TIBCO EBX® Digital Asset Manager Add-on

Version 2.3.1
April 2022

TIBCO®

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

ANY SOFTWARE ITEM IDENTIFIED AS THIRD PARTY LIBRARY IS AVAILABLE UNDER SEPARATE SOFTWARE LICENSE TERMS AND IS NOT PART OF A TIBCO PRODUCT. AS SUCH, THESE SOFTWARE ITEMS ARE NOT COVERED BY THE TERMS OF YOUR AGREEMENT WITH TIBCO, INCLUDING ANY TERMS CONCERNING SUPPORT, MAINTENANCE, WARRANTIES, AND INDEMNITIES. DOWNLOAD AND USE OF THESE ITEMS IS SOLELY AT YOUR OWN DISCRETION AND SUBJECT TO THE LICENSE TERMS APPLICABLE TO THEM. BY PROCEEDING TO DOWNLOAD, INSTALL OR USE ANY OF THESE ITEMS, YOU ACKNOWLEDGE THE FOREGOING DISTINCTIONS BETWEEN THESE ITEMS AND TIBCO PRODUCTS.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO and TIBCO EBX are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. Please see the readme.txt file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (*https://www.tibco.com/patents*) for details.

Copyright 2006-2022. TIBCO Software Inc. All rights reserved.

Table of contents

User Guide

1. About the add-on....10

Getting started

2. Configuration process overview....14
3. Configuring the data model....15
4. Creating a Drive....19
5. Linking a Drive and data model field....23
6. Uploading and attaching assets....27
7. Previewing assets....35
8. Configuring asset filters....39

Managing digital assets

9. Management overview....44
10. Editing asset information....51
11. Tags and tag clouds....57
12. Managing digital asset versions....59
13. Additional configuration options....61
14. Finding and removing unused assets....69
15. Enabling asset management from perspectives....71
16. Configuring an external Drive....73
17. Connecting to an Amazon S3 bucket....75
18. Performing bulk upload and attach....77

Developer Guide

19. REST service operations....82

Alternative storage locations

20. Connecting to alternative storage locations....92
21. Example Implementation....95

Reference Guide

22. Add-on specific services..... 104
23. Digital asset group..... 107
24. Drive group..... 111
25. D.A.C. group..... 115
26. Classification group..... 119
27. Reference data group..... 121

Release Notes

28. Version 2.3.1.....126
29. All release notes.....130

User Guide

CHAPTER 1

About the add-on

This chapter contains the following topics:

1. Concepts overview
2. How this guide is organized

1.1 Concepts overview

The TIBCO EBX® Digital Asset Manager Add-on allows you to manage digital asset storage, classification, and versioning. Once you've created a storage location, you can use the add-on to upload assets and enrich them with metadata.

> **Note**
>
> You can also use the add-on's API to connect to a third-party digital asset management system. This allows you to use the third-party tool for management purposes, but still link assets to TIBCO EBX® data model fields.

The asset governance process can include adding descriptive metadata to assets, creating versions, deactivating/deleting and backing up and restoring assets. The add-on allows you to perform these tasks from simple-to-navigate screens. See Managing digital assets [p 44] for more information.

Asset storage in Drives

The EBX® Digital Asset Manager Add-on uses *Drives* as physical storage locations for assets. You can use Drives as part of asset governance by creating one for asset types, assets from different sources or, any other organizational requirement. For information on creating a, and uploading assets into a Drive, see Getting started with the add-on. [p 14]

Digital asset components

A *Digital asset component (D.A.C.)* links a storage location (Drive) to a data model field configured to use the add-on. You can then attach assets from the Drive to the field. For example, a data model table containing product information could have a field containing assets that show one or more views of the product.

1.2 How this guide is organized

This guide contains the following sections:

Section	Contents
Getting started	This section covers initial add-on configuration. Most of the steps in this section require administrative privileges.
Uploading and attaching assets	This chapter describes how to upload assets to a Drive and attach them to a data model field. Add-on configuration must be complete prior to performing these tasks.
Previewing assets	This chapter describes behavior when previewing assets.
Configuring asset filters	This chapter outlines the different types of filters the add-on offers.
Managing digital assets	Digital asset governance can include managing basic asset information, tags, metadata and versions—all topics covered in this section. Additionally, this section discusses how to create a backup of, and restore assets.

Getting started

CHAPTER 2

Configuration process overview

Assets uploaded using the add-on can benefit from its management capabilities. In order to upload, store and link assets with a data model, you must complete the configuration tasks below:

- Add a media field to a data model using the **mediaType** data type. For detailed steps to complete this task, see Configuring the data model [p 15].
- Create an asset storage location, or *Drive*. For detailed steps to complete this task, see Creating a Drive [p 19].
- Use a *Digital asset component (D.A.C.)* to link the media field and Drive. For detailed steps to complete this task, see Linking a Drive and data model field [p 23].

The following map summarizes the tasks required to complete this process:

Initial configuration process task map

CHAPTER 3

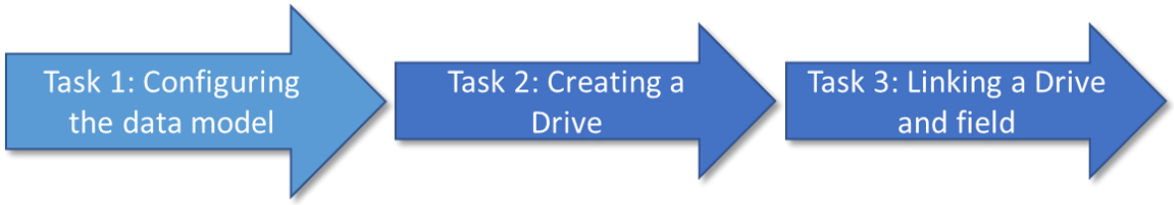
Configuring the data model

This chapter contains the following topics:

1. Configuration task map
2. Steps to complete Task 1

3.1 Configuration task map

The following map shows where you are in the add-on configuration process:



The following section outlines the steps to complete Task 1: Configuring the data model.

See also

Creating a Drive *[p 19]*

Linking a Drive and data model field *[p 23]*

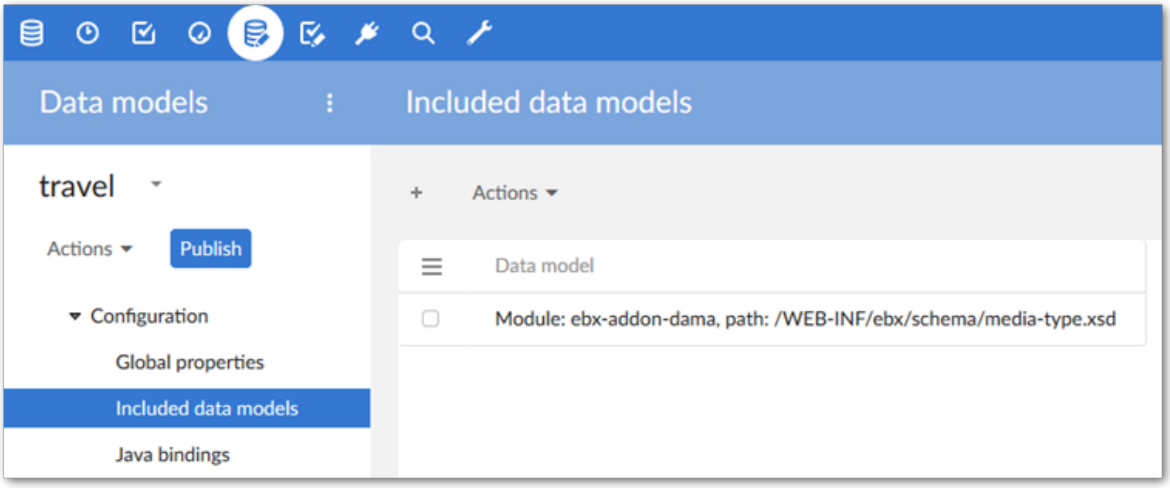
Configuration process overview *[p 14]*

3.2 Steps to complete Task 1

To attach a digital asset to a record, you must add a field to the data model. This field's data type must be set to **mediaType**. The following steps describe how to accomplish this:

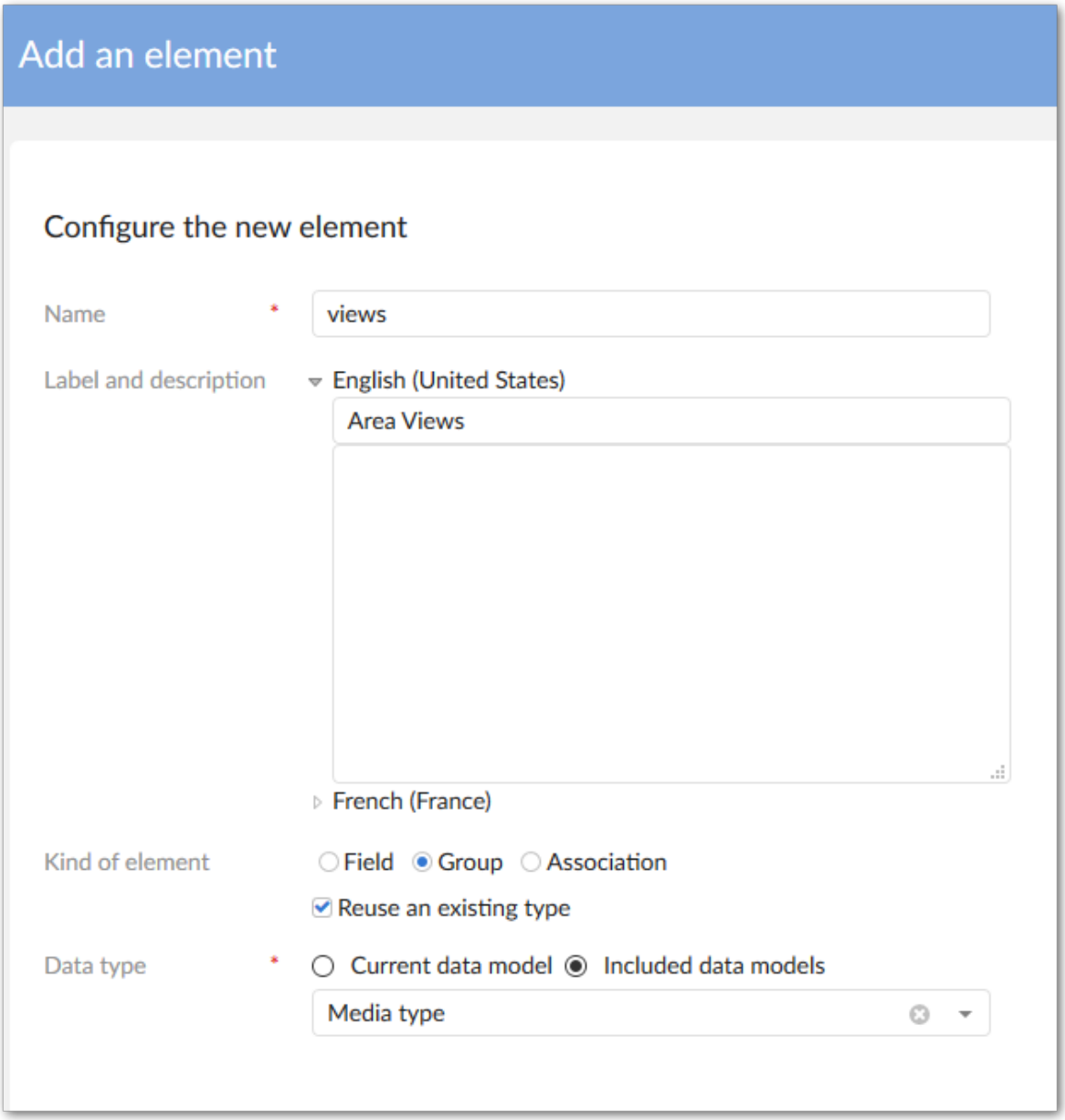
1. Locate and open an existing, or create a new data model to which you want to add asset management capabilities.
2. In the **Navigation** pane, open *Configuration* > *Included data models* and click the + icon to add a data model.

3. Select the **Packaged in module** option and in the **Model properties** group, choose **ebx-addon-dama** for the **Module name** and select **/WEB-INF/ebx/schema/media-type.xsd** for the **Path in module** field.



4. After saving and closing, create a field using the **mediaType** data type in the desired table by selecting **Create child** from the table's services menu and using the following options:
   - Enter a name and optionally a label and description.
   - For the **Kind of element**, select **Group**.
   - Select **Reuse an existing type**.

- Select **Included data models** as the **Data type** and choose **Media type** from the drop-down menu.



5. Use the group's **Minimum/Maximum number of values** properties to specify the number of assets that users can attach to this field.

> **Attention**
> If you configure options for this field's **Default views and tools**, it can prevent users from accessing asset management functionality. Additionally, the value of a Path variable for a **Media type** field must specify an absolute path. If not, a `PathAccesException` is thrown.

6. After clicking **Create**, you can publish the data model and, if needed, create a dataset.

See also

Creating a Drive *[p 19]*

Linking a Drive and data model field *[p 23]*

Configuration process overview *[p 14]*

CHAPTER 4

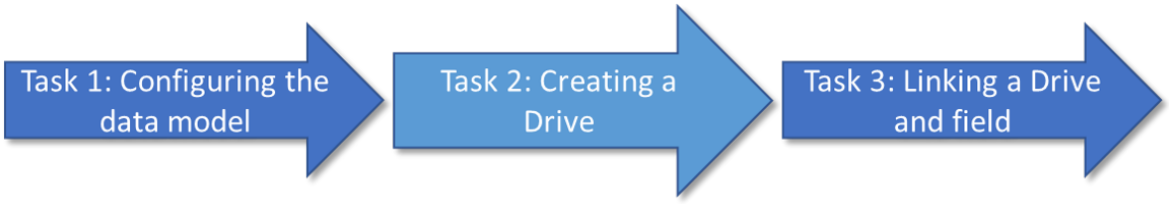
Creating a Drive

This chapter contains the following topics:

1. Configuration task map
2. Steps to complete Task 2
3. Troubleshooting Drive error messages

4.1 Configuration task map

The following map shows where you are in the add-on configuration process:



The following section outlines the steps to complete Task 2: Creating a Drive.

See also

Linking a Drive and data model field *[p 23]*

Configuring the data model *[p 15]*

Configuration process overview *[p 14]*

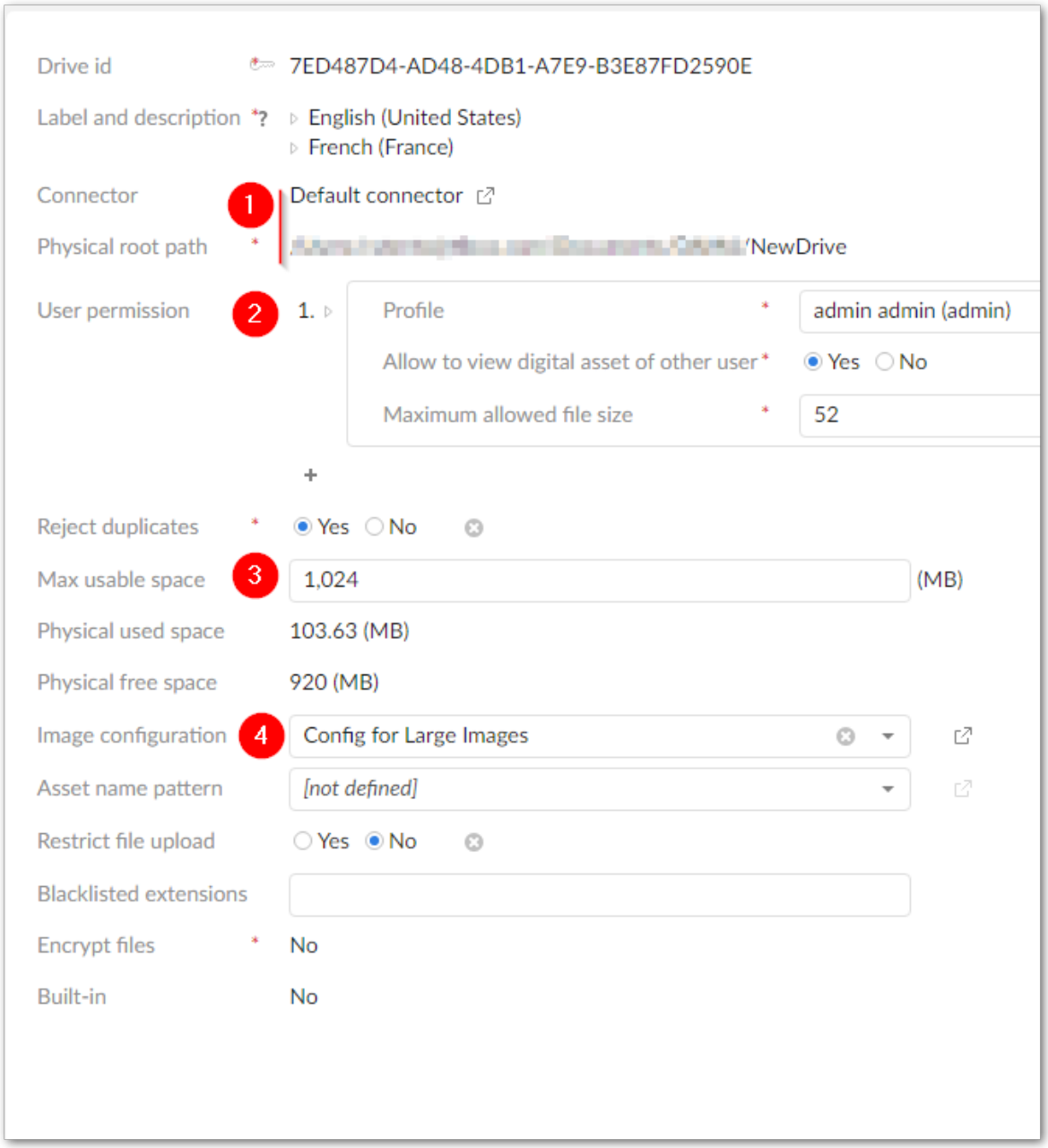
4.2 Steps to complete Task 2

A Drive specifies an asset storage location and options related to acceptable file size and storage limitations. The following example points to a location on the local machine, however the API also allows you to point to a cloud storage location. For more information, see the *Java API* documentation.

To create a Drive:

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *Drive* and open the **Drive** table. This table stores asset storage configuration information.

2. Create a new *Drive* configuration. When filling required fields, you can hover your mouse over the property and click the **?** icon to view online help, or you can refer to the Drive group [p 111] section in this guide. The following image and list highlight some configuration options.



1) **Connector:** Set the type of connector to use for this Drive:

- **Amazon S3:** Allows you to create a Drive where assets are stored and managed in an Amazon S3 bucket.

	• **Custom connector:** Allows you to connect to an externally managed location by using a custom implementation. • **External manager:** Allows you to use the API to connect to an externally managed storage location. • **Default connector:** Allows you to create a default Drive where asset storage and management tasks are handled by the add-on. **Physical root path:** (For **Default connector**) Specify the physical storage location for assets. **External management class:** (For **External manager**) Specify the class that defines an externally managed storage location. See Connecting to alternative storage locations [p 92] for more information.
2)	Specify Drive level permissions.
3)	Determine the maximum amount of physical disk space this Drive can use for asset storage.
4)	Create a custom image configuration to set limits on the physical size (width/height) of uploaded images.

4.3 Troubleshooting Drive error messages

The following table provides troubleshooting tips for *Drive* configuration error messages.

Message	Cause
Your path is a file; not a folder	The path specified in the Physical root path field points to a file when it should point to a folder.
You do not have write permission in this folder	The add-on does not have sufficient system permission to manage assets in the specified location. You may need a system administrator to raise permission levels, or use a different storage location.
This folder is not empty	You will see this message if the storage location you choose already contains files. When creating a new Drive, the storage location should be empty to avoid potential duplication of an asset's unique identifier.
Cannot create folder using this path	If the folder you specify doesn't exist, the add-on checks whether it can create the folder using the specified path. This message displays when it cannot create the folder automatically.
Comment	Displays comments applied to this asset version.
Current version	Indicates whether this version is considered the current version.
Digital asset	The digital asset of which this is a version.

See also

Linking a Drive and data model field *[p 23]*

Configuring the data model *[p 15]*

Configuration process overview *[p 14]*

CHAPTER 5

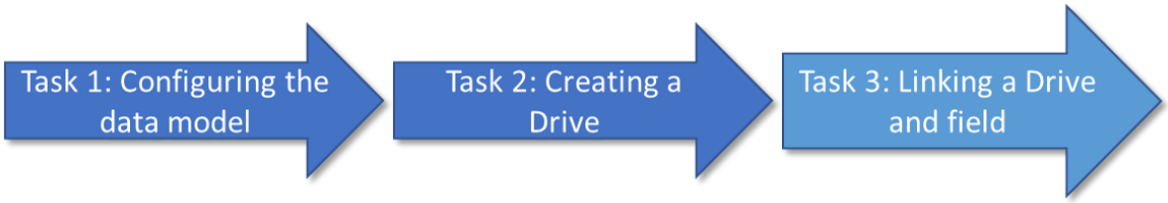
Linking a Drive and data model field

This chapter contains the following topics:

1. Configuration task map
2. Steps to complete Task 3

5.1 Configuration task map

The following map shows where you are in the add-on configuration process:



The following section outlines the steps to complete Task 3: Linking a Drive and field.

See also

Creating a Drive *[p 19]*

Configuring the data model *[p 15]*

Configuration process overview *[p 14]*

5.2 Steps to complete Task 3

To complete the final configuration task you link the data model field and Drive created in Tasks 1 and 2, respectively.

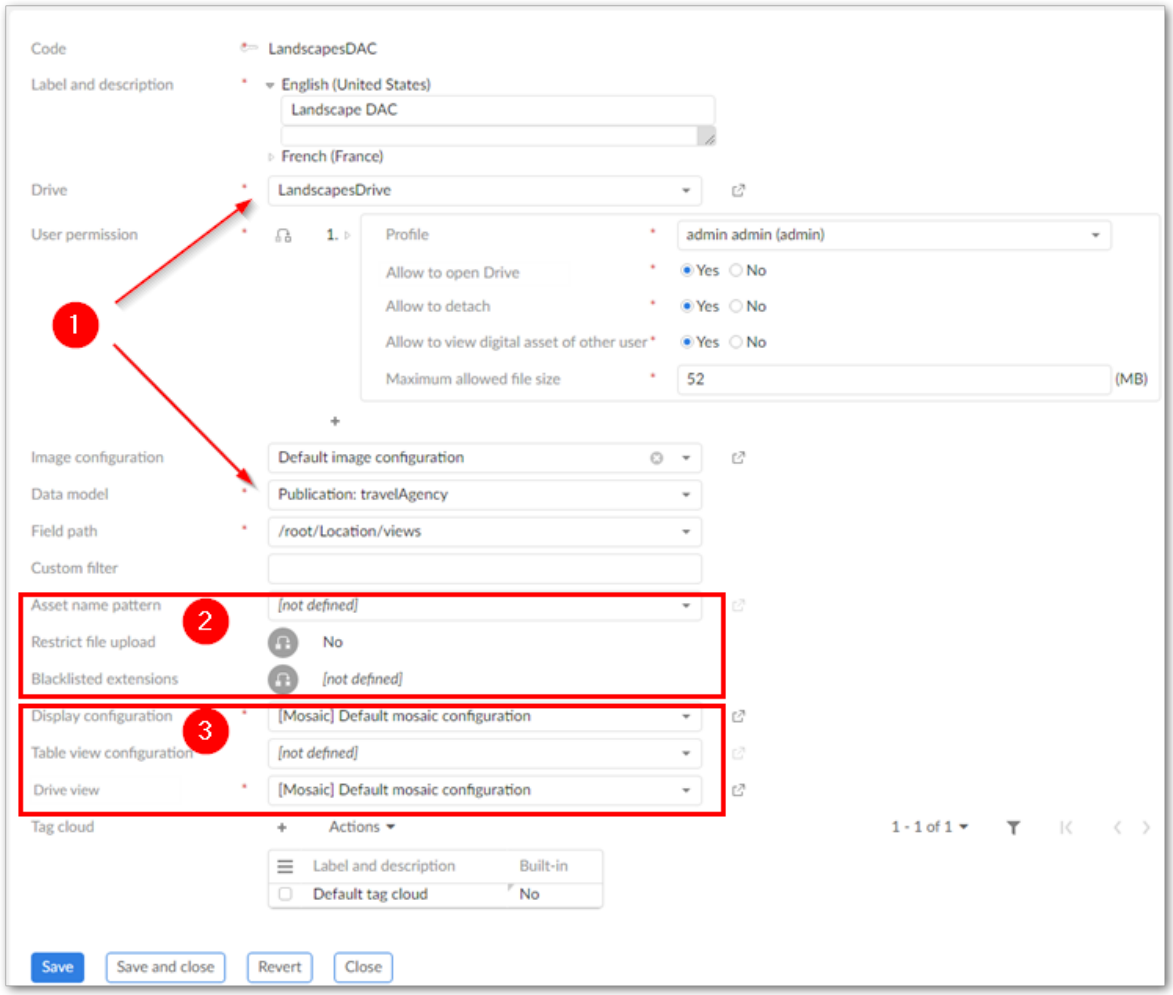
The following steps show how to link the media field you created in the data model with the Drive (storage location):

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *D.A.C* and open the **Digital asset component** table and create a new record.
2. Fill the required fields, save and close. It is important to note that by default no restrictions apply to the types of files users can upload. If you want to exclude certain file types, you can enable the **Restrict file upload** option and select acceptable types in **Supported extensions** field. The add-on includes a set of default image extensions. If you want to allow other file types, you can

create them in the **File extension** table located in the **Reference data** domain. Additionally, you can use the **Blacklisted extensions** field to manually exclude available file types.

Additionally, the final required property, **Tag cloud**, isn't available to edit until you've saved the initial configuration.

The following image highlights D.A.C. functionality:



1)	Link the physical storage location selected in the **Drive** menu to the Media-type field in the data model.
2)	Options to govern asset upload: • **Asset name pattern:** Select a regular expression to enforce a naming convention for asset physical file names. The add-on will prevent users from uploading files if their names do not match this pattern. • **Restrict file upload:** When enabled, the **Supported extensions** group displays and allows you to specify an approved set of file extensions for upload. • **Blacklisted extensions:** Enter a list of file extensions you want the add-on to blacklist.
3)	These options determine:

- How assets display after users attach them to data model fields.
- Whether the asset thumbnails display in the table views and how they display in the **Drive** view.

Enabling thumbnail display and upload in the tabular view

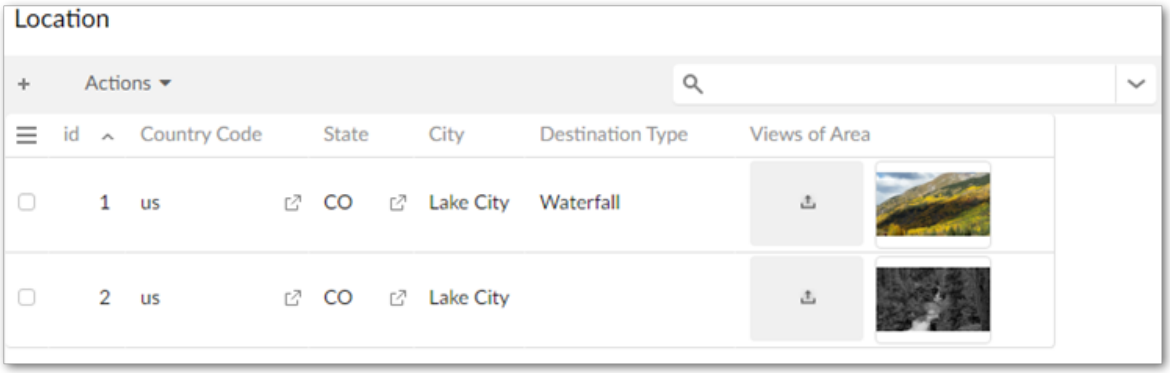
You can specify that an asset's thumbnail and an icon to upload and attach assets display when viewing a table. You can also customize the thumbnail display size. Enabling upload of assets from the tabular view is convenient for end users since they won't have to open each record to attach assets.

Thumbnail display behaves as follows:

- When no assets are attached, the field is left blank.
- If multiple assets are attached, the first thumbnail displays. Or if multiple types of assets are attached, the first asset whose type supports thumbnails displays.
- If the asset type does not support thumbnails, the type's icon displays. When no icon is specified for the asset type, the default icon displays.

To enable table display of thumbnails and allow upload and attach:

1. If one does not already exist, create a view configuration and set the **View mode** to **Table view**. For instructions on creating a view configuration, see Specifying the view mode [p 62].
2. Open the D.A.C. associated with the data model field that you want to display thumbnails.
3. Use the **EBX tabular view** drop-down menu to select the desired view configuration.
4. Save and close. The following image shows an example thumbnail display:



See also

Creating a Drive [p 19]

Configuring the data model [p 15]

Configuration process overview [p 14]

CHAPTER 6

Uploading and attaching assets

This chapter contains the following topics:

1. Overview
2. Using quick upload mode
3. Using standard upload mode

6.1 Overview

The add-on provides standard and quick-upload functionality. Administrators can use permission settings to determine this behavior. See Setting the upload mode [p 27] for more information. Even when quick-upload is available, users can still access the **Drive** to upload and attach assets. See the following sections for more information:

- Using quick upload mode [p 27]
- Using standard upload mode [p 28]

Setting the upload mode

Administrators can specify which upload mode users can access. If you enable quick-upload, users will not see notification of duplicate assets, or errors indicating upload of invalid assets. For instance, if a user attempts to upload a file that does not meet conditions specified in D.A.C. settings, the file displays in the field with a red border and will not be attached to the field when saving.

To change the upload mode:

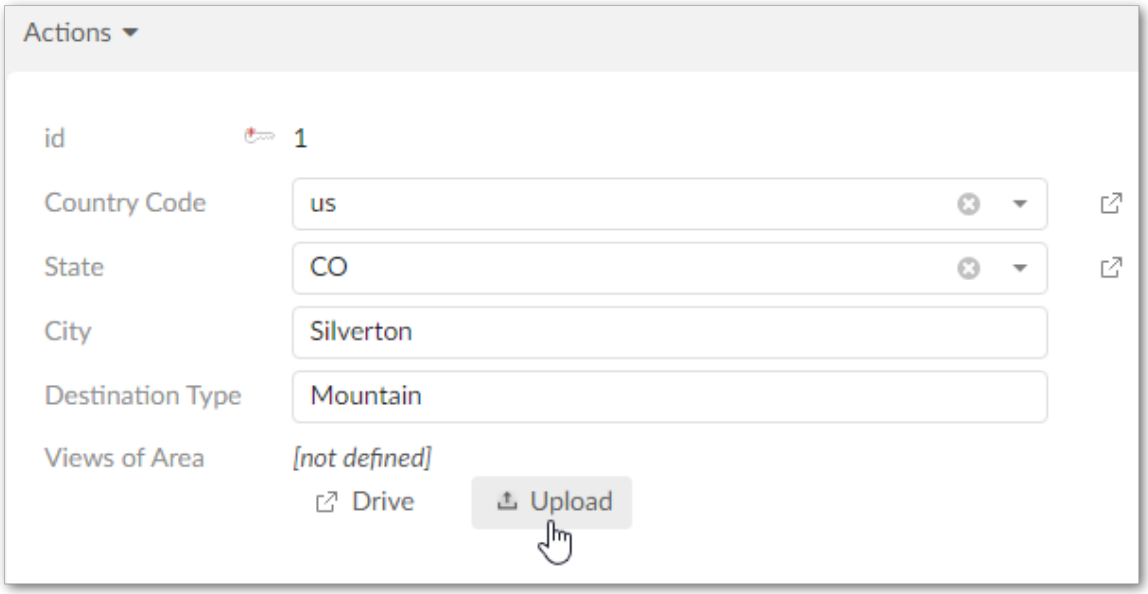
1. In the add-on's administrative area, navigate to and open the D.A.C. on which you want to adjust settings.
2. Open the appropriate user permission group and set the **Allow to quick upload and attach** property to the appropriate setting. Save and close.

6.2 Using quick upload mode

Use the following steps to upload and automatically attach an asset to a field:

1. Create a new, or open an existing record in the dataset containing the **media type** field to which you want to attach an asset.

2. Click the **Upload** button that displays in the **media type** field. Note: You cannot use the **Upload** option when the maximum number of assets are already attached to the field.



3. In the pop-up window that displays, you can browse to select the files to upload and attach to the field. Once the add-on finishes attaching assets, you can edit their information. See Management overview [p 44] for more information. Also, note that the quick-upload option displays no errors, or warnings during the upload process. However, if you've attempted to attach an assets that violates requirements for this field, it displays with a red outline indicating an error and it will not be attached to the field when saving. You can mouse over the asset to see information pertaining to the error.
4. After assets have been attached to the field, be sure to save before exiting.

6.3 Using standard upload mode

The following topics describe how to upload assets and attach them to a data model field:

- Upload and automatically attach assets [p 28]
- Upload and attach in a Drive [p 30]
- Upload assets via administrative access [p 31]
- Attaching existing assets to a field [p 31]

Attention

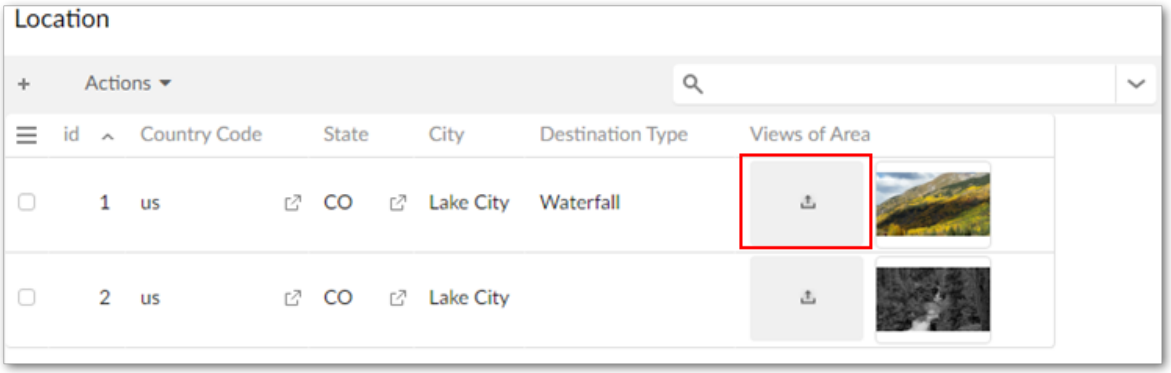
The field and Drive must be properly configured before uploading and attaching assets. See Linking a Drive and data model field [p 23] for more information.

Upload and automatically attach assets

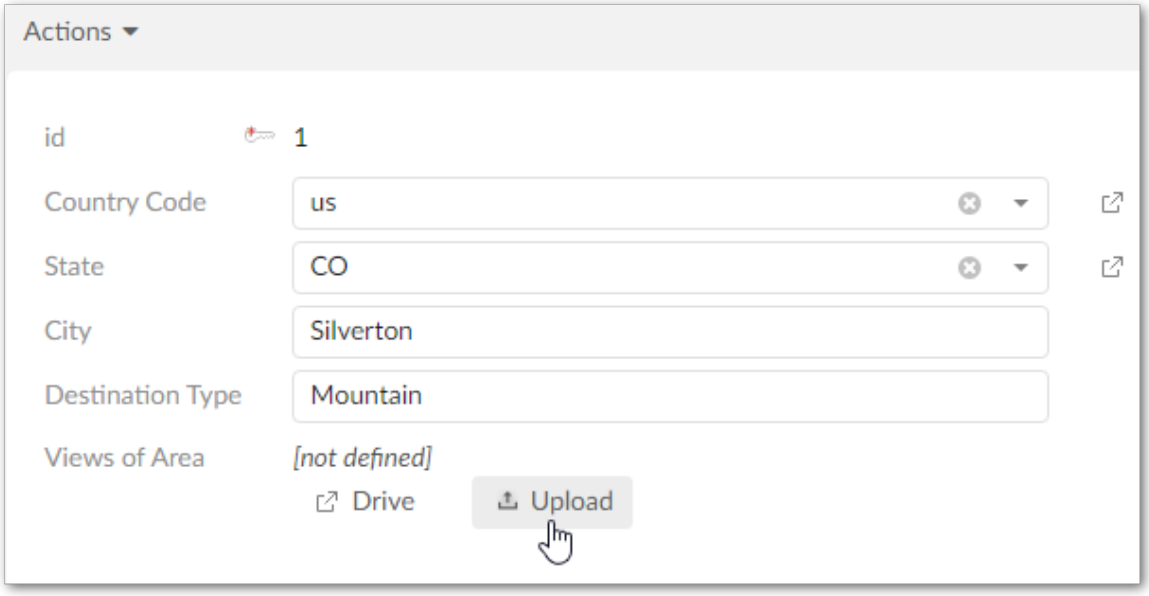
If the Drive linked with a field doesn't contain the desired asset, use the following steps to upload and automatically attach an asset to a field:

1. Initiate the upload process in one of the following ways:

- If enabled by an administrator, select the upload icon that displays in the tabular view.

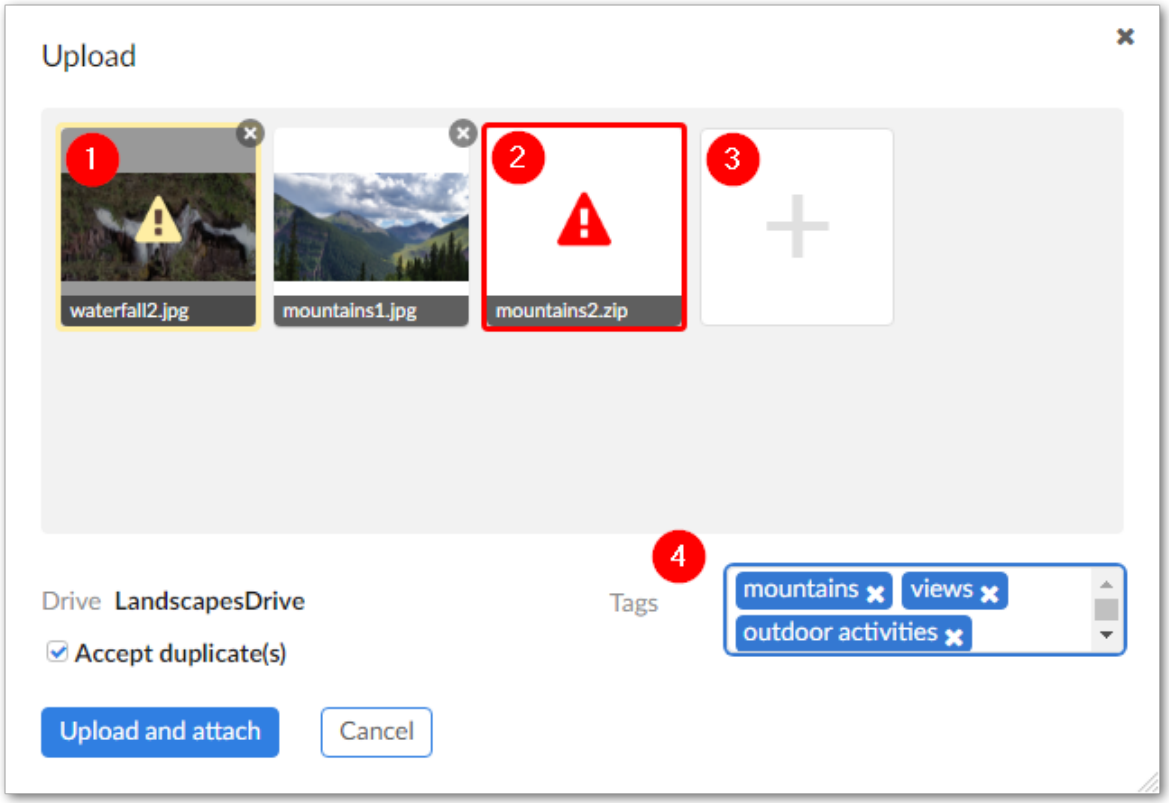


- Create a new, or open an existing record and click the **Upload** button that displays in the **media type** field. Note: You cannot use the **Upload** option when the maximum number of assets are already attached to the field.



2. In the pop-up window that displays, you can drag and drop, or browse to select the files to upload and attach to the field. The following image highlights features in the upload screen. Once the

add-on finishes attaching assets, you can edit their information. See Management overview [p 44] for more information.



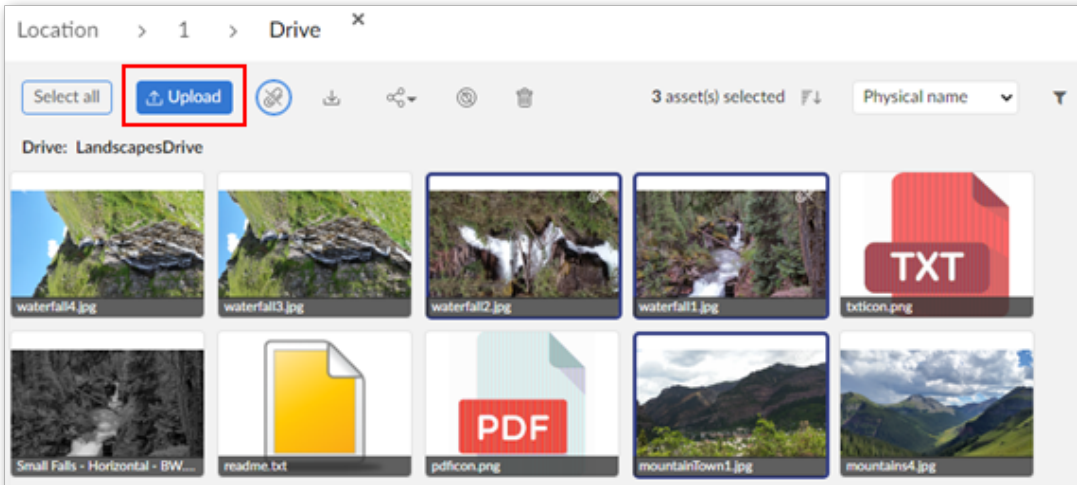
1)	The yellow icon overlay indicates an identical asset exists in the Drive. Use the **Accept duplicate(s)** option below to accept or reject upload of duplicates.
2)	The red icon overlay indicates the asset does not meet D.A.C requirements. If you mouse-over the icon, a list of acceptable file types displays.
3)	Select to browse you file system for another file to upload.
4)	Enter the tags to apply to the assets included in this upload.

Upload and attach in a Drive

The following steps describe how to upload an asset and attach it to a field using the **Drive** view:

1. Create a new, or open an existing record in the dataset containing the **media type** field to which you want to attach an asset.
2. Click the **Drive** button that displays in the **media type** field to open the **Drive** view.

3. In the **Digital assets** pane click the **Upload** button.



4. In the pop-up window that displays, you can drag and drop, or browse to select the files to upload. If you've dropped files into the upload window and want to browse to select additional files, just click the large plus thumbnail. The upload window also allows you to add tags which the add-on applies to each asset in this upload. When adding tags, you can select from existing, or create new tags by adding text and clicking the **Enter** key.
5. Click **Attach and upload**. Note that you have the additional option here of selecting **Upload** to upload assets to the Drive without attaching them to the field. Once assets have uploaded, you can select them to perform additional actions. The Management overview [p 44] section covers related topics in more detail.

Upload assets via administrative access

If you have administrative privileges, you can use perform the following steps to upload an asset:

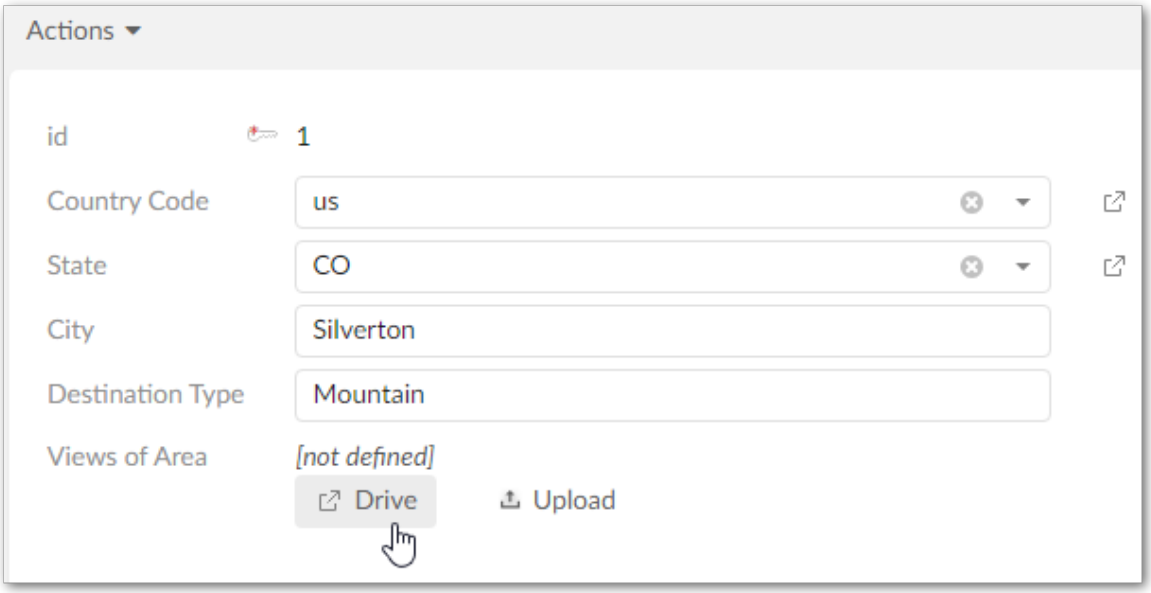
1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *Digital asset* and from the **Digital asset** table's **Actions** menu and select **Upload a new digital asset**.
2. Browse to select the desired file.
3. You can create a label, description, version name, comments and tags. When adding tags, you can select from existing, or create new tags by adding text and pressing the **Enter** key.
4. Use the **Drive** drop-down list to indicate the Drive to which you want to add this asset.
5. Click **Upload**. After the process completes, the upload window refreshes and you can repeat these steps to upload additional assets.

Once you've uploaded assets, you can manage information such as label, description, state, tags and other metadata. The section, Management overview [p 44], covers the asset management topic in detail.

Attaching existing assets to a field

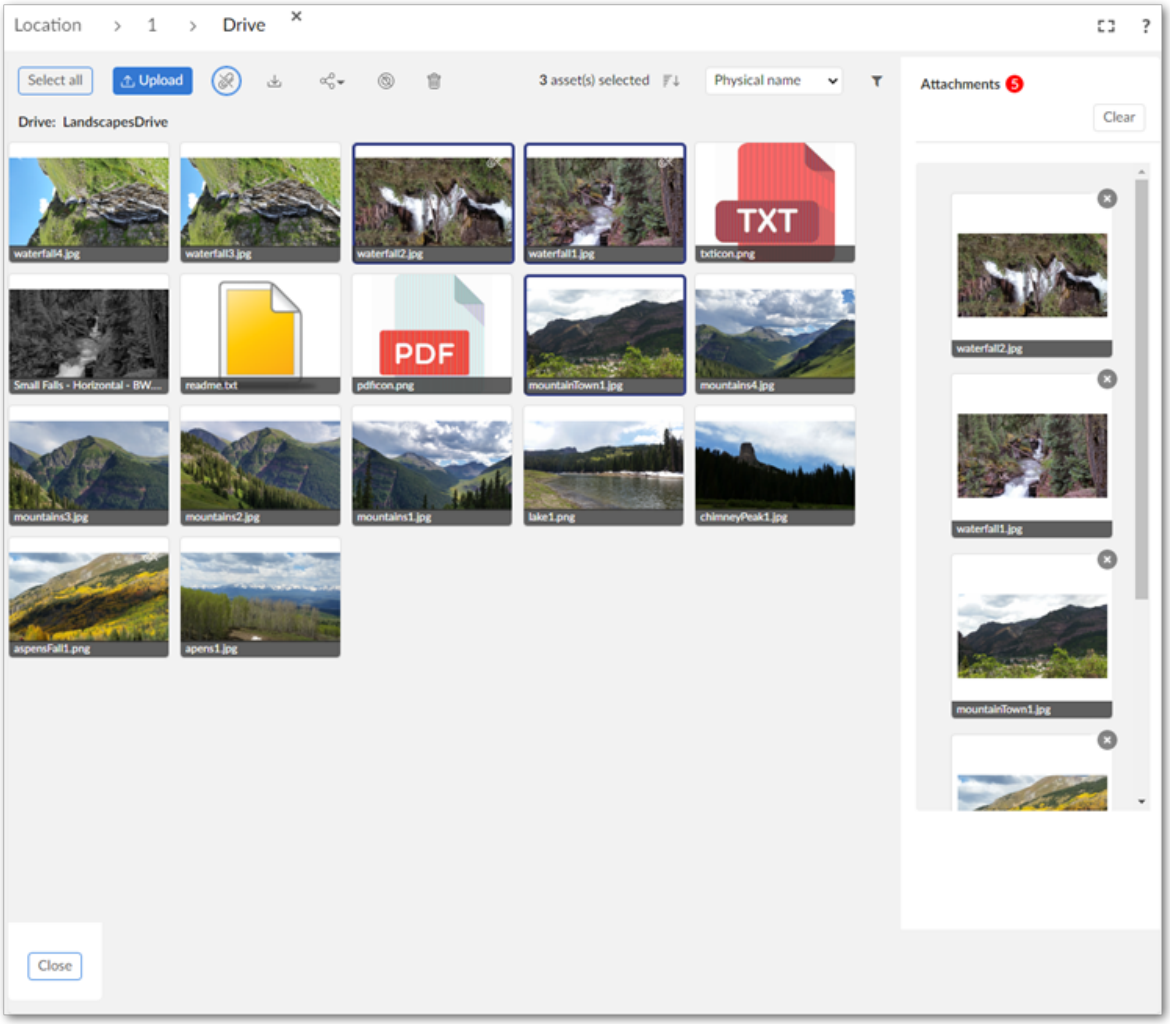
If you have previously used the add-on to upload files to a Drive, you can quickly attach them to the field by:

1. Clicking **Drive** to open the **Drive** view.



2. Attach assets:
   - Individual assets: Hover your mouse over an asset and select the paperclip icon.
   - Multiple assets: Select multiple assets and click the paperclip icon at the top of the screen.

3. Select one or more assets to attach. Select **Attachments** to show thumbnails of any assets attached to the field. When viewing the attached assets, you can detach all assets by selecting **Clear**. You can also download individual assets from the attachment view.

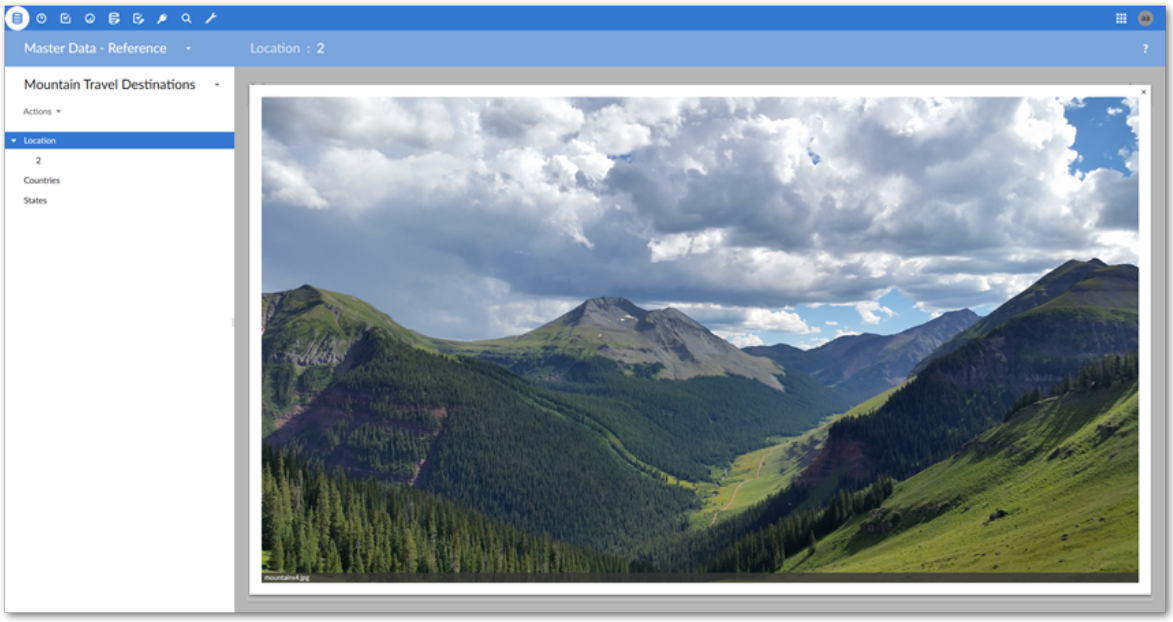


CHAPTER 7

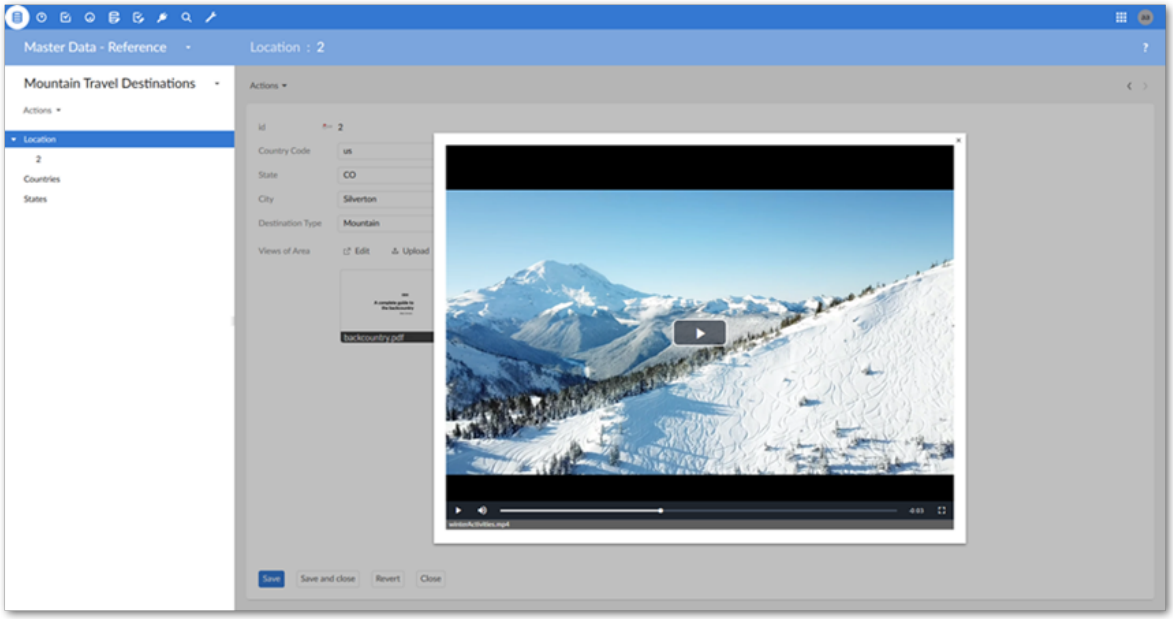
Previewing assets

For supported file types, the add-on opens a preview when you hover your mouse over an asset and select the icon. If you preview:

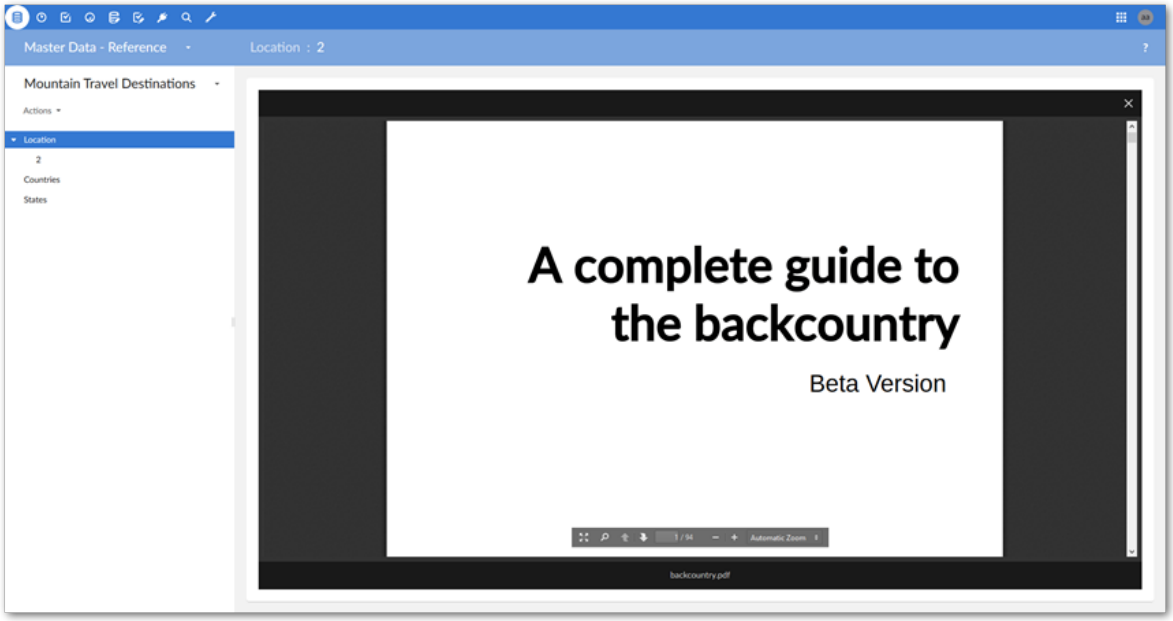
- an image: the add-on displays a full-size view and you can click to zoom.



- a video: the add-on automatically plays the video and enables basic playback controls.



- a document: the add-on loads the document for viewing. From the preview, you can access standard viewer features such as zoom, navigation, and search.



Management overview [p 44] describes additional options available when previewing an asset from the **Drive** view. The following table provides additional information pertaining to asset previews:

Asset type	Notes
MP4	Currently, the add-on supports previewing and thumbnail generation for videos with the H.264 codec and AAC, or MP3 audio format.
PDF	The add-on does not support: • Text search for older PDFs that use JPEG2000, or JBIG formats. • Previewing files larger than 5MB.
Documents (DOC, DOCX, PPT, PPTX, XLS, XLSX)	The add-on performs some conversion to display document previews. Certain style elements may display differently in the preview. If the add-on cannot create a preview, and error message displays and it allows you to download the asset. When viewing PPT or PPTX previews, text is not selectable and the search function is disabled. Also, note that previews of charts in these document types do not display.

CHAPTER 8

Configuring asset filters

This chapter contains the following topics:

1. Overview
2. Filtering records containing assets
3. Adding a custom filter

8.1 Overview

The add-on includes the following types of filters that administrators can configure:

- Filters can apply to tables and display records based on the presence of attached digital assets. Once configured, users can filter records using the EBX® table filter. For more information, see Filtering records containing assets [p 39].
- A Java class can implement a programmatic filter that only allows users to view assets that meet certain criteria. Once a developer implements the class, an administrator can include it as part of a D.A.C. configuration. The filter is essentially transparent to users as the add-on automatically applies it when they view assets. For more information, see Adding a custom filter [p 40].

8.2 Filtering records containing assets

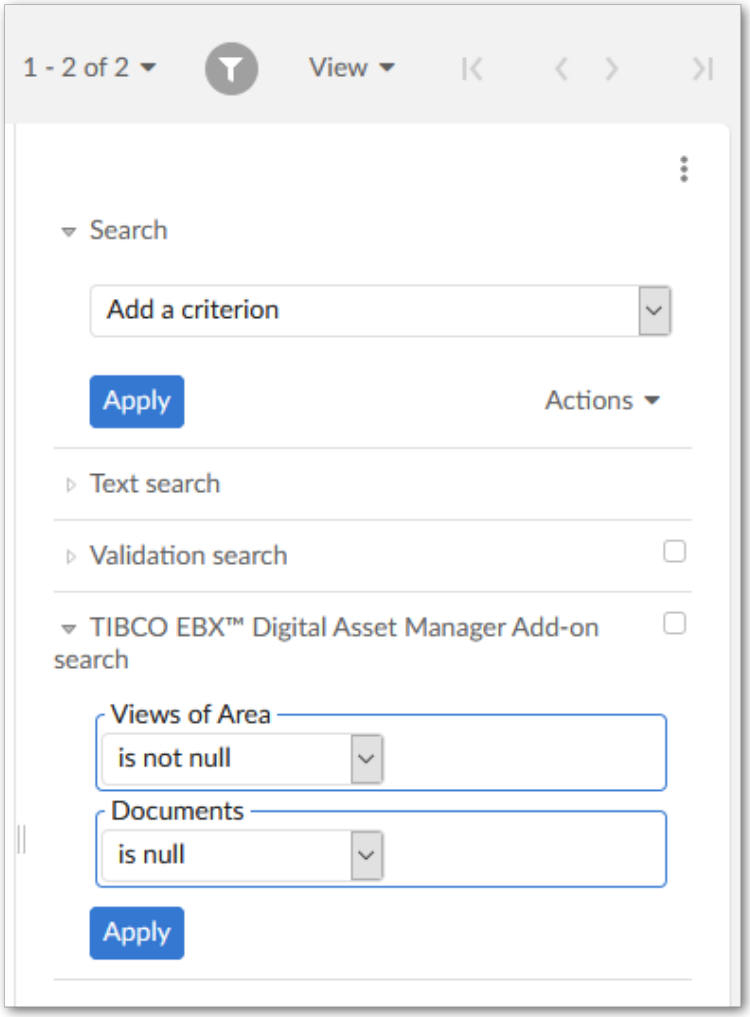
Administrators can activate a filter that allows users to display records based on the presence of attached assets. To enable filtering:

1. Open the desired model in the DMA.
2. Navigate to *Configuration* > *Add-ons* and create a new record.
3. In the **Name** field, select **Digital Asset Manager.**
4. Save, close, and publish the data model.

Users can access filter functionality via the EBX® table filter. The filter's **TIBCO EBX® Digital Asset Manager Add-on search** group displays all fields in the current table configured to accept asset attachments. For each field specify a value of:

- **is null** to show records without attachments in this field.

- **is not null** to show records with attachments in this field.

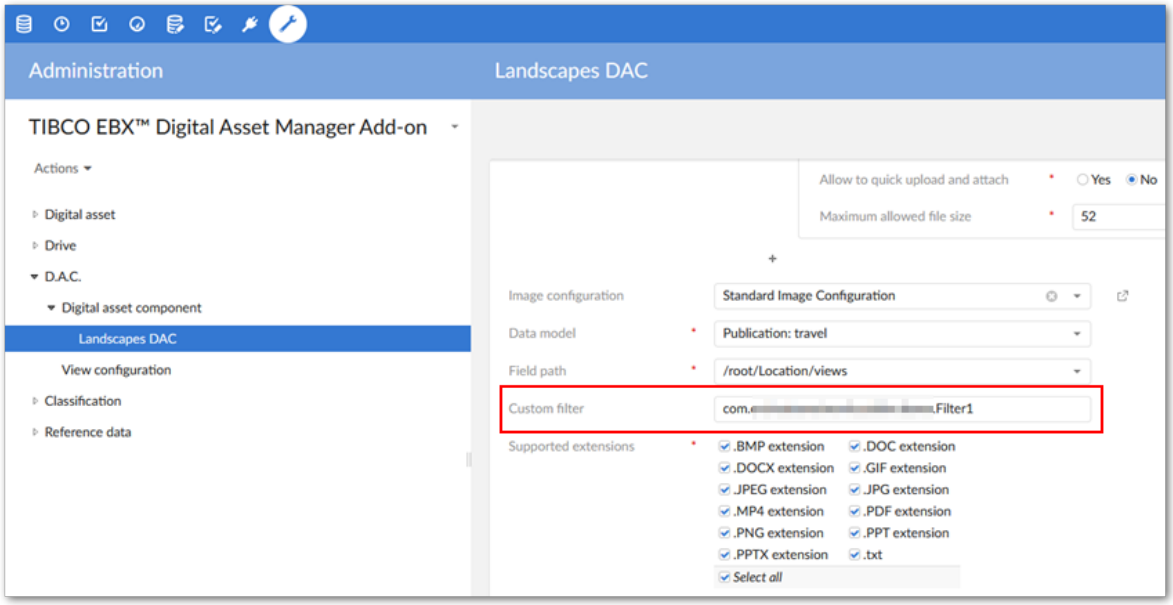


8.3 **Adding a custom filter**

A custom filter implemented in a Java class can provide fine-grained control over what users can view in the add-on's **Drive** view. See the API documentation for information on the interface and methods required for this type of filter. Once the class has been created, configuration only requires that you include the filter class in a D.A.C. configuration. The process is outlined below:

1. Open the desired record in the **Digital asset component** table.

2. In the **Custom filter** field, enter the fully qualified name of the Java class that implements the filter.



3. After saving and closing filter, the add-on applies filter functionality. The assets displayed for users in the **Drive** view are based on their permissions and the filter.

Managing digital assets

CHAPTER 9

Management overview

This chapter contains the following topics:

1. Overview of digital asset management

9.1 Overview of digital asset management

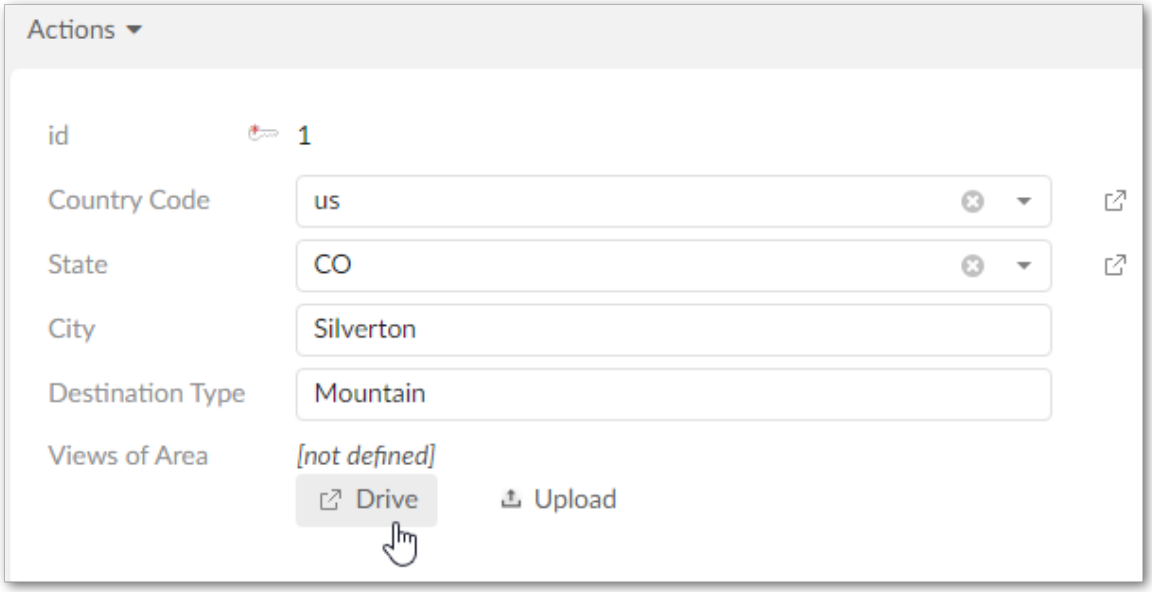
The add-on makes it quick and easy to find the specific digital asset you need with sophisticated searching, sorting, and information editing capabilities. In addition to options available to administrators, users can launch the following activities from the **Drive** view:

- Upload new assets.
- Attach assets to a data model field.
- Use criteria to sort and filter assets.
- Select assets to view their details.
- Download, share, deactivate and delete assets.

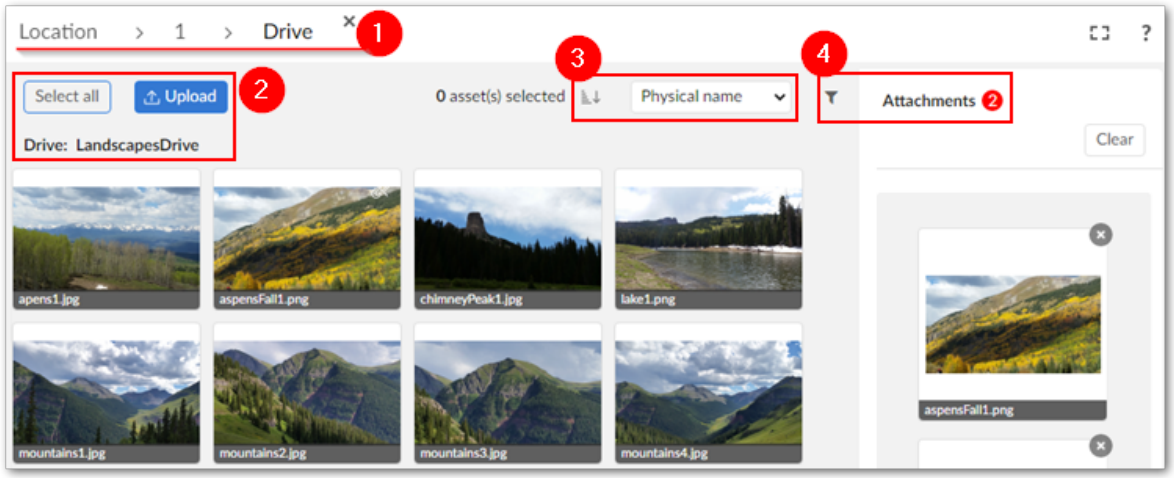
To access the **Drive** view:

- Navigate to and open a data model record containing a field attached to a Drive.

- As shown below, click the **Drive** button.



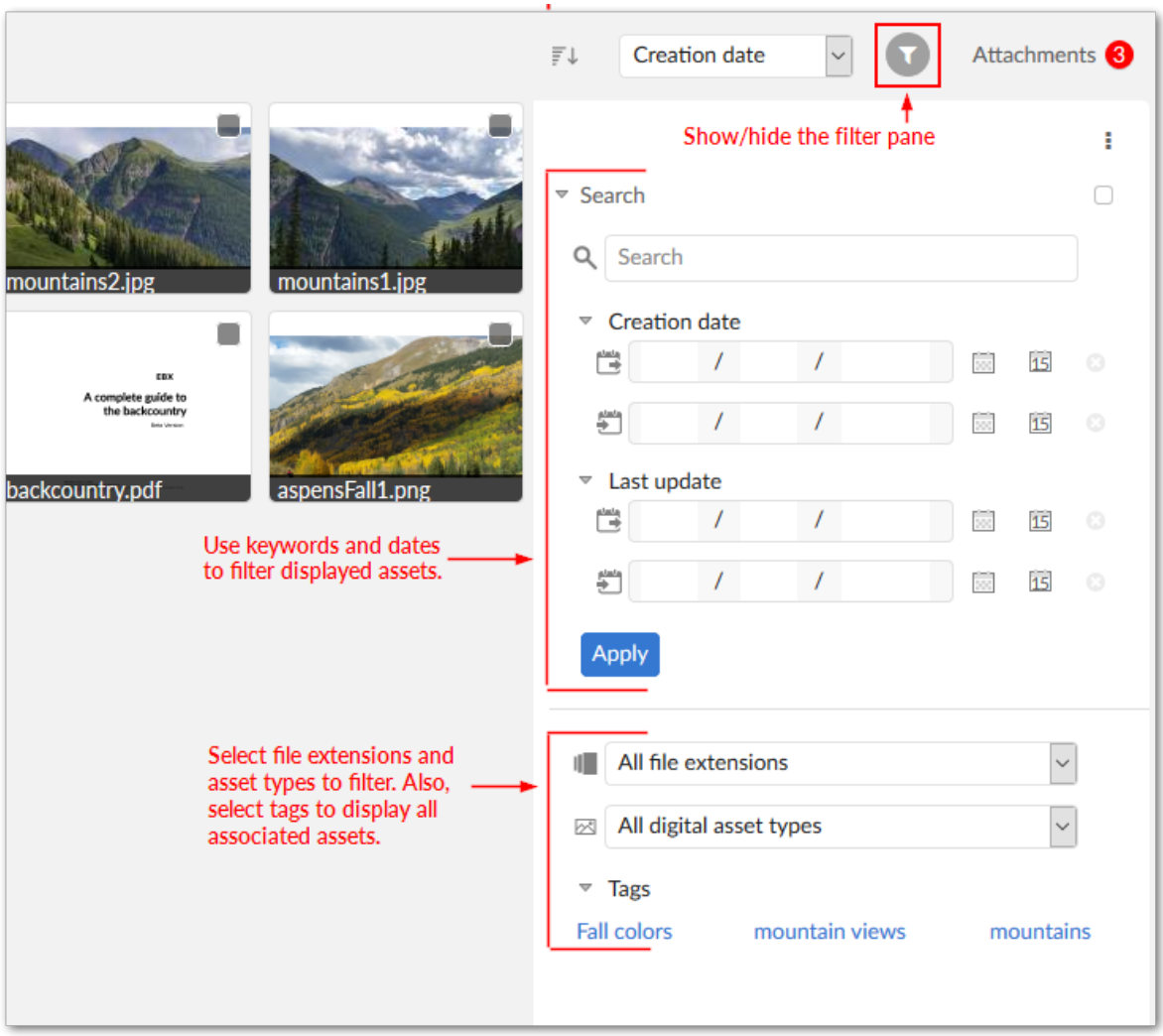
The following image and list highlight some components of this view:



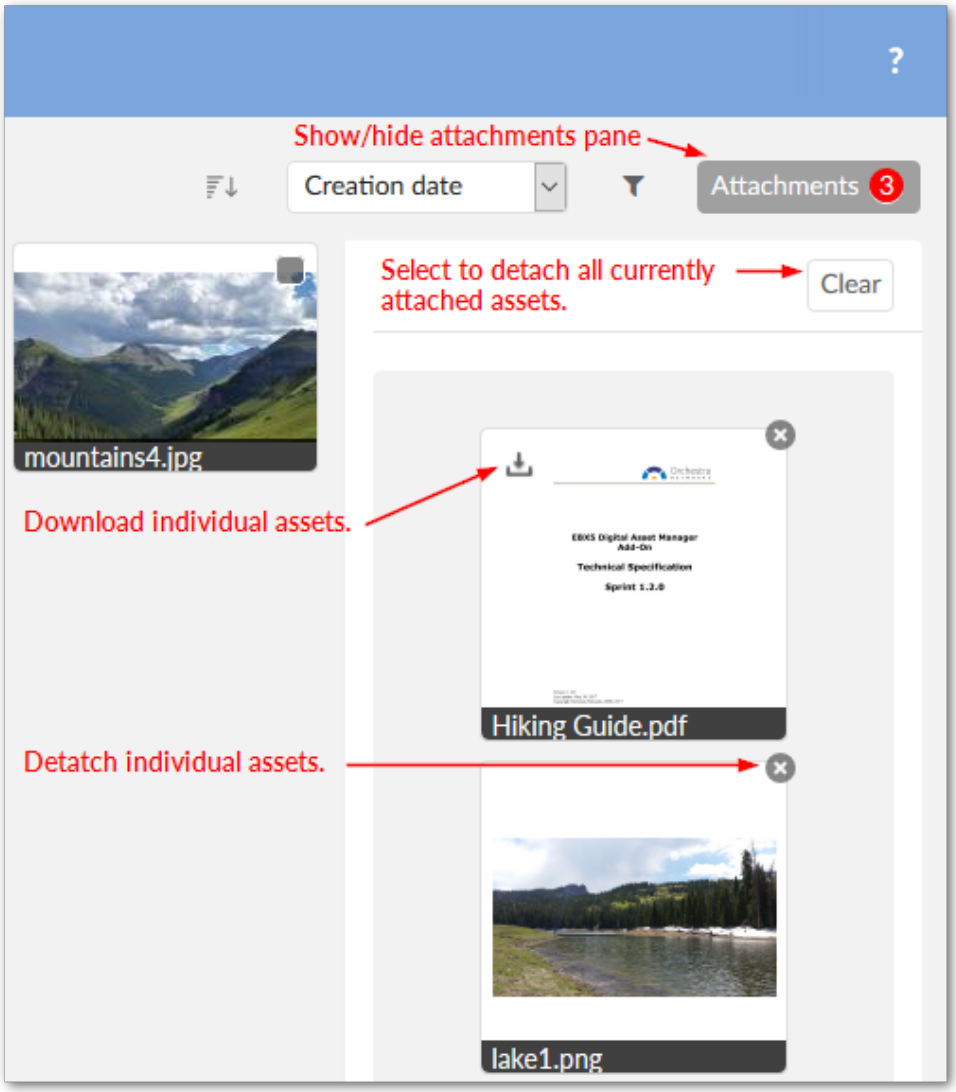
1)	The breadcrumb shows the current table and record.
2)	If it doesn't conflict with the number of allowed attachments, selecting all assets attaches all the assets to the current field. Select **Upload** to upload additional assets to the current Drive, and optionally attach them to the current field.
3)	Set the sort order to ascending or descending and use the drop-down menu to specify the sort criteria.
4)	Use the filter and **Attachments** pane to locate assets and view assets attached to the current field, respectively.

As shown below, you can use the filter to locate assets. The upper portion of the filter allows you to enter keywords and searches the asset's label, description, and physical name. You can also use a

combination of asset creation and last update dates. The lower half of the filter pane includes options to file extensions, asset types, and tags applied to assets.



You can use the **Attachments** pane to display and manage assets attached to the current field:



When you select an asset in the **Attachments** pane, an enlarged view displays. From this view you can use the icons in the top-right corner to:

- Download the selected asset. A window displays and allows you to choose the download location.
- Deactivate the asset. After deactivating an asset, only administrators can re-activate it. A deactivated asset no longer displays in the **Drive** view. To re-activate the asset: Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *Digital asset*. In the **Digital asset** table, double-click the asset to activate. Use the **State** drop-down list to update the asset, save, and close.
- Permanently delete the asset. When you permanently delete an asset, the add-on physically deletes it from the Drive.
- Share via email. Opens your email client with the selected asset attached to a new email.

Administrators can perform the same actions mentioned above. Additionally, they can use the add-on's administrative tools to backup/restore assets. The following sections describe how to:

- Edit an asset's basic information. See Editing asset information [p 51].

- Create *Tag clouds* and *tags*, and apply them to assets. See Tags and tag clouds [p 57].
- Create new versions of an asset and choose the asset considered the current version. See Managing digital asset versions [p 59].
- Create metadata categories and apply them to assets, backup and restore assets, configure additional Drive options, and set user permissions. See Additional configuration options [p 61].

CHAPTER 10

Editing asset information

This chapter contains the following topics:

1. Overview
2. Editing asset metadata

10.1 Overview

Modifications to asset metadata are either isolated, or applied globally. Your access rights, and where you make the change from determine their scope. When changes are isolated, they only impact the current dataspace. For example, when you make changes in a child dataspace, they do not propagate to the parent. Global changes are propagated to any location where an asset is used. If you:

- Are a business user and access edit functionality from a record's details view, any changes you make are isolated to the current dataspace.
- Have administrative privileges and access edit functionality using the **Manage digital assets** service, changes apply globally. When editing assets with this method, the **Manage digital assets** screen might not reflect changes users have made using the **Drive** view. Note that as an administrator you can still edit from a record's details view if you want to keep your changes local.

  When you delete a D.A.C, any asset modifications made in the local dataspace (that are related to the deleted D.A.C) are lost. Assets are reset to their initial state.

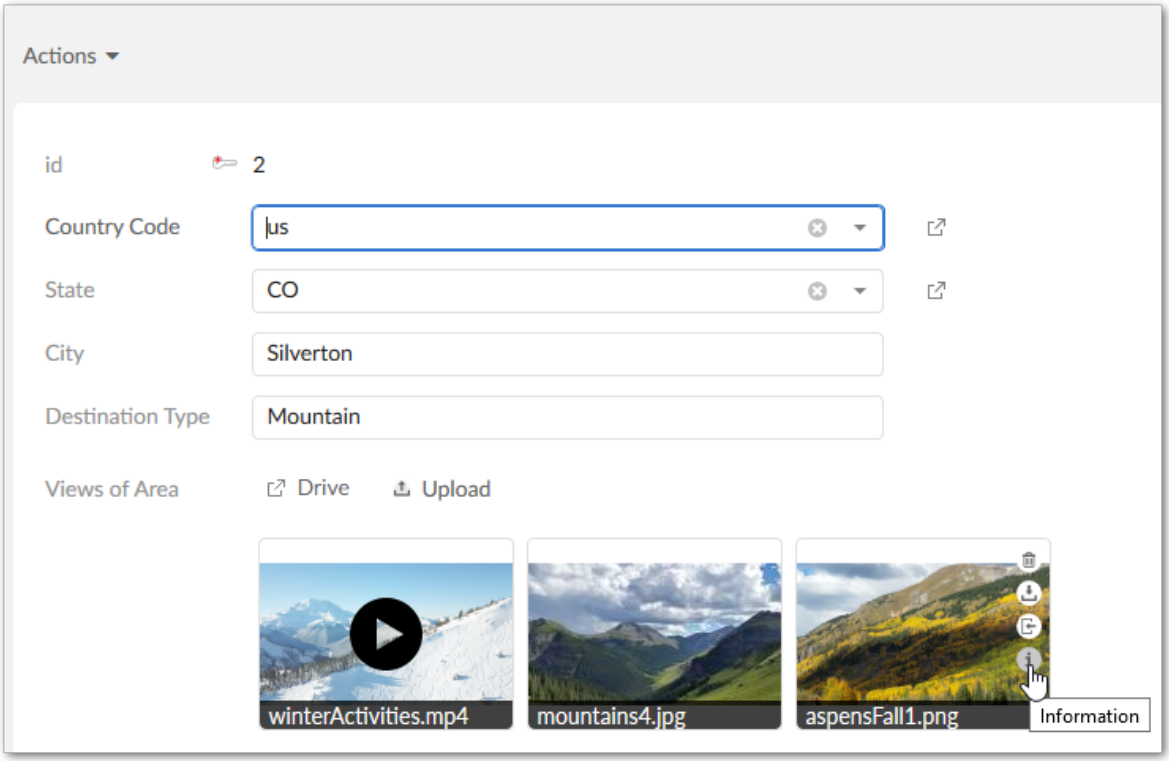
> **Note**
>
> The above behavior applies to asset versions, labels and descriptions. Changes to tags, state, and custom metadata are always applied globally.

10.2 Editing asset metadata

To edit an uploaded asset's general information, tags, metadata, and manage versions:

- As shown below, in the mediaType field, users can hover their mouse over an asset and select the information icon. If the record is read-only, its information displays, but actions aren't permitted.

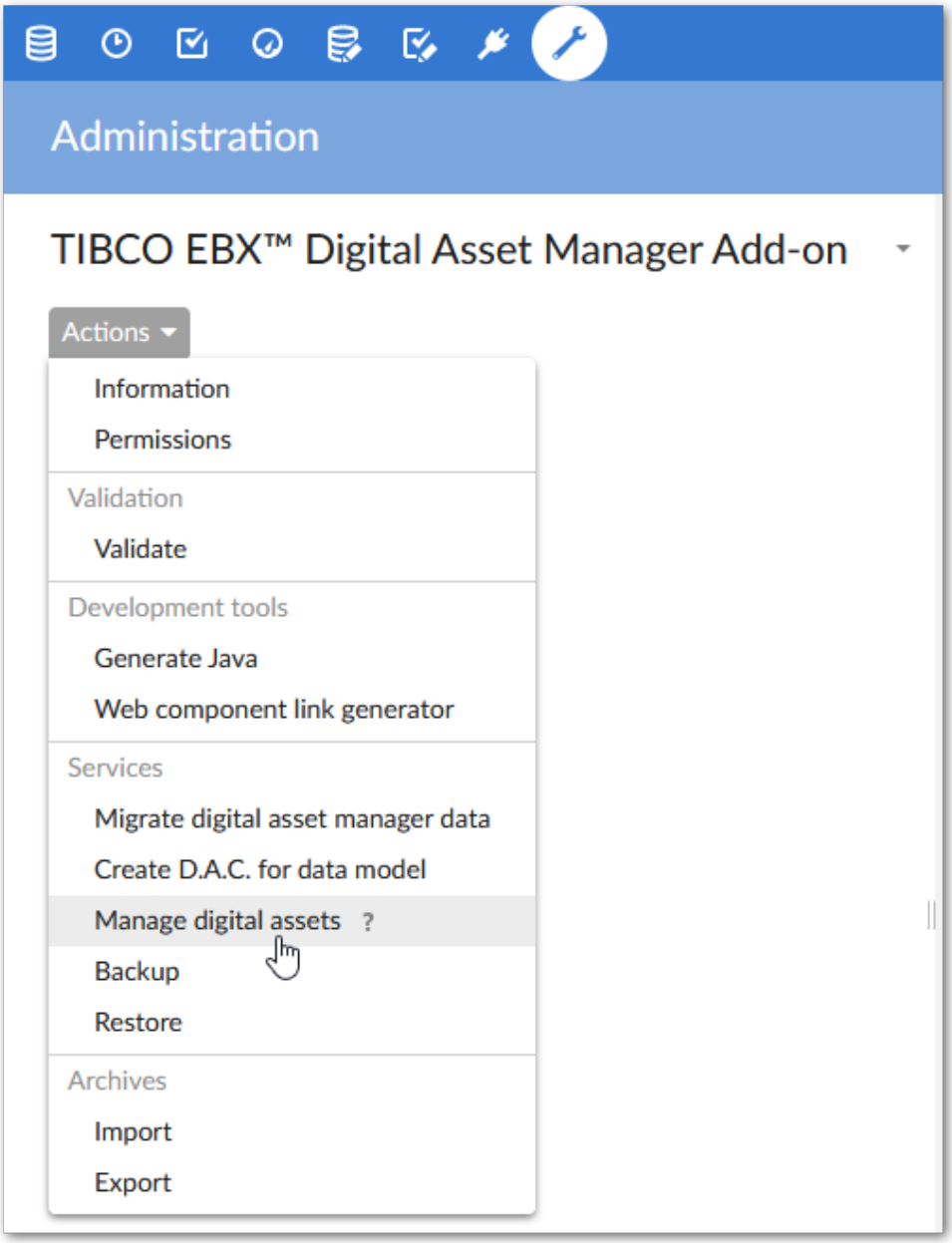
> **Attention**
> Changes made using this method are isolated to the current dataspace.



- From **TIBCO EBX® Digital Asset Manager Add-on**, administrators can select **Manage digital assets** from the **Actions** menu. This view allows you to switch between Drives to edit any asset.

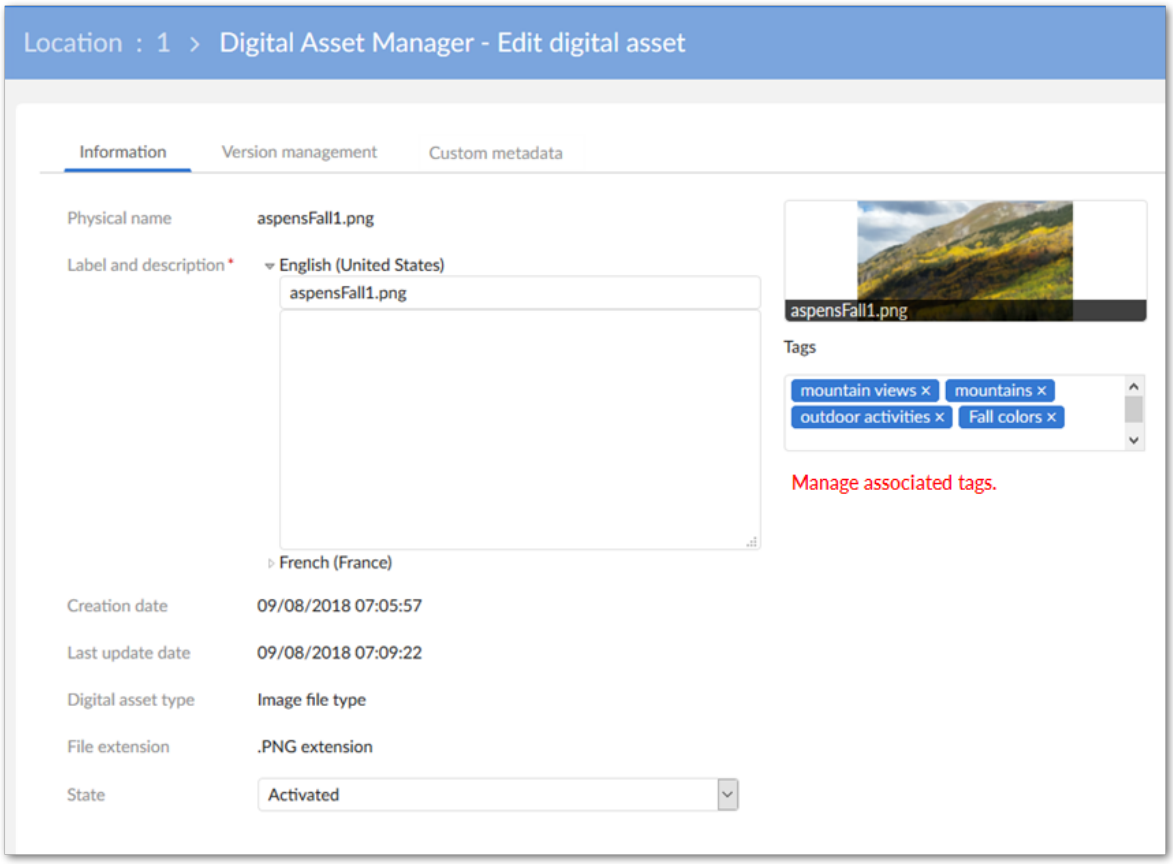
Once you've located it, click the desired asset and in the **Details** pane, click the **Edit** icon above the image.

> **Attention**
> Changes made using this method are propagated to all dataspaces.

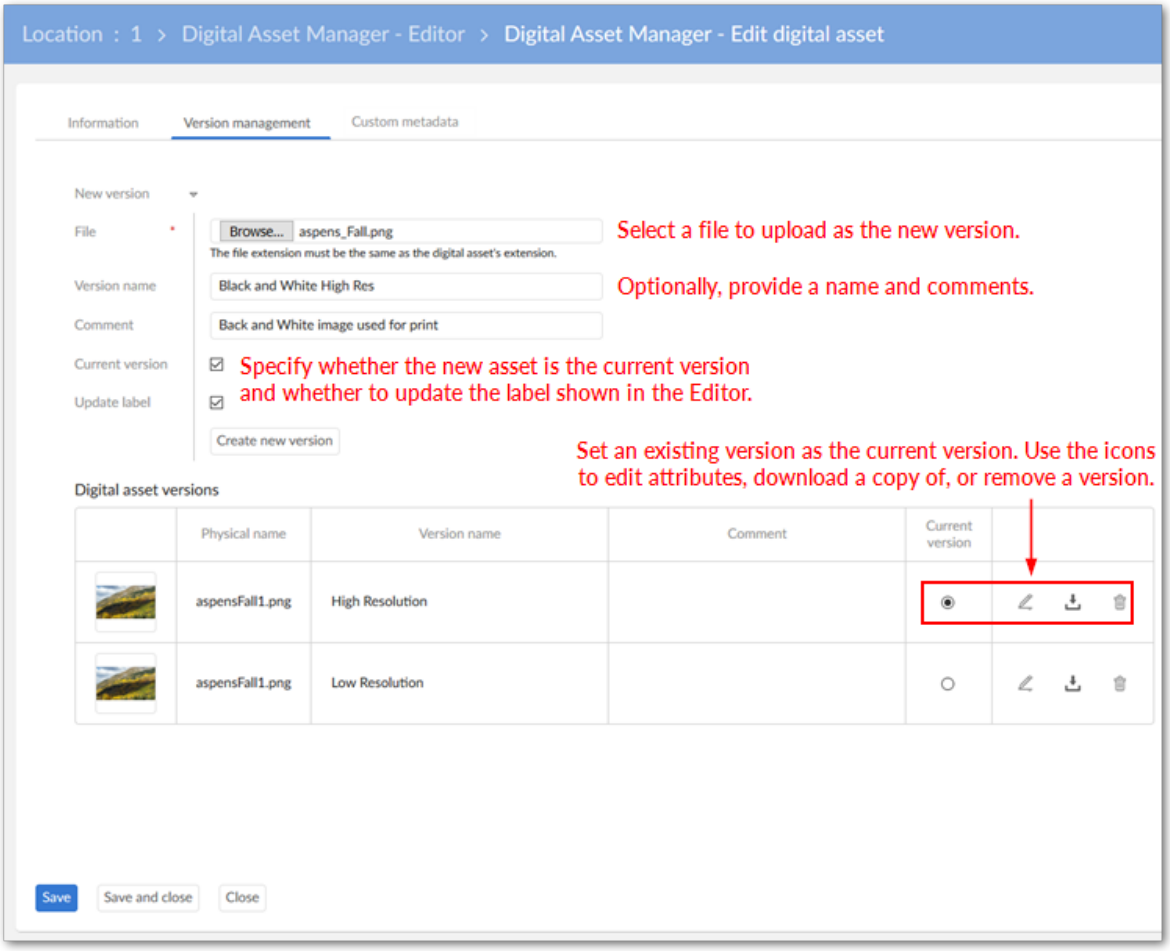


The following three tabs display in the **Edit digital asset** screen:

- The **Information** tab allows you to view and edit the asset's label, description, state and tag information. You can only edit the asset's state if you have sufficient permissions. The following section, Tags and tag clouds [p 57], describes these features in more detail.



- The **Version management** tab allows you to create new asset versions, edit the version name and comment attributes, delete asset versions, download versions, and select the current version. The Managing digital asset versions [p 59] section covers these topics.



- The **Custom metadata** tab allows you to add custom metadata to an asset. See Creating metadata categories [p 61] for more information.

CHAPTER 11

Tags and tag clouds

This chapter contains the following topics:

1. Working with tags
2. Using tag clouds

11.1 Working with tags

Tags provide descriptive metadata about assets. By creating tags based on relevant keywords that accurately define and describe assets, you enable filters and searches to retrieve the best results. To ensure consistency and accuracy, it's a good practice to double-check all asset metadata against an expected outcome.

A *tag cloud* facilitates grouping of related tags. A single tag can belong to multiple clouds. Administrators can make one or more tag clouds available to each D.A.C.

As described in the following section, you can create tags dynamically when uploading assets, or predefine tags to apply to assets at anytime.

Creating tags and applying them to assets

You can create tags in the following ways:

- When uploading an asset, you can enter a new, or select an existing tag. If you enter a new tag, the add-on automatically adds it to the asset and stores the new tag for later use. The add-on stores these tags in the tag clouds associated with this field through the D.A.C.
- If you have access to EBX® administrative features, you can navigate to *Administration > User interface > TIBCO EBX® Digital Asset Manager Add-on > Classification* and use the **Tag** and **Tag cloud** tables for creation. These tags will be available to users when uploading or editing an asset.

In addition to applying tags when uploading an asset, you can add tags when:

- Editing an asset:
  - When viewing an asset, click the **Edit** icon to display the **Edit digital asset** screen.
  - From the **Information** tab, you can add, or remove tags. Just be sure to save your changes and refresh the asset when finished.
- Managing assets as an administrator. While under *Administration > User interface > TIBCO EBX® Digital Asset Manager Add-on*, open the **Actions** menu and select **Manage digital assets**. Select the desired asset and click the **Edit** icon to add, or remove tags.

11.2 Using tag clouds

If you have administrative privileges, you can create tag clouds and manage which tags are included in each cloud. The following steps describe the process:

1. In the **TIBCO EBX® Digital Asset Manager Add-on** administration area, navigate to *Classification > Tag cloud*.
2. Click the '+' icon to create a new *tag cloud* and provide a label and description.
3. After saving, you can add tags to this cloud by creating new, or using the **Actions** menu to associate existing tags with this cloud.

You can open any tag cloud and use its **Actions** menu to:

- Associate existing tags with this cloud.
- Delete tags from the system.
- Remove tags from the cloud.
- Duplicate tags for modification (tags must be unique)
- Move tags from this cloud to a different cloud.

CHAPTER 12

Managing digital asset versions

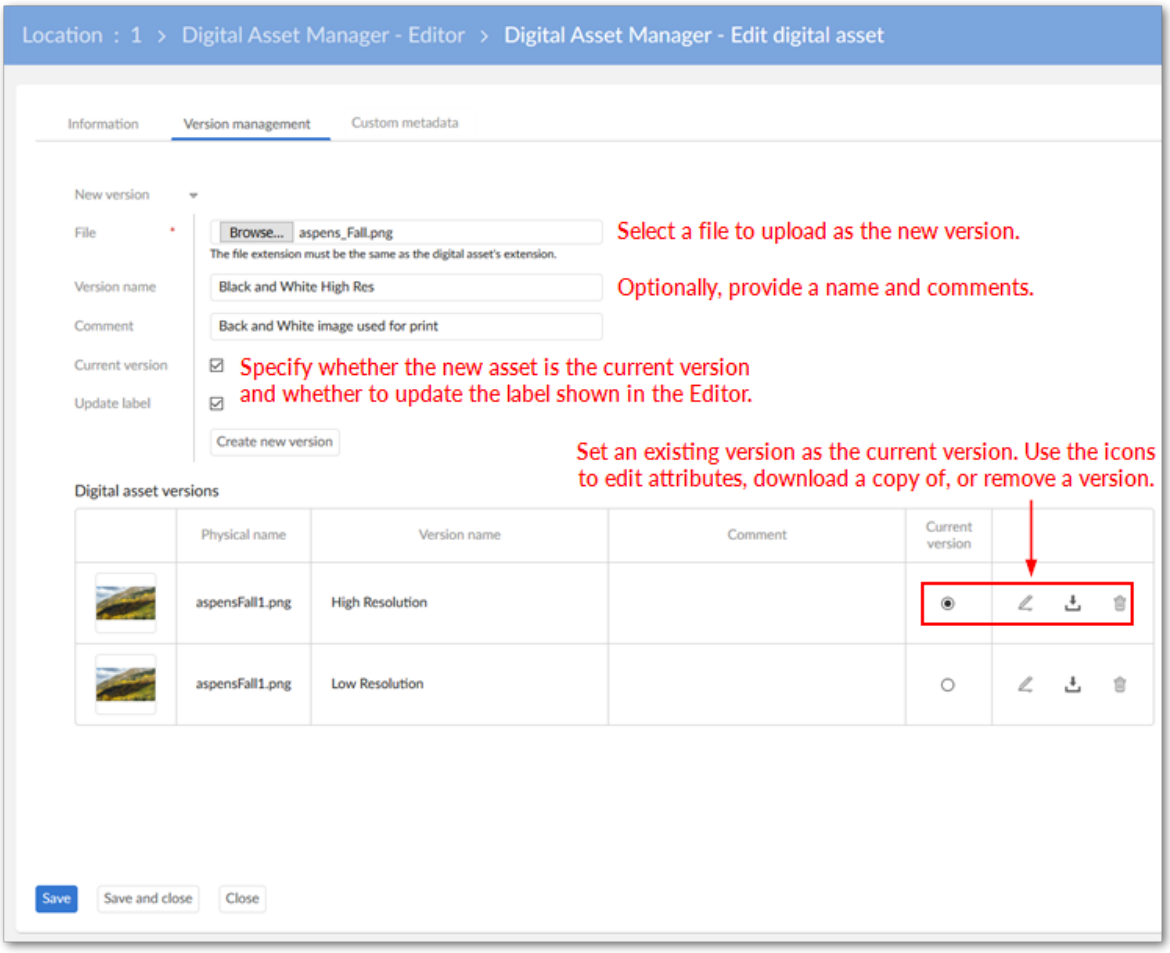
Keeping versions of digital assets allows you to re-purpose versions, or revert to an older version if needed. Even though multiple asset versions may exist, only versions designated as the current version display in the **Mosaic** and **Drive** views.

> **Note**
>
> You can attach duplicate files to the same field. When the add-on sees that the asset is a duplicate, a screen displays that allows you to preview the asset and choose whether to continue, or cancel.

To create a new version of an asset or change the current version:

1. Access the **Edit digital asset** view:
   - When viewing a record with attached assets, select an asset's icon. Any changes made to the asset's version are isolated to the current dataspace.
   - Administrators can select **Manage digital assets** from a dataset's **Actions** menu. Then select an asset to open an expanded view and select the icon. Any changes made to the asset's version are propagated to all dataspaces.
2. Select the **Version management** tab. (If you do not need to upload a new file and are only switching to a different version, tick the radio button in the **Current version** column of the desired asset, and skip to step 7).
3. Browse to upload a new file. The uploaded file must have the same file extension as the original asset.
4. Optionally, add a **Version name**, **Comment** and select the **Current version** checkbox to automatically make this the current version. If you designate this asset as the current version, you

can also enable the **Update label** option. This option updates the asset's label shown in the **Drive** view to the current version's file name.



5. Click **Create new version**. The add-on automatically applies any existing tags and metadata to the new version.
6. Save and close to exit.

CHAPTER 13

Additional configuration options

This chapter contains the following topics:

1. Creating metadata categories
2. Backing up and restoring
3. Allocating Drive disk space
4. Limiting image size
5. Specifying the view mode
6. Whitelisting and blacklisting file formats
7. Setting user permissions

13.1 Creating metadata categories

In addition to tags, the add-on allows you to create your own categories of metadata to enrich assets. After applying metadata categories to assets, you can further describe the assets by adding metadata values. To perform these tasks:

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *Reference data* > *Metadata nature* and create a new record to define a category of metadata.
2. Access the **Edit digital asset** view:
   - When viewing a record with attached assets, select an asset's icon.
   - Administrators can select **Manage digital assets** from a dataset's **Actions** menu. Then select an asset to open an expanded view and select the icon.
3. Select the **Custom metadata** tab and add a new value. While in this tab, you can also edit and remove existing metadata. Be sure to save and refresh after completing your changes.

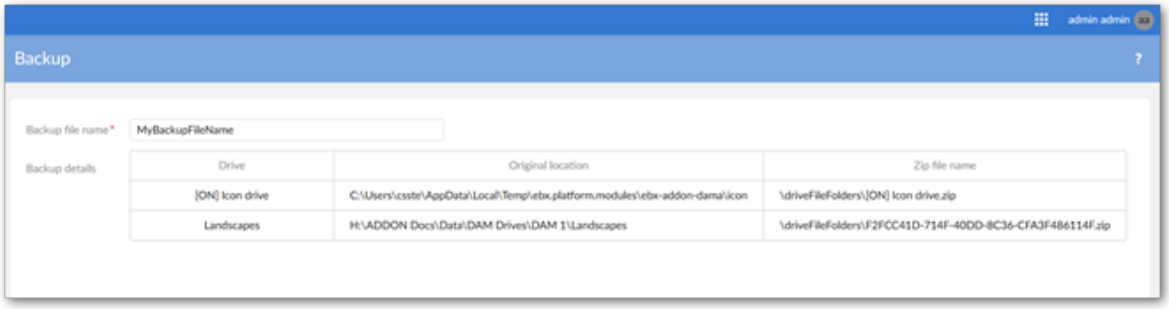
13.2 Backing up and restoring

The add-on allows you to take a single action to create a backup of all the assets it manages. When you restore the files, you'll have to enter new paths for Drives included in the backup.

To create a backup:

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on*.
2. From the *Actions* menu > *Services* group, select **Backup**.

3. Enter a name for the zip file and click **Backup**.



4. After the process completes, you can select **Download** to save a copy of the backup file.

To restore assets:

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on*.
2. From the *Actions* menu > *Services* group, select **Restore**.
3. Click **Choose file** to navigate to the backup file's location.
4. After selecting the file click **Next**.
5. A table listing all backed up Drives displays. Enter new paths, and click the **Restore** checkbox for Drives you want to restore. An important note: *If you choose not to restore specific Drives, they will no longer be available.*
6. Click **Restore**.

13.3 Allocating Drive disk space

When you create a Drive configuration, you point to a system folder to use as the *Drive's physical location*. You can specify how much space you want to allocate to the Drive with the **Max usable space** property. Once asset storage reaches this threshold, you can no longer upload assets to this *Drive*, unless you physically delete assets to make room.

13.4 Limiting image size

You can use an image configuration to limit the physical size (width/height) of images. Once you've created a configuration, you can add it to any Drive, or D.A.C. The configuration used by a D.A.C. cannot allow for larger images than that used by the Drive.

13.5 Specifying the view mode

A view configuration determines how assets display when attached to a data model field. Administrators can create configurations and use them in D.A.C.s. To create a view configuration:

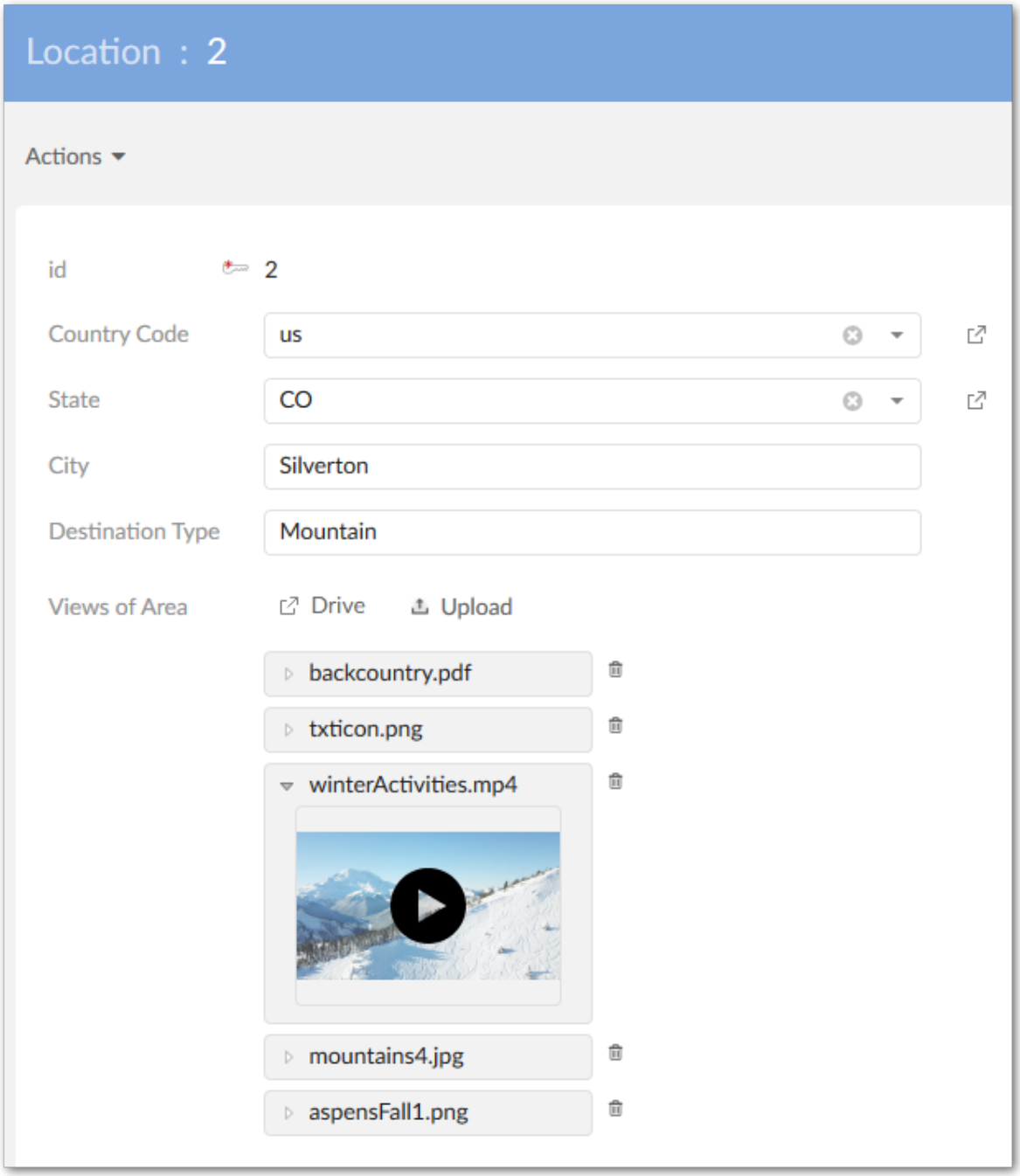
1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *D.A.C.* > *View configuration*.
2. When you open an existing, or create a new record you can choose from the following options:
   - **Mosaic view**: Displays a tiled view of the assets. When you select this option, you'll be able to set the number of assets that display in rows and columns. You can also choose whether the view scrolls vertically, or horizontally to display additional assets.

- **List view**: Displays assets in vertical list.
- **Table view**: Enables display asset thumbnails when viewing a table. Use the available options to set the thumbnail size.

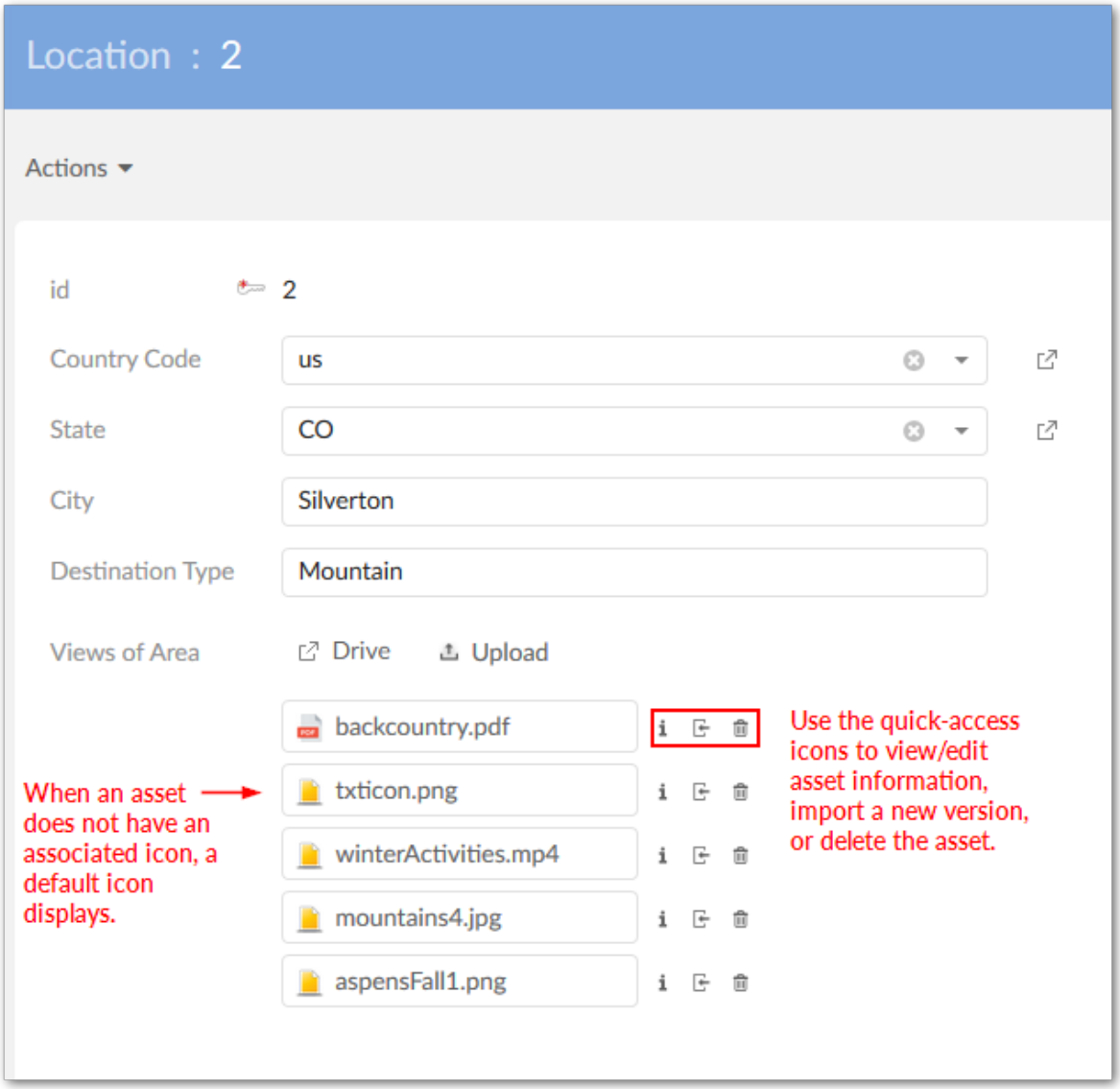
3. Decide whether assets display as their associated icons, or as thumbnails. When you choose:
   - To display icons, the assets display with their associated icon (if one exists).
   - To display thumbnails, you can set the thumbnail size and whether users can select an asset to expand it.
4. Choose whether quick action icons always display, or only display when users mouse over an assets thumbnail.

The following list of sample images show the outcomes of using the options mentioned above.

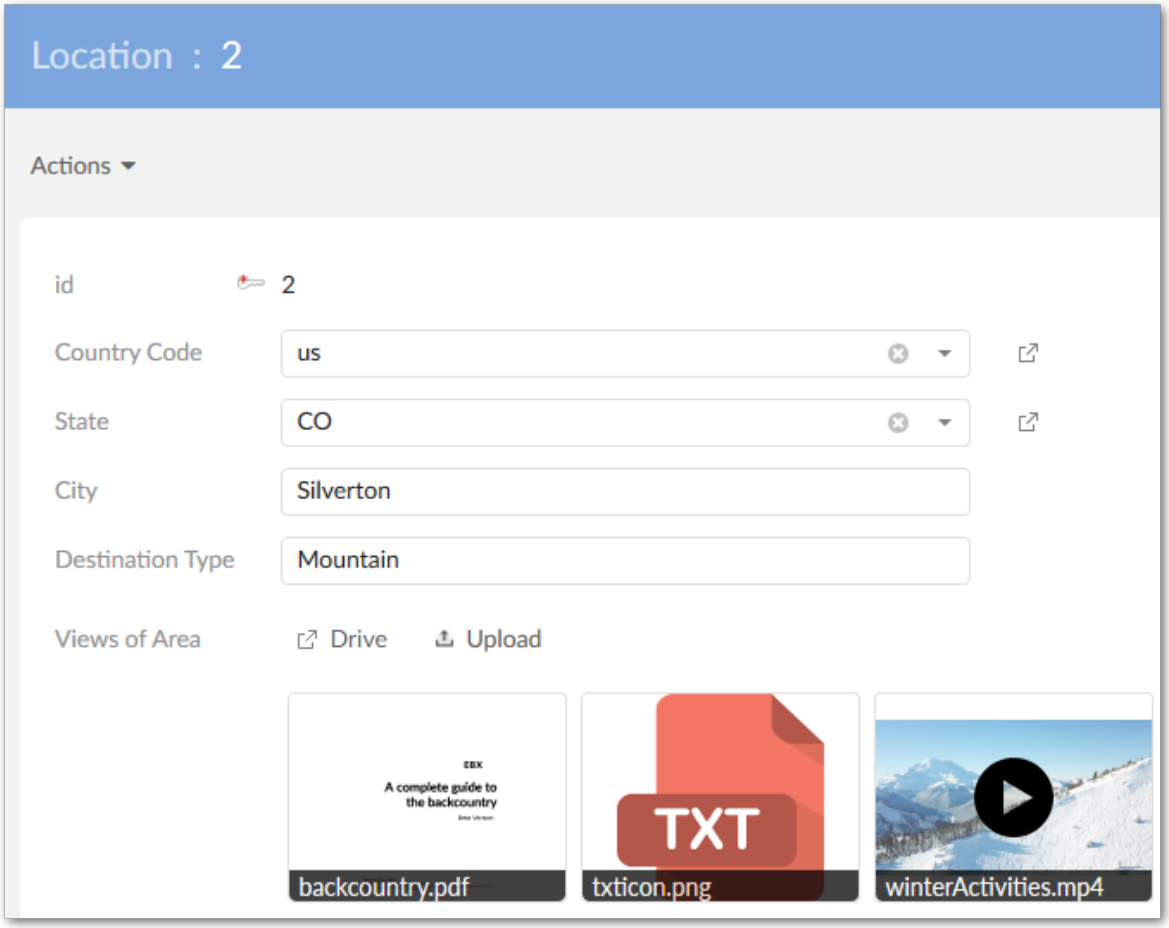
- List view using thumbnails:



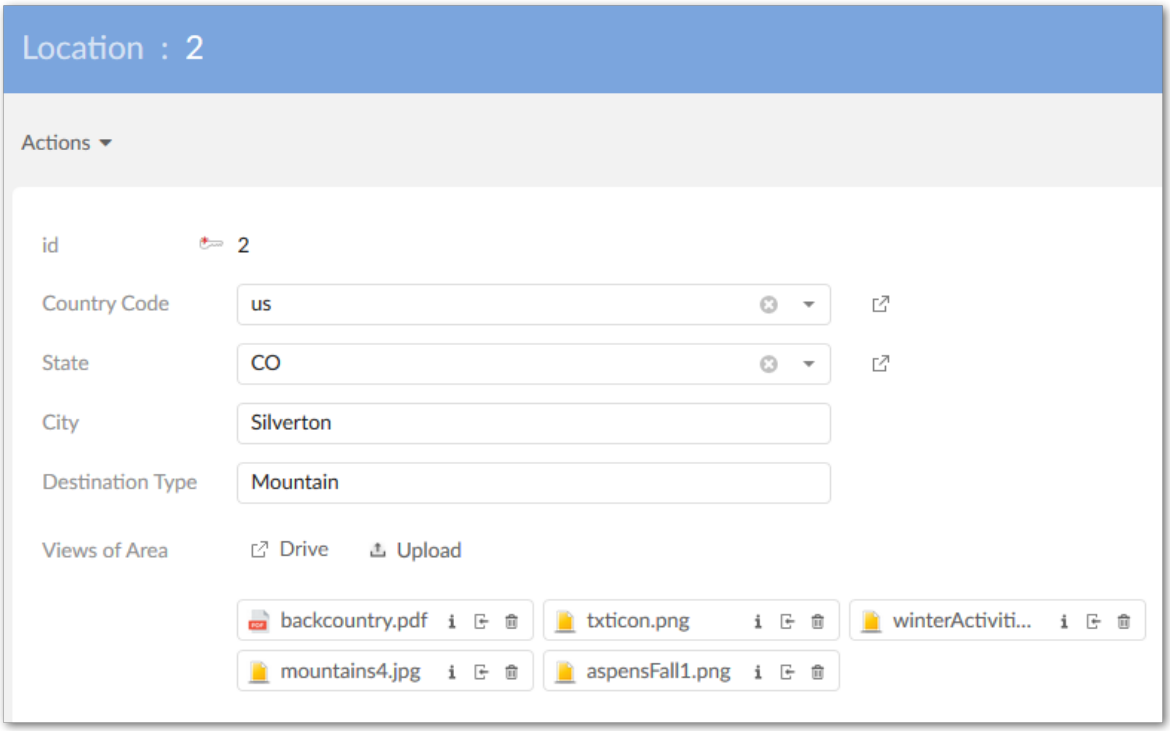
- List view using icons:



- Mosaic view with thumbnails:



- Mosaic view with icons:



13.6 Whitelisting and blacklisting file formats

By default, the add-on allows upload of all file types. However, you may want to allow a set of file formats and block others, whitelist and blacklist respectively. The add-on allows you to add to a set of predefined file extensions. At the Drive level, you can restrict asset upload to only these file extensions. Alternatively, you could manually enter file extensions to blacklist. The D.A.C. level inherits permissions specified at the Drive level. However, you can override these settings. For example, you may have two different D.A.C.s linked to the same Drive. You could change permissions in the D.A.C.s so that only one of them can upload PDF files.

The following two settings are located in *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* in the **Drive** and **Digital asset component** tables:

- When enabled, the **Restrict file upload** option in a Drive record restricts asset upload to only the asset file types listed in the *Reference data* > *File extension* table. When enabled in a D.A.C. record, you can further refine restrictions by selecting options in the **Supported extensions** field. If you would like to include additional file extensions in the **Supported extensions** list, see the Adding new file extensions [p 68] section below.
- The **Blacklisted extensions** option only displays when the **Restrict file upload** option is disabled. This field allows you to manually enter file extensions to exclude from a Drive or D.A.C. You can use commas, semi-colons, periods, or whitespace to separate a list of extensions to blacklist. For example: `exe`, `bat`, `msi`, etc.

Adding new file extensions

To enable upload and management of additional file formats:

> **Note**
>
> The following steps require administrative privileges.

1. Navigate to *Administration > User interface > TIBCO EBX® Digital Asset Manager Add-on > Reference data > File extension*.
2. Create a new record and fill in the following fields:
   - **Business code**: A user friendly name for this file extension.
   - **Filename extension**: Enter the filename suffix used to identify this file type.
   - **Digital asset type**: Choose an asset type from the drop-down list. This list populates with options from the **Digital asset type** table. You can create types in the table, or click **+ Create** in the drop-down list to create a new type.
   - **Icon** → *Select*, or upload an icon used to identify this file extension type. The add-on stores all icons in the **[ON] Icon drive**.
3. Save and close.

13.7 **Setting user permissions**

If you have administrative access, you can set user permissions. Drives and D.A.C.s apply permission settings to user profiles and grant/restrict viewing, uploading, downloading, versioning and deleting. With the ability to apply permissions at two levels *(Drive/D.A.C.*), you can fine-tune asset management.

As an example, you may want allow a user to upload assets to a Drive. However, you may not want this same user to upload, attach, or edit any assets attached to a particular field. To accomplish this, you can grant appropriate access for the Drive and restrict access with the *D.A.C.*

> **Note**
>
> The add-on supports EBX® restriction policy functionality. For more in-depth information on restriction policies, see the *EBX® User Guide* and *Reference Guide*.

CHAPTER 14

Finding and removing unused assets

The add-on provides functionality to locate and remove assets not attached to a field—an effective way to tidy-up the **Digital asset** table. As described below, administrators use a two part process to identify unused assets and select assets for removal:

1. Navigate to *Administration > User interface > TIBCO EBX® Digital Asset Manager Add-on > Digital asset*

   and select the **Digital asset** table.
2. From the table's **Actions** menu, select **Identify unused assets**. After analysis, the add-on displays only assets not attached to a field.
3. Select any assets you wish to remove and from the table's **Actions** menu, select **Permanently Delete (Physically delete)**. Press **OK** in the warning dialog to remove all selected assets.

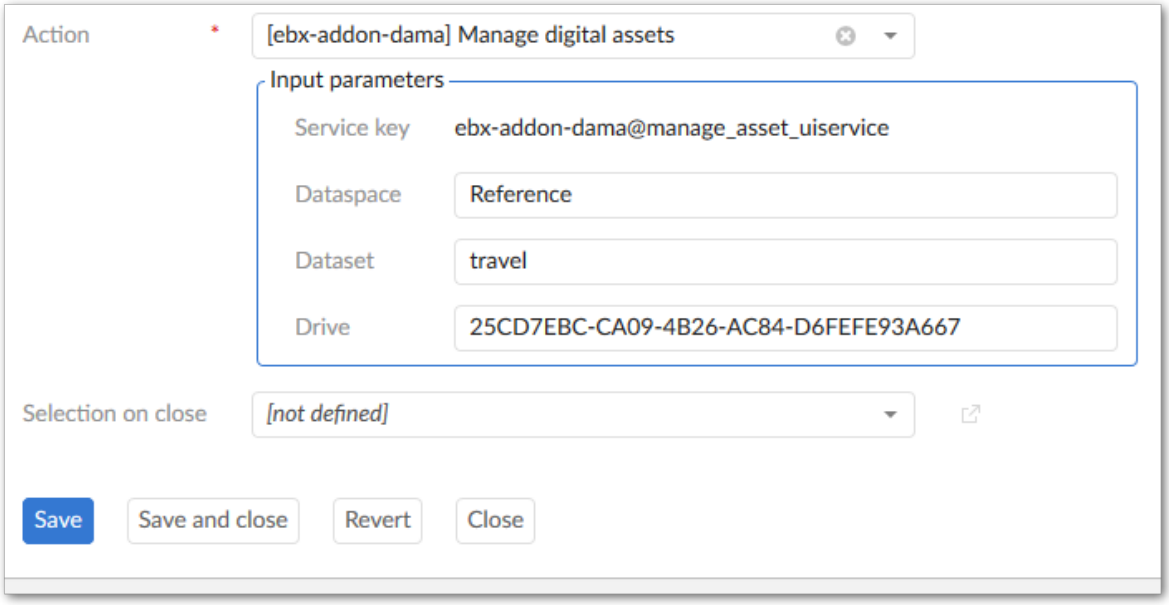
CHAPTER 15

Enabling asset management from perspectives

Administrators can attach the **Manage digital asset** service to EBX® perspectives. This service allows full control over all assets contained in a Drive. See the EBX® product documentation for more information on creating perspectives and how they work.

To add add-on functionality to a perspective:

1. Select **[ebx-addon-dama] Manage digital assets** from the **Action** menu.



2. Use the required **Dataspace** and **Dataset** fields to specify the location of the media type fields.
3. Enter one or more Drive id's—which can be found in the add-on's **Drive** table—in the **Input parameters** group **Drive** field. When entering multiple Drives, separate the entries using a semicolon. If you leave this field blank, the perspective will be able to access any Drive associated with the specified dataset.

CHAPTER 16

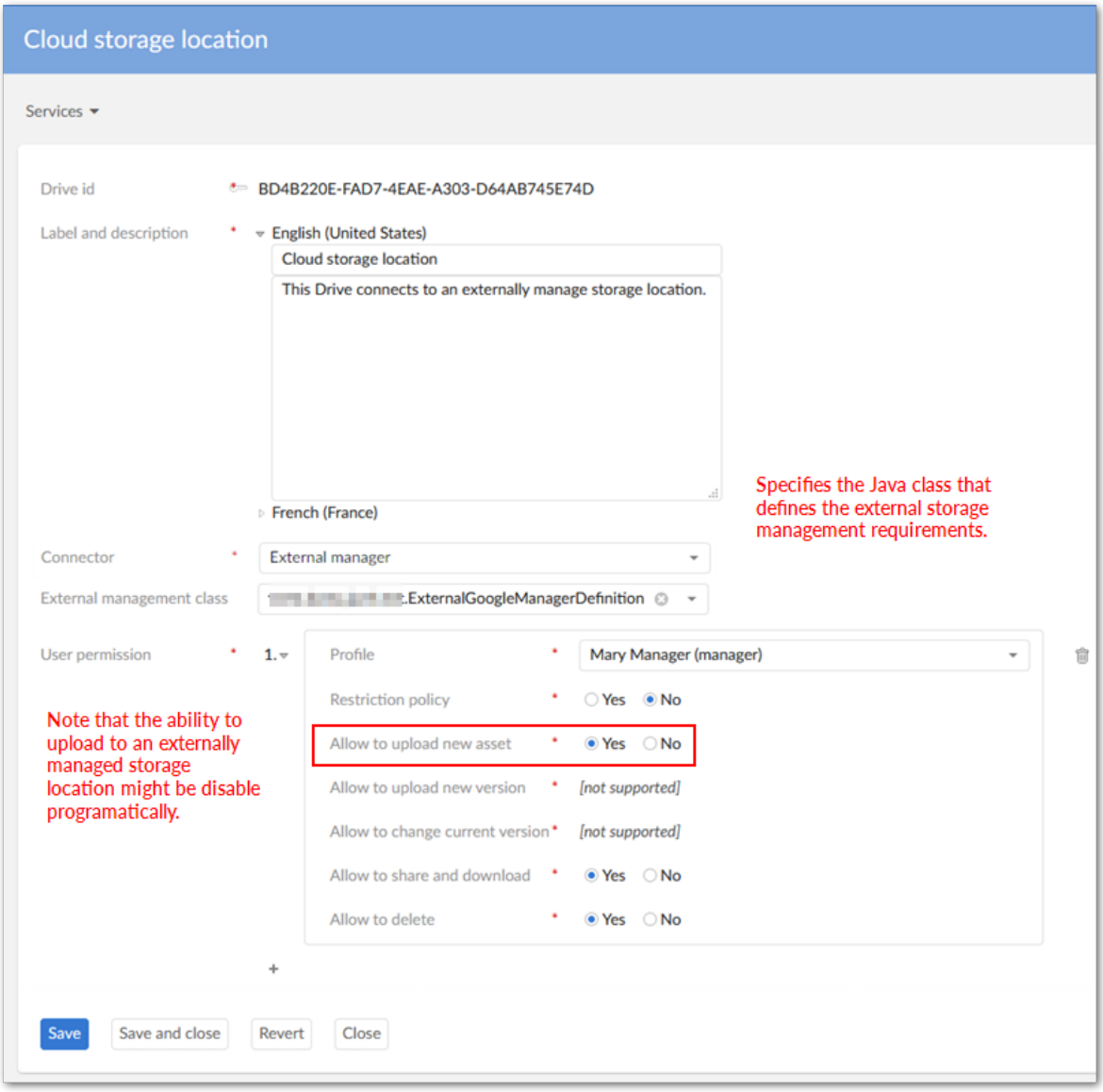
Configuring an external Drive

The add-on allows administrators to create a Drive that connects to an externally managed asset storage location. For example, you could connect to a cloud storage location. This allows externally managed assets to be attached to EBX® data model fields.

A developer must programatically implement connection details and any third-party requirements using the add-on's API. For details and examples, see Connecting to alternative storage locations [p 92]. Once necessary implementation steps are complete, you can create a Drive that connects to external storage by:

1. Navigating to *Administration > User interface > TIBCO EBX® Digital Asset Manager Add-on > Drive* and open the **Drive** table. This table stores asset storage configuration information.
2. Creating a new *Drive* configuration. When filling required fields, you can hover your mouse over the property and click the **?** icon to view online help, or you can refer to the Drive group [p 111]

section in this guide. The following image highlights some configuration options for an external Drive.



3. After saving and closing, you can link the Drive to a data model field using a Digital Asset Component (D.A.C.). For instructions, please see Configuring the data model [p 15] and Linking a Drive and data model field [p 23].

CHAPTER 17

Connecting to an Amazon S3 bucket

This chapter contains the following topics:

1. Concepts overview
2. Configuring a Drive and Amazon S3 connector

17.1 Concepts overview

Administrators can create a Drive that connects to an Amazon S3 storage location. This allows you to attach assets to fields in EBX® while managing their storage outside of EBX®. Most add-on features are available when using this type of Drive. However, the add-on does not handle versioning in this case and its features are disabled.

Setup requires updating EBX® configuration properties in addition to add-on configuration tasks. The following section provides the steps to complete these tasks.

Attention

Some steps require access to the server where EBX® is running and others require administrative access to add-on configuration settings.

17.2 Configuring a Drive and Amazon S3 connector

To configure a Drive to use an Amazon S3 connector:

1. Add the following properties to your `ebx.properties` file:
   - `ebx.addon.dama.aws.accessKey=<YourAccessKey>`
   - `ebx.addon.dama.aws.secretKey=<YourSecretKey>`

   Note

   You might be required to stop and restart your server to complete this step.

2. From the **Administration** panel, select: *TIBCO EBX® Digital Asset Manager Add-on > Drive > Drive*.
3. Create a new record.
4. After adding a label and description, use the **Connector** menu to select **Amazon S3**.

Fill in the required fields. If you have questions about one of the properties, hover your mouse over it and select the '?' to view its tooltip.

The **Bucket name** you enter must match your S3 bucket.

5. After saving and closing, you must create a D.A.C. [p 23] to connect the Drive to a data model field.

CHAPTER 18

Performing bulk upload and attach

This chapter contains the following topics:

1. Overview
2. Bulk upload
3. Attaching assets with batch processing

18.1 Overview

Uploading and attaching assets with the add-on's **Upload** and **Attach** services works well when you have a limited number of assets because:

- You are limited to uploading 300 assets at one time.
- You need to open each record to attach assets.

To save time and effort when you are working with a large number of assets, you can perform a bulk upload of zipped assets and then optionally attach assets automatically with batch processing. The following sections cover these topics:

- Bulk upload [p 77]
- Attaching assets with batch processing [p 78]

Attention

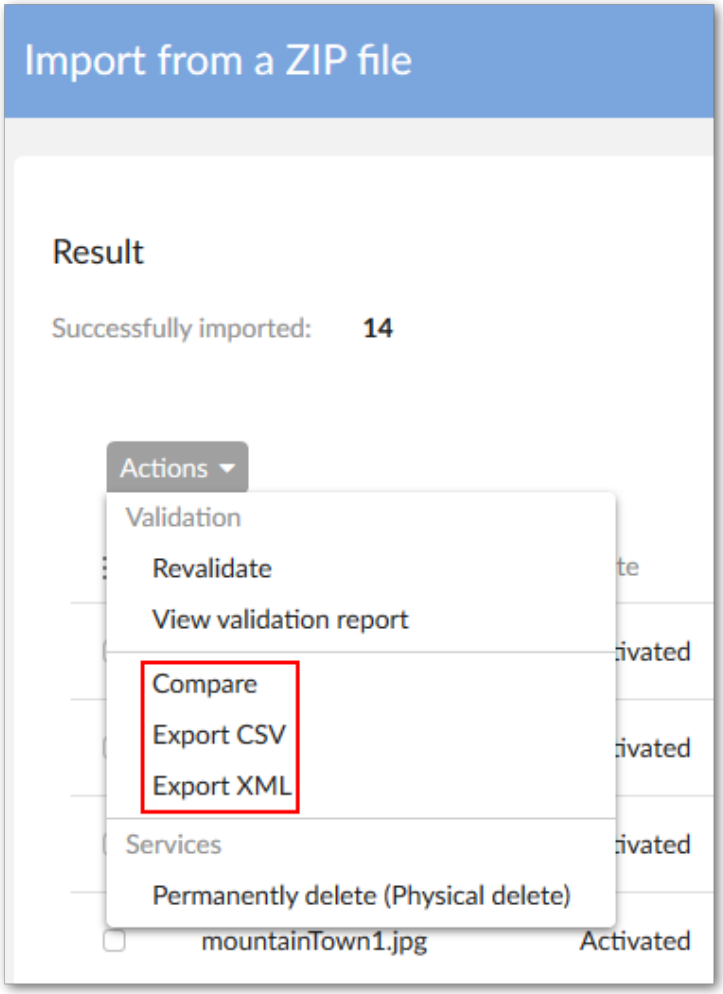
To perform a bulk upload of assets, you must have access to the **Administration** panel. Additionally, to automatically attach assets, you must have access to the most recent version of the EBX® Data Exchange Add-on (New).

18.2 Bulk upload

To perform a bulk upload:

1. Navigate to *Administration* > *User interface* > *TIBCO EBX® Digital Asset Manager Add-on* > *Digital asset* and selecting the **Digital asset** table.
2. Open the **Actions** menu and select the **Import from a ZIP file** service.
3. Browse to select the desired file, and specify the upload location using the **Drive** drop-down list.

4. After clicking **Import**, the add-on keeps you apprised of its progress and displays a result page upon successful completion.
5. From the **Result** page's **Actions** menu (shown below), you can export a report of the results, or select multiple records to compare. Also, if the import fails, you can download a copy of the log file.



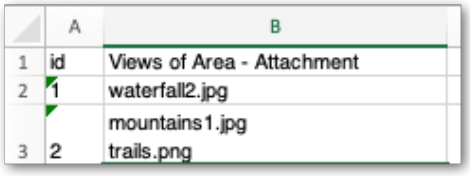
18.3 Attaching assets with batch processing

The EBX® Data Exchange Add-on (New) can automatically attach assets using the CSV or Excel import services. When performing the import, you use the **Digital asset UUID lookup** transformation function in the **Mapping** screen.

Pre-requisites:

- A working EBX® Digital Asset Manager Add-on configuration, including Drive and D.A.C., for the table location where you want to attach assets.
- The Drive linked to the table must already contain the assets you want to attach.
- The latest version of the EBX® Data Exchange Add-on (New)
- A CSV or Excel file to map record primary keys and physical file names of assets. This file specifies where to attach assets and must include the following:

- Columns that correspond to the primary key and Media-type fields of the records that you want to attach assets to. For example, in the following image the first column is the table's PK and the second is the media-type field.



	A	B
1	id	Views of Area - Attachment
2	1	waterfall2.jpg
		mountains1.jpg
3	2	trails.png

- The fields in the Media-type column must contain a list of the file names of assets already uploaded to a Drive. Only exact names are valid and must include the file extension. Use the list separator defined in the EBX® Data Exchange Add-on to distinguish between file names in the list.

> **Note**
>
> To obtain a CSV or Excel file with all primary keys and the columns ready for assets, you can perform an export with only the PK and media-type fields mapped.

To use an import operation to attach assets:

1. Navigate to the table where you want to import assets.
2. From the table's **Actions** menu, select *Data Exchange (New)* > *Import CSV* or *Import Excel*.
3. Use the **Source** option select the file containing the primary keys and lists of assets.
4. Select **Mapping**.
5. Apply the **Digital asset UUID lookup** transformation function:

   Note that if a mapping was automatically generated, you must delete it before completing the following steps.

   1. From the list **Transformation functions**, under **Cross reference**, drag the **Digital asset UUID lookup** function to the mapping area.
   2. In the source table on the left, drag the connection point from the field containing the list of assets to the input (left side) of the transformation function.

      Connect the output of the transformation function to the Media-type field in the target table.
6. Select **Import**.

   If an error occurs for individual records or assets, the import will continue with the other records and assets. Any errors that occur display on the import result screen.

Developer Guide

CHAPTER 19

REST service operations

This chapter contains the following topics:

1. Overview

19.1 Overview

The TIBCO EBX® Digital Asset Manager Add-on provides features that allow you to use REST services to get asset information. Responses are returned in the JSON format shown below where `status` indicates completion success, `message` returns any messages, and `data` is the response data if successful and null otherwise:

```
{
    "status": "success",
    "message": "Get success",
    "data": { }
}
```

See the following sections for details and examples:

- Retrieve an asset's URL [p 83]
- Get all registered mediaType fields [p 84]
- Getting media type field attachment info [p 85]
- Getting all assets in a Drive [p 86]
- Attach assets to a record [p 87]
- Detach assets from a record [p 88]
- Upload an asset from a record [p 89]

Retrieve an asset's URL

The following table provides an example of asset URL retrieval:

> **Note**
>
> This REST operation was deprecated in the TIBCO EBX® Digital Asset Manager Add-on 2.0.0 release.

Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/getAssetInformation?assetUUID=<anAssetPrimaryKey>`
Request parameters	`anAssetPrimaryKey`: The asset's primary key.

Sample response	{
	"status": "success",
	"message": "Get success",
	"data": {
	"uuid": "A3CE402E-0A27-4151-9093-49366491892E",
	"name": {
	"locale": "en_US",
	"label": "asset_01.PNG",
	"description": ""
	},
	"preview_url": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=A3C...892E&versionId=B8D...A9B"
	"&height=150&width=150&original=true&uuid=fa6...0ca&moduleName=ebx-addon-dama&contentId=B8D...A9B",
	"URL": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=A3C...892E&versionId=B8D...A9B"
	"&uuid=fa6...0ca&moduleName=ebx-addon-dama&contentId=B8D...A9B&download=true",
	"defaultIcon": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=B7R...656V&versionId=SFA...A4T"
	"&uuid=fa6...0ca&moduleName=ebx-addon-dama&contentId=SFA...A4T",
	"creation_date": "01/01/2019 17:08:33",
	"last_updated_date": "01/01/2019 17:08:33",
	"versions": [
	{
	"uuid_version": "B8DCB43A-55C0-4968-9635-84EA9C5FEA9B",
	"version_name": "",
	"comment": "",
	"preview_url": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=A3C...892E&versionId=B8D...A9B"
	"&height=150&width=150&original=true&uuid=fa6...0ca&moduleName=ebx-addon-dama&contentId=B8D...A9B",
	"URL": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=A3C...892E&versionId=B8D...A9B"
	"&uuid=fa6...0ca&moduleName=ebx-addon-dama&contentId=B8D...A9B&download=true",
	}
	],
	"metadata": []
	}

	`}`
Response parameters	`uuid`: The asset's primary key. `name`: The label and description based on the request header language (defaults to EN). `URL`: The asset's URL. `creation_date`: The asset's creation date. `last_update_date`: The asset's most recent update date. `versions`: The asset's version information. `metadata`: All of the assets' metadata values. `preview_url`: The URL of the asset preview. `defaulticon`: The URL of the default file extension icon.
Note	When using the addon-Token mechanism, the URL stays live for 30 minutes after the initial call. In other cases the URL is authenticated by TIBCO EBX®. (HTTP Request, or login/password)

Get all registered mediaType fields

The example in the following table shows retrieval of all mediaType fields registered in the EBX® Digital Asset Manager Add-on:

> **Note**
>
> This REST operation was deprecated in the TIBCO EBX® Digital Asset Manager Add-on 2.0.0 release.

Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/getMediaFields/<dataspaceKey>/<datasetName>?tablePath=<aTablePath>`
Request parameters	`dataspaceKey`: An encoded dataspace key of the user's dataspace. `datasetName`: An encoded dataset name of the user's dataset. `aTablePath`: An encoded table path in the schema to get all media fields.

Sample response	<pre>{
	    "status": "success",
	    "message": "Get success",
	    "data": [
	        {
	            "fieldPath": "/fieldComplexCase/Group1/Group2/media",
	            "dacCode":
	"_root_TestComplexCase_fieldComplexCase_Group1_Group2_media",
	            "message": ""
	        },
	        {
	            "fieldPath": "/fieldComplexCase2/group1/media1",
	            "dacCode": "_root_TestComplexCase_fieldComplexCase2_group1_media1",
	            "message": ""
	        }
	    ]
	}</pre>

Response parameters	`fieldPath`: The mediaType field's path in the adaptation. `dacCode`: The dacCode registered in the mediaType field's D.A.C. `message`: The message returned when the mediaType field isn't registered in the D.A.C.

Getting media type field attachment info

The following table contains an example of getting information for assets attached to a mediaType field:

> **Note**
>
> This REST operation was deprecated in the TIBCO EBX® Digital Asset Manager Add-on 2.0.0 release.

Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/getAttachedAssetsInformation/<dataspaceKey>/<datasetName>?tablePath=<aTablePath>&primaryKey=<aPrimaryKey>&fieldPath=<aFieldPath>&useAddonToken=<useAddonToken>&pageSize=<aPageSize>&pageIndex=<aPageIndex>`
Request parameters	`dataspaceKey`: An encoded dataspace key of the user dataspace. `datasetName`: An encoded dataset name of the user dataset. `aTablePath`: An encoded table path (in the schema). `aPrimaryKey`: The current record's primary key. `aFieldPath`: An encoded field path (in the adaptation). `useAddonToken`: (optional) Determines whether the asset token will contain the add-on token. Defaults to false. `aPageSize`: (optional) Number of assets per page. Defaults to 10. `aPageIndex`: (optional) The page index. Defaults to 1.
Sample response	`{ "fieldPath": "/fieldSimpleCase", "viewMode": "mosaic_view", "permissions": { "change_current_version": true, "delete": true, "detach": true, "edit": true, "share_and_download": true, "upload": true, "upload_new_version": true, "view_digital_asset_of_other_user": true }, "count": 10, "assetsInfo": [{ "asset_status": 0, "asset_status_message": "", "uuid": "3896DE7A-BEC8-4E2A-BF2F-BC41790CF349", "name": { "locale": "en_US", "label": "asset_01.png", "description": "" }, "URL": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=389...349&versionId=523...000", "defaultIcon": "", "creation_date": "1512456124962", "last_updated_date": "1512456124971", "mimeType": "png", "fileSize": "0.1", "metadata": [] }] }`
Response parameters	`viewMode`: The configuration view mode (mosaic view/ list view). `permissions`: The permissions on the D.A.C. `count`: The number of assets attached to the field.

	`assetInfo`: The attached asset information.

Getting all assets in a Drive

The following table contains an example of getting all assets in a Drive:

> **Note**
>
> This REST operation was deprecated in the TIBCO EBX® Digital Asset Manager Add-on 2.0.0 release.

Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/getAllAssets/<dataspaceKey>/<datasetName>?tablePath=<aTablePath>&fieldPath=<aFieldPath>&useAddonToken=<useAddonToken>&pageSize=<aPageSize>&pageIndex=<aPageIndex>`
Request parameters	`dataspaceKey`: An encoded dataspace key of the user dataspace. `datasetName`: An encoded dataset name of the user dataset. `aTablePath`: An encoded table path (in the schema). `aFieldPath`: An encoded field path (in the adaptation). `useAddonToken`: (optional) Determines whether the asset token will contain the add-on token. Defaults to false. `aPageSize`: (optional) Number of assets per page. Defaults to 10. `aPageIndex`: (optional) The page index. Defaults to 1.
Sample response	`{ "asset_status": 0, "asset_status_message": "", "uuid": "110...741", "name": { "locale": "en_US", "label": "asset_01.jpg", "description": "" }, "URL": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=110...741&versionId=5BB...15A", "defaultIcon": "", "creation_date": "1519617088917", "last_updated_date": "1519627827201", "mimeType": "jpg", "fileSize": "0.2", "metadata": [], "attached" : true }`

Attach assets to a record

The following table contains an example of attaching assets to a record:

Attention

This REST operation has been deprecated in the EBX® Digital Asset Manager Add-on 1.5.0 release.

Method type:	`POST`
Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/attachAssets/<dataspaceKey>/<datasetName>`
Request parameters	`dataspaceKey`: An encoded dataspace key of the user dataspace. `datasetName`: An encoded dataset name of the user dataset.
Request body	Content-type: `application/x-www-form-urlencoded` Parameters: `aTablePath`: An encoded table path (in the schema). `aFieldPath`: An encoded field path (in the adaptation). `primaryKey`: The current record's primary key. `uuids`: The UUIDs of assets to attach. `overwrite`: (optional) If true, existing assets are replaced. If false, new assets are attached alongside existing assets. Defaults to false if undefined.

Sample responses	If one attachment fails when attaching multiple assets, no assets will be attached and an error message is returned.
	If successful:
	`{ "status": "success" }`
	If unsuccessful (all message content will follow the request locale):
	`{ "status": "unsuccess", "message": { "type": "error", "code": "1", "messageContent": "has error ..." } }`

Detach assets from a record

The following table contains an example of detaching assets from a record:

Attention

This REST operation has been deprecated in the EBX® Digital Asset Manager Add-on 1.5.0 release.

Method type:	`POST`
Request URL	`http://<host>[:<port>]/ebx-addon-dama/rest-service/detachAssets/<dataspaceKey>/<datasetName>`
Request parameters	`dataspaceKey`: An encoded dataspace key of the user dataspace. `datasetName`: An encoded dataset name of the user dataset.
Request body	Content-type: `application/x-www-form-urlencoded` Parameters: `aTablePath`: An encoded table path (in the schema). `aFieldPath`: An encoded field path (in the adaptation). `primaryKey`: The current record's primary key. `uuids`: The UUIDs of assets to attach. `detachAll`: (optional) If true, all assets attached to a media field will be detached. You do not need to provide asset UUIDs in this case. If false, only assets identified by the UUIDs are detached. Defaults to false if undefined.

Sample responses	If successful:
	`{ "status": "success" }`
	If unsuccessful (all message content will follow the request locale):
	`{ "status": "unsuccess", "message": { "type": "error", "code": "1", "messageContent": "has error ..." } }`

Upload an asset from a record

The following table provides an example of uploading an asset via REST:

> **Note**
>
> This REST operation was deprecated in the TIBCO EBX® Digital Asset Manager Add-on 2.0.0 release.

Method type	`POST`
URL	`http://<host>[:<port>]/ebx-addon-dama/rest/asset/v1/<dataspaceKey>/<datasetName>:upload?tablePath=<aTablePath>&fieldPath=<aFieldPath>&primaryKey=<primaryKey>`
Parameters	`dataspaceKey`: An encoded dataspace key of the user dataspace. `datasetName`: An encoded dataset name of the user dataset. `aTablePath`: An encoded table path (in the schema). `aFieldPath`: An encoded field path (in the adaptation). `primaryKey`: The record's primary key.
Request headers	`Authorization`: The EBX®'s authorization key. `Content-Type`: `application/x-www-form-urlencoded`
Request body	`file`: The binary file (required). `name`: The asset's name (required). `tags`: (optional) The assets' tags. This is an array of tags, separated by comma. Example: "tag1, tag2, tag3". `attach`: (optional) If `true`, the asset will be uploaded and attached into record. Otherwise, it will be uploaded without being attached.

Sample response	<pre>{
	    "URL": "http://<host>[:<port>]/ebx-addon-dama/AssetProvider?assetPath=520...D61&versionId=FB3...8C2",
	    "asset_status": "0",
	    "asset_status_message": "",
	    "creation_date": "01/01/2019 10:54:14",
	    "fileSize": "0.2",
	    "last_update_date": "01/01/2019 10:54:14",
	    "logicalName": "asset_01.png",
	    "metaData": [],
	    "mimeType": "png",
	    "name": {
	        "label": "asset_01",
	        "locale": "en"
	    },
	    "tags": [
	        "tag1",
	        "tag2",
	        "tag3"
	    ],
	    "uuid": "520A6F13-60B0-4EAA-901E-A1D28D677D61",
	    "versionId": "FB34C7B4-1595-47FD-BAA9-B14387DB98C2"
	}</pre>

Response parameters	`URL`: The asset's URL. `asset_status`: The asset's status. `asset_status_message`: The asset's status message. `creation_date`: The asset's creation date.

	`fileSize`: The asset's creation date. `last_update_date`: The asset's most recent update date. `logicalName`: The asset's logical name. `metadata`: The assets' metadata values. `mimeType`: The asset's mime type. `name`: The asset's name. `tags`: The assets' tags values. `uuid`: The asset's primary key. `versions`: The asset's version information.

Alternative storage locations

CHAPTER 20

Connecting to alternative storage locations

This chapter contains the following topics:

1. Overview
2. High-level flow
3. Implementing and registering a class and definition

20.1 Overview

The add-on's API allows you to access alternative asset storage locations. An "alternative" storage location could be one which requires credentials and a third-party API to access. For example, assets might be stored in the cloud on Google Drive or AWS. Alternatively, you may already use a third-party tool to manage digital assets but, you want to enable data model access to these assets. Since these types of locations and use cases require management outside of EBX®, we refer to these as *externally managed storage locations*.

The headings below outline high-level implementation steps. See Example Implementation [p 95] for an example implementation.

20.2 High-level flow

Several steps are require to implement access to an externally managed storage location. Those in a developer role perform most steps; as noted below, EBX® administrators complete others:

- Implement the following:
  - A class that provides add-on functionality for external storage locations. This class can also specify asset tags and search functionality.
  - A definition of the class that implements the add-on's API. EBX® must register this definition class on repository startup or from a user-run service.
  - Any classes required by third-party tools. For example, helper classes that contain connection credential and error handling information.
- Ensure that any required libraries are deployed in the same location as your `ebx.jar` file. This would include any JAR files containing your classes that implement the add-on's API. See the EBX® *Development Guide* for more information on deployment requirements.

- Steps an EBX® administrator must complete in the UI:
  - If required, run the service that registers the external storage management class.
  - Create an add-on Drive and configure it to use external storage. Ensure proper settings for user permissions.
  - Create a D.A.C. to link the data model field and Drive. Ensure proper settings for user permissions.

  For instructions, see Configuring an external Drive [p 73].

20.3 Implementing and registering a class and definition

As shown below, the class you write to specify Drive options for an externally managed location must implement the `ExternalManager` interface. The methods in this interface enable operations on assets stored in an external location.

```
import com.orchestranetworks.addon.dama.ext.exception.*;
import com.orchestranetworks.addon.dama.ext.resource.*;
import com.orchestranetworks.addon.dama.externalmanagement.*;
import com.orchestranetworks.addon.dama.externalmanagement.bean.*;
import com.orchestranetworks.addon.dama.externalmanagement.request.*;
import com.orchestranetworks.addon.dama.externalmanagement.response.*;

public class ExternalStorageDrive implements ExternalManager
{
 // TODO populate with required methods.
}
```

The below definition class which extends the add-on's `ExternalManagerDefinition` can be used to register your external management class. Note that in this class you can also determine whether users can upload assets to the external storage location from the add-on's **Drive** view.

```
import com.orchestranetworks.addon.dama.externalmanagement.*;

public final class ExternalStorageDriveDefinition extends ExternalManagerDefinition
{
 // Creates a new ExternalManager based on your implementation
 public ExternalManager getExternalManager()
 {
  return new ExternalStorageDrive();
 }

 // Determines whether upload to the external storage location is enabled
 public boolean allowUpload()
 {
  return true;
 }
}
```

To make your external storage location implementation available as an option in the UI, it must be registered. You can register the external storage management upon EBX® startup, or by implementing a service. For more information on module and service registration, see the EBX® documentation. To register the external storage manager, add the following code to your registration servlet or service dispatcher:

```
ExternalManagerCatalog.add(new ExternalStorageDriveDefinition());
```

CHAPTER 21

Example Implementation

This chapter contains the following topics:

1. Overview
2. Sample external manager class
3. Sample definition class
4. Registering your manager class
5. Including third-party resources

21.1 Overview

This section provides a sample configuration by connecting to a Google Drive. Note these requirements are specific to Google Drive at the time of publishing and your requirements will vary depending on how you plan to implement the API.

21.2 Sample external manager class

As shown below, you can begin with a manager class that implements the `ExternalManager` interface. The sample includes search functionality, asset display, and other features. Keep in mind that although you can code most add-on features including download, upload and delete, administrators can still define permissions on these actions when creating the Drive and D.A.C in the UI.

> **Attention**
> The samples below have several references to a helper class (not shown) that has been defined to meet requirements specific to making Google Drive API calls. The contents of your helper class will depend on implementation requirements.

The sections below provide the following samples:

- Class implementation [p 96]
- Search filter implementation [p 97]
- Asset creation [p 98]
- Asset deletion [p 98]
- Asset retrieval [p 98]

- Asset tags [p 100]

Class implementation

```
import java.io.*;
import java.math.*;
import java.net.*;
import java.util.*;

import javax.activation.*;

import com.onwbp.base.text.bean.*;
import com.orchestranetworks.addon.dama.*;
import com.orchestranetworks.addon.dama.common.*;
import com.orchestranetworks.addon.dama.ext.bean.*;
import com.orchestranetworks.addon.dama.ext.exception.*;
import com.orchestranetworks.addon.dama.ext.exception.DAMException;
import com.orchestranetworks.addon.dama.ext.resource.*;
import com.orchestranetworks.addon.dama.externalmanagement.*;
import com.orchestranetworks.addon.dama.externalmanagement.bean.*;
import com.orchestranetworks.addon.dama.externalmanagement.request.*;
import com.orchestranetworks.addon.dama.externalmanagement.response.*;
import com.orchestranetworks.addon.dama.models.*;
import com.orchestranetworks.addon.utils.*;

/**
 */
public class ExternalGoogleManager implements ExternalManager
{
 //Instance variable to access the helper class for the Google Drive
 private Drive googleDrive;

 public ExternalGoogleManager()
 {
  try
  {
   this.googleDrive = GoogleDriveAPIHelper.getInstance().getDriveService();
  }
  catch (Exception ex)
  {
   throw new RuntimeException(ex);
  }
 }

 //Sample method used to pass asset URL information to other methods
 private String getUrl(String externalId, String fileName, String extension, boolean isDownload)
  throws UnsupportedEncodingException
 {
  //The network location where EBX is running
  String host = "localhost:8080";

  String providerLink = "http://" + host
   + "/<your module name>/<your service dispatcher>?service=<your service name>";
  StringBuilder urlDownload = new StringBuilder(providerLink);
  urlDownload.append("&external_id=").append(URLEncoder.encode(externalId, "UTF-8"));
  urlDownload.append("&fileName=")
   .append(URLEncoder.encode(this.getFileName(fileName, extension), "UTF-8"));
  urlDownload.append("&extension=").append(extension);
  if (isDownload)
  {
   urlDownload.append("&download=").append(Boolean.TRUE.toString());
  }

  return urlDownload.toString();
 }

 private LabelDescription getLabelDescription(String name, List<Locale> locales)
 {
  if (AddonStringUtils.isEmpty(name))
  {
   name = DAMConstants.EMPTY_STRING;
  }

  List<LabelDescriptionForLocale> labelDescriptionForLocales = new ArrayList<>();
  for (Locale locale : locales)
  {
   LabelDescriptionForLocale labelDesLocale = new LabelDescriptionForLocale();
   labelDesLocale.setLocale(locale);
   labelDesLocale.setLabel(name);
   labelDesLocale.setDescription(name);
   labelDescriptionForLocales.add(labelDesLocale);
  }
```

```
  LabelDescription labelDescription = new LabelDescription();
  labelDescription.setLocalizedDocumentations(labelDescriptionForLocales);

  return labelDescription;
 }
}
```

Search filter implementation

```
public ExternalSearchAssetsResult searchAssets(ExternalSearchRequest searchRequest)
{
 ExternalSearchAssetsResult result = new ExternalSearchAssetsResult();
 SearchFilter searchFilter = searchRequest.getSearchFilter();
 String keyword = (searchFilter == null) ? DAMConstants.EMPTY_STRING
  : searchFilter.getKeyword();
 try
 {
  FileList fileList = this.googleDrive.files().list().setFields("*").execute();
  List<File> allFiles = fileList.getFiles();
  List<File> filteredFiles = new ArrayList<>();

  for (File file : allFiles)
  {
   if (AddonStringUtils.isEmpty(keyword))
   {
    filteredFiles.add(file);
   }
   else
   {
    if (file.getName().contains(keyword)
     || file.getOriginalFilename().contains(keyword))
    {
     filteredFiles.add(file);
    }
   }
  }
  int size = filteredFiles.size();
  int pageIndex = (searchRequest.getPageIndex() < 1) ? 1 : searchRequest.getPageIndex();
  int pageSize = (searchRequest.getPageSize() < 10) ? 10 : searchRequest.getPageSize();
  int startIndex = (pageIndex - 1) * pageSize;
  int endIndex = startIndex + pageSize - 1;
  if (endIndex > size - 1)
  {
   endIndex = size - 1;
  }

  List<ExternalDigitalAsset> externalAssets = new ArrayList<>();
  for (int i = startIndex; i <= endIndex; i++)
  {
   File googleFile = filteredFiles.get(i);
   String downloadUrl = this.getUrl(
    googleFile.getId(),
    googleFile.getName(),
    googleFile.getFileExtension(),
    true);
   String previewUrl = this.getUrl(
    googleFile.getId(),
    googleFile.getName(),
    googleFile.getFileExtension(),
    false);
   ExternalDigitalAsset externalAsset = new ExternalDigitalAsset();
   externalAsset.setLabelDescription(
    this.getLabelDescription(
     googleFile.getName(),
     searchRequest.getHome().getRepository().getLocales()));
   externalAsset.setURL(previewUrl);
   externalAsset.setThumbnailURL(googleFile.getThumbnailLink());
   externalAsset.setDownloadURL(downloadUrl);
   externalAsset.setExtension(googleFile.getFileExtension());
   externalAsset.setAssetType(googleFile.getFileExtension());
   externalAsset.setId(googleFile.getId());
   externalAsset.setPhysicalName(googleFile.getName());
   externalAsset.setCreationDate(new Date(googleFile.getCreatedTime().getValue()));
   externalAsset.setLastUpdatedDate(new Date(googleFile.getModifiedTime().getValue()));
   externalAsset.setFileSize(new BigDecimal(googleFile.getSize().longValue()));
   Map<String, String> metaDataMap = new HashMap<>();
   metaDataMap.put("mKey1", "mValue1");
   metaDataMap.put("mKey2", "mValue2");
   externalAsset.setMetaDatas(metaDataMap);
   externalAsset.setWidth(new Integer(300));
   externalAsset.setHeight(new Integer(200));
```

```
   externalAssets.add(externalAsset);
  }

  result.setAssets(externalAssets);
 }
 catch (Exception ex)
 {
  throw new RuntimeException(ex);
 }

 return result;
}
```

Asset creation

```
public ExternalDigitalAsset createAsset(ExternalUploadAssetRequest request)
{
 ExternalDigitalAsset asset = new ExternalDigitalAsset();
 File googleFile = new File();
 FileResource fileResource = request.getFileResource();
 String fileName = fileResource.getFile().getName();
 googleFile.setName(fileName);
 FileContent mediaContent = new FileContent(
  new MimetypesFileTypeMap().getContentType(fileName),
  fileResource.getFile());
 try
 {
  File googleFileUploaded = this.googleDrive.files()
   .create(googleFile, mediaContent)
   .setFields("*")
   .execute();
  asset.setId(googleFileUploaded.getId());
  asset.setPhysicalName(googleFileUploaded.getOriginalFilename());
  asset.setThumbnailURL(googleFileUploaded.getThumbnailLink());
  asset.setLabelDescription(
   this.getLabelDescription(
    googleFileUploaded.getName(),
    request.getHome().getRepository().getLocales()));
 }
 catch (IOException ex)
 {
  throw new RuntimeException(ex);
 }

 return asset;
}
```

Asset deletion

```
public OperationExecutionStatus deleteAsset(
   ExternalSingularRequest request,
   boolean isPhysicalDelete)
{
 OperationExecutionStatus status = new OperationExecutionStatus();
 try
 {
  this.googleDrive.files().delete(request.getExternalId()).execute();
 }
 catch (IOException ex)
 {
  status = new OperationExecutionStatus(new DAMException(ex.getMessage()));
 }

 return status;
}
```

Asset retrieval

```
public ExternalDigitalAsset getAsset(ExternalSingularRequest request) throws DAMException
{
 ExternalDigitalAsset digitalAsset = new ExternalDigitalAsset();
 try
 {
  File file = this.googleDrive.files()
   .get(request.getExternalId())
   .setFields("*")
   .execute();
  String urlDownload = this
```

```
    .getUrl(file.getId(), file.getName(), file.getFileExtension(), true);
   String previewUrl = this
    .getUrl(file.getId(), file.getName(), file.getFileExtension(), false);
   digitalAsset.setId(file.getId());
   digitalAsset.setPhysicalName(file.getOriginalFilename());
   digitalAsset.setLabelDescription(
    this.getLabelDescription(
     file.getName(),
     request.getHome().getRepository().getLocales()));
   digitalAsset.setURL(previewUrl);

   boolean isImage = DigitalAssetExtension.parseExtension(file.getFileExtension())
    .getDigitalAssetContentType()
    .isImage();
   if (isImage)
   {
    digitalAsset.setThumbnailURL((previewUrl));
   }
   else
   {
    digitalAsset.setThumbnailURL(file.getThumbnailLink());
   }
   digitalAsset.setDownloadURL(urlDownload);
   digitalAsset.setAssetType(file.getFileExtension());
   digitalAsset.setExtension(file.getFileExtension());
   digitalAsset.setCreationDate(new Date(file.getCreatedTime().getValue()));
   digitalAsset.setLastUpdatedDate(new Date(file.getModifiedTime().getValue()));
   digitalAsset.setFileSize(new BigDecimal(file.getSize().longValue()));
   Map<String, String> metaDataMap = new HashMap<>();
   metaDataMap.put("mKey1", "mValue1");
   metaDataMap.put("mKey2", "mValue2");
   digitalAsset.setMetaDatas(metaDataMap);
   digitalAsset.setWidth(new Integer(300));
   digitalAsset.setHeight(new Integer(200));
  }
  catch (IOException ex)
  {
   throw new DAMException(ex);
  }

  return digitalAsset;
 }

 public List<ExternalDigitalAsset> getAssets(ExternalPluralRequest externalRequest)
 {
  List<ExternalDigitalAsset> externalAssets = new ArrayList<>();
  for (String externalId : externalRequest.getExternalIds())
  {
   try
   {
    File file;
    try
    {
     file = this.googleDrive.files().get(externalId).setFields("*").execute();
    }
    catch (Exception ex)
    {
     continue;
    }

    ExternalDigitalAsset externalDigitalAsset = new ExternalDigitalAsset();
    externalDigitalAsset.setId(file.getId());
    externalDigitalAsset.setPhysicalName(file.getOriginalFilename());
    externalDigitalAsset.setCreationDate(new Date(file.getCreatedTime().getValue()));
    externalDigitalAsset
     .setLastUpdatedDate(new Date(file.getModifiedTime().getValue()));
    externalDigitalAsset.setLabelDescription(
     this.getLabelDescription(
      file.getName(),
      externalRequest.getHome().getRepository().getLocales()));
    String urlDownload = this
     .getUrl(file.getId(), file.getName(), file.getFileExtension(), true);
    String previewUrl = this
     .getUrl(file.getId(), file.getName(), file.getFileExtension(), false);
    externalDigitalAsset.setURL(previewUrl);
    externalDigitalAsset.setThumbnailURL(file.getThumbnailLink());
    externalDigitalAsset.setWidth(new Integer(300));
    externalDigitalAsset.setHeight(new Integer(200));
    externalDigitalAsset.setDownloadURL(urlDownload);
    externalDigitalAsset.setAssetType(file.getFileExtension());
    externalDigitalAsset.setExtension(file.getFileExtension());

    Map<String, String> metaDatas = new HashMap<>();
    metaDatas.put("metadata1", "metadata1");
    metaDatas.put("metadata2", "metadata2");
```

```
   externalDigitalAsset.setMetaDatas(metaDatas);

   externalDigitalAsset.setTags(Arrays.asList("tag1", "tag2"));

   externalAssets.add(externalDigitalAsset);
  }
  catch (IOException ex)
  {
   throw new RuntimeException(ex);
  }
 }

 return externalAssets;
}
```

Asset tags

```
public ExternalTagResult getTags(ExternalCommonRequest request)
{
 ExternalTagResult externalTagResult = new ExternalTagResult();

 Tag tag1 = new Tag();
 tag1.setCode("tag1");
 tag1.setFontSize("14");
 externalTagResult.getTags().add(tag1);

 Tag tag2 = new Tag();
 tag2.setCode("tag2");
 tag2.setFontSize("16");
 externalTagResult.getTags().add(tag2);

 return externalTagResult;
}
```

21.3 **Sample definition class**

The following code sample shows how to instantiate the class you implemented above. Additionally, note that you can use the `allowUpload()` to determine whether users can upload to the external storage location.

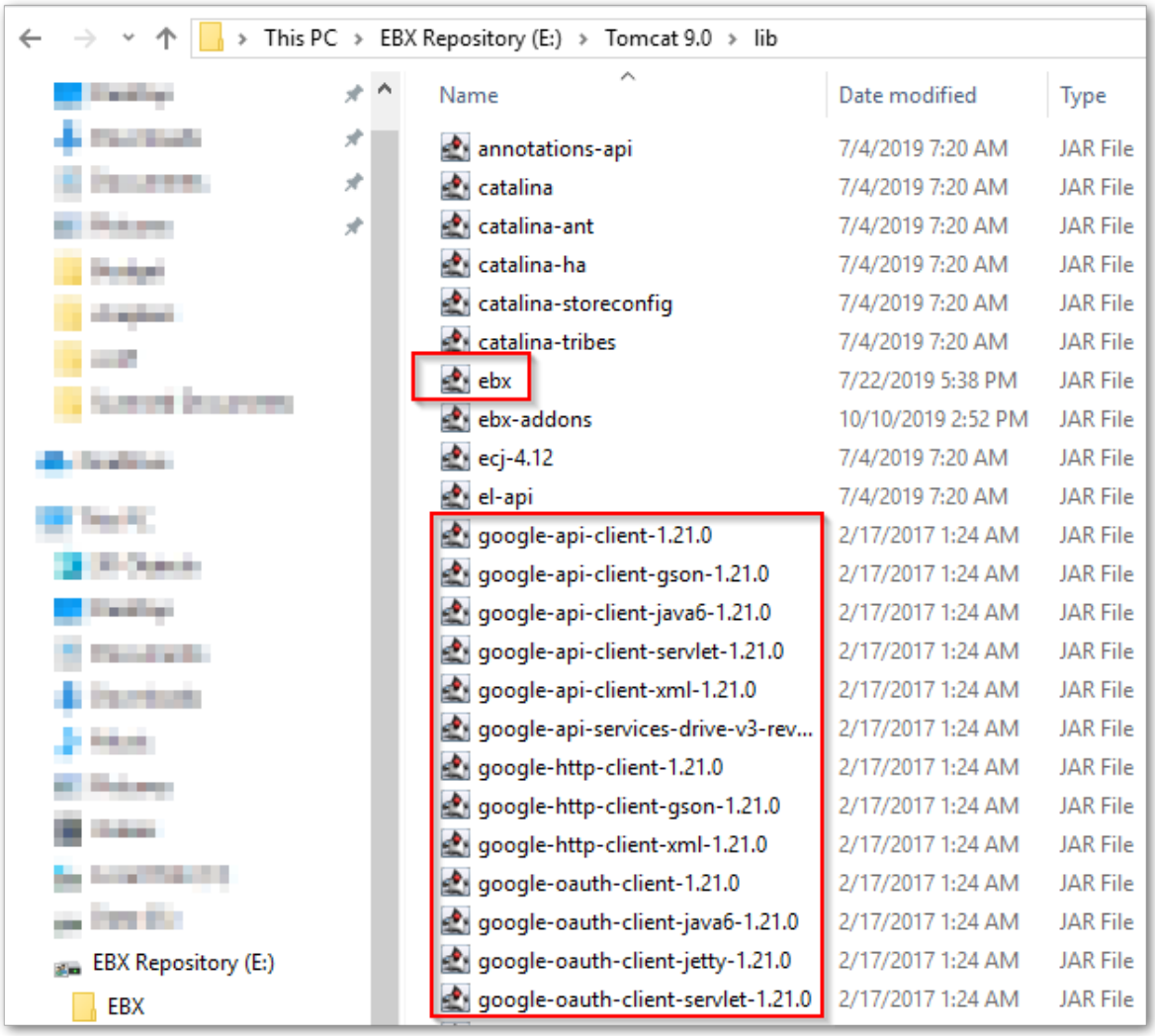
```
import com.orchestranetworks.addon.dama.externalmanagement.*;

public final class ExternalGoogleManagerDefinition extends ExternalManagerDefinition
{
 // Creates a new ExternalManager based on your implementation
 public ExternalManager getExternalManager()
 {
  return new ExternalGoogleManager();
 }

 // Determines whether upload to the external storage location is enabled
 public boolean allowUpload()
 {
  return true;
 }
}
```

21.4 **Registering your manager class**

You can register your custom class either on repository startup or by running a service in the UI. For information on how to declare a module (for startup registration) or declare a user service, please see the *TIBCO EBX® Developer Guide*.

21.5 Including third-party resources

Any resources required by a third-party should be added to the same location as your `ebx.jar` file. As shown below the required libraries for the Google Drive API are deployed in Tomcat along with EBX®.



Reference Guide

CHAPTER 22

Add-on specific services

This chapter contains the following topics:

1. Overview of add-on specific services

22.1 Overview of add-on specific services

The TIBCO EBX® Digital Asset Manager Add-on provides several custom services that allow you to perform asset management-specific actions. To access these services open the **Actions** menu for the specified dataset, or table. The following describes the location of each service and its scope:

Name/Location/Scope	Description
TIBCO EBX® Digital Asset Manager Add-on dataset	
The **Migrate digital asset manager data** service applies to the dataset.	Migrates previous add-on versions to the latest version.
The **Create D.A.C. for data model** service applies to the dataset.	Automatically creates a D.A.C. for a data model you specify. Before running the service, ensure that the data model contains a 'media type' field.
The **Manage digital assets** service applies to all assets in all Drives.	Allows you full control over any asset in any Drive.
The **Backup** service applies to the dataset.	Creates a backup of your digital assets.
The **Restore** service applies to the dataset.	Allows you to restore digital assets.
Drive table	
The **Delete all digital assets in selected drive(s)** service applies to selected records.	Deletes all digital assets contained in the selected Drives.
Change physical root path	Allows you to change the selected Drive's physical root path.
Digital asset table	
Identify unused assets	Locates any unused assets and filters the table display so that only those records display. An unused asset is one that is not attached to a field.
Import from a ZIP file	Allows you to perform a bulk import of assets from a ZIP file.
The **Create new version** service applies to selected records.	Creates a new version of the selected asset.
The **Upload new digital asset** service applies to the table.	Allows you to upload a new digital asset.
The **Activate** service applies to the selected assets.	Allows you to active an asset from this screen.

Name/Location/Scope	Description
The **Deactivate (Logical delete)** service applies to selected records.	Changes the selected asset's state to 'Deactivated'. You can return the asset to 'Activated', or another state by opening the asset and using the 'State' drop-down list.
The **Permanently delete (Physical delete)** service applies to selected records.	Physically removes the digital asset from the Drive.
Digital asset version table	
The **Create new version** service applies to the table.	Allows you to upload a new version of an asset.
The **Delete digital asset version** applies to selected records.	Physically removes the digital asset version.

CHAPTER 23

Digital asset group

This chapter contains the following topics:

1. Overview

23.1 Overview

This group contains information about any digital asset under add-on management.

Digital asset, Digital asset version, and Metadata value tables

When you open a record in the **Digital asset** table, tabs display that contain the same information as the corresponding table. The following table describes each property.

Property	Definition
Information tab/**Digital Asset** table	
Physical name	The asset's original file name.
Label and description	An asset's label and description.
Creation date	Shows the asset creation date — the date when the asset was added to the add-on control.
Last update date	Date of the last update to this asset.
Digital asset type	Indicates the asset type.
File extension	Displays the asset's file extension.
State	An asset's current state value.
Drive	Location of this asset.
Version management tab/**Digital asset version** table	
Physical name	Displays the asset's file name.
Storage date	Date on which the asset was uploaded to a Drive.
Digital asset	The digital asset of which this is a version.
User profile author	Indicates who added this asset to the add-on.
Version name	Lists any versioning applied to this asset (1.0, 2.0, etc.).
Comment	Displays comments applied to this asset version.
Current version	Indicates whether this version is considered the current version.
Custom metadata tab/**Metadata value** table	
Digital asset	The digital asset to which this metadata applies.
Metadata nature	Displays the metadata nature applied to this digital asset.

Property	Definition
Value	Holds the metadata value applied to this asset.
Tags tab	
Tags	Displays the tags attached to the selected asset.

CHAPTER 24

Drive group

This chapter contains the following topics:

1. Overview

24.1 Overview

The **Drive group** contains tables for Drive and image configuration.

Drive table

This table defines a Drive, or asset storage location. After creating a Drive, you can link it with a data model field using a **Digital asset component**. This enables the selected field access to assets stored in the Drive.

Property	Definition
Drive id	Id of drive.
Label and description	Defines the label and description for this Drive.
Connector	Specifies whether the Drive points to a local or network location, or to an externally managed storage location. Connections to external locations, require implementation of the add-on's API and any additional third-party requirements. See the add-on's *Developer Guide* for more details.
Physical root path	Specifies the path to this Drive's storage location. This location stores all assets and asset versions uploaded to this Drive.
User permission group	
Restriction policy	Defines whether the permission is restrictive or not. For all user-defined permission rules that match the current user session: If some rules having restrictions are defined, the minimum permissions of these restricted rules are applied; if no rule having restrictions are defined, the maximum permissions of all matching rules are applied.
Profile	User profile to which this set of permissions applies.
Allow to upload new asset	Determines whether the chosen profile can upload files to this Drive.
Allow to upload new version	Determines whether users associated with the specified profile can upload new versions of assets.
Allow to change current version	Determines whether users associated with the specified profile can choose which asset is considered the current version. If users are allowed to upload new assets, this property defaults to 'Yes'. If you set this to 'No', the most recently uploaded asset is considered the current version.
Allow to share and download	Determines whether users associated with the specified profile can download and/or share the asset via email.
Allow to delete	Determines whether the chosen profile can delete files in this Drive.
Allow to view digital asset of other user	Determines whether users associated with the specified profile can view assets uploaded by other users.
Maximum allowed file size	Specifies the largest file size (in MBs) that the selected profile can upload to this Drive.
Reject duplicates	Specifies whether upload of duplicate assets is rejected by default. When disabled, users can manually choose when uploading assets.

Property	Definition
Max usable space	Caps the amount of usable disk space for a Drive that uses this configuration. Once storage reaches this threshold, users cannot upload additional files, unless files are physically deleted.
Physical used space	Amount of disk space used by assets in this Drive.
Physical free space	Amount of disk space remaining.
Image configuration	Specifies the image configuration used by this Drive. The image configuration sets a maximum width and height for uploaded images.
Asset name pattern	Specifies a predefined regular expression that enforces compliance of asset physical file names with a defined pattern. If the name doesn't follow the pattern, users cannot upload the asset.
Restrict file upload	Determines whether file type restrictions apply by default for a Digital Asset Component (D.A.C.) that uses this Drive. By default, the add-on does not limit upload to specific file types. With this option enabled, you can select allowed file types when creating a D.A.C. that uses this Drive. Note that you can override this option when creating a D.A.C.
Blacklisted extensions	The add-on blocks any extensions you list here. Separate entries using a comma, semicolon, period or whitespace. For example: `exe, msi, sh, bat`
Encrypt files	When activated, the add-on encrypts files and thumbnails before storing in the file system. The add-on automatically decrypts these files before displaying or allowing users to download. Note that you cannot modify this field after saving.

Image configuration

This table allows you to create configurations that set limits on the width and height of uploaded images. These settings can be applied to *Drive configurations*.

Property	Definition
Code	Specifies a unique value to identify this configuration.
Label and description	Defines the label and description for this configuration.
Max width	Sets the maximum width for images uploaded to a Drive that uses this configuration.
Max height	Sets the maximum height for images uploaded to a Drive that uses this configuration.
Min width	Sets the minimum width for images uploaded to a Drive that uses this configuration.
Min height	Sets the minimum height for images uploaded to a Drive that uses this configuration.

CHAPTER 25

D.A.C. group

This chapter contains the following topics:

1. Overview

25.1 Overview

The **Digital asset component (D.A.C.)** group allows you to create a link between a data model field to which you want to attach assets and a physical storage location. Additionally, you can configure mosaic look and feel.

Digital asset component table

This table allows you to configure a **Digital asset component**, which links a data model field with a Drive.

Property	Definition
Code	Specifies a unique value to identify this D.A.C.
Label and description	Defines the label and description for this D.A.C.
Drive	Specifies the Drive used as the storage location for this configuration.
User permission group: Permissions can inherit from the selected Drive, or you can define additional permissions that apply to this D.A.C.	
Restriction policy	Defines whether the permission is restrictive or not. For all user-defined permission rules that match the current user session: If some rules having restrictions are defined, the minimum permissions of these restricted rules are applied; if no rule having restrictions are defined, the maximum permissions of all matching rules are applied.
Profile	User profile to which this set of permissions applies.
Allow to upload new asset	Determines whether the chosen profile can upload files to this Drive.
Allow to upload new version	Determines whether users associated with the specified profile can upload new versions of assets.
Allow to change current version	Determines whether users associated with the specified profile can choose which asset is considered the current version. If users are allowed to upload new assets, this property defaults to 'Yes'. If you set this to 'No', the most recently uploaded asset is considered the current version.
Allow to open Drive	Determines whether users associated with the specified profile can open the Drive view.
Allow to share and download	Determines whether users associated with the specified profile can download and/or share the asset via email.
Allow to delete	Determines whether the chosen profile can delete files in this Drive.
Allow to detach	Determines whether a user can detach assets from fields.
Allow to view digital asset of other user	Determines whether users associated with the specified profile can view assets uploaded by other users.
Allow to quick upload and attach	Determines whether users can choose to skip intermediate steps when uploading and attaching assets.
Maximum allowed file size	Specifies the largest file size (in MBs) that the selected profile can upload to this Drive.

Property	Definition
Image configuration	Specifies the image configuration used by this D.A.C. The image configuration sets a maximum width and height for uploaded images.
Data model	Specifies the data model containing the field you want to connect to the specified Drive.
Field path	Defines the path to the field you want to connect to the specified Drive.
Custom filter	This optional property accepts a custom filter implemented in a Java class. The filter can specify that only assets meeting certain criteria display in the **Drive** view. The filter class you create your must implement the predefined interface (`com.orchestranetworks.addon.dama.ext.EditorFilter`). This interface's `accept()` method filters assets from the **Drive** view. If you decide to create a custom filter class, set the class value to D.A.C record and be sure the class implements the `EditorFilter` interface. The value filter depends on the record's fields.
Asset name pattern	Specifies a predefined regular expression that enforces compliance of asset physical file names with a defined pattern. If the name doesn't follow the pattern, users cannot upload the asset.
Restrict file upload	Determines whether to apply file type restrictions. By default, the add-on does not limit upload to specific file types. With this option enabled, you can select allowed file types when creating a D.A.C. that uses this Drive.
Blacklisted extensions	The add-on blocks any extensions you list here. Separate entries using a comma, semicolon, period or whitespace. For example: `exe, msi, sh, bat`
Supported extensions	Enables upload of the selected extensions. This field only displays when you enable the **Restrict file upload** option.
Display configuration	Sets the display configuration used by this D.A.C. which determines how assets display when attached to the data model field.
EBX tabular view	The configuration for view digital asset on the table.
Drive view	Determines the default layout for the **Drive** view.
Tag cloud	Specifies which tag cloud is used for this D.A.C.

View configuration table

This table allows you to set asset display options.

Property	Definition
Label and description	Defines this configuration's label and description.
View mode	Determines whether assets display in a tiled mosaic, list view, or table view.
Display attached assets	Determines whether assets display as their associated icon, or thumbnails of the asset.
Always display quick actions	Determines whether the quick actions buttons are always visible on asset thumbnails.
Expand attached asset	Allows users to expand/collapse asset thumbnails.
Thumbnail width	Determines the width of the asset thumbnail.
Thumbnail height	Determines the height of the asset thumbnail.
Display label under thumbnail	Determines whether the assets label displays in the mosaic.
Vertical scroll	Determines whether the mosaic scrolls vertically or horizontally.
Number of assets per row	Sets the number of assets displayed in the mosaic's rows. Note that this property can affect overall width of the mosaic.
Number of assets per column	Sets the number of assets displayed in the mosaic's columns. If additional assets are loaded, a scroll bar enables navigation.

CHAPTER 26

Classification group

This chapter contains the following topics:

1. Overview

26.1 Overview

The **Classification group** stores all tags and tag clouds that you can apply to assets and D.A.C.'s. The **Digital asset by tag** table also provides you with a hierarchical view of each tag and the assets to which they have been applied.

Tag table

This table stores all tags. If you have administrative privileges, you can create new tags by adding records to this table. All tags must be unique.

Property	Definition
Code	Defines this tag.

Digital asset by tag

Contains relationships to show a hierarchical view of tags and related assets.

Property	Definition
Digital asset	Identifies the digital asset.
Tag	Associates this tag with the digital asset.

Tag cloud table

The **Tag cloud** table stores the label and description for each tag cloud. A tag can be associated with multiple tag clouds. When you open a tag cloud, a view of associated tags displays. From this view, you can create new tags and use the **Actions** menu to:

- Attach an existing tag to this tag cloud. Only tags not currently attached to a cloud display for attachment.

- Detach one or more selected tags from this tag cloud.
- Delete one or more selected tags.
- Duplicate a tag. However, each tag must be unique. Before saving the tag, you must update it.
- Move one or more selected tags to a different tag cloud.

Property	Definition
Label and description	Applies the label and description to this tag cloud.

CHAPTER 27

Reference data group

This chapter contains the following topics:

1. Overview

27.1 Overview

Tables in the **Reference data group** define values that other EBX® Digital Asset Manager Add-on components can use.

Metadata nature table

This table allows you to create categories of metadata. You can add these categories to digital assets. After applying metadata nature you can further enrich the asset with a data value.

Property	Definition
Code	Specifies a unique value to identify this metadata nature.
Label and description	Defines the label and description for the metadata category.

Digital asset type

This table contains the available types of digital assets.

Property	Definition
Code	Specifies a unique value to identify a type of digital asset.
Business code	A business friendly identifier for a digital asset type.
Label and description	Label and description used by a digital asset type.

Connector table

This table lists available Drive types.

Property	Definition
Label and description	User-friendly name and description of the source
Storage type	Defines the storage type for this connector.
Built in	If set to: • **Yes**: This is provided by the add-on. • **No**: This is not provided by the add-on.

File extension table

This table stores available file extensions. The **Supported extensions** field in a D.A.C. specifies which extensions users can attach to fields. This table includes several default extensions, but allows you to create additional extensions to meet your business requirements.

Property	Definition
Business code	A business identifier for a file extension.
Filename extension	Specifies a unique value to identify the file extension format.
Label and description	Defines the file extension's label and description.
Digital asset type	Specifies what type of digital asset this file extension type is.
Mime type	Optional information for this file extension.
Icon	Sets the icon for this type of file extension. You can upload and store files for use as icons.

Pattern table

This table stores regular expressions used to ensure asset file names follow a specified pattern. This property is available to set at the Drive and D.A.C levels.

Property	Definition
Id	The auto-incremented identifier for this record.
Label and description	Defines the pattern's label and description.
Pattern	Defines a regular expression pattern to validate the asset's name when uploading assets to the drive. For example: '^[a-zA-Z0-9]$', '^[0-9]$', '^[a-z][a-z]([0-9]+)?$', etc.

State table

This table lists available states. States determine whether an asset is active, or inactive. Only active assets can be viewed in the **Drive** view or from the **Manage digital assets** view.

Property	Definition
Code	Specifies a unique value to identify a state.
Label and description	Defines the label and description for a state.

History table

The history table stores actions related to a digital asset.

Property	Definition
User profile	User profile that made changes to the specified asset.
Digital asset	Asset changes were applied to.
Old state	The asset's previous state.
New state	The asset's new state.
Changed date	Date changes were made to the asset.

Release Notes

CHAPTER 28

Version 2.3.1

Released: March 2022

This chapter contains the following topics:

1. New features
2. Changes in Functionality
3. Changes to third-party libraries
4. Closed Issues
5. Known Issues

28.1 New features

This release contains no new features.

Attention
As the add-on is undergoing revision, the documentation's images and descriptions of some components might not exactly match the current UI.

28.2 Changes in Functionality

This release includes no changes to functionality.

28.3 Changes to third-party libraries

This release contains the following updates to third-party libraries:

- The Apache PDFBox and Apache FontBox were updated to version 2.0.25.
- The jQuery UI and UI widget libraries were updated to version 1.13.1.
- The jQuery library was updated to version 3.6.0.
- The Apache XML Graphics Commons library was updated to version 2.7.
- The debugger.js component was removed from the PDF.js library.
- The FasterXML/jackson-databind library was updated to version 2.13.2.1.
- The Zip4j library was updated to version 2.9.1.

28.4 Closed Issues

[EBXDAMA-1662] An error occurs in the media type field when the table has 2 primary keys and one data type is date.

28.5 Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

CHAPTER 29

All release notes

This chapter contains the following topics:

1. Version 2.3.1
2. Version 2.3.0
3. Version 2.2.1
4. Version 2.2.0
5. Version 2.1.1
6. Version 2.1.0
7. Version 2.0.1
8. Version 2.0.0
9. Version 1.8.8
10. Version 1.8.7
11. Release Notes 1.8.6
12. Release Notes 1.8.5
13. Release Notes 1.8.4
14. Release Notes 1.8.3
15. Release Notes 1.8.2
16. Release Notes 1.8.1
17. Release Notes 1.8.0
18. Release Notes 1.7.0
19. Release Notes 1.6.0
20. Release Notes 1.5.1
21. Release Notes 1.5.0
22. Release Notes 1.4.3
23. Release Notes 1.4.2
24. Release Notes 1.4.1
25. Release Notes 1.4.0
26. Release Note 1.3.0

27. Release Note 1.2.2
28. Release Note 1.2.1
29. Release Note 1.2.0
30. Release Note 1.1.1
31. Release Note 1.1.0
32. Release Note 1.0.2 fix 001
33. Release Note 1.0.3
34. Release Note 1.0.2
35. Release Note 1.0.1
36. Release Note 1.0.0

29.1 Version 2.3.1

Released: March 2022

New features

This release contains no new features.

> **Attention**
> As the add-on is undergoing revision, the documentation's images and descriptions of some components might not exactly match the current UI.

Changes in Functionality

This release includes no changes to functionality.

Changes to third-party libraries

This release contains the following updates to third-party libraries:

- The Apache PDFBox and Apache FontBox were updated to version 2.0.25.
- The jQuery UI and UI widget libraries were updated to version 1.13.1.
- The jQuery library was updated to version 3.6.0.
- The Apache XML Graphics Commons library was updated to version 2.7.
- The debugger.js component was removed from the PDF.js library.
- The FasterXML/jackson-databind library was updated to version 2.13.2.1.
- The Zip4j library was updated to version 2.9.1.

Closed Issues

[EBXDAMA-1662] An error occurs in the media type field when the table has 2 primary keys and one data type is date.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.2 **Version 2.3.0**

Released: February 2022

New features

This release contains the following new features:

- A new Drive configuration property allows administrators to choose whether to allow upload of duplicate assets to the given Drive.
- A new connector type allows a Drive location to be set to an Amazon S3 bucket. This allows you to attach assets to fields in EBX® while managing their storage outside of EBX®.
- Actions such as deletion, deactivation, downloading, and sharing can now be performed on multiple assets at once.
- When configured by an administrator, users now have the option of uploading and attaching assets directly from the tabular view. This prevents having to open each record to attach assets.
- The **Digital asset** table now displays a thumbnail of each digital asset.
- When uploading files, any files that have errors can now be removed before uploading.
- A new API is available that allows you to create a custom connector to link a field containing assets with a third-party tool.

> **Attention**
> As the add-on is undergoing revision, the documentation's images and descriptions of some components might not exactly match the current UI.

Changes in Functionality

This release includes the following changes to functionality:

- The **Class** field is now hidden in the **Pattern** table.

- Two files with the same name can now be uploaded to a Drive. The add-on automatically appends a number increment to the file name.
- The `File not found` error no longer displays when all assets are removed from a Drive.
- When you delete a D.A.C, any asset modifications made in the local dataspace (that are related to the deleted D.A.C) are lost. Assets are reset to their initial state.

Changes to third-party libraries

This release contains no updates to third-party libraries.

Closed Issues

This release contains the following closed issues:

- *[EBXDAMA-591]* An error message displays when opening a record that has an existing asset in the blacklisted extension.
- *[EBXDAMA-730]* Error message should be shown as pop-up when deleting asset in Digital asset version table that does not have another version.
- *[EBXDAMA-761]* Create new record submission is NOT prevented when there are errors in the **Pattern** table.
- *[EBXDAMA-774]* In the **Upload** screen, the information related to the maximum allowed file size and restricted file upload are not displayed even though they are set up in the Drive/ DAC.
- *[EBXDAMA-795]* The order of assets is changed after selecting another Drive.
- *[EBXDAMA-824]* The original file name is changed when a user downloads an external asset.
- *[EBXDAMA-833]* An incorrect error message is shown when a user tries to preview an asset from a deleted Drive.
- *[EBXDAMA-835]* Can Input the letter or special character in the search page in the **Manage digital asset** screen.
- *[EBXDAMA-844]* XLS and XLSX files don't display correctly when changing the asset's zoom level.
- *[EBXDAMA-875]* Assets displayed in the **Drive** view don't display according to the [Mosaic] view set in the D.A.C.
- *[EBXDAMA-878]* The display of the tag value is incorrect.
- *[EBXDAMA-885]* Unable to change Drive of existing asset via the **Digital asset** table in the **Administration** panel.
- *[EBXDAMA-893]* An unexpected error message displays when leaving fields blank in the image configuration screen.
- *[EBXDAMA-906]* The dropdown menu displays incorrectly when a Drive name is too long.
- *[EBXDAMA-912]* The wrong screen is displayed when closing the **Edit digital asset** screen of sharing assets.
- *[EBXDAMA-917]* SVG images do not display correctly.
- *[EBXDAMA-927]* The error message is missing the rule pattern and it is not translated into French.
- *[EBXDAMA-945]* The **Accept duplicate(s)** checkbox does not display when uploading existing assets.

- *[EBXDAMA-952]* BMP images cannot be previewed even when smaller than 5MB.
- [EBXDAMA-1397][Rest API] The `getAssetInformation` endpoint returns the incorrct calculated URL for `preview_url`, `defaultIcon`, `versions.URL` and `versions.preview_url`.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.3 Version 2.2.1

Released: December 2021

New features

This release contains no new features.

Changes in Functionality

This release includes no changes to functionality.

Changes to third-party libraries

This release contains the following updates to third-party libraries:

- The jQuery UI library was updated to version 1.13.0.
- The jQuery File Upload library was updated to the latest version.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.

- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.4 Version 2.2.0

Released: November 2021

New features

This release contains no new features.

Changes in Functionality

This release includes the following changes to functionality:

- When configuring a Drive, you now select from a list of available connectors instead of Drive types. The following connectors are available in this release:
  - **Default connector:** Allows you to create a default Drive where asset storage and management tasks are handled by the add-on.
  - **External manager:** Allows you to use the API to connect to an externally managed storage location.
- The **Editor** is now the **Drive** view and you can no longer edit asset metadata from this view. All functionality, apart from editing, in this view remains the same. You can now access editing features by:
  - Selecting an asset's information icon when viewing a record.
  - Using the **Manage digital assets** service (available to administrators).
- Modifications to assets are no longer autocratically global in scope. Changes are now either isolated, or applied globally depending on your access rights, and where you initiate changes. When changes are isolated, they only impact the current dataspace. For example, when you make changes in a child dataspace, they do not propagate to the parent. Global changes are propagated to any location where an asset is used. If you:
  - Are a business user and access edit functionality from a record's details view, any changes you make are isolated to the current dataspace.
  - Have administrative privileges and access edit functionality using the **Manage digital assets** service, changes apply globally. Note that as an administrator you can still edit from a record's details view if you want to keep your changes local.

    > **Note**
    >
    > The above behavior applies to all asset-specific information, except tags. Changes to tags are always applied globally.

Changes to third-party libraries

This release contains no updates to third-party libraries.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.5 Version 2.1.1

Released: September 2021

New features

This release includes no new features.

Changes in Functionality

This release includes no changes in functionality.

Changes to third-party libraries

The Apache Commons Compress library was upgraded to version 1.21.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- When updating asset information in a child dataset, the parent dataset automatically updates also.

- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.6 Version 2.1.0

Released: July 2021

New features

This release includes the following new features:

- Use batch processing to automatically attach assets. Note that this feature requires the EBX® Data Exchange (New) Add-on. See Performing bulk upload and attach [p 77] for more information.
- You can now specify that assets must meet minimum height and width requirements for upload.
- A regular expression can now be defined in a D.A.C or Drive configuration to enforce compliance of asset physical file names. If the name doesn't meet the specified pattern, users cannot upload the asset.
- Performance improvements were made when:
  - Importing assets with tags.
  - Importing assets from a zip file.
- The **Upload** screen now displays additional information about the Drive and D.A.C.

Changes in Functionality

Instead of uploading assets and then validating them, the add-on saves time by preventing upload of invalid assets. The following asset attributes are validated before upload to ensure compliance with configuration settings:

- File size
- File extension
- Image resolution
- Max Drive size on bulk upload
- File name compliance with specified regex patterns

Changes to third-party libraries

The Apache PDFBox, Preflight, pdfbox-tools, and XmpBox libraries were updated to version 2.0.24.

Closed Issues

This release contains the following closed issue: *[EBXDMA-796]* Users can see assets uploaded by other users even when permission settings prohibit this behavior.

Known Issues

This release contains the following known issues:

- Searching for assets in the media-type field using quick search is not supported. However, when previewing some file types, such as PDFs, you can search for text within the file.
- When updating asset information in a child dataset, the parent dataset automatically updates also.
- The configuration is not updated for the supported file extensions after users access the Digital Asset Manager widget.
- Document previews cannot display Vietnamese text.
- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Thumbnails of some uploaded SVG files do not display correctly.
- Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.

29.7 **Version 2.0.1**

Released: June 2021

Updates

The following library updates were made in this release:

- The Apache PDFBox, FontBox, pdfbox-tools, Preflight, and XmpBox libraries were updated to version 2.0.23.
- The Apache HTTPClient library was updated to version 4.5.13.
- The Apache Commons IO library was updated to version 2.8.0.
- The JAI ImageIO library was updated to version 1.4.0.
- The Zip4j library was updated to version 1.3.3.
- The Google Guava library was updated to version 30.
- The scripts.bundle.js and Google Guava version 29 were removed.

Bug fixes

This release contains the following bug fixes:

- **[EBXDAMA5-567]** An attached asset is hidden when its field is set to read-only in the dataset.
- **[EBXDAMA5-753]** When a media type field is set to 'Read-only', add-on features are still available when viewing record details.

29.8 Version 2.0.0

Released: March 2021

New features and updates

This release contains the following new features and updates:

- The TIBCO EBX® Digital Asset Manager Add-on has been updated to ensure compatibility with the TIBCO EBX® 6.0.0 GA release.
- Support was added to enable the preview of XLS and XLSX files.
- The API was updated to allow you to enable and disable certain features when managing an external Drive.
- The following updates were made to known limitations:
  - The following limitation was removed: The **Deactivated** state is still available for user although user does not have the right to deactivate any asset.
  - The following limitation was added: Previews of PPT, PPTX, DOC and DOCX files cannot display charts due to a limitation of the library used to generate the preview.
- The following libraries were updated or removed:
  - dom4j to v2.0.3
  - Jackson-databind to v2.10.0
  - Apache PDFBox (Fontbox, etc …) v2.0.20
  - jaxb-api-2.1.jar to v2.1 and jaxb-impl-2.1.3.jar to v2.1.3
  - common-fileupload to v1.4
  - poi to v4.1.2
  - batik-all to v1.13
  - jQuery_fileupload to v10.31
  - log4j and slf4j were removed

- APIs that were deprecated in previous versions of the add-on are no longer available. If you use any deprecated APIs, please refer to the list below for their replacements:

> **Attention**
> All APIs in the following table are located in the `com.orchestranetworks.addon.dama` package.

Deprecated	Replacement
`AccessPermission`	`ext.enumeration.AccessPermission`
`DAMException`	`ext.exception.DAMException`
`DateRange`	`ext.bean.DateRange`
`DigitalAsset getAssetUUID()`	`ext.bean.DigitalAsset getAssetId()`
`DigitalAssetKey`	`ext.bean.DigitalAssetKey`
`AbstractDigitalAssetSpec`	`DigitalAssetSpec`
`DigitalAssetState`	`ext.enumeration.DigitalAssetState`
`DigitalAssetVersionKey`	`ext.bean.DigitalAssetVersionKey`
`DigitalAssetVersionSpec`	`ext.bean.DigitalAssetVersionSpec`
`DriveFactory`	`ext.factory`
`DriveManager`	`ext.DriveManager`
`DriveType`	`ext.bean.DriveType`
`MetaData` `OperationExecutionStatus`	`ext.bean.MetaData` `ext.exception.OperationExecutionStatus`
`ResourceIdentifier`	`ext.ressource.ResourceIdentifier`
`ResourceIdentifierFactory`	`ext.resource.ResourceIdentifierFactory`
`SearchContext` `ext.bean.SearchFilter`	`externalmanagement.search.SearchConfig` `externalmanagement.search.SearchFeature`
`Size`	`ext.bean.Size`
`SortBy`	`ext.enumeration.SortBy`

Deprecated	Replacement
`StorageManager`	`ext.StorageManager`
`Version`	`ext.bean.DigitalAssetVersion`

Note

A validation error might prevent the **Backup** service from executing properly. If this occurs, open the validation report and select **Reinitialize validation report**. Then, re-execute the **Backup** service.

29.9 Version 1.8.8

Released: February 2021

Product updates

Google Guava was updated to version 30.1.

Bug fixes

[EBXDAMA5-1657] An exception occurs in the add-on when using the **Manage digital asset** service.

29.10 Version 1.8.7

Released: January 2021

Product updates

This release contains the following library updates:

- Apache Standard Taglibs to version 1.2.3.
- Jackson Databind to version 2.11.2.
- Batik XML utility library to version 1.13.
- Apache Common Codec to version 1.13.

Bug fixes

[EBXDAMA5-1651] Absolute URLs are not calculated as expected.

[EBXDAMA5-1656] A NullPointerException is thrown while starting EBX.

29.11 Release Notes 1.8.6

Release Date: October 20, 2020

Bug fixes

[EBXDAMA5-1590] A complete URL is required when sharing an asset by email.

29.12 Release Notes 1.8.5

Release Date: September 18, 2020

Updated features

This release contains the following feature updates:

- The following library updates were made: jQuery to v3.5.0, jQuery UI to v1.12.1, jQuery fileupload to v10.31, and Apache POI to v4.1.2. Additionally, DOCConverter, OfficeConverter, PPTConverter, and PPTXConverter were updated.
- Support for the OpenJDK8 and Open JDK11 libraries has been included.

Bug fixes

[EBXDAMA5-1583] The log was updated from INFO to DEBUG mode to display more readable results.

29.13 Release Notes 1.8.4

Release Date: June 23, 2020

New features and updates

This release contains the following feature updates:

- The add-on now provides administrators the option to activate assets when viewing them from the **Digital asset** screen. To perform this action, select the **Activated** option from the **Actions** drop-down menu.
- In the **Manage digital assets** screen, the **Drive** drop-down list is now sorted alphabetically.
- File formats listed in the **Blacklisted extensions** property cannot be attached from the **Editor** screen.
- Behavior has been updated to improve consistency in the way error messages display.
- When an administrator moves assets from an encrypted Drive to an unencrypted Drive, the add-on automatically decrypts assets. This functionality applies when moving assets in the opposite manner—decrypted to encrypted.
- The error message shown when uses are not able to attach an asset has been updated.
- Support for JDK8 and JDK11 has been updated. Additionally, the docx4j, pdfbox, and jQuery libraries have been updated.

Bug fixes

This release contains the following bug fixes:

- **[EBXDAMA5-1180]** [IE] The Esc key does not work correctly when previewing a document file in full-view mode.
- **[EBXDAMA5-1181]** [IE] The search page service does not work correctly when previewing document assets.

- **[EBXDAMA5-1206]** [IE] The first page of a document file cannot be completely displayed in full-view in the **Editor** or **Manage digital assets** screens.
- **[EBXDAMA5-1557]** With assets that have tags, the **Last updated date** property is updated when saving and closing—even if there is no change to the asset.
- **[EBXDAMA5-1558]** In the **Manage Digital Assets** screen, selected tags are not cleared after changing the Drive.
- **[EBXDAMA5-1559]** Loading issue exceptions that link to the DOCX4J dependency prevent preview of DOCX files on Drives.
- **[EBXDAMA5-1561]** The documentation and tool tips for the **Blacklisted extensions** should reflect the field behavior that listed extensions can also be separated by whitespace, semicolons, or dots.
- **[EBXDAMA5-1566]** An unexpected message is given when users upload DOC/DOCX files (about 300kb) in the UI.
- **[EBXDAMA5-1581]** All Drives are displayed in the **Manage digital assets** screen instead of the defined Drive when running the add-on service on a perspective.
- **[EBXDAMA5-1589]** A NullPointerException is raised when the thumbnail displays an SVG default icon.
- **[EBXDAMA5-1590]** There should be a complete URL for an asset which is shared via email.

29.14 Release Notes 1.8.3

Release Date: April 20, 2020

Updated features

When you use the REST API to retrieve an asset's URL, the returned JSON object includes the following new response parameters:

- `preview_url`: The URL of the asset preview.
- `defaulticon`: The URL of the default file extension icon.

Bug fixes

This release contains the following bug fixes:

- [EBXDAMA5-1571] When using the add-on's REST API, the value returned in the URL field must be the URL of the original file and not the file preview's URL.
- [EBXDAMA5-1579] If the uploaded file is >5MB, a thumbnail image based on the source file must not be generated.
- [EBXDAMA5-1582] The DAMException log is updated when attempting to delete an asset that has been removed.
- [EBXDAMA5-1584] An asset version cannot be deleted if its binary file has already been deleted.

29.15 Release Notes 1.8.2

Release Date: February 20, 2020

New features and updates

This release contains the following feature updates:

- Tooltips have been added for the **Drive** and **Tags** properties when uploading assets.
- PDF files that are less than 5MB can now be previewed.
- When using the API to get an asset's URL, the parameter in the returned JSON object now matches the protocol used in the request.
- The **Editor** can now be used to edit assets when using a TIBCO EBX® Cloud Instance.
- When viewing assets from a storage location managed outside of EBX®, PDF files can now be previewed in the **Editor**.
- If using an externally managed Drive location, the **State** field is now hidden when viewing information about an asset.
- The number of assets that can be included in a single upload has been increased to 300.
- When there are multiple versions of an asset, the displayed file size now updates to the current version of the asset.
- The thumbnail and preview display for SVG files has been updated. If the file includes animation, users can interact with it when previewing.

Bug fixes

This release contains the following bug fixes:

- **[EBXDAMA5-1223]** Duplicated tags are not hidden when users add a tag with a space character, which already exists.
- **[EBXDAMA5-1363]** A `TableTrigger` is fired in the add-on even when there is no change in the record.
- **[EBXDAMA5-1384]** An error message should be raised when uploading assets on the **Editor** screen through a deleted D.A.C.
- **[EBXDAMA5-1435]** A deactivated asset is missing in the attachment pane when it is deactivated from the **Manage digital assets** screen.
- **[EBXDAMA5-1458]** The loading icons are not aligned with other thumbnails in the list when quickly uploading multiple assets from the **Edit digital asset** screen.
- **[EBXDAMA5-1459]** Unexpected behavior occurs when uploading a file to create a new version from the Media type field.
- **[EBXDAMA5-1465]** The API's `UIWidget` to disable modifications in the editor does not work.
- **[EBXDAMA5-1470]** The system is not responsive after double-clicking on assets.
- **[EBXDAMA5-1471]** A null pointer exception is thrown when the `ExternalManager`'s `getTags` API returns null.
- **[EBXDAMA5-1472]** Users cannot preview assets in an external Drive from the **Edit digital asset** screen.
- **[EBXDAMA5-1473]** After changing the label of a dataset, the download button is not available on the **Edit digital asset** screen.

- **[EBXDAMA5-1475]** Incorrect behavior occurs when accessing a perspective using the **[ebx-addon-dama] Manage digital assets** service.
- **[EBXDAMA5-1485]** [API - External Management] Change the default value for **All file extensions** from **0** to **""**.
- **[EBXDAMA5-1496]** A blank screen briefly appears when attaching a large number of assets.
- **[EBXDAMA5-1502]** An asset cannot be shared via email on the **Digital Asset Manager - Editor** screen on a Relational model.

29.16 Release Notes 1.8.1

Release Date: December 10, 2019

Updated behavior

The add-on no longer refreshes data model schemas by retrieving a list of all registered models in the repository.

29.17 Release Notes 1.8.0

Release Date: November 8, 2019

New features and updates

The following sections highlight the new features and updates for this release:

- Features and enhancements [p 145]
- API updates [p 146]

Features and enhancements

- You now have the option of enabling encryption upon Drive creation. This option encrypts all assets in a Drive so they can only be accessed by a user logged in to EBX®.
- When uploading multiple files, asset thumbnails now display immediately after upload of that asset is complete.
- A Drive can now be configured to access an externally managed storage location. For example, assets might be stored in the cloud on Google Drive or AWS. Alternatively, you might already use a third-party tool to manage digital assets but, you want to enable data model access to these assets. Note that API implementation is required for these types of use cases.
- To improve user experience in certain deployment environments, Spring configuration has been moved from hybrid to pure Java.
- It is no longer mandatory for Digital Asset Components to have associated tag clouds.
- A result page now displays after running the **Delete all assets in Drive** service.
- By default all file types are now supported and enabled for upload by the add-on. However, you can use two new properties **Restrict file upload** and **Blacklisted extensions** to whitelist and blacklist file extensions, respectively. See, Whitelisting and blacklisting file formats [p 67] for more details.
- SVG files are now supported.

API updates

Using the add-on's API you can now access externally managed storage locations. These locations can include assets managed by third-party tools. In addition to linking to and displaying these assets in EBX® data model fields, you can:

- Search for and filter assets.
- Enable upload and deletion of assets.
- Attach and detach externally managed assets to EBX® data model fields.
- Display externally managed tags.

Known limitations

This release contains the following limitations:

- When accessing assets in an external Drive, error messages may display when selecting all assets. This can be the result of access limitations put in place by the third-party tool.
- Without width and height measurements, the thumbnail and content of an SVG asset are not properly displayed.

Bug fixes

This release contains the following bug fixes:

- **[DAMA-1330]** The Vietnamese title of a downloaded file is wrong.
- **[DAMA-1335]** A thumbnail label that contains HTML elements is not correctly displayed in the **Edit digital asset** screen.
- **[DAMA-1336]** On the list view mode the thumbnail of an asset overlaps the outer border of the media type field.
- **[DAMA-1337] Label and description** value is not saved after being modified in the **Digital asset** table.
- **[DAMA-1338]** An user-unfriendly error page displays in the **Editor** and **Manage digital assets** screens when users attempt to download a non-existent asset.
- **[DAMA-1348]** The error icon displays incorrectly when using quick upload.
- **[DAMA-1359]** The **Select all** button does not work on the **Editor** screen after assets have been detached.
- **[DAMA-1376]** The loading icon is not aligned with other thumbnails in the row when quickly uploading an asset.
- **[DAMA-1381]** Assets cannot be displayed on the field when they are attached from the **Editor** screen and the view mode has been changed in the D.A.C.
- **[DAMA-1391]** [Chrome] Incorrect behavior is presented when user selects the **Apply** checkbox on the search pane.
- **[DAMA-1420]** A redundant header appears on the upload modal dialog from the **Icon** field on **File extension** record.

29.18 Release Notes 1.7.0

Release Date: June 20, 2019

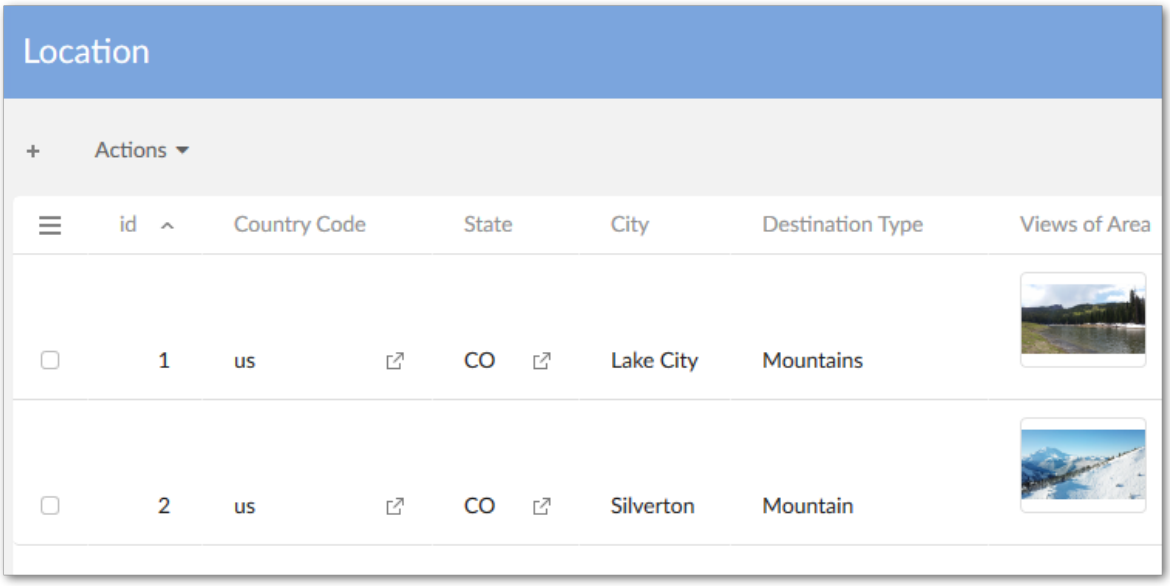
New features and updates

The following sections highlight the new features and updates for this release:

- Features and enhancements
- API updates

Features and enhancements

- The **Manage digital assets** service can now be launched from EBX® perspectives. The service can allow users control over assets linked to one or more Drives.
- When choosing a new asset as the current version, the new asset's file name can be automatically populated to the **Editor** using the new **Update label** option.
- Add-on performance has been improved when a schema refresh is required.
- The Jackson library has been upgraded due to a security issue.
- The PhotoSwipe library has been removed from the add-on.
- When playing asset videos from within the add-on, the video is now streamed instead of downloaded.
- An asset thumbnail can now be displayed in EBX® tables. See Enabling thumbnail display and upload in the tabular view [p 26] for more information. The following image shows an example of this feature:



API updates

The API now includes the following; please refer to the *Java API* documentation for additional information and samples:

- The API now permits assets managed by third-party tools to be linked to data model fields.
- A custom filter can now be implemented that determines the assets displayed when users enter the **Editor**.
- Sample code has been added to demonstrate implementing cloud storage for assets.
- The Digital Asset `UIBean` has been replaced by `CustomWidget`.

Bug fixes

This release contains the following bug fixes:

- **[DAMA-1028]** Text overlap occurs in the PDF version.
- **[DAMA-1048]** Grant permission while calling DAM API in Script tasks.
- **[DAMA-1188]** The loading icon persists when previewing a large video (more than 164MB).
- **[DAMA-1196]** A blank page is shown when users access document files on UI Bean when the drive has been deleted by another user.
- **[DAMA-1197]** An error is raised when accessing assets on UI Bean if the D.A.C. has been deleted by another user.
- **[DAMA-1249]** A blank space appears on the 'Edit digital asset' and 'Import a new version' screens.
- **[DAMA-1255]** The error message must be unified when user accesses an asset on DAM UI Bean in case the asset or the D.A.C. is deleted.
- **[DAMA-1256]** An unreasonable message is given when user downloads assets from different screen.
- **[DAMA-1259]** The scroll bar does not move down to the newest point on the upload modal screen.
- **[DAMA-1269]** The search page box does not work when previewing document assets in Internet Explorer.
- **[DAMA-1285]** The error message must be unified when a user accesses an asset on the **Editor** if the asset or the D.A.C. is deleted.
- **[DAMA-1287]** Large videos (more than 164MB) cannot be downloaded.
- **[DAMA-1292]** Error log at runtime occurs when deploy an add-on using UI Framework.
- **[DAMA-1294]** A tag cannot be created when a user re-selects an existing tag on upload modal screen.
- **[DAMA-1298]** A `NullPointerException` occurs while registering `DriveManager`.

29.19 **Release Notes 1.6.0**

Release Date: March 25, 2019

New features and updates

The following sections highlight the new features and updates for this release:

- General updates [p 149]
- Updates to the UI [p 149]

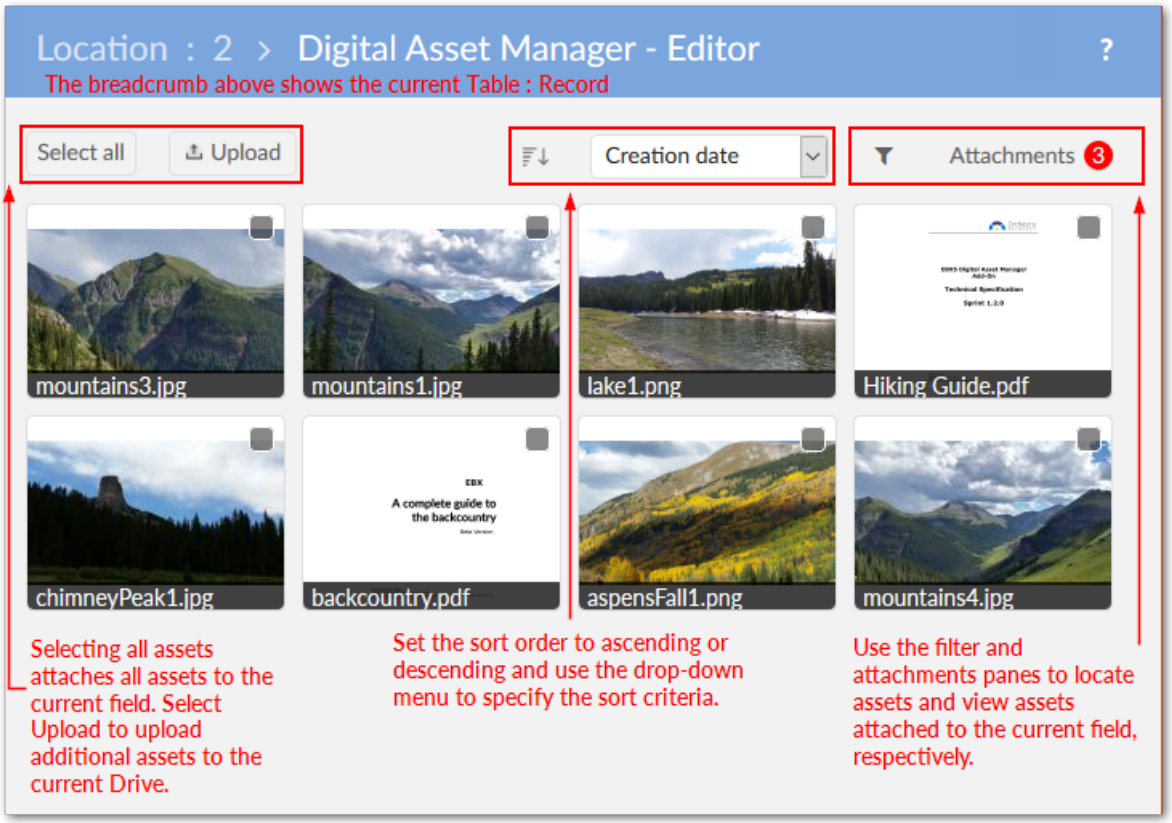
General updates

This release contains the following new features and updates:

- The add-on now supports TIBCO EBX® restriction policy functionality for user permissions.
- A programmatic filter can now be used to filter assets displayed in the editor.
- The `baseURL` parameter has been removed from the `DigitalAsset.getURL()` method.
- A public REST service has been provided to upload an asset.
- Documented code samples have been updated.
- Support has been added to take into account the new version of the TIBCO EBX™ Information Search Add-on (2.4.0).

Updates to the UI

- The **Manage digital assets** UI has been updated.
- The following updates were made to the **Editor** UI:
  - The UI is now responsive and automatically resizes based on the window or screen size.
  - Infinite scroll has been added to the **Editor** for assets and tags.
  - As highlighted in the image below, the UI layout has also been updated:



Bug fixes

This release contains the following bug fixes:

- **[DAMA-745]** The title of the download pop-up is incorrect when running the **Import from a ZIP file** service.
- **[DAMA-1064]** A JavaScript error is raised in Internet Explorer when cancelling an upload.
- **[DAMA-1154]** No message is given when user imports a ZIP file exceeding available server space.
- **[DAMA-1187]** The user can change the current version without having sufficient permission.
- **[DAMA-1200]** [IE] The error icon's red border no longer displays when a user hovers their mouse over the icon.

29.20 Release Notes 1.5.1

Release Date: December 14, 2018

Featured updates

The Creative Commons license has been moved from ECI to BAS.

Bug fixes

This release contains the following bug fixes:

- **[36575]** The **Identify unused asset** service does not work with assets on the media type field under a group.
- **[36577]** The **Identify unused asset** service does not work with assets on the media type field under a table belonging to a group.

29.21 Release Notes 1.5.0

Release Date: October 26, 2018

New features and updates

This release contains the following new features and updates:

- The new **Change physical root path** service allows you to update a Drive's storage location. Once updated all add-on related assets are migrated to the new location.
- REST services for TIBCO EBX™ GO Add-on have been implemented.
- Significant updates have been included to ensure compatibility with the EBX® 5.9.0 GA release.

Bug fixes

This release contains the following bug fixes:

- **[20798]** The **Permanently delete digital asset** and **Delete digital asset version** services do not work on a hierarchy view.
- **[26790]** Lack of a border and a text overflow when a tag has long text.
- **[28668]** Users cannot create tags containing quotation marks on the **Upload new digital assets** dialog.
- **[28993]** Users are not allowed to close the **Upload** pop up while uploading assets.

- **[29229]** Inconsistent display of **Upload** and **Edit** buttons between a media type field using **Quick upload and attach** and the UI.
- **[29300]** The display of service buttons on the thumbnail of asset on the Digital Asset Manager UI Bean are inconsistent.
- **[30085]** An error message occurs when users click **Select all** after detaching assets including the **File not found** items.
- **[32131]** The upload files process does not work properly.
- **[33465]** The search pane's tooltip on the **Editor** / **Manage digital asset** screen is wrong.
- **[33545]** An incorrect tooltip is displayed for the **Cancel** button in the **Importing assets** screen.

29.22 Release Notes 1.4.3

Release Date: November 29, 2018

Featured updates

The Creative Commons license has been moved from ECI to BAS.

Bug fixes

This release contains the following bug fixes:

- **[36578]** The **Identify unused asset** service does not work with assets on the media type field under a group.
- **[36581]** The **Identify unused asset** service does not work with assets on the media type field under a table belonging to a group.

29.23 Release Notes 1.4.2

Release Date: September 13, 2018

Featured updates

The Digital Asset Manager Add-on has undergone updates to ensure compatibility with the EBX® 5.8.1 fix M release.

29.24 Release Notes 1.4.1

Release Date: June 22, 2018

Bug fixes

[33653] Files cannot be uploaded due to a conflict with the jQuery library.

29.25 Release Notes 1.4.0

Release Date: May 2, 2018

New features

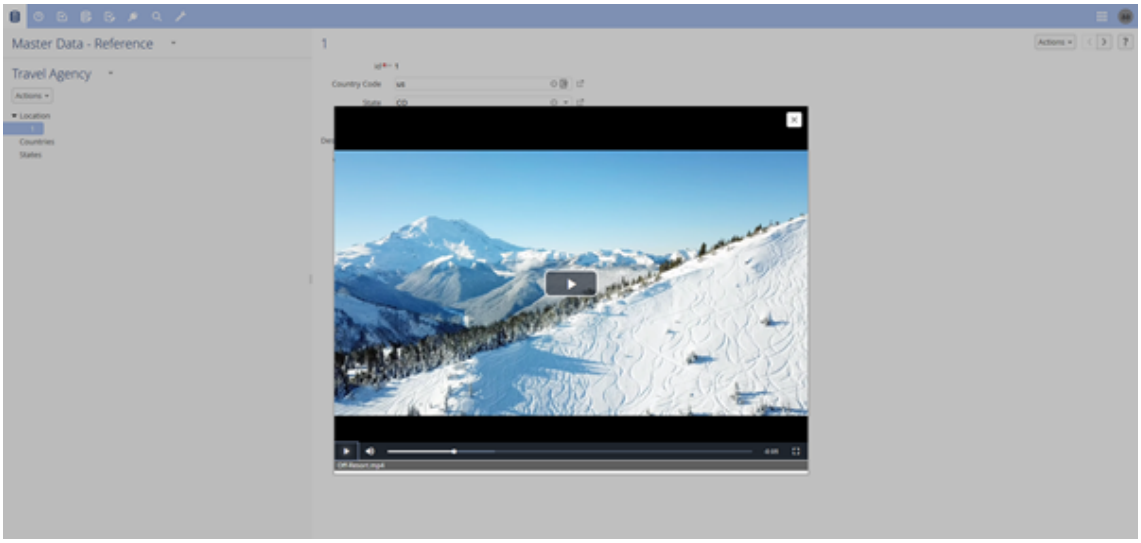
The following sections describe new features for the GA 1.4.0 release:

- Previewing assets [p 152]
- Importing assets in bulk [p 153]
- Record filtering [p 153]
- REST services [p 155]

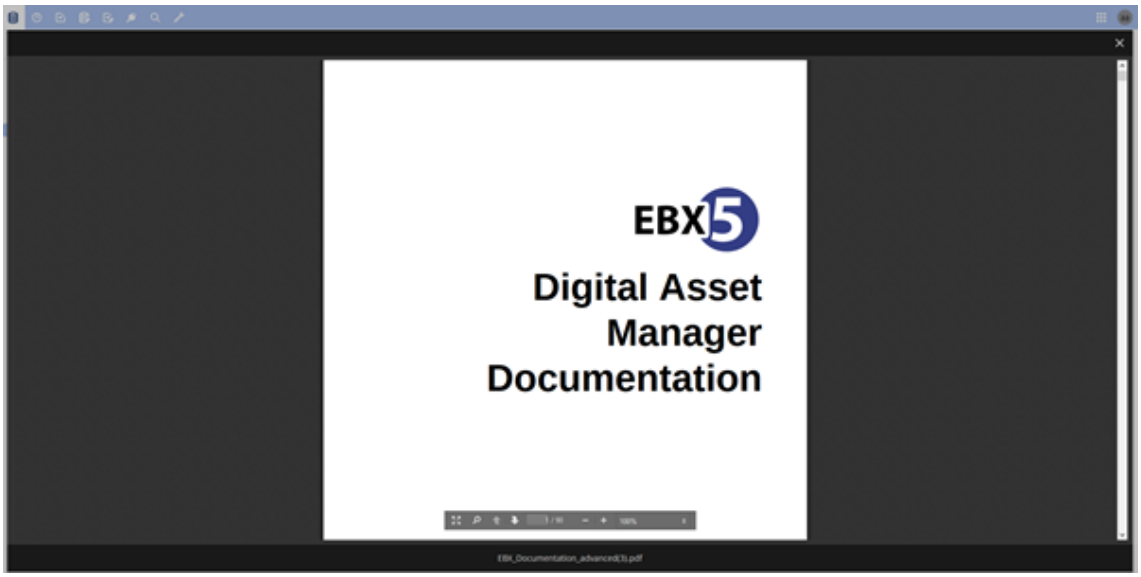
Previewing assets

The add-on now includes the ability to preview additional types of assets. See Previewing assets [p 35] for more detailed information. The newly supported types are:

- MP4 files: When previewing, the add-on automatically starts the video and enables basic playback control.



- Document files: When previewing PDF, DOC, DOCX, PPT, or PPTX assets you can use standard document viewer features, such as zoom, navigation, and search.



Importing assets in bulk

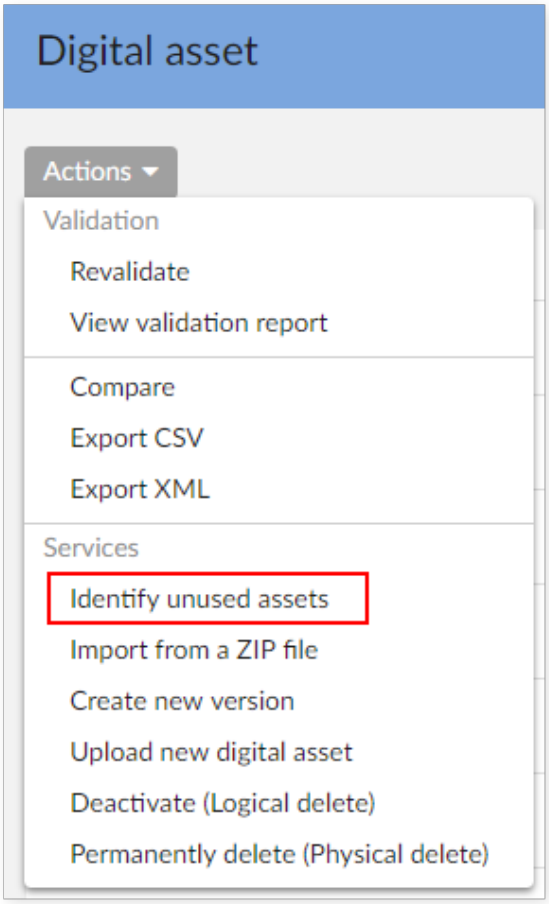
Administrators now have the option to perform a bulk upload of assets by importing a ZIP file. After import, the **Result** page's **Actions** menu enables:

- Download of a log file if the import failed.
- Comparison of two or more imported assets.
- Export of the report in CSV or XML format if the import was successful.

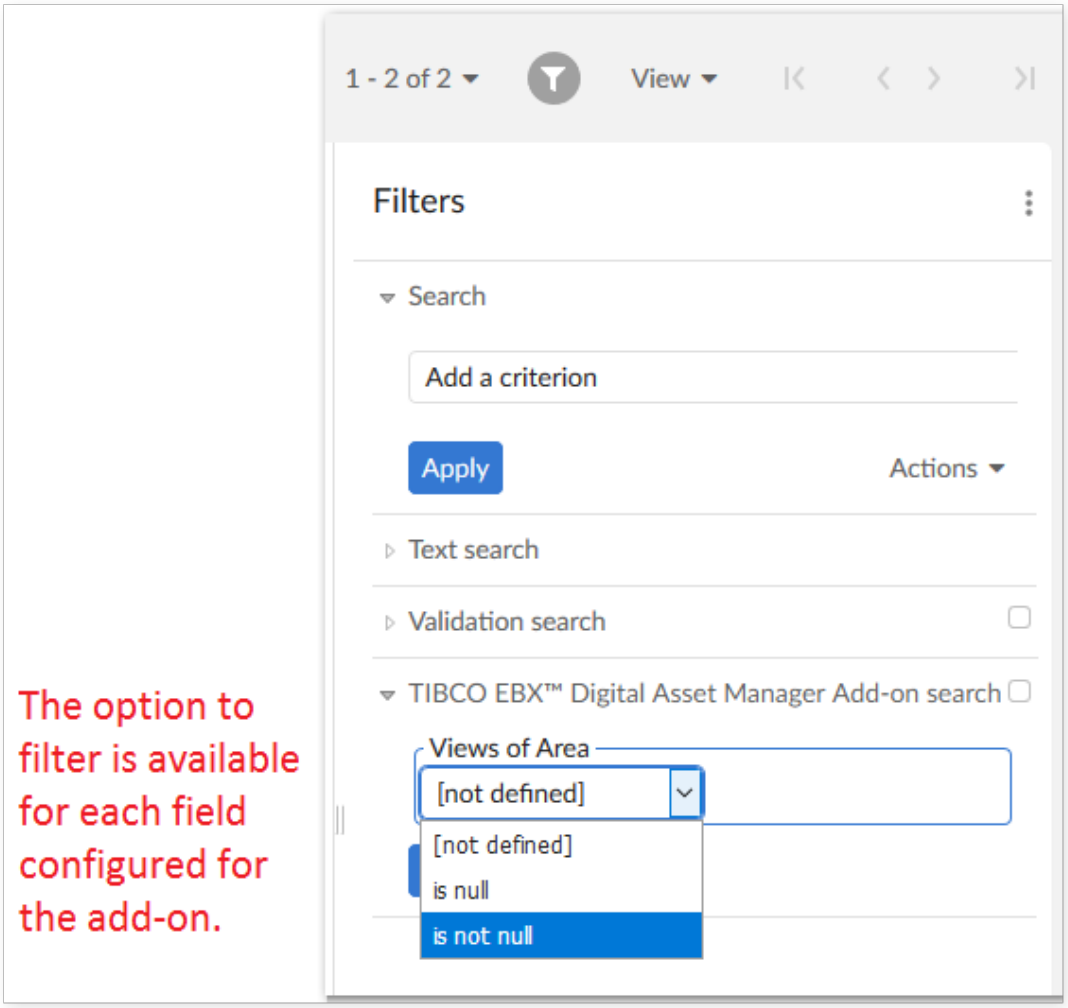
Record filtering

The add-on now provides the ability to filter the following types of records:

- Asset records that are not attached to a field. The new **Identify unused assets** service locates and displays all assets not attached to a field. From this view you can compare assets and export reports in CSV and XML formats. For more information, see Finding and removing unused assets [p 69] .



- Table records with no attached assets. The EBX® filter now contains the **Digital Asset Manager Add-on search** filter. The filter can be used to show/hide records that have attached assets. For more information, see Configuring asset filters [p 39] .



REST services

REST services are now available to:

- Get all digital assets from a Drive.
- Attach and detach assets from a field.

See REST service operations [p 82] for examples.

Known limitations

This release contains the following limitations:

- A JavaScript error is raised in Internet Explorer when canceling an upload.
- The progress bar does not work properly when importing from a ZIP file.
- Document previews cannot display Vietnamese text.
- The video seek bar does not work properly when a user previews MP4 files in Google Chrome.

Bug fixes

This release contains the following bug fixes:

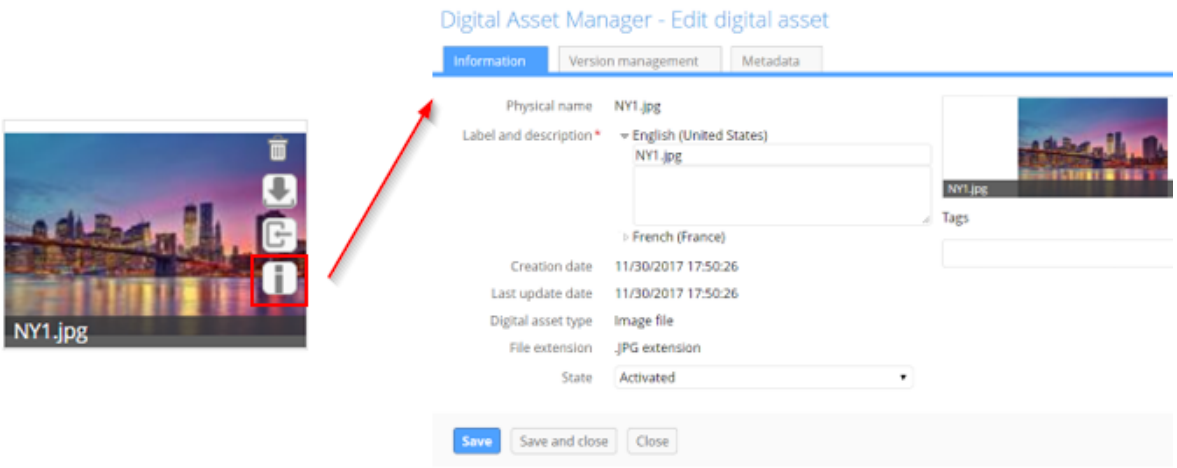
- **[30757]** Users can still view assets in the **Editor** after another user has deactivated the asset.
- **[30769]** Users can view other's assets even without sufficient permission in the D.A.C. or Drive.
- **[32140]** An error occurs on the loading page when clicking the 'Reset filter' button on the **Editor** screen.

29.26 **Release Note 1.3.0**

Release Date: December 15, 2017

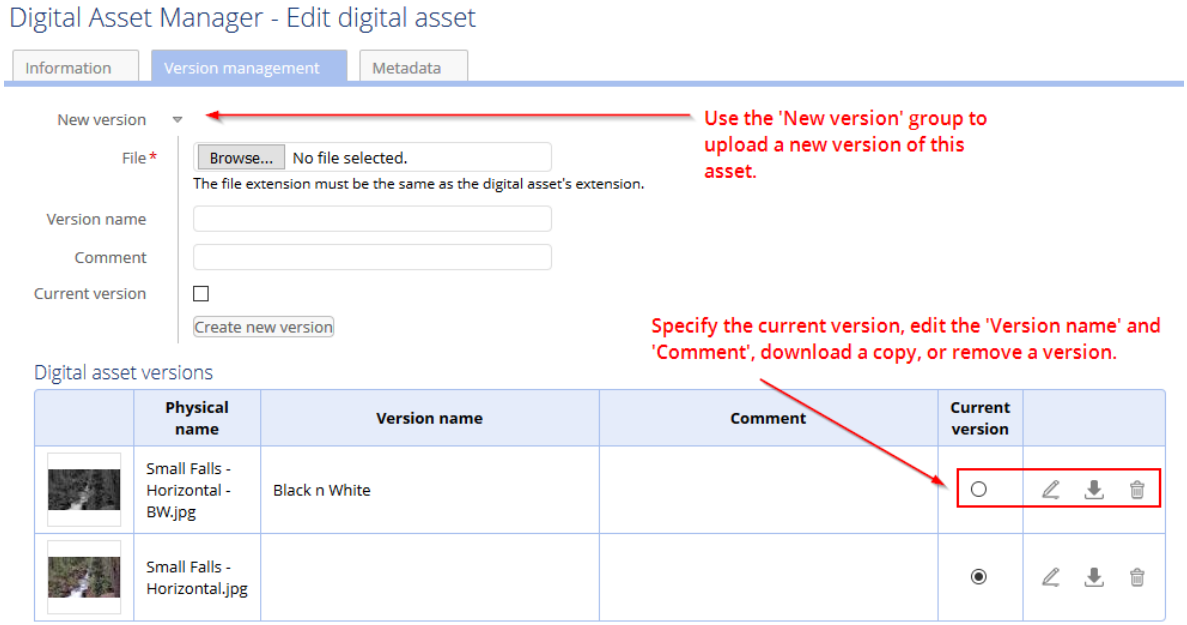
Overview of features and enhancements

- Usability improvements [p 157]
- A new **Information** icon is available on asset thumbnails. Clicking this icon opens the **Edit digital asset** screen where you can view detailed information about the asset, its versions, and its metadata.



- You now have the option of determining whether quick action icons always display on assets, or only when users mouse over the asset.

- When viewing asset versions, you can now download individual versions for comparison.



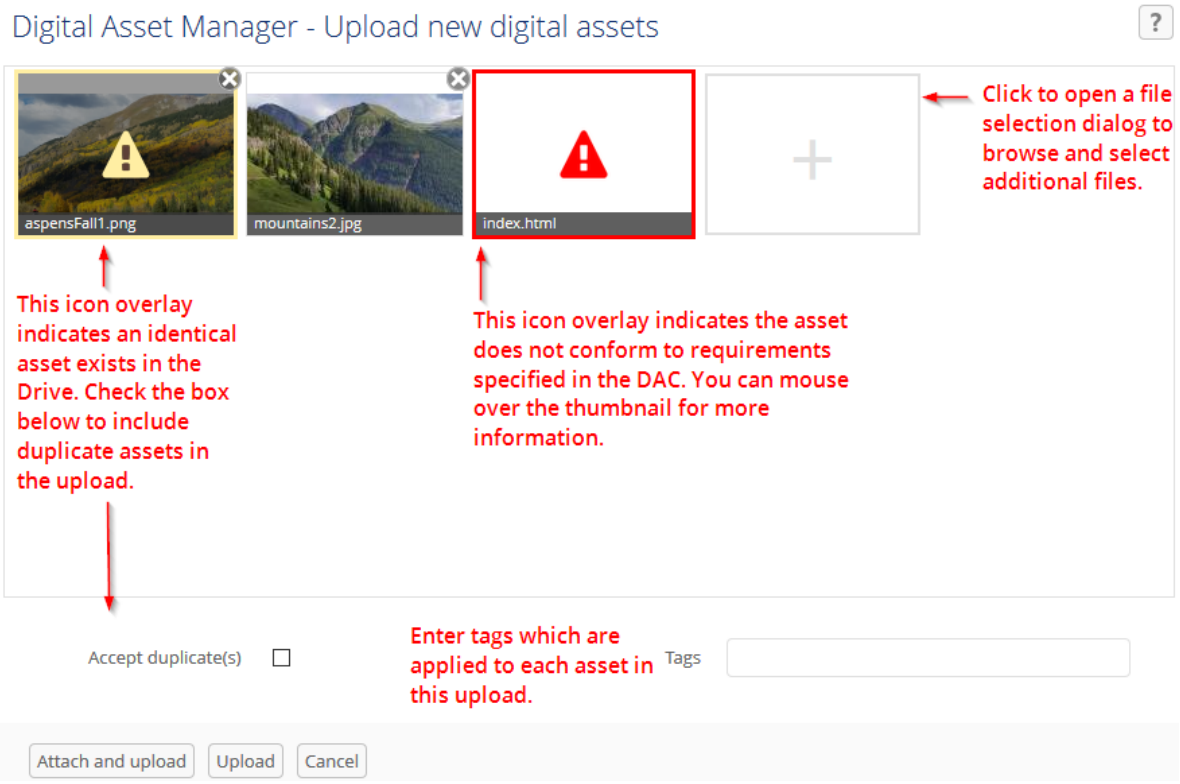
- New REST services are available that allow you to:
  - Retrieve an asset's URL.
  - Get all mediaType fields registered in Digital Asset Manager Add-on.
  - Get information about assets attached to a mediaType field.

Usability improvements

A large portion of the asset management process involves getting your assets into the manager. With this in mind, the latest version of the Digital Asset Manager Add-on provides the following usability improvements focused on helping you during the upload process (see Uploading and attaching assets [p 27] for more information):

- Instead of bringing in assets individually, you can now upload and attach more than one asset at a time.
- A new quick-upload mode allows users to bypass pop-up dialogs and directly upload and attach assets. This mode integrates well with workflows.
- In addition to browsing to select files, drag-n-drop functionality is now included in the upload screen.
- Duplicate asset management during upload has been improved. The upload screen now indicates which assets are duplicate and you can choose whether to keep, or remove these assets during upload.
- When browsing for files to select during upload, the add-on applies a filter and only displays those extensions supported by the current D.A.C. configuration.

The following image highlights some of the aforementioned features:



Bug fixes

- **[28586]** The 'Tag' text box disappears in the 'Manage digital assets' screen after removing tag on an asset.
- **[28623]** The 'Drive' field is null and Tags are displayed after changing value of 'Drive' in the D.A.C.
- **[28675]** An error occurs on the Digital Asset Manager UI Bean and an exception is thrown in the log when users use an uncompleted list view.
- **[28839]** An error occurs and an exception is thrown in the log when running 'Attach' or 'Detach with an uncompleted list view.
- **[28846]** A record cannot be created by using 'Duplicate record' on 'Drive' table in the 'TIBCO EBX™ Digital Asset Manager Add-on' dataset.
- **[28871]** An error occurs when the media type field is in a group with multiple elements.
- **[28877]** A newly uploaded asset is attached into the first occurrence in spite of being uploaded from another media type field.
- **[28883]** Overlap occurs in the 'Editor' screen on IE8.
- **[28886]** Users can create an image configuration even when the mandatory fields of 'Max width' and 'Max height' are empty.
- **[28887]** The 'Bin' icon is displayed without thumbnail on the Digital Asset Manager UI bean after detaching the asset on IE8.
- **[29304]** A user can still open the Editor screen without sufficient permission.

- **[29722]** An unsupported file can be attached to the 'File attachments' pane on the Editor screen.
- **[29836]** An exception is thrown in the log when inputting a value into the 'Metadata' table without choosing 'Metadata nature'.

29.27 Release Note 1.2.2

Release Date: November 7, 2017

New features

- You can now download assets contained in read-only records.
- It is now possible to upload a new asset version from the list view.
- A new REST service is available to retrieve a digital asset's URL.

29.28 Release Note 1.2.1

Release Date: October 16, 2017

New features

- Take into account the new version of TIBCO EBX™ Information Search Add-on (2.0.0).

29.29 Release Note 1.2.0

Release Date: September 6, 2017

New features

UI updates and improvements

- The Editor layout has been improved.
- Assets can be displayed in fields using a combination of settings that include using thumbnails, or icons and using a list, or tiled mosaic view.
- Display of asset thumbnails has been improved and also retains an image's original aspect ratio.

Functionality improvements

- User permissions have been expanded and can now be set at the Drive and D.A.C. levels.
- In the Editor, assets can be selected to shown an expanded view. Several options for downloading, editing, and sharing are available from the expanded view.
- Search, sort, and filter options available in the Editor have been updated.
- An asset's label description, version name and comments, and tags can be modified during upload.
- The following actions can be performed when mousing over an asset thumbnail: detach, download, or import of a new version.
- Tag domains have been replaced with tag clouds. Tag clouds can be used to organize tags and display them in the Editor.
- Duplicate assets can now be uploaded.

- Uploaded assets are saved in sub-folders so that the limitations of certain file systems don't prevent the upload of additional assets.

Bug fixes

- **[25631]** Users cannot run the Restore service when they do not have delete permission on a Drive contained in the backup file.
- **[26113]** There is a redundant 'Upload new digital asset' dialog when users upload an asset.
- **[26114]** There is an error when clicking a thumbnail that has been accidentally deleted in the file system.
- **[26115]** An asset's label displays incorrectly when users input HTML characters.
- **[26116]** A 403 page error occurs when users click on the download button from Digital Asset Manager UI Bean.
- **[26117]** Multiple errors occur when users input HTML characters into Domain and Tag domain.
- **[26118]** The tooltip of an asset is not updated after running the Refresh service.
- **[26119]** [Configuration] An error occurs after clicking on the 'Create a record' button in the 'History' table.
- **[26120]** [IE] The label and textbox of the 'Domain' field are not positioned correctly.
- **[27962]** An unknown record is automatically created in the 'Digital Asset Type' table.

29.30 **Release Note 1.1.1**

Release Date: August 16, 2017

Bug fixes

- **[27520]** A 'Drive type' that connects to cloud storage cannot be created.

29.31 **Release Note 1.1.0**

Release Date: April 18, 2017

New features

- A new public API is available to create, update, and delete digital assets and versions. It also allows connection to additional types of drives.

Bug fixes

- **[25406]** The system returns incorrect results when filtering label and domain.
- **[25413]** The 'Last update date' is not updated after users update domain and tag of an asset.
- **[25438]** Unauthorized access to resources is possible.

Warnings

- **Customers are strongly advised to upgrade to the latest version which patches the security issue.**

29.32 Release Note 1.0.2 fix 001

Release Date: April 18, 2017

Bug fixes

- **[25508]** Unauthorized access to resources is possible.

Warnings

- **Customers are strongly advised to upgrade to the latest version which patches the security issue.**

29.33 Release Note 1.0.3

Release Date: April 3, 2017

Bug fixes

- **[22873]** Even when users deactivate a digital asset, it still appears in the 'Digital asset list' pane, 'Attached' pane, and 'Media' field.
- **[22879]** The wrong current version of digital asset is downloaded.
- **[22917]** There is an error when previewing the digital asset when the file has been accidentally deleted in the file system.
- **[24361]** The digital assets cannot be loaded when the 'Digital asset type' option is set to 'image'.
- **[24661]** After modifying business version or comment, the system uploads the previously selected file again.
- **[24677]** Domain for a digital asset cannot be created when the 'Domain' field contains special characters.
- **[24723]** An unexpected error occurs in version management tab when displaying a digital asset without setting the current version.
- **[24825]** An unexpected error occurs when a digital asset has no label.
- **[24871]** Digital asset is not deleted after users click the 'Cancel upload' button.
- **[25116]** An unexpected error occurs when users modify ebx.locales.available=en in ebx.properties.

29.34 Release Note 1.0.2

Release Date: January 23, 2017

New features

- When specifying a Drive's storage location using the 'Physical root path' property it is now possible to use a relative path.

Bug fixes

- **[24115]** Abnormal behavior is recorded when importing an EBX® file.

29.35 **Release Note 1.0.1**

Release Date: October 28, 2016

New features

- Images that use the CMYK color space can be managed with the Digital Asset Manager Add-on.

29.36 **Release Note 1.0.0**

Release Date: October 12, 2016

New features

- A 'Media-type' field can be added to a data model to enable digital asset management.
- Assets can be uploaded using the add-on's services supplied at the administrative, dataset and field levels.
- An editor is available that allows you to perform the following actions on digital assets: upload, select, deselect, search, filter, edit, preview, download, share, deactivate and delete.
- Several custom services are available that allow you to perform asset management-specific actions in. The services are covered in more detail in the main documentation.
- It is possible to create a backup of, and restore the Digital Asset Manager Add-on's configuration settings and assets.