

# Perceptive Capture Connector

## Installation and Setup Guide

Version: 5.0.x

Written by: Documentation Team, R&D  
Date: December 2021

Hyland

## Copyright

Information in this document is subject to change without notice. The software described in this document is furnished only under a separate license agreement and may be used or copied only according to the terms of such agreement. It is against the law to copy the software except as specifically allowed in the license agreement. This document or accompanying materials contains certain information which is confidential information of Hyland Software, Inc. and its affiliates, and which is subject to the confidentiality provisions agreed to by you.

All data, names, and formats used in this document's examples are fictitious unless noted otherwise. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright law, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Hyland Software, Inc. or one of its affiliates.

Hyland® and Hyland Software®, as well as Hyland product names, are registered and/or unregistered trademarks of Hyland Software, Inc. and its affiliates in the United States and other countries. All other trademarks, service marks, trade names and products of other companies are the property of their respective owners.

© 2021 Hyland Software, Inc. and its affiliates. All rights reserved.

## Table of Contents

<b>About Perceptive Capture Connector .....</b>	<b>5</b>
Overview of the setup process .....	5
<b>Install the connector .....</b>	<b>6</b>
Install the connector using the drag-and-drop feature .....	6
Install the connector manually .....	6
<b>Upgrade the connector .....</b>	<b>7</b>
Upgrade the connector using the drag-and-drop feature .....	7
Upgrade the connector manually .....	8
<i>Uninstall the existing version of the connector .....</i>	<i>8</i>
<i>Install the latest version of the connector .....</i>	<i>8</i>
<b>Configure the connector .....</b>	<b>9</b>
PCC File Conversion Configuration .....	9
<b>Configure the connector for Brainware for Invoices.....</b>	<b>9</b>
Configure Brainware for Invoices for content integration .....	10
Configure the Content Connect Service.....	10
Create the workflow process .....	10
Create the queues .....	11
Create Brainware for Invoices File Output channels.....	11
<i>Channel requirements.....</i>	<i>11</i>
<i>Create and configure a File Output channel .....</i>	<i>11</i>
Create Brainware for Invoices Status Update channels .....	12
<i>Channel requirements.....</i>	<i>12</i>
<i>Configure and Import a BFI Status Update channel.....</i>	<i>13</i>
Create Brainware for Invoices Export channels .....	13
<i>Channel requirements.....</i>	<i>14</i>
<i>Configure and Import a Brainware for Invoices(BFI) Export channel.....</i>	<i>14</i>
<b>View Report Details.....</b>	<b>15</b>
<b>Triggers .....</b>	<b>16</b>
Document trigger .....	16
Status Update trigger.....	16
<i>Use trigger values in a channel input or output mapping .....</i>	<i>17</i>
<b>Actions .....</b>	<b>17</b>
Capture Document action.....	17

File System Document Export action .....	18
<i>Parameter Descriptions</i> .....	19
<b>Writers</b> .....	<b>21</b>
DCExport Subobject writer .....	21
Document Property writer .....	22
<i>Custom Properties</i> .....	22
Image writer .....	24
<b>Example of Perceptive Content workflow for Brainware for Invoices</b> .....	<b>26</b>

## About Perceptive Capture Connector

Perceptive Capture Connector (PC Connector) allows you to create, configure, and enable channels that map data and functionality between Perceptive Content and Brainware for Invoices (BFI). With a combination of channels, you can create a solution that enables exchange of files between Perceptive Content and BFI, routes documents in Perceptive Content according to their status code, and imports the extracted XML data into Perceptive Content eForm fields, document properties, or other objects. Each PC Connector channel consists of a trigger, an action, and a results phase.

A trigger is an event that initiates the channel. The PC Connector provides two triggers.

- **Document Trigger.** This trigger allows you to create, export, and soft delete documents in Content and create channels that pass XML data back to your original documents in Perceptive Content.
- **Status Update Trigger.** This trigger allows you to create channels that trigger when BFI provides status codes during document processing

An action is a task that the channel performs when the trigger event occurs. The PC Connector provides the following actions.

- **Capture Document Action.** This action allows you to create a document in Perceptive Content using the configured Capture Profiles and Application Plans.
- **File System Document Export.** This action exports any Perceptive Content documents from an Connect ASQ to a specific network location. . For example, you can use this action to create the TIFF files that Brainware for Invoices uses for data extraction.
- **The results phase** lets you specify what to do with the data that is available after the action completes. For example, you can output extracted XML values to Perceptive Content document keys, custom properties, eForm fields, and the DCExport subobject.

This guide outlines the installation and configuration procedures for the PC Connector. It also includes basic instructions for creating and configuring each channel type.

The PC Connector files include several sample XML files to help you understand the way you can configure your channel mappings.

For more information about other required Perceptive Connect Runtime components, refer to the Perceptive Connect Runtime Installation Online Help.

## Overview of the setup process

You can install the connector and set up the channels for enabling the connector to function for Brainware for Invoices.

To install the connector and set up File Output, Status Update, and Export channels, review or complete the following sections as needed.

- [Install the connector](#)
- [Create Brainware for Invoices File Output channels](#)
- [Create Brainware for Invoices Status Update channels](#)
- [Create Brainware for Invoices Export channels](#)

## Install the connector

You can install Perceptive Capture Connector, version 5.0.0 and related files manually using Perceptive Connect Runtime Web Console or from the “Upload new bundles” section in Perceptive Connect Runtime. You must install the connector on the same computer as the Perceptive Connect Runtime service.

### Install the connector using the drag-and-drop feature

You can obtain the Perceptive Capture Connector installation files and store them in a temporary directory on your computer for installation purposes.

To install the connector using Perceptive Connect Runtime, complete the following steps.

1. In a browser, type the address in the format `http://<Perceptive Connect Runtime machine name>:<port>`.
2. In the left pane, under **Manage**, click **Install a Connector**.
3. In **Perceptive Connect Runtime**, on the **Upload new bundles** page, drag the **Perceptive CaptureConnector 5.0.0-all.zip** file to the area designated by the box. You can also click **Manual upload** to upload the files manually.

**Result** Complete, queued, and processing files appear at the bottom of the page. After Perceptive Connect Runtime processes an item, a summary of the processed files, as well as an overview describing how many channels were affected by the installation, is displayed.

**Note:** The affected channel summary contains the following lists.

- **Fixed.** Channels that previously failed to validate and became valid after the installation.
  - **Broken.** Channels that previously validated and became invalid after the installation.
  - **Unknown.** Channels that PCR cannot verify as valid or invalid for the report. Usually, this is a byproduct of older channels with trigger parameters that are not stored in the database as plain text.
4. Perform one of the following actions:
    - Click **Accept** to accept the installation.
    - Click **Roll back** to undo the installation.

**Note:** You must accept or roll back the installation before PCR can process the next item. The bundle installation summary contains the following categories.

- **Installed.** Bundles installed from the file or archive and uploaded by the user.
- **Failed.** Bundles that failed to install. This includes a message with the reason for failure.
- **Skipped.** Files that PCR cannot install, such as non-JAR files.

### Install the connector manually

You can install the Perceptive Capture Connector from the Perceptive Connect Runtime Web Console page on Perceptive Connect Runtime. You can also install the connector by clicking “Manual upload” in the “Upload new bundles” page.

To install the PC connector, complete the following steps.

1. In a browser, type the Perceptive Connect Runtime Web Console URL in the format `http://<Perceptive Connect Runtime machine name or IP>:<port>`.
2. In the browser dialog box, enter the Web Console user name and password.

**Note:** The default user name and password are `admin`. However, the administrator can change the defaults during the Perceptive Connect Runtime installation process.

3. In the **OSGI Bundles** page, click **Install/Update**.
4. In the **Upload/Install Bundles** dialog box, complete the following substeps.
  1. Select the **Start Bundle** check box.
  2. Clear the **Refresh Packages** check box.

**Note:** The **Start Level** remains at the default value.

5. To install the JAR files included in the connector ZIP file, complete the following substeps.
  1. Click **Choose Files** and navigate to the connector in your temporary directory.
  2. Select the **capture-connector.content.mapping-5.0.0** and **capture-connector.endpoint-5.0.0** JAR files, one at a time, and click **Open**.
  3. Click **Install or Update**.

## Upgrade the connector

You can upgrade Perceptive Capture Connector, version 5.0.0 and related files manually using Perceptive Connect Runtime Web Console or from the “Upload new bundles” section in Perceptive Connect Runtime.

### Upgrade the connector using the drag-and-drop feature

To upgrade the connector using the drag-and-drop feature in Perceptive Connect Runtime, complete the following steps.

1. In a browser, type the address in the format `http://<Perceptive Connect Runtime machine name>:<port>`.
2. In the left pane, under **Manage**, click **Install a Connector**.
3. In **Perceptive Connect Runtime**, on the **Upload new bundles** page, drag the **Perceptive Capture Connector 5.0.0-all.zip** file to the area designated by the box. You can also click **Manual upload** to upload the files manually.

**Result** Complete, queued, and processing files are displayed at the bottom of the page. After Perceptive Connect Runtime processes an item, a summary of the processed files, as well as an overview describing how many channels were affected by the installation is displayed.

**Note:** The affected channel summary contains the following lists.

- **Fixed.** Channels that previously failed to validate and became valid after the installation.
  - **Broken.** Channels that previously validated and became invalid after the installation.
  - **Unknown.** Channels that PCR cannot verify as valid or invalid for the report. Usually, this is a byproduct of older channels with trigger parameters that are not stored in the database as plain text.
4. Perform one of the following actions.

1. Click **Accept** to accept the installation.
2. Click **Roll back** to undo the installation.

**Note:** You must accept or roll back the installation before PCR can process the next item. The bundle installation summary contains the following categories.

- **Installed.** Bundles installed from the file or archive and uploaded by the user.
- **Failed.** Bundles that failed to install. This includes a message with the reason for the failure.
- **Skipped.** Files that PCR cannot install, such as non-JAR files.



## Upgrade the connector manually

To upgrade from a previous version of Perceptive Capture Connector, complete the following steps. You can also upgrade the connector by clicking "Manual upload" in the "Upload new bundles" page.

1. Uninstall the existing version of the connector
2. Install the latest version of the connector

### Uninstall the existing version of the connector

3. To uninstall the existing version of the connector, complete the following steps.

1. In the left pane, under **TROUBLESHOOT**, click **List OSGi Bundles**.
2. In the **OSGi Bundles** page, click **Stop**  next to the following bundles.
  - Perceptive Capture Connector Content Mapping
  - Perceptive Capture Connector Endpoint
  - Perceptive Capture Connector Writer
3. Click **Uninstall**  next to each of the stopped bundles.


### Install the latest version of the connector

4. To install the latest version of Perceptive Capture Connector, complete the following steps.
  1. Locate the temporary directory you created in the "Download and extract the files" section.
  2. In the left pane, under **TROUBLESHOOT**, click **List OSGi Bundles**.
  3. In the **OSGi Bundles** page, click **Install/Update** and complete the following substeps.
    1. In the **Upload/Install Bundles** dialog box, click **Choose Files**.
    2. Navigate to the **capture-connector.content.mapping-5.0.0.jar** and **captureconnector.endpoint-5.0.0.jar** files in the extracted directory, click **Open**, select the **Start Bundle** check box, and then click **Install or Update**.
    3. Restart Perceptive Connect Runtime service.



## Configure the connector

To configure the Perceptive Capture Connector, complete the following steps.

1. In the **Connect Runtime Web Console**, in the left pane, under **Manage**, click **Configure**.
2. In the **Name** column, under **Perceptive Capture Connector**, click **Edit the configuration values** .
3. Select the **Secure API** checkbox to enable the API security.
4. In the **Client Hostname** box, type the hostname or IP address of the system where Brainware for Invoices(BFI) is installed.
5. In the **Channel Timeout** box, type the maximum time in seconds, for which the client will wait for the channels to finish execution.
6. In the **Supported Mime Types** box, enter a semicolon separated list of supported Mime Types for use with Capture Document Action.
7. Click **Save**.

## PCC File Conversion Configuration

To configure the Perceptive Capture Connector File Conversion Configuration, complete the following steps.

1. In the **File Image Service Base Address** box, enter the valid base address of file image service.
2. In the **Conversion API URI** box, enter the conversion API URI part of the file image service URL.
3. In the **Connection Manager Time To Live** box, enter the time to live value for Connection Manager Pool.
4. In the **Connection Manager Default Max Per Route** box, enter the maximum value per route for Connection Manager Pool.
5. In the **Connection Manager Max Total** box, enter the time to live value for Connection Manager Pool.
6. In the **Channel Timeout** box, enter the value for connection time out.

## Configure the connector for Brainware for Invoices

Perceptive Content workflow with Envoy services enables Perceptive Content to communicate with Perceptive Connect Runtime (PCR), and the file export directory. Brainware for Invoices collects the document from the file export directory. The Integration automated system queue you implement in your workflow process sends a web service notification using the Envoy service operation name you define. Based on the format of the document to be processed, PCR receives the document from Perceptive Content. After this process is complete, PCR sends a success response using Perceptive Integration Server. Perceptive Content can route the file forward in the Perceptive Content workflow for additional processing.

To configure Perceptive Content to use the connector, complete the following procedures.

1. Configure Brainware for Invoices for Content integration

2. Configure the Content Connect Service
3. Create the workflow process
4. Create the queues

## Configure Brainware for Invoices for content integration

You must configure Brainware for Invoices for content integration by updating the Brainware for Invoices database settings.

To update the database settings, complete the following step.

- In **Solution Configuration Manager**, go to **Project Settings** and navigate to the **Processing Settings** tree hierarchy to locate **Amount Settings** node.
- Set `Export Thousand Separator` to the value `""` and `Export Decimal Separator` to the value `"."`.
- Save the settings.

## Configure the Content Connect Service

To configure the Perceptive Content Connector service, complete the following steps.

1. Navigate to the location where the Content Server is installed and enter the `{install location}\etc` directory.
2. Open the `inserverWorkflow.ini` file and make the following changes in configuration settings.
  - Configure the `connect.uri` setting and set it to your Connect Runtime instance with the format `http://{Connect Runtime host name}:{port}/rs/workflowTrigger`.
  - Configure the `connect.timeout` to your desired expiration time.
3. Save the file.
4. Optional. The setting is automatically loaded after a short period. To reload the configuration immediately, restart the Content Server service.

## Create the workflow process

To create a workflow process, complete the following steps.

1. In **Perceptive Content Management Console**, in the left pane, click **Workflow**.
2. In the right pane, on the **Workflow** tab, click **New**.
3. In the **Add Process** dialog box, complete the following substeps.
  1. In the **Name** box, type a name.
  2. In the **Description** box, type the description of the process that appears in the tooltip for the process.
4. Click **OK**, select the process, and then click **Modify**. Use **Perceptive Content Workflow Designer** to create queues, routes, and workflow rules.

## Create the queues

Based on how you want to process your invoices, create the connect ASQ, and the work queues. The workflow you create using the connect ASQ, and work queues ascertains how a document is processed. The ASQ integrates the workflow process with Perceptive Connect Runtime and uses the subsequent work queues to indicate the processing status of the document.

For information about creating a workflow process and queues, refer to the following help topics.

- Create a Connect ASQ
- Create a work queue

**Note:** For a sample workflow, refer to the Example of Perceptive Content workflow for Brainware section.

## Create Brainware for Invoices File Output channels

File Output channel exports copies of documents from Perceptive Content to a local or network directory, triggered by a Connect ASQ. Brainware for Invoices is configured to monitor the directory and pull incoming TIFF files, and then performs classification, extraction, and validation on each image.

This section provides the steps for creating and configuring a File Output channel by specifying the ASQ used for the trigger, the destination directory, the separator character, and the file name parameters. The file name always includes the Content document ID, and can include a separator character, such as ^. The file name can also include up to 99 value segments, as needed, using any of the following options.

- A literal string
- A Perceptive Content document property, which can include the document name and document version number
- A Perceptive Content document key value
- A Perceptive Content custom property value

You can create multiple File Output channels with differing value segments or destination directories as needed for your solution. Before you begin configuring a channel, we recommend that you open the *Perceptive Connect Runtime Installation Online Help* to use as reference material.

### Channel requirements

A File Output channel requires the following components.

- A Connect ASQ ID from Perceptive Content
- The Integration ASQ trigger and the Document Property reader provided by the Perceptive Content Connector
- The File System Document Export action provided by this connector
- The Trigger reader provided by the Perceptive Connect Runtime

### Create and configure a File Output channel

To create and configure a File Output channel, complete the following steps.

1. Navigate to **Sample Mappings** folder > **Brainware for Invoices** folder > **File System Document Export** folder.

2. In the **File System Document Export.xml** file which is available inside the **File System Document Export** folder, replace the `Workflow Queue ID` with the `Submit Queue ID` from the Perceptive Content Workflow Designer.
3. Configure the `DOCUMENT_ID`, `ExportDirectory`, `Separator`, and extra file name segment parameters, as needed. For details, refer to information available in the File System Document Export action section.
4. In a browser, enter the channel wizard URL in the format `http://<Perceptive Connect Runtime machine>:<port>`.
5. In the left pane, under **Manage**, click **Import Channels**.
6. Select one or more channel configuration files in the **File System Document Export** folder under **Sample Mappings**.
7. On the **Upload New Channels** page, drag and drop the **File System Document Export.xml** file in the area designated by the box.

**Note** All files processed are displayed at the bottom of the page in a list view. When a file is expanded from the files list view, the following information is available. A list of all the files processed as per the file that was imported. For details, refer to View Report Details section.

8. Click the expand icon above the report to view the details of the entire report.

## Create Brainware for Invoices Status Update channels

Brainware for Invoices Status Update channels route documents in Perceptive Content workflow as they pass through different stages of Brainware processing.

These channels use the Brainware for Invoices project name, client ID, and the status code to route documents to specified queues. For example, you can create a channel that routes documents that failed the extraction process, such as status code 250, to a queue called IC Processing Error. Your users can then review documents in the queue and resubmit them for processing.

You can create one channel per combination of project name, client ID, and status code.

You cannot replace or modify an enabled channel. For assistance with modifying an Brainware Status Update channel, contact your Hyland representative.

**Note:** Before you begin configuring a channel, we recommend that you open the *Perceptive Connect Runtime Installation Online Help* to use as reference material. For more information on client ID, refer to Brainware for Invoices documentation.

### Channel requirements

A BFI Status Update channel requires the following components.

- A BFI project name, client ID, and status code
- Two Perceptive Content queue names
- The Status Update trigger, provided by this connector
- The `RoutelImageNowWorkflowItem` action, provided by the Perceptive Content Connector

## Configure and Import a BFI Status Update channel

To create and configure a Status Update channel, complete the following steps.

1. Navigate to **Sample Mappings** folder > **Brainware for Invoices** folder > **Status Update** folder.
2. Edit the values of the trigger parameters in these files for one or more channels:
  1. Replace the value of the Project Name parameter with the BFI project that processes the documents you want to route, such as `Invoices`. Do not include the file extension.
3. **Note** The BFI project name is set in the `ProjectName` column of the **BRWGRL** table of Brainware for Invoices configuration database. In **Solution Configuration Manager**, this setting is available under **General Settings** under the **Global Settings** category.
2. Replace the value of the Project Name parameter with the BFI client ID. Client IDs are non-negative integers.
3. Replace the value of the Status Code parameter with the BFI status code that you want this channel to handle, such as 550. Status codes are numbers between 0-999.
3. **Optional.** Edit the **SuccessQueueName** and **FailureQueueName** values to match the appropriate Perceptive Content workflow queues in the input mapping section if required.
4. In a browser, enter the channel wizard URL in the format `http://<Perceptive Connect Runtime machine>:<port>`.
5. In the left pane, under **Manage**, click **Import Channels**.
6. Select one or more channel configuration files in the **Status Update** folder under **Sample Mappings**.
7. On the **Upload New Channels** page, drag and drop the **XML** files available in the **Status Update** folder in the area designated by the box.

**Note** All files processed are displayed at the bottom of the page in a list view. When a file is expanded from the files list view, the following information is available. A list of all the files processed as per the file that was imported. For details, refer to View Report Details section.

8. Click the expand icon above the report to view the details of the entire report.

**Note** You can further create status update channels for any other status codes following the template provided in any of these sample configuration files.

## Create Brainware for Invoices Export channels

Brainware for Invoices Export channels pass XML data, extracted by Brainware, back to your original documents in Perceptive Content. With a Brainware for Invoices Export channel, you can write this XML data to document keys, custom properties, eForm fields, or the DCExport subobject. The channel uses the XML transformer, a specialized reader, to migrate the extracted data to an XML template or schema.

These channels use the Brainware for Invoices project name, client ID, and document class to identify which data to export from

Brainware for Invoices to Perceptive Content. For example, you can create a channel that uses extracted information from the image to populate an eForm or update indexes in Perceptive Content.

**Note:** You cannot replace an enabled channel. For assistance with modifying an Brainware Export channel, contact your Hyland representative. For more information on the client ID, refer to Brainware for Invoices documentation.

## Channel requirements

A Brainware for Invoices(BFI) Export channel requires the following components.

- A BFI project name, client ID, and document class
- The Document trigger, DCExport Subobject writer, and Document Property writer provided by this connector
- The XML reader and XML transformer, provided by the Perceptive Connect Runtime
- The RouteImageNowWorkflowItem action, eForm Data Definition reader, and eForm writer provided by the Perceptive Content Connector

## Configure and Import a Brainware for Invoices(BFI) Export channel

This section provides a high-level overview of how to create BFI Export channels. When you create a BFI Export channel, you must configure the input and output XML data to map the extracted Brainware data to your specific data definition or schema. This process is highly variable depending on your target. As you read this overview, we recommend that you refer to the detailed appendices for the required components.

To configure and import a BFI Export channel, complete the following steps.

1. Navigate to **Sample Mappings** folder > **Brainware for Invoices** folder > **Export** folder.
2. Edit the values of the trigger parameters in the **Export – Invoices.xml** file:
  1. Replace the value of the Project Name parameter with the BFI project that processes the documents you want to route, such as `Invoices`. Do not include the file extension.

**Note** The BFI project name is set in the `ProjectName` column of the **BRWGRL** table of Brainware for Invoices configuration database. In **Solution Configuration Manager**, this setting is available under **General Settings** under the **Global Settings** category.

  2. Replace the value of the Project Name parameter with the BFI client ID. Client IDs are non-negative integers.
3. In the **Inputs** section perform the following modifications if required:
  1. **Optional.** Edit the **SuccessQueueName** and **FailureQueueName** values to match the appropriate Perceptive Content workflow queues in the input mapping section if required.
  2. Ensure that all pre-configured XPath paths referring to the BFI XML are correct, in the input mapping section.
  3. Map any additional XML fields like Invoice Number or Invoice Amount, if required. These will be used in the output mapping section.
  4. Configure the eForm Data Definition reader, `eFormDataDefinitionSource`, to extract the data definition from your Perceptive Content eForm. The provided sample XML is pre-configured to write to AP Invoice eForm.
  5. Configure the XML transformer to map the extracted BFI values to your form's data definition. This maps the additional values extracted from the BFI XML to the data definition extracted from your form. The provided sample XML is pre-configured to map values to the AP Invoice eForm data definition.
4. In the **Outputs** section perform the following modifications if required:

1. Configure the eForm writer to write the transformed XML values into your eForm fields. The writer references the name of the XML transformer. The provided sample XML is pre-configured to write the transformed XML in the inputs section to the document's corresponding AP Invoice eForm in Perceptive Content.
  2. Configure the Document Property writer to write the previously read values from the BFI XML to the document's keys and custom properties in Perceptive Content. The provided sample XML is pre-configured with some common Invoice headers. You may configure more as per your requirement.
  3. **Optional.** Configure the DCEExport Subobject writer to write the BFI XML, `ic_xml`, to Perceptive Content's DCEExport Subobject.
  5. In a browser, enter the channel wizard URL in the format `http://<Perceptive Connect Runtime machine>:<port>`.
  6. In the left pane, under **Manage**, click **Import Channels**.
  7. Select one or more channel configuration files in the **Export** folder under **Sample Mappings**.
  8. On the **Upload New Channels** page, drag and drop the **Export - Invoices.xml** file in the area designated by the box.
- Note** All files processed are displayed at the bottom of the page in a list view. When a file is expanded from the files list view, the following information is available. A list of all the files processed as per the file that was imported. For details, refer to View Report Details section.
9. Click the expand icon above the report to view the details of the entire report.
  10. In the similar way, as mentioned above, follow the steps with the **Export – Void.xml** file to import the corresponding channel.

## View Report Details

During importing channel, on the **Upload New Channels** page, as you drag and drop the relevant XML files (as discussed above) in the area designated by the box, at the bottom of the page in a list view all files processed are displayed. When a file is expanded from the files list view, the following information is available. A list of all the files processed as per the file that was imported. You can select each file by clicking on the file name. When a file is selected, a report appears on the right.

To view the report details, perform the following task.

- Click the expand icon above the report to view the details of the entire report.

A report may contain the following information based on results from the import.

- Channel information
- ID
- Name
- Action
- General errors
- Trigger errors
- Input errors
- Output errors

## Triggers

A trigger is an event that causes a channel to execute. Each channel has a single trigger, so the event defined by the trigger and its inputs is the only entry point into a given channel. Triggers can also bring data into the channel. This data is available for input mapping for the channel action or for the results output. For additional information about triggers, refer to the “Appendix A: Triggers” section of the Perceptive Content Connector Installation Guide.

All of the triggers for the Perceptive Capture Connector use REST calls. Each trigger includes output values that you can use in the channel data context for either input or output mappings.

The Perceptive Capture Connector provides the following triggers and associated data.

- Document trigger
- Status Update trigger

### Document trigger

Document Trigger allows you to create channels for creating and soft deleting documents in Perceptive Content. You can also use this trigger to create channels to pass XML content back to documents in Perceptive Content.

This trigger includes the following input identifiers.

- Project Name. The BFI Project Name.
- Client Id. The client ID configured in BFI.
- Document Class. The document class configured in Brainware for Invoices. If no document class is configured in BFI, configure the document class with “GLOBAL”.
- Mode. The three possible modes are CREATE, EXPORT and DELETE

In CREATE mode the following data is made available to the channel by the trigger.

- A standard document image with the key STANDARD\_IMAGE
- An optional original image with the key ORIGINAL\_IMAGE
- A metadata XML with the key XML\_DOC

In EXPORT mode only the Perceptive Content document ID with the key DocumentID is made available to the channel by the trigger, in addition to the data provided by the CREATE mode. This data can be read using the Trigger Reader provided by Perceptive Connect Runtime.

In DELETE mode only the Perceptive Content document ID with the key DocumentID is made available to the channel by the trigger.

### Status Update trigger

The Status Update trigger allows you to create Status Update channels for BFI.

This trigger includes the following input identifiers.

- ProjectName. The BFI Project Name.
- ClientId. The client ID configured in BFI.
- StatusCode. The BFI status code which is a non-negative integer between 0-999.



## Use trigger values in a channel input or output mapping

To use a Status Update trigger value in the input mapping, use the Perceptive Content Connector Trigger reader. In this instance, the trigger node contains the output value from the selected trigger and the trigger name is any string that you use as a reference in another node.

### Example

```
<c:parameter>
  <c:name>DocumentId</c:name>
  <c:trigger>DOCUMNET_ID</c:trigger>
</c:parameter>
```

To use a trigger value in the output mapping, such as for the DC Export Sub-object, reference the value directly.

### Example

```
<c:parameter>
  <c:name>ICXmlDoc</c:name>
  <in:DCEExportTarget>DocumentId</in:DCEExportTarget>
</c:parameter>
```

## Actions

An action is a connector-defined task configured in the channel that executes when the channel triggers. You can configure the various inputs that the action requires when you create the channel. The action uses these inputs to complete its task in an application outside of Perceptive Connect Runtime, as defined in the connector. For additional information about actions, refer to the “Appendix B: Actions” section of the Perceptive Content Connector Installation Guide.

The Perceptive Capture Connector provides the following actions.

- Capture Document action
- File System Document Export action

### Capture Document action

The Capture Document Action allows you to create a document in Perceptive Content using the configured Capture Profiles and Application Plans. After receiving an image and metadata for document keys and custom properties, the Capture Profile name, and other configuration parameters, this action creates a document with all the information in Perceptive Content and routes it to the workflow queue configured in the Capture Profile.

When you select the Capture Document action, Perceptive Connect Runtime automatically populates the following XML configuration template in the Inputs box.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<c:inputs xmlns:in="http://www.perceptivesoftware.com/pif/imagenow"
xmlns:c="http://www.perceptivesoftware.com/pif/mapping">
  <c:parameter>
    <c:name>StandardImage</c:name>
    <c:none/>
  </c:parameter>
  <c:parameter>
    <c:name>OriginalImage</c:name>
    <c:none/>
  </c:parameter>
</c:inputs>
```

```

</c:parameter>
<c:parameter>
  <c:name>CaptureProfileData</c:name>
  <c:none/>
</c:parameter>
<c:parameter>
  <c:name>CaptureProfile</c:name>
  <c:none/>
</c:parameter>
</c:inputs>

```

## File System Document Export action

This action exports any Perceptive Content documents from an Integration ASQ to a specific network location, such as the Brainware import directory. With the File System Document Export action, you can perform the following actions.

- File name format configuration. You can include Perceptive Content values or literal values in the file naming convention and pass it to Brainware.
- File splitting. You can split a multi-page document into smaller documents and export these as separate files to the file system.
- File merging. You can merge multiple documents into a single document.
- TIFF conversion. If you have multiple documents with variable format, you can convert all the documents into TIFF format before exporting the documents.

When you select the File System Document Export action, Perceptive Connect Runtime automatically populates the following XML configuration template in the Inputs box.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<c:inputs xmlns:ns2="http://www.perceptivesoftware.com/pie/mapping/sample"
xmlns:c="http://www.perceptivesoftware.com/pif/mapping">
  <c:parameter>
    <c:name>WfId</c:name>
    <c:trigger>WorkflowItemId</c:trigger>
  </c:parameter>
  <c:parameter>
    <c:name>DOCUMENT_ID</c:name>
    <in:workflowItem
xmlns:in="http://www.perceptivesoftware.com/pif/imagenow">
      <in:reference>WfId</in:reference>
      <in:objectType>WORKFLOW</in:objectType>
      <in:objectField>OBJECT_ID</in:objectField>
    </in:workflowItem>
  </c:parameter>
  <c:parameter>
    <c:name>ExportDirectory</c:name>
    <c:literal>E:\Temp</c:literal>
  </c:parameter>
  <c:parameter>
    <c:name>Separator</c:name>
    <c:none/>
  </c:parameter>
  <c:parameter>
    <c:name>EnableSubDirectory</c:name>

```

```

    <c:literal>>true</c:literal>
  </c:parameter>
  <c:parameter>
    <c:name>SinglePageDocFlag</c:name>
    <c:literal>>true</c:literal>
  </c:parameter>
  <c:parameter>
    <c:name>Convert</c:name>
    <c:none/>
  </c:parameter>
  <c:parameter>
    <c:name>Segment1</c:name>
    <c:literal>S1</c:literal>
  </c:parameter>

```

The action requires that you specify the `DOCUMENT_ID` and `ExportDirectory` parameters. `DOCUMENT_ID` retrieves the ID of the Perceptive Content document that the action is exporting. The ID is always the first element of the file name. The Integration ASQ Trigger provides the Doc ID. However, if you use this action with another trigger that does not provide the Doc ID, the Workflow Item reader can use the workflow ID to retrieve the Doc ID.

## Parameter Descriptions

`Convert` allows you to specify whether to process a document in the original format or TIFF format. To process a document as TIFF, add `TRUE` to the literal reader or add `FALSE` to keep the existing format. However, if a Perceptive Content document has both, PDF and TIFF files, and the value of the `SinglePageDocFlag` parameter is set to `FALSE`, PCC Connector converts the PDF files to TIFF irrespective of whether you specify `TRUE` or `FALSE` value to this parameter.

`EnableSubDirectory` enables you to configure the connector to create a subdirectory in the output directory for the processed Perceptive Content document. To create a subdirectory in the export directory add `TRUE` to the literal reader.

`SinglePageDocFlag` enables you to configure the connector to create individual files for each page that makes up a Perceptive Content document. To create individual files for each page of a file add `TRUE` to the literal reader. For example, if one TIFF file and one PDF file with ten pages each are associated with a Perceptive Content document, ten PDF files and ten TIFF files are created separately in the export directory if you set the literal reader for `SinglePageDocFlag` as `TRUE`. The files created inherit their native file type.

`ExportDirectory` sets the output location on the file system. If the export directory is on a network location, the Perceptive Connect Runtime user must specify the network path of the export directory under the `ExportDirectory` parameter. To specify the export directory, replace `<c:none/>` with an empty set of literal reader from Perceptive Connect Runtime, and provide the network path value. The following example describes the `ExportDirectory` parameter.

```

<c:parameter>
  <c:name>ExportDirectory</c:name>
  <c:literal>//192.168.45.101/Shared/Export</c:literal>
</c:parameter>

```

**Note:** The Perceptive Connect Runtime user must have write access to the export directory.

`Separator` sets the separator character for the file name. If you do not specify a character, the default is `^`. Refer to the following list for acceptable separator characters.

```
~ ! @ # $ % ^ & ( ) - _ + [ ] { } ; ' , .
```

With the `Segment` parameters, you can configure up to 99 additional segments in the file name. Keep in mind your system file name character limits.

To insert a Perceptive Content document key value, custom property value, or Perceptive Content document name, in the `Segment` parameters, replace `<none\>` with the element `<objectPropertyReader>`. The element `<objectPropertyReader>` is described below.

```
<in:objectPropertyReader>
  <in:name>Field4</in:name>
  <in:objectIdRef>DOCUMENT_ID</in:objectIdRef>
  <in:objectType>document</in:objectType>
  <in:propertyType>KEY</in:propertyType>
</in:objectPropertyReader>
```

Modify `<objectPropertyReader>` using any of the following options.

- To insert a literal string, use the literal reader from Perceptive Connect Runtime. For example, insert a company name or business unit. To insert an empty segment, use the literal reader from Perceptive Connect Runtime with an empty set.

```
<c:literal></c:literal>
```

**Note:** You cannot use the characters `< > : " / \ | ? *` in the `Segment` values or as the separator.

- To insert a Perceptive Content document key value, in the element `<in:propertyType>`, type `KEY`, and in the element `<in:name>`, specify a key value. Allowed values are `field1`, `field2`, `field3`, `field4`, `field5`, `folder`, `tab`, `drawer`, and `type`.
- To insert a Perceptive Content custom property, in the element `<in:propertyType>`, type `CUSTOMPROPERTY`, and in the element `<in:name>`, specify the name of the custom property.
- To insert the Perceptive Content document name, in the element `<in:propertyType>`, type `DOCPROPERTY`, and in the element `<in:name>`, specify the document name.

```
<c:parameter>
  <c:name>DOCUMENT_ID</c:name>
  <in:workflowItem>
    <in:reference>WfId</in:reference>
    <in:objectType>WORKFLOW</in:objectType>
    <in:objectField>OBJECT_ID</in:objectField>
  </in:workflowItem>
</c:parameter>
<c:parameter>
  <c:name>WfId</c:name>
  <c:trigger>WorkflowItemId</c:trigger>
</c:parameter>
<c:parameter>
  <c:name>ExportDirectory</c:name>
  <c:literal>C:\Program Files\Perceptive\Projects\AP\Export</c:literal>
</c:parameter>
<c:parameter>
  <c:name>Separator</c:name>
  <c:literal>#</c:literal>
</c:parameter>
<c:parameter>
  <c:name>Segment1</c:name>
  <c:literal>ID</c:literal>
</c:parameter>
<c:parameter>
  <c:name>Segment2</c:name>
  <in:objectPropertyReader>
```

```

<in:name>Field4</in:name>
<in:objectIdRef>DOCUMENT_ID</in:objectIdRef>
<in:objectType>document</in:objectType>
<in:propertyType>KEY</in:propertyType>
</in:objectPropertyReader>
</c:parameter>

```

**Note:** The system user must have write access permission to the ExportDirectory location you mention.

## Writers

Writers are interfaces, provided by connectors, which let you configure channels to output data to applications outside of Perceptive Connect Runtime. You use writers to configure the results output mapping, which lets you specify what to do with any data resulting from the action execution. You can invoke writers using specific XML tags, defined per writer. For additional information about writers, refer to the “Appendix D:Writers” section of the *Perceptive Content Connector Installation Guide*.

The Perceptive Capture Connector provides the following writers.

- DCExport Subobject writer
- Document Property writer
- Image writer

### DCExport Subobject writer

This writer lets you save an XML document (w3c.Document Java type) to the DCExport subobject of a document in Perceptive Content. Perceptive Content uses the `workingName` parameter to identify objects attached to the DCExport subobject. The `workingName` value is the root node from the XML document to be stored, such as `BrainwareDocument`. If an object with the same `workingName` already exists, the writer replaces the data with the latest XML document.

To output an XML document from a channel to a Perceptive Content document, include the following XML template in the channel output mapping, and enter the appropriate values.

```

<c:name></name>
<in:DCExportTarget></in:DCExportTarget>

```

The `name` field references the name node of the Perceptive Content Connector Trigger reader. Refer to the “Trigger” section in the *Perceptive Connect Runtime Installation and Setup Guide*.

In the `DCExportTarget` field, reference the Perceptive Content document ID to which you want to attach the XML. For example, if you used the Status Update trigger from this connector, you can use the trigger’s output value `DocumentId`. If you used the Integration ASQ Trigger, from the Perceptive Content Connector, you would reference the Workflow Item reader name, such as `DOCUMENT_ID`.

DCExport subobjects only exist at the document level. You cannot write this subobject to a Perceptive Content folder, or to an individual page within a Perceptive Content document.

#### Example

```

<c:parameter>
  <c:name>ICXmlDoc</c:name>
  <in:DCExportTarget>DocumentId</in:DCExportTarget>
</c:parameter>

```

## Document Property writer

The Document Property writer enables you to output a value to a document property in Perceptive Content.

To write a value from the channel to a document property, include the following XML template in the channel output mapping, with the appropriate values.

```
<in:documentPropertyWriter>
  <in:keyMappings>
    <in:keyMapping>
      <in:source></in:source>
      <in:target></in:target>
    </in:keyMapping>
  </in:keyMappings>
  <in:customPropertyMappings>
    <in:customPropertyMapping>
      <in:source></in:source>
      <in:target></in:target>
    </in:customPropertyMapping>
  </in:customPropertyMappings>
  <in:errorCustomProperty></in:errorCustomProperty>
  <in:ignoreMissingProperty></in:ignoreMissingProperty>
</in:documentPropertyWriter>
```

When configured, the `keyMappings` section must have minimum one and maximum seven unique `keyMapping` values. The `source` field contains a reference to the value to be stored against the property.

The `target` field contains the key name and is restricted to `FIELD1`, `FIELD2`, `FIELD3`, `FIELD4`, `FIELD5`, `TYPE` and `DRAWER`.

When configured, the `customPropertyMappings` section can have one or more `customPropertyMapping` values. The `source` field contains a reference to the value to be stored against the property. The `target` field contains the name of the custom property to be populated.

The `errorCustomProperty` section contains the name of a custom property to which you can write error messages when an error occurs while updating document keys and custom properties. Alternatively you can leave the field blank if you don't want to write error messages.

The `ignoreMissingProperty` section contains a `TRUE` or `FALSE` value. When this value is set to `TRUE`, the writer logs a warning message that the custom property could not be found and continues processing any unprocessed custom properties. When this value is set to `FALSE`, the writer shows an error that the property could not be found and any unprocessed custom properties are not processed. When this property is not present in the configuration, it is set to `FALSE` by default.

## Custom Properties

Perceptive Content identifies the custom property type. The type need not be passed to the writer, but the writer internally translates values in the context to `String` for the `STRING`, `NUMBER`, `DATE` and `FLAG` types and the `String` should be in a specific format. The following custom property types are acceptable.

**STRING** This is stored in the custom property just as it exists in the context.

**NUMBER** The number is parsed. If the string is not a number, the writer will fail.

**DATE** The date must be in the format `MM/DD/YYYY`.

**FLAG** The following values will be parsed into the corresponding flag value. Any other value will cause the writer to fail.

- 0 False
- 1 True
- false False
- true True

The writer does not support writing to custom properties of types “USER\_GROUP”, “USER\_LIST”, “COMPOSITE” and “ARRAY”.

Examples:

In the below sample mapping, the writer updates both document keys and custom properties.

```
<c:parameter>
  <c:name>DOCUMENT_ID</c:name>
  <in:documentPropertyWriter>
    <in:keyMappings>
      <in:keyMapping>
        <in:source>InvoiceNumber</in:source>
        <in:target>FIELD1</in:target>
      </in:keyMapping>
      <in:keyMapping>
        <in:source>VendorName</in:source>
        <in:target>FIELD2</in:target>
      </in:keyMapping>
      <in:keyMapping>
        <in:source>PONumber</in:source>
        <in:target>FIELD3</in:target>
      </in:keyMapping>
    </in:keyMappings>
    <in:customPropertyMappings>
      <in:customPropertyMapping>
        <in:source>VendorNumber</in:source>
        <in:target>Vendor ID</in:target>
      </in:customPropertyMapping>
      <in:customPropertyMapping>
        <in:source>VendorName</in:source>
        <in:target>Vendor Name</in:target>
      </in:customPropertyMapping>
      <in:customPropertyMapping>
        <in:source>InvoiceDate</in:source>
        <in:target>Invoice Date</in:target>
      </in:customPropertyMapping>
    </in:customPropertyMappings>
    <in:errorCustomProperty>Error</in:errorCustomProperty>
    <in:ignoreMissingProperty>true</in:ignoreMissingProperty>
  </in:documentPropertyWriter>
</c:parameter>
```

In the below sample mapping, the writer is configured to update only document custom properties.

```
<c:parameter>
  <c:name>DOCUMENT_ID</c:name>
  <in:documentPropertyWriter>
    <in:customPropertyMappings>
      <in:customPropertyMapping>
        <in:source>VendorNumber</in:source>
        <in:target>Vendor ID</in:target>
      </in:customPropertyMapping>
    </in:customPropertyMappings>
  </in:documentPropertyWriter>
</c:parameter>
```

```

        <in:customPropertyMapping>
            <in:source>VendorName</in:source>
            <in:target>Vendor Name</in:target>
        </in:customPropertyMapping>
    </in:customPropertyMapping>
    <in:customPropertyMapping>
        <in:source>InvoiceDate</in:source>
        <in:target>Invoice Date</in:target>
    </in:customPropertyMapping>
</in:customPropertyMappings>
<in:errorCustomProperty>Error</in:errorCustomProperty>
<in:ignoreMissingProperty>true</in:ignoreMissingProperty>
</in:documentPropertyWriter>
</c:parameter>

```

In the below sample mapping, the writer is configured to update only document keys.

```

<c:parameter>
    <c:name>DOCUMENT_ID</c:name>
    <in:documentPropertyWriter>
        <in:keyMappings>
            <in:keyMapping>
                <in:source>InvoiceNumber</in:source>
                <in:target>FIELD1</in:target>
            </in:keyMapping>
            <in:keyMapping>
                <in:source>VendorName</in:source>
                <in:target>FIELD2</in:target>
            </in:keyMapping>
            <in:keyMapping>
                <in:source>PONumber</in:source>
                <in:target>FIELD3</in:target>
            </in:keyMapping>
        </in:keyMappings>
        <in:errorCustomProperty>Error</in:errorCustomProperty>
        <in:ignoreMissingProperty>true</in:ignoreMissingProperty>
    </in:documentPropertyWriter>
</c:parameter>

```

## Image writer

The Image writer enables you to split a multi-page document, convert it to TIFF, and replace it with the existing pages in a Perceptive Content document.

You can either provide a new image as input on which the operations are performed or the image can be fetched from Perceptive Content based on an input Document ID. You can also add an 'original' file or image to the document as its last page. You cannot perform any split or conversion operations on this 'original' image or file.

To perform the above operations add the following XML template to the output mapping of a channel.

```

<in:imageWriter>
    <in:replaceWith></in:replaceWith>
    <in:convertToTiff></in:convertToTiff>
    <in:splitMultiPageTiff></in:splitMultiPageTiff>
    <in:original></in:original>
</in:imageWriter>

```

The `replaceWith` field contains a reference to the image on which you perform the split and conversion operations. If you leave this field blank, the image is fetched from Perceptive Content and all operations are performed on this fetched image.



The `convertToTiff` field contains a TRUE or FALSE value. Setting this value to TRUE converts any image file format to TIFF.

The `splitMultiPageTiff` field contains a TRUE or FALSE value. Setting this value to true splits a multipage file into individual files, each consisting of a single page. The resulting files are TIFF files, irrespective of whether `convertToTiff` is set to TRUE or FALSE.

The `original` field contains a reference to an image or a file that you would like to add as the last page of a document.

#### Example

```
<c:parameter>
  <c:name>DOCUMENT_ID</c:name>
  <in:imageWriter>
    <in:replaceWith>StandardImage</in:replaceWith>
    <in:convertToTiff>true</in:convertToTiff>
    <in:splitMultiPageTiff>true</in:splitMultiPageTiff>
    <in:original>OriginalImage</in:original>
  </in:imageWriter>
</c:parameter>
```

Here, the writer splits `StandardImage` (a multi-page image file in any format) into multiple TIFF files each consisting of a single page. The existing pages in the Perceptive Content document get replaced with these new files. The `OriginalImage` file is added as the last page of the document.

The `StandardImage` and `OriginalImage` files are usually made available in the Perceptive Capture Connector channel by Document Trigger and can be read using Perceptive Capture Connector Trigger Reader.

```
<c:parameter>
  <c:name>DOCUMENT_ID</c:name>
  <in:imageWriter>
    <in:replaceWith></in:replaceWith>
    <in:convertToTiff>true</in:convertToTiff>
    <in:splitMultiPageTiff>true</in:splitMultiPageTiff>
    <in:original></in:original>
  </in:imageWriter>
</c:parameter>
```

In this example, the writer fetches the images from the Content document, performs the operations on these images, and replaces the old pages in Perceptive Content with the new pages. This writer does not add an original image as the last page as the configuration is not set. When used in a late archiving scenario, in the event of duplicate indexes, this configuration retains all existing pages in Perceptive Content. However, the original image from previous conflicting create calls may be split or converted to TIFF based on the configuration. The original image from the last create call remains unaffected.

## Example of Perceptive Content workflow for Brainware for Invoices

The following image provides an example layout of the Perceptive Content queues provided for the Brainware for Invoices workflow.

