAE	"ar-AE"
String	Arabic_Yemen	"ar-YE"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Armenian	"hy-AM"
String	Assamese	"as-IN"
String	Basque	"eu-ES"
String	Belarusian	"be-BY"
String	Bengali_Bangladesh	"bn-BD"
String	Bengali_India	"bn-IN"
String	Bosnian_Bosnia_Herzegovina	"bs-BA"
String	Bulgarian	"bg-BG"
String	Burmese	"my-MM"
String	Catalan	"ca-ES"
String	Chinese_Hong_Kong	"zh-HK"
String	Chinese_Macao	"zh-MO"
String	Chinese_PRC	"zh-CN"
String	Chinese_Singapore	"zh-SG"
String	Chinese_Taiwan	"zh-TW"
String	Croatian	"hr-HR"
String	Croatian_Bosnia_Herzegovina	"hr-BA"
String	Czech	"cs-CZ"
String	Danish	"da-DK"
String	Divehi	"dv-MV"
String	Dutch	"nl-NL"
String	Dutch_Belgium	"nl-BE"
String	English_Australia	"en-AU"
String	English_Belize	"en-BZ"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	English_Canada	"en-CA"
String	English_HongKong	"en-HK"
String	English_India	"en-IN"
String	English_Indonesia	"en-ID"
String	English_Ireland	"en-IE"
String	English_Jamaica	"en-JM"
String	English_Malaysia	"en-MY"
String	English_New_Zealand	"en-NZ"
String	English_Philippines	"en-PH"
String	English_Singapore	"en-SG"
String	English_South_Africa	"en-ZA"
String	English_Trinidad	"en-TT"
String	English_UK	"en-GB"
String	English_US	"en-US"
String	English_Zimbabwe	"en-ZW"
String	Estonian	"et-EE"
String	Faroese	"fo-FO"
String	Farsi	"fa-IR"
String	Filipino	"fil-PH"
String	Finnish	"fi-FI"
String	French	"fr-FR"
String	French_Belgium	"fr-BE"
String	French_Cameroon	"fr-CM"
String	French_Canada	"fr-CA"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	French_Cote_d_Ivoire	"fr-CI"
String	French_Democratic_Rep_Congo	"fr-CD"
String	French_Haiti	"fr-HT"
String	French_Luxembourg	"fr-LU"
String	French_Mali	"fr-ML"
String	French_Monaco	"fr-MC"
String	French_Morocco	"fr-MA"
String	French_Reunion	"fr-RE"
String	French_Senegal	"fr-SN"
String	French_Switzerland	"fr-CH"
String	Frisian_Netherlands	"fy-NK"
String	Fulfulde_Nigeria	"ff-NG"
String	FYRO_Macedonian	"mk-MK"
String	Gaelic_Ireland	"gd-IE"
String	Gaelic_Scotland	"gd-GB"
String	Gallegan	"gl-ES"
String	Georgian	"ka-GE"
String	German	"de-DE"
String	German_Austria	"de-AT"
String	German_Liechtenstein	"de-LI"
String	German_Luxembourg	"de-LU"
String	German_Switzerland	"de-CH"
String	Greek	"el-GR"
String	Guarani	"gn-PY"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Gujarati	"gu-IN"
String	Hausa	"ha-NE"
String	Hawaiian	"haw-US"
String	Hebrew	"he-IL"
String	Hindi	"hi-IN"
String	Hungarian	"hu-HU"
String	Icelandic	"is-IS"
String	Igbo	"ig-NG"
String	Indonesian	"id-ID"
String	Inuktitut	"iu-CA"
String	Italian	"it-IT"
String	Italian_Switzerland	"it-CH"
String	Japanese	"ja-JP"
String	Kannada	"kn-IN"
String	Kanuri	"kr-TD"
String	Kashmiri	"ks-IN"
String	Kazakh	"kk-KZ"
String	Khmer	"km-KH"
String	Konkani	"kok-IN"
String	Korean	"ko-KR"
String	Kyrgyz	"ky-KZ"
String	Lao	"lo-LA"
String	Latin	"la-XL"
String	Latvian	"lv-LV"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Lithuanian	"lt-LT"
String	Malay	"ms-MY"
String	Malay_Brunei_Darussalam	"ms-BN"
String	Malayalam	"ml-IN"
String	Maltese	"mt-MT"
String	Maori	"mi-NZ"
String	Marathi	"mr-IN"
String	Mongolian	"mn-MN"
String	Nepali	"ne-NP"
String	Nepali_India	"ne-IN"
String	Norwegian	"nb-NO"
String	Norwegian_Nynorsk	"nn-NO"
String	Oriya	"or-IN"
String	Oromo	"om-ET"
String	Panjabi	"pa-PK"
String	Polish	"pl-PL"
String	Portuguese	"pt-PT"
String	Portuguese_Brazil	"pt-BR"
String	Punjabi_Pakistan	"pa-PK"
String	Pushto	"ps-AF"
String	Quechua_Ecuador	"qu-EC"
String	Quechua_Peru	"qu-PE"
String	Rhaeto_Romance	"rm-IT"
String	Romanian	"ro-RO"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Romanian_Moldova	"ro-MD"
String	Russian	"ru-RU"
String	Russian_Moldava	"ru-MD"
String	Sami	"se-NO"
String	Sanskrit	"sa-IN"
String	Serbian_Cyrillic	"sr-RS"
String	Serbian_Latin	"sr-SP"
String	Sindhi_India	"sd-IN"
String	Sindhi_Pakistan	"sd-PK"
String	Sinhala	"si-LK"
String	Slovak	"sk-SK"
String	Slovenian	"sl-SI"
String	Somali	"so-ET"
String	Sorbian	"wen-DE"
String	Spanish	"es-ES"
String	Spanish_Argentina	"es-AR"
String	Spanish_Bolivia	"es-BO"
String	Spanish_Chile	"es-CL"
String	Spanish_Colombia	"es-CO"
String	Spanish_Costa_Rica	"es-CR"
String	Spanish_Dominican_Republic	"es-DO"
String	Spanish_Ecuador	"es-EC"
String	Spanish_El_Salvador	"es-SV"
String	Spanish_Honduras	"es-HN"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Spanish_LatinAmerica	"es-XL"
String	Spanish_Mexico	"es-MX"
String	Spanish_Nicaragua	"es-NI"
String	Spanish_Panama	"es-PA"
String	Spanish_Paraguay	"es-PY"
String	Spanish_Peru	"es-PE"
String	Spanish_Puerto_Rico	"es-PR"
String	Spanish_Uruguay	"es-UY"
String	Spanish_US	"es-US"
String	Spanish_Venezuela	"es-VE"
String	Swahili	"sw-TZ"
String	Swedish	"sv-SE"
String	Swedish_Finland	"sv-FI"
String	Syriac	"syr-SY"
String	Tajik	"tg-TJ"
String	Tamil	"ta-IN"
String	Tatar	"tt-RU"
String	Telugu	"te-IN"
String	Thai	"th-TH"
String	Tibetan	"bo-CN"
String	Tigrinya_Eritrea	"ti-ER"
String	Tigrinya_Ethiopia	"ti-ET"
String	Tsonga	"ts-ZA"
String	Tswana	"tn-BW"

6 Appendix: Language Codes

Type	Language Identifier	Language Code
String	Turkish	"tr-TR"
String	Turkmen	"tk-TM"
String	Uighur	"ug-CN"
String	Ukrainian	"uk-UA"
String	Urdu	"ur-PK"
String	Urdu_India	"ur-IN"
String	Uzbek	"uz-UZ"
String	Venda	"ve-ZA"
String	Vietnamese	"vi-VN"
String	Welsh	"cy-GB"
String	Xhosa	"xh-ZA"
String	Yi	"ii-CN"
String	Yiddish	"yi-MD"
String	Yoruba	"yo-NG"
String	Zulu	"zu-ZA"

Index

A

asset packaging, configuring 33

B

background jobs, activating 30

C

Clay Tablet Platform 7
configuration overview 26

F

firewall, configuring network settings 16

G

Gadget, adding 26
guide 8

H

heartbeat job, activating 28

I

installation instructions 18
  updating the Connector 20
introduction 5

L

language codes 43
language mapping, configuring 33
license, configuring 31
Lionbridge Connector Support 8
  getting a Zendesk password 13
  information to include in a ticket 9
  signing up for a Zendesk account 11
  submit a ticket 9
  viewing and updating your ticket 10
logging, configuring
  DXC cloud environment 39
  local installation 37

M

multiple users, configuring 41

N

network settings, configuring for a firewall 16

O

overview 7
  configuration 26
  pre-installation 16

P

pre-installation overview 16

R

run interval, setting 27

S

scheduled job, activating 27
source.xml, configuring 31
support 8
  getting a Zendesk password 13

information to include in a ticket 9
signing up for a Zendesk account 11
submit a ticket 9
viewing and updating your ticket 10
system date, setting 16
system requirements 16
system time zone, setting 16
system time, setting 16

T

target.xml, configuring 31
terminology 5
translation heartbeat job, activating 28
translation workflows, configuring 41

U

uninstalling the Connector 25
update instructions 20
Use grouped asset handling check box 35
users, configuring multiple 41

W

workflows, configuring 41