opentext™

OpenText™ Information Archive

Administration Guide

Set up storage, learn about logging and auditing, and perform administrative actions against ingested SIP and table data.

EARCORE250400-AGD-EN-01

OpenText™ Information Archive
Administration Guide
EARCORE250400-AGD-EN-01
Rev.: 2025-Sept-26

This documentation has been created for OpenText™ Information Archive CE 25.4.
It is also valid for subsequent software releases unless OpenText has made newer documentation available with the product, on an OpenText website, or by any other means.

Open Text Corporation

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111
Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440
Fax: +1-519-888-0677
Support: https://support.opentext.com
For more information, visit https://www.opentext.com

Table of Contents

Chapter 1

Introduction

Administering OpenText Information Archive consists of performing day-to-day administration and optimization tasks. This document discusses concepts related to the archiving process and the OAIS data model. For more information, see section 2.4 “The archiving process” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

This document also covers the following topics:

How to configure system-wide settings

The Configuring system-wide settings chapter includes detailed information about setting up storage for structured and unstructured data. For more information about these two data types, see section 1.1 “What can be archived?” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

How to use audits to track events related to the system, tenant, and the applications that contain archived data.

Along with the information in this guide, see the following for additional information about audits and how to use them:

- Section 7.5 “Auditing” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.
- Section 9.18 “Audits” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

Learn how to install the language pack

The language pack allows you to change the language of the IA Web App interface.

Day-to-day administration

A store is a storage configuration object that contains properties for linking a space with a File System Folder or Bucket. This guide delves into how to configure stores for an application. To learn more about applications, see Section 4 “Applications and data” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

How to manage SIP and table applications

To learn about the differences between SIP and table archives, see Section 2.3 “Table archiving and SIP archiving” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

How to use jobs to perform various tasks for compliance, cleanup, and upgrades

For more information, see Section 8.1 “Jobs” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

How to map groups and manage permissions

The system defines specific roles for users that will interact with it. A user's role is determined by membership in a group, determines the actions that the user can perform. For more information, see Section 7.3 “User roles and actions” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)* and Section 7.4 “Configuring groups to access OpenText Information Archive” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

How to use various dashboards

For more information, see the following:

- License dashboard
- Storage dashboard
- For more information about the Compliance dashboard, see Section 9.15 “Running and viewing metrics with the compliance dashboard” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

How to configure the level of logging

This guide describes the various log files used to monitor activity in your system.

Chapter 2

Configuring system-wide settings

This chapter describes how to create configuration objects for storage systems, stores, data nodes, and databases. Configuration objects (also known as configuration resources) allow you to bind a configuration in OpenText Information Archive with physical systems. For example, you can create a data node configuration object to establish a connection with an existing physical data node.

For more information about configuration objects, see Section 4.4 “How OpenText Information Archive is configured for data” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)* and Section 4.4.1 “Configuration objects” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

2.1 Setting up storage for structured data

Structured data is managed differently depending on the archive type of the application. SIP applications store structured data using a Lucene index in dedicated Lucene-based content stores backed by a filesystem. This is like how unstructured data is managed overall, except that Lucene content stores require a (Local) filesystem and unlike unstructured content cannot use cloud-based storage.

Table applications store structured data in PostgreSQL managed by means of a data node and database configuration.

The **Administration > Data Nodes** tab allows the user in an Administrator role to:

- Register a data node to act as a container for RDB databases. You can also edit an existing data node from this screen.
- Create or register an RDB database to hold the structured content of archived records for a table application.

For more information, see Section 3.1 “Configuring how data is stored in PostgreSQL” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

2.1.1 Data nodes

Existing data nodes are displayed in a table that contains the following information:

▸	Click the arrow beside a specific data node to expand to view the database(s) and applications associated with the selected data node.
Data Node Name	Indicates the name of the data node.
⊙	Click the action button to: • Edit an existing data node. • Create an RDB database using the selected data node. • Register an RDB data node.
Bootstrap	Indicates the JDBC URL used to connect to the database.
In Use	Indicates whether the data node is currently referenced by an RDB database. Even if a data node cannot be deleted, it can always be edited.

To register an RDB data node:

1. On the **Data Nodes** tab, click **+**. The Register Data Node page is displayed.
2. Enter the following information:

Data Node Name
Enter a unique name for the data node.

Username
Enter the username of the PostGreSQL account that will be used to connect.

Password
Enter a password for the data node (it will be checked).

Connection URL
Enter the JDBC connection string to the data node. For example:

```
jdbc:postgresql://10.1.2.3:5432/
```

SSL
Select if the data node has Secure Sockets Layer (SSL) enabled.

SSL Mode
If SSL is enabled, select the mode. While there are different settings, it is recommended that you use one of the following settings:

- `verify-full`: The server hostname is verified to ensure it matches the name stored in the server certificate.
- `verify-ca`: Verifies that the server is valid by checking the certificate chain up to the root certificate on the client.
- `require`: Makes the encryption mandatory and also requires the connection to fail if it cannot be encrypted.

Refer to PostgreSQL documentation for more information about the other available settings:

- `prefer`
- `allow`
- `disable`
- `default`

3. If SSL is required, enter the **Root certificate**. Enter the root SSL certificate for the data node. In PostgreSQL documentation, this value is referred to as `sslrootcert`, which is the filename of the SSL root certificate and represents the full file path which must be accessible for each IA Server.

   Complete the following fields, if applicable:

   Client certificate
   If you wish to use a client certificate for authentication, you can enter the client certificate instead of using the username and password. The path entered must be accessible for all IA Servers. In PostgreSQL documentation, this value is referred to as the `sslcert` and represents the full file path for the certificate file accessible by each IA Server.

   Client key
   Enter the client key information. In PostgreSQL documentation, this value is referred to as the `sslkey` and represents the full file path for the key file accessible by each IA Server.

   Key password
   Enter the key store password. The value is masked. This is optional but is recommended to be set. In PostgreSQL documentation, this value is referred to as `sslpassword`.

4. Optional Click **Test Connection** to ensure the connection between the data node and bootstrap works. The system will inform you if the connection could be made or not.

   Note: If you receive the message `No suitable driver found for <text>/postgres(08001)`, you likely forgot to add this prefix `jdbc:postgresql://` to the Connection URL.

5. Click **Register** to complete the set up.

Caution
You cannot register the same bootstrap for multiple data nodes.

A relational database (RDB) is a set of data sets organized by tables, records and columns. RDBs use a clear link between database tables. The method in which tables share information helps improve data searchability, organization and reporting.

An RDB database is a storage configuration object that represents a database in PostgreSQL. It contains a set of properties to access the physical database.

An RDB database must be registered with OpenText Information Archive before it can be used:

- If the DB is created by OpenText Information Archive, it is registered automatically.
- If the DB was created outside OpenText Information Archive, it must be registered. For more information, see Registering a database in a data node.

In the **Data Nodes** tab, click the arrow beside one of the existing data node names to view the following related database information:

Database Name
Indicates the name of the database.

Data Node Name
Indicates the name of the associated data node.

Bootstrap
Indicates the JDBC connection string.

Applications
The list of applications that use the database.

In Use
Indicates whether the database is currently being used by a space.

Note: Creating a database is only possible if the username associated with the data node has the permission in PostgreSQL.

2.1.2 Creating/Editing a database

To create a database:

1. On the **Data Nodes** tab, click the action button beside the applicable data node name and select **Create Database**. The **Create Database** page is displayed.

   Note: If the database owner does not have the permission to create users and create databases, you cannot use this procedure. You must create both the users and database in PostgreSQL and instead register the database. For more information, see Section 3.1 “Configuring how data is stored in PostgreSQL” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

2. Enter the following information:

   Database Name
   Enter a unique name for the database. The database name should not contain spaces.

Username
Enter the PostgresSQL account that will be the owner for this database. If the account does not exist, it will be created.

Password
Enter the password for the PostgresSQL account. If you decide to use a client certificate, both a username and password must be entered but will not be used.

Confirm Password
Reenter the password to confirm it is correct. This field is only displayed when a database is being created. It does not appear if you are registering or editing a database.

Client certificate
If you wish to use a client certificate for authentication, you can enter the client certificate instead of using the username and password. The path entered must be accessible for all IA Servers.

Client key
Enter the client key information.

Key password
Optional: Enter the key store password if the client key is password protected.

3. Click **Create**.

To edit a database:

1. On the **Data Nodes** tab, click the arrow beside the applicable **Data Node Name** to display a list of the associated databases.
2. Click the action button next to the **Database Name** of the database you want to edit.
3. Click **Edit** in the drop-down list.
4. In the **Edit Database** dialog, make the necessary changes following the field descriptions provided in the table above.

   Note: That if you change the username, it does not change the owner of the database in PostgreSQL. To do this, change in PostgreSQL first and then change owner of the database using this procedure.

5. Click **Save**.

2.1.3 Registering a database in a data node

This procedure is used for registering an RDB database that was created outside OpenText Information Archive.

Note: If you are registering an RDB database, that username must already exist and is expected to the owner of the database.

To register a database:

1. On the **Data Nodes** tab, click the action button beside the applicable data node name and select **Register Database**.

   The Register Database page is displayed.

2. Enter the following information:

   Database Name
   Enter a unique name for the database.

   Username
   Enter the username of PostgreSQL user who is the database owner.

   Password
   Enter the associated password for the username mentioned above.

   Client certificate
   Enter the client certificate to ensure the server is communicating with a legitimate user.

   Client key
   Enter the client key information.

   Key password
   Optional: Enter the key store password only if the client key is password protected.

3. Click **Register**.

   Clicking on **Register** will implicitly test the connection and will register the database only when the connection is successful.

2.2 Setting up storage for unstructured data

Storage binds OpenText Information Archive with physical storage (such as file systems, PowerScale, ECS, S3, *etc.*) and refers to a storage configuration object that contains a list of properties for target storage configuration.

Type	Storage Type	Test Connection
Object Storage	Pure Storage FlashBlade	Yes
	Microsoft Azure Blob Storage	Yes
	Amazon S3 Storage	Yes
	AWS S3 with Amazon Glacier	Yes
	Dell EMC Elastic Cloud Storage	Yes
	NetApp StorageGRID	Yes
	PowerScale OneFS S3	Yes
	Scality RING	Yes
Legacy	Dell EMC CAS (Content Addressed Storage) Elastic Cloud Storage	Yes
Custom	Custom storage systems	Yes
Archive Centre	OpenText Archive Center	Yes
	OpenText Core Archive	Yes
File Storage	Dell EMC PowerScale	No
	Local File System	No

A storage system holds data, such as unstructured content for records, backups, raw XML files, ingestion logs, and so on.

In cloud-based storage, buckets are the basic containers that hold data (for Azure, however, Blobs act as the basic storage containers). The rules governing everything from naming conventions to whether buckets/Blobs can be nested, vary from storage system to storage system. Access the following links to learn about the different rules for creating buckets for various types of storage:

- For information about ECS see the Dell Technologies website
- For information about Amazon S3 see the *Amazon Simple Storage Service user guide* on the Amazon AWS website
- For information about Azure see *Naming and Referencing Containers, Blobs, and Metadata* on the Microsoft website.
- For more information about GCS see the *Cloud Storage buckets guide* on the Google Cloud documentation website.

- For more information about NetApp storage see the NetApp my support website.

2.2.1 Adding a storage system using IA Web App

Existing storage systems are displayed in a table that contains the following information:

Storage Name	Indicates the name of the storage system.
	Allows you to complete one of the following actions to the selected storage system: • **Edit**: Make changes to the selected storage system. • **Delete**: Delete the selected storage system. • **Test Connection**: Once the **Test Connection** button is pressed, OpenText Information Archive tries to establish a connection with the storage system. The system will inform you if the connection could be made or not. See Setting up storage for unstructured data to learn which storage types support this feature. If the connection is not successfully established, the error message indicates the reason why the connection failed. Make the necessary changes to the fields indicated in the error message and click the **Test Connection** button again.
Storage Type	Indicates the type of storage. Refer to Setting up storage for unstructured data to review the list of accepted storage systems.
Properties/ Storage Details	The information displayed in these columns depends on the type of storage system being used.
In Use	Indicates whether the storage system is currently being used. If a storage system cannot be deleted, it can always be edited. All actions are available in context menu for the storage system.

To add a storage system using IA Web App:

1. In the IA Web App, on the **Storage** tab, click **+**. The **Create Storage System** page is displayed.
2. Select a **Storage Type**.
3. Proceed with the steps in one of the following sections, depending on the type of storage being used:
   - Configuring the storage class for Microsoft Azure Blob Storage
   - Configuring Amazon S3 storage
   - Configuring AWS S3 with Amazon Glacier for an application
   - Configuring ECS Storage for an application
   - Configuring Dell EMC CAS Elastic Cloud Storage for an application
   - Configuring custom storage for an application

- Archive Center
- Core Archive

4. Click the **Test Connection** button to ensure the connection works. See Setting up storage for unstructured data to learn which storage types support this feature.

   Note: When editing a connection, the sensitive fields need to be re-entered.

5. Click **Create**.

2.2.1.1 Providing hardware retention to objects on NetApp

This section discusses compliance in regards to NetApp. For more information about compliance operations in OpenText Information Archive, see Section 9 "Compliance – General concepts" in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

Retention indicates how long content should be kept for compliance and is associated to archived information by applying a retention policy against the data. Once retention is applied to any objects/packages in the IA Web App, retention is also applied at the NetApp (hardware) level. Consequently, you will not be able to delete those objects via the NetApp console.

If you choose to apply retention at the hardware-level, enable the Object Lock feature when creating a NetApp bucket (refer to NetApp documentation for further instructions).

Important
Ensure that the Object Lock feature is enabled, if desired, when you are creating the bucket, as it is not possible to enable the feature once the bucket has been created.

When applying retention to any object, provide a date set in the future. Once applied, it is impossible to delete the object until this retention period expires.

To enable the Object Lock feature, when creating a NetApp bucket in OpenText Information Archive, ensure the **Push retention at the bucket level** checkbox is selected. If this box is not selected, the Object Lock feature will not be enabled.

Once the bucket has been created, you still have the option to deactivate the Object Lock feature by selecting the **Do not push retention at the hardware level** checkbox. This scenario might occur if the Object Lock feature was enabled, but you do not want to apply retention to a particular store. If the **Do not push retention at the hardware level** checkbox is selected, if a retention policy is applied to a package or object, retention is only applied at the OpenText Information Archive level and not at the NetApp level.

If you opted not to enable the Object Lock feature, the **Do not push retention at the hardware level** checkbox is automatically selected and cannot be updated.

About the PEM File

1. The certificate should be an externally generated certificate.
2. Make sure to use only one certificate per storage connection. If you want to use another certificate, it should translate to a different Core Archive Storage connection.
3. In event of certificate expiry, edit the existing storage connection, and enter a new externally generated certificate in the same PEM File Content field. Once the certificate is edited in IA Web App, login into the Core Archive Connector Business Administrator and enable the data source again. Once done, you will be able to use the storage with OpenText Information Archive again.

Note: The test connection and create storage options do not check if the certificate is still valid.

When adding a store using the IA Web App, use the same logical archive that is pre-configured in Core Archive Business Administrator.

Prior to ingesting data for the first time:

1. Log into the Core Archive Connector Business Administrator and enable the data source.
2. Once selected, you should be able to view and accept the certificate.

2.2.1.2 Providing hardware retention to objects on Scality RING and Pure Storage FlashBlade

Once retention is applied to any objects/packages in the IA Web App, retention is also applied at the Scality RING/Pure Storage FlashBlade (hardware) level. If you choose to apply retention at the hardware-level, enable the Object Lock feature when creating a Scality RING/Pure Storage FlashBlade bucket.

Important

Ensure that the Object Lock feature is enabled, if desired, when you are creating the bucket, as it is not possible to enable the feature once the bucket has been created. When applying retention to any object, provide a date set in the future. Once applied, it is impossible to delete the object until this retention period expires.

To enable the Object Lock feature, when creating a bucket in OpenText Information Archive, ensure the **Push retention at the bucket-level** checkbox is selected. If this box is not selected, the Object Lock feature will not be enabled. Once the bucket has been created, you still have the option to deactivate the Object Lock feature by selecting the **Do not push retention at the hardware-level** checkbox. This scenario might occur if the Object Lock feature was enabled, but you do not want to apply retention to a particular store. If the **Do not push retention at the hardware-level** checkbox is selected, if a retention policy is applied to a package or object, retention is only applied

at the OpenText Information Archive-level and not at the Scality RING/Pure Storage FlashBlade levels. If you opted not to enable the Object Lock feature, the **Do not push retention at the hardware-level** checkbox is automatically selected and cannot be updated.

2.2.2 Configuring the Storage Class for Microsoft Azure Blob storage

For Microsoft Azure Blob storage, OpenText Information Archive supports two storage classes:

- Hot: Used to store data that is accessed frequently.
- Cool: Used to store data that is infrequently accessed and stored for at least 30 days.

While creating an Azure container using the IA Web App, define a **Storage Class**. For the **Storage Class** field, select either **Hot** or **Cool**. The default value for the **Storage Class** field is **Hot**.

There is a **Storage Class** dropdown in the Azure store creation screen, as well. The purpose to provide this dropdown at the Azure store-level is to provide the option to override the storage class that was defined at the Azure container-level. For example, if, at the Azure container-level (the Create Container screen), a user previously configured the **Storage Class** as **Hot**. At the store-level (Azure store creation screen), however, you have configured the **Storage Class** as **Cool**. Then, the objects or packages ingested into this particular Azure container through this particular store will be stored in a Cool storage class.

The default value for this dropdown **Storage Class** on the Azure store creation screen is **Default**. The **Default** value for the dropdown indicates that there is no storage class defined at the Azure store-level. In this case, the storage class that is defined at the Azure container-level, will receive any ingested objects.

Important

When completing the **Credentials** section in the IA Web App:

- For the **Access Key** field, enter the account name.
- For the **Secret Key** field, enter the key.

2.2.3 Adding lifecycle rules with the Microsoft Azure portal

An Administrator can configure a lifecycle management rules for Amazon S3 (S3) stores and Google Cloud Storage (GCS) bucket. The policy allows customers to define lifecycle rules that move objects ingested through OpenText Information Archive in S3/GCS buckets to move from one storage class to another.

The objects transition from one storage class to another according to the transition rules defined in the lifecycle management configuration. This helps customers significantly reduce storage costs, as customers can define rules that move objects from an expensive storage class to a comparatively cheaper class.

While customers have the option to configure lifecycle management rules for S3 and GCS in OpenText Information Archive, it is not possible to configure rules for Microsoft Azure because of several constraints. It is possible, however, for customers to use the Azure portal to configure lifecycle management rules directly. The eligible objects ingested through OpenText Information Archive can then be transitioned from one access tier to another, according to the rules defined via Azure portal.

Ensure the following settings are entered in the **Blob Service > Lifecycle Management > Details** tab:

Rule scope
Ensure that the field is set to **Limit blobs with filters**. If this value is not set, the lifecycle management rule will be applicable to all containers created for that particular storage account.

Blob type
Ensure that this field is set to **Block blobs**.

Blob subtype
Ensure that this field is set to **Base blobs**.

Important
In the **Base blobs** tab, add an **if-then block** rule. OpenText Information Archive **only** supports the **Move to cool storage** setting. This transfers all eligible blobs/objects from the hot access tier to the cool one after the specified number of days have passed.

In the **Filter set** tab, add a prefix. For example, if you want to create a lifecycle management rule specific to a container, enter the name of the bucket in a prefix section. You can associate a single lifecycle management rule with multiple container/prefix paths. You can even enter the complete path of a store inside a container. Consequently, only objects contained in that store will be transitioned.

For more information about Optimizing costs by automating Azure Blob Storage see the Microsoft website.

2.2.4 Dell EMC ECS

To use Dell EMC Elastic Cloud Storage (ECS), ensure that the application has been created and it includes a space. A space is a storage configuration object that represents the relation between storage and application.

To configure ECS storage:

1. In the **Administration > Storage** tab, click + and select **Create Storage System**.
2. Select Dell EMC Elastic Cloud Storage from the **Storage Type** list.
3. Enter a **Storage Name**.
4. If desired, enter a **Description**.
5. Specify the **URL**, which is the URL of the ECS store.
6. If you are using a proxy, select the **Enable Proxy** box and enter the **Proxy Url**. If selected, you will also have to enter a **Proxy User Name** and **Proxy User Password** in the **Credentials** section.
7. In the **Credentials** section, complete the following to connect to the ECS instance:
   a. Enter the **Credential Name**.
   b. If desired, enter a **Credential Description**.
   c. Enter the **Access Key** ID.
   d. Enter the **Secret Key** ID.
8. Click **Create**. Your new storage will now be in the list of storage systems.
9. To configure an application to be the ECS Storage, navigate to the application's **Spaces** tab and create a space or edit the existing space.

   Note: For a table application, if you are creating a new space, you must first create an RDB database that hasn't been already associated with an application.

   a. For the **Structured Databases**, select a **Database** from the dropdown list.
   b. For the **Storage Systems > Storage Type** field, select **Object Storage**.
   c. In the **Storage Name** list, select the storage you want. The list of possible choices will show the name of the storage with the type of storage in parentheses (e.g., **My ECS (S3) – <url>**, where<*url*>is the URL of the ECS storage).
   d. Click **Create**.
10. To configure a store to use the ECS Storage, navigate to the application's **Stores** tab and add a store, Ensure the following:
    - For the **Space** field, select the space that you created or edited to include the ECS Storage.

- Select the applicable **Space Root** (the format for the values contained in the **Space Root** field is <ApplicationName>-space_<StoreName>),

11. Click **Create Bucket** and enter a bucket name. Refer to the following for more information about bucket names.
12. Select **Confirmation** from the **Type** list,
13. Select whether to push retention to the hardware level.
14. Click **Create**,

The store that you created will be used by the Holding Wizard for storing the unstructured content.

2.2.5 Configuring Dell EMC CAS Elastic Cloud Storage for an application

This section illustrates how to configure system objects required to ingest data into Dell EMC CAS Elastic Cloud Storage.

To configure a Dell EMC CAS Elastic Cloud Storage object for ingestion:

1. Select Dell EMC CAS Elastic Cloud Storage from the **Storage Type** list.
2. Enter a **Storage Name**.
3. If desired, enter a **Description**.
4. Specify the **Connection String**, which is the storage system's IP address or DNS domain name along with the location of the PEA file on the server.

   Set the global variable `LD_LIBRARY_PATH`, which is the path of the storage system's libraries. If these values are not set, you will not be able to access the storage system's Command Line Interface and will, instead, receive an error message.

   Note: Ensure that you are using the correct SDK version to connect with OpenText Information Archive.

5. Enter the following Pool Entry Authorization (PEA) information:

   a. Enter the **Variable**.

   b. Enter the **Content**

6. Create an application or edit an existing application to use the newly created Dell EMC CAS Elastic Cloud store.
7. Create a space under the newly created application.

   a. Select **Legacy Object Storage** in the **Storage Systems > Storage Type** field.

   b. Select the storage system created in step 1.

8. Add a store. For a SIP application, it is necessary to assign the stores at the holding level. For a table application, it is necessary to assign the stores at the database level.

    a. Select the applicable **Space Root** (the format for the values contained in the **Space Root** field is <ApplicationName>-space_<StoreName>),
    b. Create a bucket in which to store data.

Be sure to follow the post-installation steps to include the storage system's SDK libraries in the classpath for the IA Server.

2.2.6 Amazon S3 storage

Amazon S3 supports virtual hosted-style and path-style URLs to access a bucket.

Path-style URLs, however, are being deprecated, OpenText Information Archive now only supports virtual-hosted-style URLs for Amazon S3 storage.

In virtual-hosted-style URL, the bucket name is part of the domain name in the URL.

The virtual hosted-style method requires the bucket name to be DNS-compliant. For more information about the virtual hosting of buckets see the *Amazon Simple Storage Service user guide* on the Amazon Web Services website.

Click the **Register Bucket** button to register an existing bucket or access point alias. However, if you register an access point alias, you cannot apply an object lock through using the **Push retention at the bucket level** feature and, furthermore, cannot configure lifecycle management (for more information, see Configuring a lifecycle management rule for a store.

This section illustrates how to install and ingest content into Amazon S3 Storage (hereafter referred to as S3).

To use Amazon S3, download the S3 SDK for Java. Refer to the latest set of release notes to learn what versions of S3 SDK are supported.

2.2.6.1 Configuring hardware retention for Amazon S3

Amazon S3 allows you to provide hardware-level retention to stored objects.

Previously, when a user applied retention to packages present in Amazon S3 stores via the IA Web App, retention was applied only at the OpenText Information Archive level. Even though retention was applied to the packages, those packages can actually be deleted via the Amazon S3 console.

Currently, once retention is applied to any objects/packages at the OpenText Information Archive level, retention is also applied at the Amazon S3 (hardware) level. Therefore, you can no longer delete those objects from the Amazon S3 console.

The S3 retention policy is a retention mechanism applied at Amazon S3 (hardware level).

If you do want to use S3 retention, enable the Object Lock feature while creating Amazon S3 buckets. If the bucket has been created and the Object Lock feature was not enabled, it is impossible to enable it.

While applying retention to any object, provide a date in the future. Once we apply retention with this date, it is not possible to delete the object until the retention period has expired. For information about the Object Lock feature see the *Amazon Simple Storage Service user guide* on the Amazon Web Services website.

When creating a bucket for Amazon S3 in the IA Web App, you have the option to enable the Amazon S3 Object Lock feature via the **Push retention at the bucket level** checkbox.

If you create a bucket and do not enable the **Push retention at the bucket level** checkbox, the bucket will be created without the Object Lock feature. If the box is selected, however, the Amazon S3 bucket is created with the Object Lock feature enabled.

Once the Amazon S3 bucket has been created, you still have the option **Do not push retention at the hardware level** for the store.

Suppose the bucket has the capability to support retention at the hardware-level (the Object Lock feature has been enabled), but you do not want to apply S3 retention to a particular store. Then, on the Amazon S3 store creation screen, enable the **Do not push retention at the hardware level** checkbox. Then, even if any retention policy is applied to packages or objects, retention is only applied at the OpenText Information Archive level and not at Amazon S3 level. However, if you decide to apply retention at the hardware-level also (S3 retention), ensure that the **Do not push retention at the hardware level** checkbox is not checked at the store-level screen.

If you have not enabled the object retention feature at the bucket-level, the **Do not push retention at the hardware level** checkbox on the Store screen is checked and you cannot change it.

2.2.6.2 Configuring a location when creating a bucket

When you create a bucket, use the **Location** field to specify the AWS Region in which you want Amazon S3 to create the bucket. Once a bucket is created in Amazon S3, it becomes impossible to change its location. The user can see the Location of the bucket in **View Bucket** screen, but cannot change it.

AWS maintains a series of opt-in Regions with higher security requirements than other commercial regions:

- Africa (Cape Town)
- Asia Pacific (Hong Kong)
- Asia Pacific (Hyderabad)
- Asia Pacific (Jakarta)
- Asia Pacific (Melbourne)

- Canada West (Calgary)
- Europe (Milan)
- Europe (Spain)
- Europe (Zurich)
- Israel (Tel Aviv)
- Middle East (Bahrain)
- Middle East (UAE)

Create a bucket in one of the above-mentioned opt-in Regions in the Amazon S3 console and then map the bucket in OpenText Information Archive. For more information about the opt-in Regions, see *AWS Resource Explorer User Guide* on the Amazon AWS website. To see the full list of Regions, see *Amazon EC2 User Guide for Linux Instances* or *Amazon EC2 User Guide for Windows Instances* on the Amazon AWS website.

The default Amazon S3 bucket location is **US East (N. Virginia)**. For more information about creating buckets, see the *Amazon Simple Storage Service (S3) User Guide* on the Amazon AWS website.

2.2.6.3 Configuring the Storage Class

Once an object is stored in Amazon S3 storage, the object is saved as a particular **Storage Class**. OpenText Information Archive supports the following storage classes:

- Standard
- Standard-IA
- Intelligent-Tiering
- One Zone-IA
- Glacier Instant Retrieval (IR)

The default storage class for an object is always **Standard**. While creating an Amazon S3 bucket, you can define the **Storage Class** in which the objects are to be saved. Once the bucket has been created, if you want to change the storage class defined at the Amazon S3 bucket-level, change the access tier on the **View Bucket** screen.

You can override the storage class on the Amazon S3 store creation screen (store-level). If you have defined any particular storage class at the Amazon S3 store-level, it will take priority over the storage class defined at the Amazon S3 bucket-level. The objects will then be saved in the storage class defined at the Amaozn S3 store-level. For information about Amazon S3 Storage Classes see the Amazon AWS website.

2.2.6.4 Configuring a lifecycle management rule for a store

Rules can be defined to apply retention or holds to records or packages, and can be evaluated by running the associated jobs (Apply Retention via Rule, Apply Hold via Rule). For more information, see Section 9.9 "Rules" in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

While creating an Amazon S3 store, you can also configure a lifecycle management rule for the store in a bucket. In lifecycle management, objects can be transferred from one storage class to another for automatic cost savings after a certain number of days have passed (also known as *Transition*). As of this release, for the lifecycle management rule, OpenText Information Archive supports one transition. You can configure the **Rule Name**, **To** (the destination storage class), and **Age (days)**, which indicates the number of days since the object was created, as a part of that rule.

While creating a transition for an Amazon S3 lifecycle management rule, there are five storage classes to choose from:

- Standard-IA
- Intelligent-Tiering
- One Zone-IA
- Glacier
- Glacier Deep Archive

Glacier and Glacier Deep Archive are offline storage classes. If you select either one of these as the destination storage class, three more fields are displayed in the IA Web App:

- A checkbox **Offline Content** (a read-only field)
- **Restoration Rules**
- **S3 Duration**

If the object is transferred to the Glacier or Glacier Deep Archive storage classes, those objects cannot be downloaded directly. Instead, a restoration procedure needs to be initiated. The restoration configurations require **Restoration Rules** and **S3 Duration**, which are entered during Amazon S3 store creation.

For the **Restoration Rules**, if the **To** dropdown (destination storage class is **Glacier**), there are three values to choose from:

- **Standard (3-5 hours)**
- **Expedited (1-5 minutes)**
- **Bulk (5-12 hours)**

For the **Restoration Rules**, if the **To** dropdown (destination storage class is **Glacier Deep Archive**), there are two values to choose from:

- **Standard (within 12 hours)**
- **Bulk (within 48 hours)**

S3 Duration is the time duration (in days) during which the restored copy from Glacier/Glacier Deep Archive can be accessed. Once this duration has expired, the restored copy is no longer accessible/downloadable, and you need to initiate the Restoration process once again.

The creation of lifecycle management rules on the Amazon S3 store creation screen is optional. For information about object lifestyle management, see the *Amazon Simple Storage Service user guide* on the Amazon AWS website.

There are certain rules for the transitions through lifecycle management rules between the various storage classes. For example, as a part of lifecycle management, objects can be transferred from **Standard** to **Standard-IA**, **Standard/Standard-IA** to **Intelligent-Tiering**, **Standard/Standard-IA/Intelligent-Tiering** to **One Zone-IA**, **Standard/Standard-IA/Intelligent-Tiering/One Zone-IA** to **Glacier**, and **Standard/Standard-IA/Intelligent Tiering/One Zone-IA/Glacier** to **Glacier Deep Archive**, but not *vice versa*. For information about transitioning objects see the *Amazon Simple Storage Service user guide* on the Amazon AWS website.

There is, however, one limitation. For the following transitions, Amazon S3 does not transition objects that are smaller than 128 KB because it is not cost-effective:

- From the S3 Standard or S3 Standard-IA storage classes to S3 Intelligent-Tiering or S3 Glacier Instant Retrieval
- From the S3 Standard storage class to S3 Standard-IA or S3 One Zone-IA

For information about supported transitions and related constraints see the *Amazon Simple Storage Service user guide* on the Amazon AWS website.

2.2.6.5 Installing Amazon S3 SDK (AWS SDK)

To use Amazon S3, download the S3 SDK for Java. Refer to the latest set of release notes to learn which versions of S3 SDK are supported.

1. Install the OpenText Information Archive distribution.
2. Download Amazon S3 SDK for Java from the Amazon Web Services website.

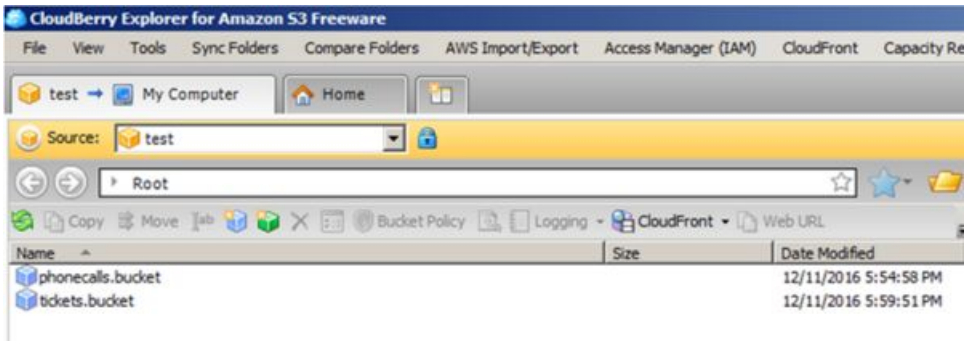
   Prior to the next step, ensure that the OpenText Information Archive server is not running.
3. Copy the `jar` file (`aws-java-sdk-<version number>.jar`) from the `lib` folder of the downloaded S3 SDK into external directory of the downloaded OpenText Information Archive distribution (`/infoarchive/lib/iaserver/external/`).
4. Start the IA Server.
5. For testing purposes, it is possible to configure a sample application with AWS S3 storage. For example, you may install the PhoneCalls (package-based

application) or Tickets (table-based application) applications from OOTB samples and then, with the help of IA Web App, reconfigure the storage to use Amazon S3.

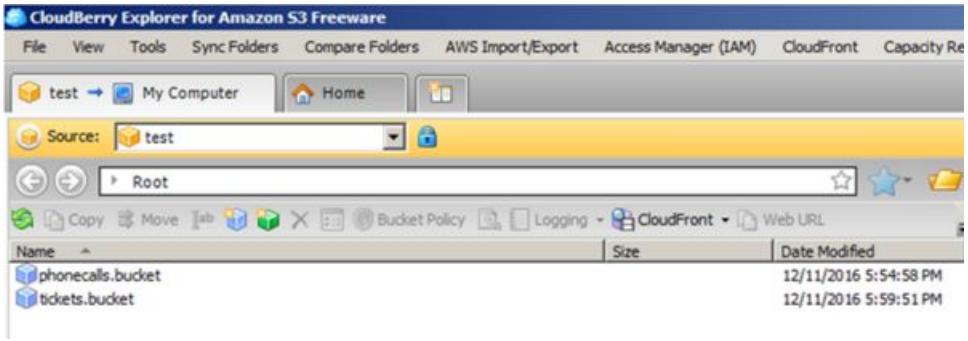
2.2.6.6 Verification of Amazon S3 storage usage

Download free tools such as CloudBerry to connect to Amazon S3 and verify the existence of the buckets.

Navigate inside the bucket and verify that the AIP and BLOB objects are stored in the bucket:



Alternatively, use the AWS console to see the bucket information:



2.2.6.7 Configuring a proxy server to connect to Amazon

The Administrator can use a proxy server setup to connect to Amazon Web Services (AWS) S3 for production deployment.

In case a proxy is required, the following demonstrates how to configure OpenText Information Archive to use Amazon S3 storage with a proxy:

- When you add or edit the storage system, ensure following information is entered:

  a. Check the **Enable Proxy** box.
  b. Enter a **Proxy Url** for the proxy server.

If other users will access the proxy server:

c. Enter **Access Key**.

d. Enter **Secret Key**.

e. If desired, enter a **Proxy User Name**.

f. If desired, enter a **Proxy User Password**.

2.2.6.8 Enabling Amazon Web Services Identity and Access Management (IAM)

The system supports the use of Amazon's IAM role authentication in addition to the standard credentials using access and secret keys. For more information about IAM, refer to the *Amazon Simple Storage Service User Guide*. The IAM role authentication mode is applicable only when the system is deployed on Amazon Web Services environments.

To enable IAM:

1. When creating or editing an existing Amazon S3 storage system, select the **Use IAM Role based authentication** check box.
2. Complete the following:
   a. Enter a **Credential Name**.
   b. Optional Enter a **Credential Description**.

   If IAM is enabled, the standard credentials of Access Key and Secret Key are not available.
3. If you are creating a storage system, click **Create**. If you are editing an existing storage system, click **Save**.

2.2.7 Configuring AWS S3 with Amazon Glacier for an application

To archive data, complete the following instructions to configure AWS S3 and AWS Glacier. The following should be configured prior to starting this procedure:

- Data Node
- Database
- Application
- Space

Caution

The general rule of thumb is that any content that the system needs to access regularly should not be stored in Amazon Glacier.

1. Configure a storage system that can be done in the Storage section of the IA Web App.

   a. For the **Storage Type** field, select S3.

   b. Provide values for the remaining fields.

      When entering the **URL**, ensure it points to the end point of the desired AWS S3 service.

      When entering the **Secret Key**, enter the AWS S3 account

2. If desired, configure the Glacier feature by selecting **Enable Glacier** so that data can be archived in AWS Glacier. The archival process is driven by the rules as specified while configuring a store. The following rules apply to all of the content objects in a bucket:

   - *S3 to Glacier Transition Rule*: Signifies the time period (in days) that starts from the creation date of the content objects of a bucket, which is also configured in the Stores section. At the end of this period, those content objects will be archived in AWS Glacier.

     Note: It may take a while before the archive process is completed.

   - *Glacier to S3 Restoration Rule*: Decides how long the restoration of an archived object may take so that it can be read. Users can specify one of the three options:

     – Expedited (fastest: 1-5 minutes for retrieval),

     – Standard (3-5 hours for retrieval) and

     – Bulk (5-12 hours for retrieval).

   - *Rule Name*: Provide a unique name for configuration rule.

   - *S3 Duration*: Signifies the number of days until a restored object will be kept in AWS S3.

If AWS S3 (with Glacier feature) is accessed via a proxy server, **Enable Proxy** should also be checked besides **Enable Glacier**. The relevant details of the proxy (for instance, Proxy URL, Proxy User Name and Proxy User Password are subsequently specified). Proxy User Name and Proxy User Password are optional and only specified if required by the proxy server being used.

When a content is stored on Glacier, it is not immediately available. If an end user attempts to access content, a notification appears in the Status column of the **Background Requests** tab.

2.2.7.1 Multipart

Users can upload SIP and table data in AWS S3 using a multi-part upload strategy, in which a file is uploaded in multiple parts. Two properties specific to that strategy are: part size and multi-part upload threshold, which are provided by users while configuring a `StorageEndPoint` object.

If the file size exceeds the specified threshold, the multi-part upload strategy will be used. AWS S3 splits the file into chunks, each having the size specified in the `StorageEndPoint`'s `partSize` property.

If either of the two properties is unspecified (`part size` or `multi-part upload threshold`), IA Server uses a suitable default value for the missing field. And, if the specified value is invalid, then an error is issued and the creation of a `StorageEndPoint` will be unsuccessful.

2.2.8 OpenText Archive Center

Prerequisite: As a prerequisite step prior to connecting to Archive Center from OpenText Information Archive, the logical archive must be created in advance.

To configure Archive Center (AC) as a storage service in OpenText Information Archive, users need to add a storage system with an Archive Center type, specifying the following:

- `ArchiveCenterEndPoint`
  - `Name`: Enter the name of the `ArchiveCenterEndPoint` instance.
  - `Hostname`: Enter the server name that the Archive Center server connects to.
  - `port`: Enter the port that the Archive Center (AC) server connects to. Always specify the shortform before using it.
  - `SSL`: An optional field that needs to be enabled if communication between the IA Server and AC is to be encrypted. If the provided port is encrypted, then the SSL flag should be enabled. For more information, see "Setting up TLS/SSL between OpenText Information Archive and OpenText Archive Center" on page 34.
- `ArchiveCenterEndPointCredential`
  - `Name`: Enter the name of the `ArchiveCenterEndPointCredential` instance.
  - `Description`: Enter the description of the `ArchiveCenterEndPointCredential` instance.
  - `pemFile`: Enter the content comprising private key and certificate (this field does not represent a file path).

Caution

Known Issue: When the incorrect PEM certificate (syntax is valid, but wrong PEM certificate) is entered in the Storage tab, the Test

Connection operation is successful. However, ingestion and all other CRUD operations will fail.

For more information about encrypting communication between OpenText Information Archive and Archive Center, see "Setting up TLS/SSL between OpenText Information Archive and OpenText Archive Center" on page 34.

2.2.8.1 Store configuration

When creating a bucket in IA Web App, use the same logical archive (pre-configured in the Archive Center administration client).

Note: Prior to ingesting data for the first time, log into the Archive Center administration client and enable the certificate.

2.2.8.2 About the PEM File

1. The certificate should be an externally generated certificate.
2. Make sure to use only one certificate per storage connection. If you want to use another certificate, it should translate to a different Archive Center Storage connection.
3. In event of certificate expiry, edit the existing storage connection, and enter a new externally generated certificate in the same PEM File Content field. Once the certificate is edited in IA Web App, login into the Archive Center administration client and enable the logical archive again. Once done, you will be able to use OpenText Information Archive again.

2.2.8.3 Setting up TLS/SSL between OpenText Information Archive and OpenText Archive Center

You can protect the data that is sent between OpenText Information Archive components and Archive Center by setting up a TLS 1.2 connection. (TLS is the successor of SSL.) For more information about configuring TLS/SSL for Archive Center , see the *OpenText Archive Center Installation Guide for Windows (Extended Components Installer) (AR-ICLU)*.

Before you begin: Install Archive Center.

To set up TLS/SSL between OpenText Information Archive and Archive Center :

1. Stop Archive Center's Apache Tomcat server by using the following command:

   ```
   net stop <TOMCAT_NAME>
   ```

   where <TOMCAT_NAME> is the name that the Tomcat server process is registered as.

2. Stop the Archive Spawner process by using the following command:

   ```
   net stop <SPAWNER_NAME>
   ```

where <SPAWNER_NAME> is the name that the Archive Spawner process is registered as.

3. In a text editor, open the server configuration file for Tomcat, <TOMCAT_ROOT>/conf/server.xml.

4. Change the parameters in the file as follows:

```
<Connector port="8080" protocol="HTTP/1.1"
   connectionTimeout="20000"
   redirectPort="<PORT_NUMBER>" />
<Connector port="<PORT_NUMBER>"
protocol="org.apache.coyote.http11.Http11NioProtocol"
   maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
   clientAuth="false" sslProtocol="TLS" keystoreFile="<KEYSTORE_FILE>"
   keystorePass="<TOMCAT_PASSWORD>" />
```

where <PORT_NUMBER> is the port that you want to use for HTTPS connections (for example, 443), and <KEYSTORE_FILE> is the location of your keystore relative to the <TOMCAT_ROOT> directory (for example, conf/.keystore).

5. Start the Tomcat server by using the following command:

```
net start <TOMCAT_NAME>
```

6. Start the Archive Spawner process by using the following command:

```
net start <SPAWNER_NAME>
```

7. In the OpenText Information Archive Administration Client, open the properties of the Archive Center server, which are listed under shared services.

8. In the **Port** field, specify the port number that you specified in step 4 for Tomcat, and change the **Protocol** to **https (strongly recommended)**.

9. Sign out of the Archive Center server.

10. Restart the Archive Spawner process.

11. Sign in to the Archive Center server. Under its Configuration section, the ArchiveLink HTTPS Port should be the same port as the port that you specified in step 4 for Tomcat. If it is not the same port, then do the following:

    a. Change the properties of ArchiveLink HTTPS Port to specify the port number.

    b. Restart the Tomcat server.

    c. Restart the Archive Spawner process.

12. Open admInfo by running the following command on the computer that hosts Archive Center:

```
https://<IP_OF_ARCHIVE_CENTER_SERVER>/archive?admInfo&pVersion=0046&resultAs=html
```

    Verify that the SSL port is the port that you specified.

13. Configure ArchiveCenterEndPoint by providing the SSL port in its **port** field and set the **ssl** boolean field to **true**.

2.2.9 Core Archive

Complete the following prerequisite steps prior to connecting to Core Archive from OpenText Information Archive:

1. Contact OpenText to obtain the Core Archive tenant
2. Install the Core Archive Connector, which internally communicates with the tenant created in the previous step.
3. The collection and logical archives should be configured with help of a Core Archive Connector Business Administrator.

To configure Core Archive as a storage service in OpenText Information Archive, users need to add a storage system with a Core Archive type, specifying the following:

- `CoreArchiveEndPoint`:
  - `Name`: Enter the name of the `CoreArchiveEndPoint` instance.
  - `Hostname`: Enter the host name (the location where the Core Archive Connector is installed).
  - `port`: Enter the port the Core Archive Connector runs on.
  - `SSL`: An optional field that needs to be enabled if communication between the IA Server and Core Archive is to be encrypted. If the provided port is encrypted, the SSL flag should be enabled.
- `CoreArchiveEndPointCredential`
  - `Name`: Enter the name of the `CoreArchiveEndPointCredential` instance.
  - `Description`: Enter the description of the `CoreArchiveEndPointCredential` instance.
  - `pemFile`: Enter the content comprising the private key and certificate (this field does not represent a file path).

Caution

Known Issue: When the incorrect PEM certificate (syntax is valid, but wrong PEM certificate) is entered in the Storage tab, the Test Connection operation is successful. However, ingestion and all other CRUD operations will fail.

Refer to About the PEM file for more information.

2.2.10 Custom storage

This section illustrates the process of:

- Configuring custom storage
- Implementing an OpenText Information Archive content store
- Setting up the environment

OpenText Information Archive includes example extensions in the `<IA_ROOT>/extensions/iaserver` directory. These unsupported examples can help you understand how extensions work, and they include README files that you can read for more information.

1. Configure the custom store.

   To configure a custom store, a bag of properties and the name of the Spring service, which is used by OpenText Information Archive to find the customer implementation of ContentStoreFactory, can be specified by the clients via the REST API. It can be done by following the REST relation `http://identifiers.emc.com/custom-storages`, which is available under the Services link `http://identifiers.emc.com/services`. The properties can be used by the clients in implementing the custom store.

2. Customize the implementation. The following three interfaces can be used by customers to define the implementation:

   - `com.emc.ia.content.store.ContentStore`: Use this interface to define methods that perform CRUD operations on unstructured content.

     The implementation should also set the location of an object through `Content#setPath(String path)`, where the path contains the location of the object. If the aforementioned method is not used in combination with `Content#getPath()`, the implementation can never change the algorithm for path construction, since it needs to locate previously stored content. If the implementation uses `Content#setPath()` at the end of a write, then on a read, it can use `Content#getPath()` to retrieve the stored path to locate the object to read. If not, the implementation will have to construct the path on each read using the other Content fields.

     Note: The getPath()/setPath() feature was not available in the 4.2 Custom ContentStore API. Hence, for any read of Content objects (in a Custom ContentStore) created with OpenText Information Archive 4.2, `Content#getPath()` will always be blank. The implementation will need to take that into account and fall back to resolving against its initial path construction scheme for such Content objects. OpenText Information Archive cannot migrate this for you, since it has no knowledge about the path construction scheme of your Custom ContentStore implementation.

   - `com.emc.ia.content.store.ContentStoreFactory`: It is implemented as a Spring service. A service name, custom storage name and properties

(optional) are provided to OpenText Information Archive while configuring a custom store. The following is an example of what the payload of adding a custom store looks like:

```
{
  “name” : “customStorageName”,
  “factory”: “CustomContentStoreFactoryImpl”,
  “properties” : {
    “key1” : “value1”,
    “key2” : “value2”.
   }
}
```

The following is an example of defining `ContentStoreFactory` as a service:

```
@Service("CustomContentStoreFactoryImpl")
public class ContentStoreFactoryImpl implements ContentStoreFactory {
@Override
public ContentStore newStore(Bucket bucket, Map<String,
String> properties) {
// return instance of your implementation of ContentStore.
}
}
```

In the above example, the service annotation contains `CustomContentStoreFactoryImpl`. It should be identical to what is specified during custom storage configuration (see the custom store's payload example).

- `com.emc.ia.content.store.ContentStoreRetention`: It can be implemented if the customer wants to push retention to the hardware level. The following two interfaces can be used to get information regarding the content store that might be useful in implementing custom store:
  - `com.emc.ia.content.store.Bucket`: It returns the name of the bucket as specified during configuration of content store (see step 1).
  - `com.emc.ia.content.store.Content`: It returns information that can be used by customers in implementing the custom store.

3. Set up the environment.

Note: If you are deploying several different extensions for the IA Server, ensure that different names for configuration classes are used.

Customers have to define a Spring configuration class with a condition: Its package name must be prefixed with `com.emc.ia`. This enables OpenText Information Archive to find customer-defined Spring services that also includes one for the `ContentStore`. For example:

```
package com.emc.ia.my.content;
@Configuration
@ComponentScan(basePackages = { "com.my.content.store" })
public class CustomStoreConfig {
}
```

The above configuration can be used by customers to include packages that contain implementation classes. In addition to the configuration class, OpenText Information Archive requires you to add implementation (as a jar) on its class-path. To achieve that, the jar will be copied into the external directory of the

OpenText Information Archive distribution, which is `<IA_ROOT>/lib/iaserver/external`.

2.2.11 PowerScale OneFS S3

To configure PowerScale OneFS S3 storage:

1. In the **Storage** tab, click + and select **Create Storage System**.
2. Select PowerScale OneFS S3 from the **Storage Type** list.
3. Enter a **Storage Name**.
4. If desired, enter a **Description**.
5. Specify the **URL**, which is the URL of the PowerScale OneFS S3 store.
6. If you are using a proxy, select the **Enable Proxy** box and enter the **Proxy Url**. If selected, you will also have to enter a **Proxy User Name** and **Proxy User Password** in the **Credentials** section.
7. In the **Credentials** section, complete the following to connect to the PowerScale OneFS S3 instance:
   a. Enter the **Credential Name**.
   b. If desired, enter a **Credential Description**.
   c. Enter the **Access Key** ID.
   d. Enter the **Secret Key** ID.
8. Click **Create**. Your new storage will now be in the list of storage systems.
9. To configure an application to be the PowerScale OneFS S3 storage, navigate to the application's **Spaces** tab and create a space or edit the existing space.

   Note: For a table application, if you are creating a new space, you must first create an RDB database that has not been already associated with an application.

2.2.11.1 Configuring system objects for ingesting data into PowerScale OneFS S3

This section illustrates how to configure system objects required to ingest data into PowerScale OneFS S3.

Important
To use PowerScale OneFS S3 with OpenText Information Archive, the version of OneFS must be 9.3 or above.

If you plan to use PowerScale OneFS S3 as object storage in OpenText Information Archive, enable the following configurations (one time) using the OneFS Storage Administration console:

1. In **Protocols** tab, select **Object Storage (S3)**.
2. In the **Object storage** screen, under the **Global Settings** tab, enable the **Enable S3 service** and select the **Enable S3 HTTP** checkbox.
3. • In the **Object storage** screen, under the **Key management** tab, select the user by providing `admin` as `User` and `FILE:System` as `Providers`. Select **Create new key** and you will receive an Access Key and Secret Key.

Using PowerScale OneFS S3 storage is similar to using Dell EMC Elastic Cloud Storage (ECS) in OpenText Information Archive:

1. Use the IA Web App to create the storage system first and provide an access URL, Access Key and Secret Key.
2. Use the IA Web App to create a space.
3. Then user needs to create a store. While creating the store, create buckets using either the IA Web App or in the OneFS Storage Administration console. Map them to the same bucket name in OpenText Information Archive. However, it is advisable to create the bucket from OpenText Information Archive itself.

   However, if you still want to create a bucket upfront using the OneFS Storage Administration console, ensure that the value for the path `is /ifs/<bucket_name>`, where `ifs` is the name of default directory of OneFS and `bucket_name` is the name of the bucket the customer is trying to create. You should also enable the **Create bucket path if does not exist** checkbox.

Limitation: Like Amazon S3, ECS, and NetApp StorageGRID, PowerScale OneFS S3 does not support the Hardware Retention feature yet.

2.2.12 Google Cloud Storage

This section illustrates how to configure system objects required to ingest data into Google Cloud Storage (GCS).

To configure a GCS object for ingestion:

1. Add a storage system with a GCS type.
   - Specify the **Client E-Mail**, **Project ID**, and **Private Key** to connect to the GCS storage.
2. Create an application.
3. Create a space under the newly created application.
   a. Specify Object Storage in the **Storage System** field.
   b. Select the URL created in Step 1a.
4. Add a store.
   - Create a bucket to store the data in.

The store that you created will be used by the Holding Wizard for storing the unstructured content.

2.2.12.1 Bucket configuration

Buckets are the basic containers that hold data.

Important

Each GCS Service Account has associated roles, either default roles that already exist or custom roles. Ensure that the role associated with the GCS Service Account has the following Cloud Storage permissions set:

- `resourcemanager.projects.get`
- `storage.buckets.create`
- `storage.buckets.get`
- `storage.buckets.list`
- `storage.buckets.update`
- `storage.objects.create`
- `storage.objects.delete`
- `storage.objects.get`
- `storage.objects.list`
- `storage.objects.update`

1. Click **Create Bucket**.
2. Enter the following information:

Bucket
For information about bucket naming conventions see the Google Cloud Storage website.

Location
Select the location to specify the region for the bucket being created. While the locations specified in the IA Web App might not match the locations listed in the *Create storage buckets guide* on the GCS website, locations are added to OpenText Information Archive every release, as required.

Push retention at the bucket level
Select the checkbox to enable the GCS Object Retention Lock feature. If you create a bucket and do not enable the **Push retention at the bucket level** checkbox, the bucket will be created without the Object Retention Lock feature. For more information about this feature, refer to the GCS product documentation.

Storage Class
Select the storage class of the bucket being created.

3. Click the + icon to enter a new lifecycle rule:

   From
   Move the content from the specified storage class.

   To
   Move the content to the specified storage class.

   Age
   Indicate the number of days when the content needs to be moved from the **From** storage class to the **To** storage class.

   Created Before
   Indicate the date to move the content that is created before the entered date in the **From** storage class to the **To** storage class.

4. Click **Add** to associate the lifecycle rule to the bucket.
5. Repeat Steps 3 and 4 to associate multiple lifecycle rules to the bucket.
6. Click the **X** icon to remove the associated rule from the bucket.
7. Click **Create**.

2.2.13 Scality RING

To configure Scality RING storage:

1. In the **Storage** tab, click + and select **Create Storage System**.
2. Configure the following fields:

   Storage Type
   Select **Scality RING** from the list.

   Storage Name
   Enter a name.

   Description
   Optionally, enter a description.

   URL
   Enter the URL of the Scality RING store.

   Enable Proxy
   If you are using a proxy, select this checkbox and enter the Proxy URL. If selected, you will also have to enter a **Proxy User Name** and **Proxy User Password** in the **Credentials** section.

   Enable Path Style Access
   If the **Enable Path Style Access** checkbox is checked, OpenText Information Archive will use path style access to connect to Scality RING.

   If the **Enable Path Style Access** checkbox is clear, OpenText Information Archive will use virtual-host style access to connect to Scality RING.

- Configure the following in the **Credentials** section to connect to the Scality RING instance:
  - Enter the **Credential Name**.
  - Optionally, enter a **Credential Description**.
  - Enter the **Access Key**.
  - Enter the **Secret Key**.

3. Click **Create**. Your new storage will now be in the list of storage systems.
4. To configure an application to be the Scality RING storage, navigate to the application's **Spaces** tab and create a space or edit an existing space.

Note: For a table application, if you are creating a new space, you must first create an RDB database that has not been already associated with an application.

2.2.13.1 Configuring system objects for ingesting data into Scality RING

This section illustrates how to configure system objects required to ingest data into Scality RING.

To use Scality RING with OpenText Information Archive, the version of Scality RING must be 9.2.0 or above.

Using Scality RING storage is similar to using NetApp StorageGRID in OpenText Information Archive:

To configure system objects for ingesting data into Scality RING:

1. Use the IA Web App to create the storage system first and provide an access **URL**, **Access Key**, and **Secret Key**.
2. Use the IA Web App to create a space.
3. Create a store. While creating the store, create buckets using either the IA Web App or in the Scality RING console. Map them to the same bucket name in OpenText Information Archive. However, it is advisable to create the bucket from OpenText Information Archive itself.

2.2.14 Pure Storage FlashBlade

To configure Pure Storage FlashBlade storage:

1. In the **Storage** tab, click + and select **Create Storage System**.
2. Configure the following fields:

Storage Type
Select **PureStorage FlashBlade** from the list.

Storage Name
Enter a name.

Description
Optionally, enter a description.

URL
Enter the URL of the Pure Storage FlashBlade store.

Enable Proxy
If you are using a proxy, select this checkbox and enter the Proxy URL. If selected, you will also have to enter a **Proxy User Name** and **Proxy User Password** in the **Credentials** section.

- Configure the following in the **Credentials** section to connect to the Pure Storage FlashBlade instance:
  - Enter the **Credential Name**.
  - Optionally, enter a **Credential Description**.
  - Enter the **Access Key**.
  - Enter the **Secret Key**.

3. Click **Create**. Your new storage will now be in the list of storage systems.
4. To configure an application to be the Pure Storage FlashBlade storage, navigate to the application's **Spaces** tab and create a space or edit an existing space.

Note: For a table application, if you are creating a new space, you must first create an RDB database that has not been already associated with an application.

2.2.14.1 Configuring system objects for ingesting data into Pure Storage FlashBlade

This section illustrates how to configure system objects required to ingest data into Pure Storage FlashBlade.

To use Pure Storage FlashBlade with OpenText Information Archive, the version of Pure Storage FlashBlade must be 4.5 or above.

To configure system objects for ingesting data into Pure Storage FlashBlade:

1. Use the IA Web App to create the storage system first and provide an access **URL**, **Access Key**, and **Secret Key**.
2. Use the IA Web App to create a space.
3. Create a store. While creating the store, create buckets using either the IA Web App or in the Pure Storage FlashBlade console. Map them to the same bucket name in OpenText Information Archive. However, it is advisable to create the bucket from OpenText Information Archive itself.

2.3 Enabling a delete marker when creating a bucket for Amazon S3, Scality RING, NetApp StorageGRID, and Pure Storage FlashBlade

This section discusses how to enable a delete marker for the Amazon S3, Scality RING, NetApp StorageGRID, and Pure Storage FlashBlade storage systems. For more information about delete markers, refer to the respective product documentation.

To use a delete marker, select the **Enable Delete Marker** checkbox.

2.4 Configuring background processing

The background processing in the IA Server can be configured using specific 'per node' settings in the `application.yml` file. By default, all background processing is enabled on every node. The `no-background-processing` profile can be enabled to disable most of the background processing on individual nodes. For advanced use cases, it is also possible to disable specific types of background processing by only enabling the corresponding profiles, but this is not recommended unless an explicit need to do so has been identified. When background processing is enabled normally, the following settings control its resource usage.

The `background.taskExecutor.threadPoolSize` property controls the physical threads available for order items, batch items, and job background processing. Due to the typically expensive nature of work performed by these threads, this is a hard limit. The total number of logical threads may be higher, the difference will overflow into the thread pool queue and are processed as threads become free. Background processing is automatically distributed across all nodes that have the corresponding

background processing enabled. It is the total number of threads for all nodes combined that determine maximal concurrency of a deployment.

`background.taskExecutor.threadPoolSize`
Specifies the maximum physical number of threads available for parallel background processing on the node. This should be fine-tuned in accordance with the available CPU threads.

`background.taskExecutor.queueCapacity`
Specifies the capacity to queue the overflow of requested concurrent background processing that exceed the number of available threads. When setting this value, take into consideration how many threads are available and the number of logical threads configured per background processing type.

`background.taskExecutor.threadNamePrefix`
Used for analysis of thread dumps. There is typically no need to change the default value for this setting.

The `background.numberOfThreads` properties define logical thread limits of the individual node for the different types of background processing:

`background.numberOfThreads.orderItems`
Maximum number of order items to run concurrently on this node.

`background.numberOfThreads.batchItems`
Maximum number of batch items to run concurrently on this node.

`background.numberOfThreads.jobItems`
Maximum number of jobs to run concurrently on this node.

For batch operations, there is additional configuration in the **Global Settings** tab that allows you to configure how many batches are created per operation type and the chunk size per batch, which controls the transaction scope and duplication of work during recovery of an interrupted run. There is a default fallback for both these settings that applies to all operations for which no operation specific override has been configured. Additionally, for each batch operation type, a dedicated value can be configured should there be a need to deviate from the default values. The global settings in question are `batch.nrOfBatches.*` and `batch.chunk.*`. There is also `batch.interval.*`, which allows for defining the maximum interval at which batch order items explicitly run to perform a status update on all batches. However, this interval setting typically does not need to be changed and is primarily available for troubleshooting purposes.

The `nrOfBatches` setting defines the preferred number of batches to create per batch operation of the corresponding type, which limits the maximum concurrency of a single batch operation. All batches of a batch operation can be run in parallel as long as there are enough batch item threads available across all nodes of the deployment combined (where batch item execution is enabled). If not enough batch item threads are available, the system queues the remaining batch items and runs them as batch item threads become available. The chunk size is implicitly enforced as a minimum batch size to avoid creating multiple batches too small for the concurrent execution to provide sufficient benefit to justify the necessary management overhead.

A batch operation divides the work across the value specified for the `nrOfBatches` property but, depending on the dataset, it is possible that the system creates fewer batches because:

- For each batch operation, the system first divides the work into batches and then each batch is broken down into chunks. The number of items to be processed in each chunk will be equal to the chunk size, except for the last one, as there might be fewer items remaining.
- Items belonging to the same partition are processed by the same batch.

The priority mechanism provides a means to prioritize a certain number of order/batch items to run as early as possible instead of the order they are scheduled. The priority `numberOfThread` limits are not additive to the regular limit, but instead come at the expense of regular items on an as needed basis. To avoid starvation of regular items in favor of priority items, the number of priority items should be less than the total number of items for the corresponding type.

An example to clarify the behavior is as follows:

- Given `numberOfThreads.orderItems` = 10
- Given `numberOfThreads.priorityOrderItems` = 5.
- Given there are 20 non-priority order items already scheduled before 10 priority order items become scheduled.
- Given all these order items take an exactly an equal amount of time.
- Then the first run executes the first five regular order items and the first five priority order items.
- The second run executes the second five regular order items and the second five priority order items.
- The third run executes the remaining 10 regular order items.

`background.numberOfThreads.priorityOrderItems`
: Maximum number of priority order items to run concurrently at the expense of earlier scheduled non-priority ones.

`background.numberOfThreads.priorityBatchItems`
: Maximum number of priority batch items to run concurrently at the expense of earlier scheduled non-priority ones.

The `background.s3taskExecutor` properties control the threads available for Amazon S3 operations. These threads timeout one minute after last usage and are only active during Amazon S3 operations.

`background.s3taskExecutor.threadPoolSize`
: Specifies the number of threads to be made available for Amazon S3 operations.

`background.s3taskExecutor.threadNamePrefix`
: Used for analysis of thread dumps. There is typically no need to change the default value for this setting.

The `background.taskScheduler` property controls the threads available for internal scheduled tasks.

`background.taskScheduler.threadPoolSize`
Specifies the maximum number of threads that should be used for scheduled tasks, the interval of which is configured separately below.

The various internal scheduled tasks run at a specific interval that can be configured per individual node using the following properties:

`background.scheduledTaskInterval.orderItems`
Frequency at which the system checks for newly scheduled order items to be executed.

`background.scheduledTaskInterval.batchItems`
Frequency at which the system checks for newly scheduled batch items to be executed.

`background.scheduledTaskInterval.jobItems`
Frequency at which the system checks for newly scheduled job items to be executed.

`background.scheduledTaskInterval.abandonedOrderItems`
Frequency at which the system checks for abandoned order items to reschedule.

`background.scheduledTaskInterval.abandonedBatchItems`
Frequency at which the system checks for abandoned batch items to reschedule.

`background.scheduledTaskInterval.abandonedJobItems`
Frequency at which the system checks for abandoned job items to reschedule.

`background.scheduledTaskInterval.batchProgressCheck`
Frequency at which the system checks for batch item completion to reschedule corresponding order items.

`background.scheduledTaskInterval.managedItemBackup`
Frequency at which the system backs up changed compliance metadata.

`background.scheduledTaskInterval.newJobs`
Frequency at which the system checks for new jobs to be scheduled.

`background.scheduledTaskInterval.abandonedJobs`
Frequency at which the system checks for abandoned jobs to reschedule.

`background.scheduledTaskInterval.cleanupJobs`
Frequency at which the system checks for jobs to clean up.

`background.scheduledTaskInterval.deleteOnExitCleanup`
Frequency at which the `deleteOnExit` cache is cleaned up to avoid a memory leak.

`background.scheduledTaskInterval.backgroundSearchesCancelCheck`
Frequency at which the system checks whether background searches have been requested to be canceled and act on the request.

`background.scheduledTaskInterval.savedSearchRerun`
Frequency at which the system checks for saved searches to be rerun.

`background.scheduledTaskInterval.rollForwardObjectCleanup`
Frequency at which the system checks for rollforward objects to clean up.

`background.scheduledTaskInterval.cleanupLocks`
Frequency at which the system checks for locks to cleanup.

`background.scheduledTaskInterval.updateGlobalSettings`
Frequency at which the system checks for updated global settings to populate the local cache with.

2.4.1 Changing an In Use file storage path using IA Web App

The following procedure is only applicable for file storage systems (local file systems and Dell EMC PowerScale) It is required that file storage systems are read-write accessible to *all* IA Server nodes in the environment. Unless only a single IA Server node is used, a network mountable or sharable location must be used. In this context *Local* refers to the environment/deployment, not to the individual IA Server nodes. Non-local would typically be cloud-based storage where the actual storage is outside of the deployment environment.

In case of disaster recovery proof deployments, the replica deployment site would typically also need to rely on a replica of the file system to avoid the storage system being inaccessible to the replica site when the main site goes down. It is technically possible to have both connect to the same network-based storage at the cost of reduced failover capabilities. Failing individual IA (Server/webapp/database) nodes can be recovered from, but failing entire data centers, including the storage, would then not be recoverable.

1. Use the IA Web App to edit the existing `defaultFileSystemRoot`'s **Folder Path** field in the **Storage** tab. Update the value to the desired path (for example, from the `data\root` directory to the `c:\backup\data\root` directory).

   Note: This path needs to be a shared Windows/Linux folder location, such as `//x.x.x.x/share` in order to support more than a single IA Server node in the entire environment. The directory must be writable. In case of only a single IAS node, it is still important to ensure this directory is located outside of the installation directory of IA to ensure that upon upgrade the data remains accessible in the new installation. OOTB a relative path is specified for ease of setting up demo environments, which results in a path of `<IA_ROOT>/data/root`, but for production environments this is never appropriate and needs to be changed before installing any applications (including first-time-setup) to avoid having to do additional work when changing it afterwards.

   Tip: In the sample applications, the following is the section that uses the properties to indicate the `defaultFileSystemRoot`:

```
fileSystemRoot:
  name: defaultFileSystemRoot
  configure: create
  description: Default FileSystemRoot
  path: ${fileSystemRoot.path}
```

2. Manually copy the content of the source folder (entire directory structure exactly under the file system root path) to the target location (for example, from `<IA_ROOT>\data\root` to `c:\backup\data\root`).
3. Verify that the corresponding application's space and stores point to the new target location.
4. Verify that the retrieval and ingestion operations have not been impacted by the change.

2.5 Moving content between stores

Unstructured content in OpenText Information Archive is stored in a storage system. OpenText Information Archive supports moving all or specific formats (`eg:ci_container`, `ri_xml`) of unstructured content from one store to another.

For example, if the unstructured content is stored in a FileSystem store, the Administrator can move the unstructured content to an Amazon S3 store using the `move-content` command in IA Shell (for more information, see Section 2.10.10 “move-content” in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*).

To move unstructured content between stores, the following conditions must be met:

- Content can only be moved between the stores of the same application.
- Source and target stores are REGULAR store types.
- Source Store status is either ONLINE or READ-ONLY, and the target store status is ONLINE.

2.5.1 Compliance stores

OpenText Information Archive takes necessary precautions when moving content from/to WORM store to maintain policy compliance. The following is done as part of the content move from/to WORM stores.

- It is not possible to move unstructured content from a WORM store non-WORM store (except if the option `DO_NOT_APPLY_STORAGE_RETENTION` is used).
- When the content is moved between WORM stores, the same retention period gets applied to the moved content in the target store.

2.5.2 Offline stores

Unstructured content in offline stores (for example, Amazon S3 Glacier) is not available immediately to move to other stores. It requires a restoration process first to bring the content online. As part of the move content, OpenText Information Archive invokes the restoration process when the content is offline. Once the content is available online, then it will be moved to the target store.

OpenText Information Archive provides necessary statistics on the number of moved content objects, offline content and records failures in the background task logs. For the offline stores, the Administrator needs to re-invoke the `move-content` command again once the content is available online.

2.6 Auditing

OpenText Information Archive allows the auditing of many events on different levels. There are events related to the system, the tenant, and individual applications.

Audited events can be searched only through the Audits application once they are archived by running the Archive Audits job. The job ingests unarchived audits and applies a retention policy, so they are eligible to be disposed after a certain amount of time has passed.

2.6.1 Audit event types

Being able to prove that certain actions have been performed is important, particularly when it comes to compliance. The **Audit** tab allows the Administrator to configure which event types are audited. While OpenText Information Archive is shipped with some defaults, it is strongly recommended that you review which audit event types are enabled.

In IA Web App, Administrators can access the **Administration > Audit** tab to select which events they want audited.

Caution

It is strongly recommended that you review what audit event types are audited, test them in a staging environment, review the volume of audits generated, and then enable only the most important audits required. The defaults may not be suitable, depending on your use of OpenText Information Archive.

For example, with SIP applications, the same audit settings are applied to private and aggregate modes. When the aggregate mode is used, a high number of audits may be generated. You may want to consider disabling some SIP-specific audits, such as audits in the Ingestion events category.

All audit types that can be configured for an individual application can also be configured at the tenant-level. Turning on audits for retrieval events can impact

performance. Whenever a user gets a page of resources, an audit is generated for each item returned in the page.

Audits for enabling or disabling audit events are audited at the system-level.

Use the Application filter to view:

- System-level audits: Contains a list of audit-events that correspond to services. Audits for changing events are audited at the system level.
- Tenant-level audits: Contains a list of tenant-level audit-events and application audits that are to be configured for all applications.
- Application-specific audits: Below the separator is a list of all installed applications. Select an application to view a list of audit events related to that application.

Note: If the audit is enabled at the tenant–level, the check box for the audit is disabled.

If you want the audit disabled, first, disable it at the tenant–level. Next, disable the audit from each application. Furthermore, for any new applications created or installed, the factory defaults apply, so if you do not want the audit enabled, you must manually disable it.

The Event Category filter allows an Administrator to determine which specific events are currently being audited. The filters depend on the level of access chosen. A check mark indicates that a particular audit event is enabled to generate an audit. Click a box to add or remove a check mark. The following table outlines the applications and the event categories that can be audited:

Application	Event category
System	Background Events Compliance Events Provisioning Events Other
Tenant	Background Events Compliance Events Content Events Content Provisioning Events Ingestion Events Package Events Provisioning Events Search Events Other

Application	Event category
Applications	Background Events Compliance Events Content Events Content Provisioning Events Ingestion Events Package Events Provisioning Events Search Events Other

Events are categorized in the following manner:

- **Background Events**:

  If an order item will act on many items, batch items represent a chunk of work to do.

  - **Batch Item**: Audit for when an order item acts on many items and a batch has been created.
  - **Job Definition**: Audit for when a job has been scheduled and configured.
  - **Job Instance**: Audit specific for an instance of running a job.
  - **Order Item**: Audit for when a background task has been created.
  - **Rule**: Audit for when a rule is used to apply retention, apply holds, or trigger events.

- **Content Events**: These are events specific to accessing the unstructured content associated with a record (content information = CI). The CI can be accessed via a native browser, the OpenText™ Brava!™ viewer, or the OpenText Intelligent Viewing viewer. Content events include:

  - **Download CI**: Audit for when CI is downloaded.
  - **Viewer Download**: Audit for when CI is downloaded via a native browser, the Brava! viewer, or the Intelligent Viewing viewer.
  - **Viewer Print**: Audit for when CI is printed via a native browser, the Brava! viewer, or the Intelligent Viewing viewer.
  - **Viewer View**: Audit for when CI is viewed via a native browser, the Brava! viewer, or the Intelligent Viewing viewer.

- **Ingestion Events**: Ingestion events are about ingesting content into the archive and tracking the state of the artifacts. Apply an audit based on one of the following actions to ingestion events at the tenant- and application-level. Both tenant- and application-level event types include AIPs and purge lists. Ingestion events include:

  - **Commit**: Audit for when the AIP is committed. This is the final part of ingestion where the AIP has been confirmed to be valid and any retention

required (based on the retention class or default retention policy on the application) is applied.

  - **Confirmation**: Audit of when confirmation is done. This action is done via the Confirmation job.
  - **Confirmation Completed**: Audit for when the confirmation is completed.
  - **Ingest**: Audit for the ingestion of an AIP or table.
  - **Ingest failed**: Audit for when the ingestion of an AIP fails.
  - **Receive**: Audit for when an AIP is received.
  - **Receive failed**: Audit for when the receive fails (for an AIP). Only happens because of a failure when running the IA Shell `receive` command. For more information, see Section 2.7.11 "receive" in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*.

- **Content Provisioning Events**: Content provisioning events are standard operations for content associated with an object. Content provisioning events include:
  - **Create Content**: Audit for when content has been created.
  - **Delete Content**: Audit for when content is marked ready for deletion.
  - **Download Content**: Audit for when content binary has been downloaded.
  - **Get Content**: Audit for when content has been fetched.
  - **Move Content**: Audit for when content has been moved to another store.
  - **Partial Update Content**: Audit for when content binary has been partially updated.
  - **Restore Content**: Audit for when a restore has been requested for the content.
- **Provisioning Events**: Provisioning events are standard operations for managing resources. Most resources support provisioning events. Audits for changing events are now audited at the system level (versus the tenant). Provisioning events include:
  - **Create**: Audit for when the resource is created.
  - **Delete**: Audit for when the resource is deleted. For items that are disposed, the dispose audit is generated instead of this audit.
  - **Reset**: Audit for all system configuration setting is reset. This can only be done via IA Shell or REST API. A reset audit is not generated when a configuration setting is reset using the IA Web App; instead, a delete audit is created.
  - **Retrieve**: Audit for when the resource is retrieved. This audit is generated for each item when getting the list of resources.

Caution

Activating the retrieve audit can degrade performance.

  - **Update**: Audit for when the resource is modified. The name of the attributes changed in the resource are included in supplemental data.

- **Compliance Events**: To learn what actions you can apply tenant- and application-level audits against, see Section 9.8.1 “What are compliance events?” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

  Note: Due to performance-related issues, when applying retention to records, there is an audit per retained set or hold set rather than an individual audit per record. This audit indicates the number of items that were added per set.

- **Package Events**: These are events specific to the package that can either impact the phase of the package or act on the package. Package events include:

  - **Invalid**: Audit for when an AIP is marked invalid, which occurs when the wrong SIP was submitted, and you want to resubmit the correct SIP with the same identifier. The audit is generated before the job processes it. The system may generate this audit during system recovery if the package was in an open library and the content was no longer available.
  - **Rebuild**: Audit for when an AIP is rebuilt, which is initiated via a user’s action.
  - **Reject**: Audit for when an AIP is rejected, which is done if all AIPs that belong to the same collection are invalidated. When you reject an AIP, you cannot resubmit an AIP with the same DSS as long there is one or more rejected AIPs in the repository. The audit is generated before the job processes it.
  - **Rollback**: Audit for when an AIP is rolled back, which happens for two reasons:
    - A rollback of the AIP, or
    - A rollback of the invalidation.

- **Other**: Apply an audit to other events at the tenant- and application-level, Other events include:

  - **Archive**: Audit for when audits are archived. Archiving is done via the Archive Audit job. When this is done, the archived audits are no longer available through the audit REST APIs. but are available using the search templates in the Audit application.
  - **Cache In**: Audit for when a database or application has been cached in.
  - **Cache Out**: Audit for when a database or application has been cached out.
  - **Disable**: A system-based audit for when an audit event has been disabled.
  - **Enable**: A system-based audit for when an audit event has been activated.
  - **End**: Audit for when declarative configuration, a job instance, or recovery ends (for an order item).

  - **Export**: Audit for exporting. Currently only for configuration, purge lists, retained sets, and hold sets.
  - **Failed**: A system-based audit for when a login has failed.
  - **Inactive**: A system-based audit for when a job definition has become inactive.
  - **Purge**: Audit for when audits are purged.
  - **Recover**: Audit for when a recover was done for a resource. Currently, only available for holdings.
  - **Restart**: Audit for when a job instance is restarted.
  - **Set**: Audit for when the mek-alias has been changed.
  - **Start**: Audit for when declarative configuration, a job, or recovery is started (for an order item). This audit does not happen if there was an error during processing.
  - **Successful**: A system-based audit for when a user has successfully logged in or out of the system.
- **Search Events**: These events are specific to searches, as well as enabling debugging. Search events include:
  - **Aviator**: Tracks when Aviator is used on a set of search results.
  - **Enrich Search Result**: Tracks when the feature is used on a set of search results.
  - **Export Search Results**: Tracks when search results are exported.
  - **Fetch**: Tracks when the resource is fetched. Currently, only AICs, AIUs and table rows support this event.
  - **Reload**: Tracks when a saved search, is reloaded by a user because the search result has expired.
  - **Rerun**: This reruns the saved search and differs from a refresh, as the number of items in the saved search may change.
  - **Search**: Tracks when a user runs a primary or cross-application search.

Tenant provisioning events are configured at the System-level. Application provisioning events are configured at the Tenant-level.

Application audits can be enabled at tenant-level so that any new applications created will have the desired event types automatically enabled.

It is recommended that you configure system- and tenant-level audits that impact all applications first. If you want to only enable certain audits for a particular application, disable the audit at the tenant level and enable the audit for each application that you want the event enabled for.

When disabling an audit, ensure that it is not enabled at the tenant level and for each application.

2.6.2 General definition of fields

Name	Description	Notes
Audited Object	When applicable, indicates the ID of the object.	Not always set, especially for login events.
Application	Indicates the name of application.	If possible, the name of the application. Not set for most tenant audits, such as retention policies.
Event Source	Indicates the name of the resource.	Not always set, set to the user name for a login event.

2.6.3 Types

Audits are organized by Type. For each type, several events can apply. For example, almost every audit type has provisioning events (create, retrieve, update and delete).

System types

Audit Events
Tracks when an audit event has been enabled or disabled.

Compliance Metric
Tracks when compliance metrics have been exported.

Configuration
Specific to declarative configuration, this event allows configuration on when start and end is done for configuration, and also for exporting the configuration.

Cryptography Object
Represents a cryptography object. These objects can be referenced by applications.

Custom Storage
Refers to a custom storage mechanism.

RDB Data Node
Manages multiple RDB databases.

File System Root
Indicates a root location on a disk for file storage.

Group Definition
Configuration that defines groups, including which role each group is assigned to.

Job Definition
Configuration that defines a job, including scheduling information, and parameters required for the job to function.

Job Instance
Represents a future or past instance of a job.

Login
Indicates a login, which can be tracked through the gateway service. Includes operations done via the IA Shell or ANT scripts.

Logout
Indicates a logout. Typically, this will be for users of the IA Web App.

Recovery
Initiated when a recovery operation is done. This allows configuration on being able to audit when start and end is done for a recovery operation.

Storage Endpoint
Configuration representing a storage endpoint. This configuration may refer to a UR, for example the URL of an Amazon S3 Storage account.

Storage Endpoint Credential
Configuration storing credentials for a storage endpoint. This configuration stores credentials for accessing an Amazon S3 Storage account.

Storage Metric by Application
Audit for storage metric. The CREATE event must be enabled to have information for the Storage Reports in the Audit application.

This audit must be enabled for the Storage Report to work. This audit is created each time the Refresh Metrics job runs.

For the storage metric audit, the following values are set:

- `systemSegmentSize`: Current size in KB of the `mainDatabase` (`AIP`, `Content`, `Library`, `OrderItem`, and `JobInstance`), `rollForwadDatabase` (`AipCommit` and `recordsRollForwardOperation`) and `synchronizationDatabase`. Current size contains used size and free space.
- `auditSegmentSize`: Current size in KB of the `auditdatabse`.
- `retentionSegmentSize`: Current size in KB of the `managedItemDatabase`.
- `unstructuredContentCount`: Number of unstructured contents in every store (default file store, default result file store, Dell EMC CAS, ECS, *etc.*).
- `unstructuredCountSize`: Current size in KB in every store.
- `structuredDataLibraryCount`: Number of structured data (type `DATA` and `SEARCH_RESULTS`).
- `structuredDataLibrarySize`: Current size in KB of structured data (type `DATA` and `SEARCH_RESULTS`).
- `structuredDataLibraryIndexSize`: Current Index size of structured data (type `DATA` and `SEARCH_RESULTS`).
- `pricingStructuredData`: Size in Kilo Characters of data in pricing applications.

- `pricingUnstructuredContent`: Size in Kilo Characters of unstructured data in pricing applications.
- `unit`: Unit of size of the data (for example, KB).

Role Definition
Configuration that defines user roles, including the tasks available for each role.

Tenant
Configuration object that represents a customer. Tenants include applications.

RDB Database
A database in PostgreSQL, not to be confused with databases for table applications.

Tenant types

Access Node
Tracks when an access node has been created, deleted, retrieved, or updated.

AIC
Specific to a SIP-based application, stores information for doing queries (Archival Information Collection).

AIP
Specific to a SIP-based application, an AIP is a package that contains archived data. An AIP is a SIP that has been archived. AIPs contain AIUs.

AIU
Specific to a SIP-based application, an AIU is a record inside an AIP.

Application
An application is either SIP- or table-based, and stores information about that application.

Application Category
Used to categorize applications.

Audits
The storage of an audit record.

Batch item
If an order item will act on many items, batch items represent a chunk of work to do.

Bucket
Related to Dell EMC Elastic Cloud Storage, a bucket defines a location with a Dell EMC Elastic Cloud Storage store.

Configuration Helper
Specific to a SIP-based application. Used for configuring queries

Confirmation
Used to confirm various lifecycle transitions for the AIP.

Cross-Application Search
A cross-application search contains the components necessary to define a search. A search contains a result master, XForm, and query.

Custom Presentation Configuration
Custom presentation (part of a search).

Database
Representation of an archived (SQL) database. Databases contain one or more schemas.

Database Cryptography Object
Defines the crypto settings for a database (table-based application).

Delivery Channel
Defines what to do when the search is run. The results of a search are sent to a delivery channel.

Encryption Server Configuration
Not currently used.

Event
Events can be:

- Used to determine the date to start aging; and
- Updated by jobs

Export configuration
Export configuration (used by searches).

File System Folder
Represents a file system folder for storing unstructured content.

Hold
Prevents disposition and deletion of items, even if the retention policy indicates that the items qualify for disposition.

Hold Set
When applying a hold, items are put into a hold set.

Holding
Specific to a SIP-based application, a holding governs how AIPs are ingested and provides retention instructions.

Holding Composition
Used by the Holding Wizard, allows a Developer to create a holding that is destroyed once the configuration has been completed.

Holding Cryptography Object
Specific to a SIP-based application, defines cryptography settings for the holding.

Ingest
Reserved for future use. Part of a SIP application.

Ingest Node
Specific to a SIP-based application, an ingest node is part of a holding.

Library
Defines the SQL query for the search in table-based applications.

Managed item
Represents an item being managed for compliance.

Matter
Tracks when a matter has been created, updated, retrieved, or deleted. The update event is also used if it the matter is shared with others or a hold is applied to the matter (making it legal).

Order Item
Represents the processing of a background task. Order items can be seen from the Background Request page or from the job history.

PDI
Specific to a SIP-based application, a PDI is an OASIS term that stands for Preservation Description Information. Defines information for the indices for better query performance.

PDI Crypto
Specific to a SIP-based application, defines cryptography settings for the PDI.

PDI Schemas
Specific to a SIP-based application, defines the fields in the record.

Purge List
When items are eligible for disposition, a purge list groups related items into a list that can be approved. Part of the disposition process.

Receiver Nodes
Specific to a SIP-based application, a receiver node is part of a holding.

Result
A result is the stored query results. Results can be exported.

Result Master
Result master defines the columns to be returned for searches.

Retained Set
Specific to apply retention, the items that have retention applied are put into a set.

Retention Policy
Defines a policy for how long to keep items.

Rule
A rule can be used to apply retention, apply holds, or trigger events, Rules are evaluated when running the associated job.

Saved Search
Tracks when a saved search has been created, updated, retrieved, deleted, reloaded, or rerun,. The update event is also used if it was shared or had items removed

Schema
Specific to a table-based application, a schema contains a set of tables.

Search
A search contains the components necessary to define a search. A search contains a result master, XForm, and query.

Search Composition
The search set.

Search Debug
Configuration for enabling debugging settings for Search.

Search Group
Part of search.

Search Mapping
Tracks when a search's mapping has been created, deleted, retrieved, or updated.

Search Result Query
Tracks when a search query has been created, deleted, retrieved, or updated.

Space
Represents all storage configured for use by an application. This includes structured data for table applications in RDB and unstructured data stored on file systems or cloud storage, such as Dell EMC Elastic Cloud Storage and Amazon S3 Storage.

Space Root Folder
A space root folder ties a file system root and space

Space Root Object
Part of a space that defines the credentials and storage information.

Space Root RDB Database
Defines the library for storing structured content for table applications.

Store
Defines a location to store unstructured content.

Table
Specific to a table-based application, a table is grouped within a schema.

Table Row
Specific to a table-based application, a record within a table.

Transformation
Part of a SIP application, allows you to convert from one schema to another using XSLT.

Value Transformation
Tracks when the `ValueTransformations` information in the `application-config > searches > configuration.yml` file has been updated.

Library Policy
Part of a SIP application, defines a library policy specific to storing your application.

XForm
An XForm is part of search that defines the specified fields to narrow a search.

2.6.4 Crypto (Cryptography) keystore audits

This section discusses keystore audits related to encryption. For more information, see Section 2 "Managing cryptographic keystores" in *OpenText Information Archive - Encryption Guide (EARCORE-AGE)*.

The following two audits cannot be disabled:

- Crypto_keystore migrate, with three supplemental fields:
  - From: Which strategy was previously in use
  - To: Which strategy was migrated to
  - Label: The value defined in the `application.yml` file to avoid accidentally importing the keystore into the database twice

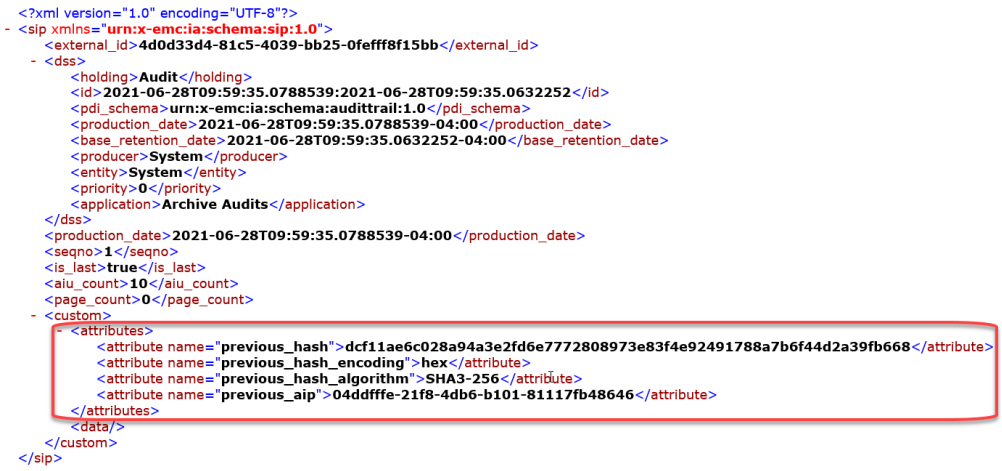
2.6.5 Chaining audits

Chaining audits is a mechanism that allows you to determine if any audits have been modified or removed. It is important to note that this feature will not provide much benefit if disposition occurs often on the Audit application.

Note: The ability to chain audits is disabled by default, but can be enabled using the `audit.chaining.enabled` property of the **Global Settings** tab.

Audits of a specific type can be chained (for example, system audits can be chained, tenant audits can be chained, and audits tied to a specific application can be chained).

When actions are applied to an AIP and audits occur, the SIP descriptor for the AIP is updated with the previous hash name, previous hash encoding, previous hash algorithm, as well as the previous name of the AIP (see the circled `<attributes>` section from a SIP descriptor file below):



These changes are also displayed in the **Metadata** tab on the right side of the screen.

The following is an example of the downloaded PDI file. Note for readability in this documentation, the XML has been formatted, but the actual XML will not have this format:

```
<previousAip>04ddfffe-21f8-4db6-b101-81117fb48646</previousAip>
<previousAipPdiHash algorithm="SHA3-256"
     encoding="hex">dcf11ae6c028a94a3e2fd6e7772808973e83f4e92491788a7b6f44d2a39fb668</
previousAipPdiHash>
```

Furthermore, each attachment in an AIP has its own encoding and hash algorithm. If any changes have been made to the attachment, the changes are logged in the AIP's PDI file.

For the Audit application, you can amend the information that appears in search results. To update, access the **Holdings** tab for the Audit application, select **Edit Ingestion > Metadata**. From here you can update, delete or add the names, labels, and types of fields related to chained audits.

2.6.6 Auditing application retention

You have some choices when auditing an application's retention.

Use the following default settings, which create audits:

- Against the retention policy when a policy is applied or removed, and
- Against the retained set when items are added or removed.

Event name	Event type
Apply	Retention Policy
Remove	Retention Policy
Add Items	Retained Set

Event name	Event type
Remove Items	Retained Set

You can also enable an audit for the AIP or table when retention is applied or removed.

Event name	Event type
Apply Retention	AIP
Remove Retention	AIP
Apply Retention	Table
Remove Retention	Table

These audits are not enabled by default and can be enabled for all applications or for specific applications. There is no audit per record when applying retention.

2.6.7 Auditing when items are added or removed from a hold set or retention is applied to multiple AIPs

The following two event names audit when items are added or removed from a hold set or when retention is applied to multiple AIPs in a single operation:

- `add_items`
- `remove_items`

The `add_items` audit is triggered in the following scenarios:

- When retention is applied to multiple AIPs in a single operation.
- When a hold is applied to a package.
- When a hold is applied to a saved search.
- When a hold is applied to a saved search, more data is subsequently ingested, and the saved search is rerun, and more records are added to the hold set.

The `remove_items` audit is triggered in the following scenarios:

- When retention is removed from multiple AIPs in a single operation.
- When a hold is removed from items in an AIP. Removing all items in an AIP triggers the `delete` audit.
- When a hold is removed from records in a saved search. Removing all items in a saved search triggers the `delete` audit.

2.6.8 Confirmation audit attachments

Attachments is a section in a confirmation audit that provides access to a downloadable file of the content of the confirmation. Attachments are generated when **Audit** is selected as the confirmation store in a holding's **Confirmation** settings.

There are three use cases related to confirmation audit attachments: ingestion-related confirmations (Storage, Receipt, Available), disposition-related confirmations (Purge), and invalidation-related confirmations (Invalidation, Rejection).

Steps to generate a Storage confirmation attachment for a Phone Calls application holding:

1. Ingest packages (refer to Setting up ingestion for a SIP application in the Configuration Guide).
2. In the Phone Calls application's **Holdings** tab, click the down arrow next to the **Holding Name** and then click **Edit Confirmation**.
3. In the **Edit Confirmation** dialog, select **Storage**, select **Audit**, and then click **Save**.
4. Run the Confirmation job.
5. Run the Archive Audits job.
6. Run the Audit application's **Application Audit** search, selecting **Phone Calls** as the **Application**, **AIP** as the **Event Type** and **Confirmation** as the **Event Name**.
7. Click the arrow next to the **Application Name** and scroll down to **Attachments**.
8. To download and view the content of the confirmation, click the **Download** link.

Steps to generate a Purge confirmation attachment for a Phone Calls application holding:

1. Ingest packages (refer to Setting up ingestion for a SIP application in the Configuration Guide).
2. In the Phone Calls application's **Holdings** tab, click the down arrow next to the **Holding Name** and then click **Edit Confirmation**.
3. In the **Edit Confirmation** dialog, select **Purge** and **Audit**.
4. Select the **Advanced Settings** tab and make sure the **AIU Query** is specified (default for the Phone Calls application).
5. Click **Save**.
6. Run the Generate Purge Candidate List job.
7. Go to the Phone Calls **Applications > Purge Lists** tab and approve the purge list.
8. Run the Dispose Purge Candidate List job scoped to the Phone Calls application.
9. Run the Archive Audits job.

10. Run the Audit application's **Application Audit** search, selecting **Phone Calls** as the **Application**, **AIP** as the **Event Type** and **Confirmation** as the **Event Name**.
11. Click the arrow next to the **Application Name** and scroll down to **Attachments**.
12. To download and view the content of the confirmation, click the **Download** link.

Steps to generate an Invalidation confirmation attachment for a Phone Calls application holding:

1. Ingest packages (refer to Setting up ingestion for a SIP application in the Configuration Guide).
2. In the Phone Calls application's **Holdings** tab, click the down arrow next to the **Holding Name** and then click **Edit Confirmation**.
3. In the **Edit Confirmation** dialog, select **Invalidation**, select **Audit**, and then click **Save**.
4. In the **Applications > Packages** tab, click a **Package Name**, click the down arrow and select **Invalidate Package**.
5. Run the Invalidation job scoped to the Phone Calls application.
6. Run the Confirmation job.
7. Run the Archive Audits job.
8. Run the Audit application's **Application Audit** search, selecting **Phone Calls** as the **Application**, **AIP** as the **Event Type** and **Confirmation** as the **Event Name**.
9. Click the arrow next to the **Application Name** and scroll down to **Attachments**.
10. To download and view the content of the confirmation, click the **Download** link.

2.6.9 Default enabled event types

Refer to the **Administration > Audit** tab to see which event types are enabled by default. It is recommended that you review all audits to determine which ones are to be enabled.

2.6.10 Auditing for storage system audits

The following table indicates what to expect for audits. The following event types are not used any more:

- OpenText Core Archive Endpoint
- OpenText Archive Center Endpoint Credential
- Dell EMC Elastic Cloud Storage Endpoint
- Dell EMC Elastic Cloud Storage Endpoint Credential

Storage	Audit event types	Supplemental data field
OpenText Core Archive	Storage Endpoint Storage Endpoint Credential	`type`, `description`, `hostname`, `port`, `ssl`
OpenText Archive Center	Storage Endpoint Storage Endpoint Credential	`type`, `description`, `hostname`, `port`, `ssl`
Dell EMC CAS Elastic Cloud Storage	Storage Endpoint Storage Endpoint Credential	`type`, `description`, `connectionString`, `pea.variable` (for each variable defined)
Custom Storage	Custom Storage	`description`, `factoryServiceName`, `properties` (for each variable defined)
Azure	Storage Endpoint Storage Endpoint Credential	`type`, `description`, `url`, `proxyUrl` `storageType`, `description`, `proxyUserName`, `accessKey`
Dell EMC Elastic Cloud Storage	Storage Endpoint Storage Endpoint Credential	`type`, `description`, `url`, `proxyUrl`
PowerScale	File System Root	`type`, `description`, `path`
Local File System	File System Root	`type`, `description`, `path`
Amazon S3 Storage	Storage Endpoint	`type`, `description`, `url`, `proxyUrl`
	Storage Endpoint Credential	`storageType`, `description`, `proxyUserName`, `accessKey`

2.6.11 How primary and cross-application searches are audited

In previous releases, primary searches were linked directly to an AIC, query, or table configuration, which was how the system tracked and audited such actions as executing searches or exporting search results. Cross-application searches, however, are not linked directly to an AIC, query, or table configuration.

To enable the system to better audit primary and cross-application searches, the following changes were made:

- An event type called Cross-Application Search has been added to Search Events audit at the tenant and application levels.
- The system audits when a user exports search results via the Search event type. This was previously declared at the AIC/Query level.

- The system audits when a user runs a search via the Search event type, the Cross-Application Search event type (or via the Search Result Query type). This was previously declared at the AIC/Query level.
- The system audits when a resource is fetched via the Search event type or via the CrossApplication Search event type. This was previously declared at the AIC/table row level.

2.6.12 Auditing for sharing saved searches

The Audit application allows you to track when a group has been granted permission to read or edit a saved search. Conversely, it also tracks when permissions have been removed for a particular group.

The supplemental fields include:

- `groupsGrantedForRead`
- `groupsGrantedForEdit`
- `groupsRemovedForRead`
- `groupsRemovedForEdit`

If a group is granted editing access to a saved search, the group is given both the `groupsGrantedForRead` and `groupsGrantedForEdit` privileges.

Use the Application Audit search to track when group permissions have been added or removed from a saved search.

2.7 Language support

2.7.1 The Language Pack

In past releases, the Language Pack had to be manually installed. Now, the Language Pack is bundled in the `infoarchive.zip` artifact in the following location: `<IA_ROOT>/lib/iawebapp/external`.

To view the language preferences that are available:

1. Log into the IA Web App.
2. Click your username in the top-right corner of the screen and select **Preferences**.
3. Under Locale, click **Language** to view the list of available languages.

2.7.2 Enabling support for changing the date and number localization of the user interface

When an Administrator enables support for changing the date and number localization of the IA Web App user interface, users can change the language settings, including the display language and the format of dates, times, and numbers. For more information, see Section 1.2.1 “Configuring locale preferences” in *OpenText Information Archive - End User Guide (EARCORE-UGD)*.

On new installations of OpenText Information Archive, this support is enabled by default. For an upgraded system, the previous setting will be used.

When this support is enabled, the number display format contains a thousands separator. For example, in English, the thousands separator is a comma: 12,345. If any columns, side panels, or inline panels in search results are set to the NUMBER data type, then by default they will show the thousands separator. However, this might not be appropriate. For example, a year column with the thousands separator enabled would show the years 2,020 and 1,976 instead of 2020 and 1976.

To change the localization support setting for language and format of the user interface:

1. In a text editor, open the `<IA_ROOT>/config/iawebapp/application.yml` file.
2. In the `infoarchive.webapp` section, set the `enableDateAndNumberLocalization` parameter to the desired value:
   ```
   enableDateAndNumberLocalization: true
   ```
3. Restart IA Web App.

   Note: A hard reset for each browser is required for the changes to take effect.

2.7.3 Adding a custom language translation to IA Web App

In addition to the languages that you can add to IA Web App using the language pack, you can add a custom language translation.

Language support is dynamic:

1. The languages that a user can choose from are decided at runtime, based on the presence of supporting language files.
2. You can add more languages by following the steps below.

A custom language uses the US format for dates, times, and numbers.

To add a custom language translation:

1. Translate the strings in the `en.json` file into the language that you want. You can find this file in the `<IA_ROOT>/lib/iawebapp/infoarchive-webapp.jar` file, in the JAR file's `WEB-INF/classes/static/languages` directory.
2. Create a directory named `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization/languages`.
3. Create a file named `languages.json` in the `languages` directory that you created in the previous step.
4. Edit the `languages.json` file and add the following content in the file:

   ```
   {
       "<LANGUAGE_CODE>": "<LANGUAGE>"
   }
   ```

   For example:

   ```
   {
       "cs": "Czech"
   }
   ```

5. Move the `<LANGUAGE_CODE>.json` file (for example, `cs.json`) into the `languages` directory.
6. Copy the `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization/languages` directory into `<IA_ROOT>/config/iawebapp/customization` directory.
7. Restart IA Web App.
8. To verify the installation of the custom language, do the following:
   a. Log into IA Web App
   b. In the top-right corner of the page, click your user name, and then select **Language Settings**.
   c. In the **Language** drop-down list, verify that the language appears.

For more information about the customization location, see Setting the customization location when deploying to Tomcat or other containers on page 72.

2.8 Custom branding

OpenText Information Archive supports limited, drop-in branding customization. Customers can define the view and display of the font, color, images and styling of IA Web App.

A sample branding customization is provided to the customer in the following directory: `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization/branding`.

To review the sample branding, copy the `customization/branding` folder and paste it into the following directory: `<IA_ROOT>/config/iawebapp/customization`.

To change the custom branding, copy the `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization` folder to the `<IA_ROOT>/config/iawebapp` folder. Start the IA Web App and copy and replace the image file to the folder, but do not change the name of the image file.

The image file and the one it is replacing need to be the same dimensions (width and height) in a PNG, GIF or JPG format.

To verify the web customization:

1. Open IA Web App in an Internet browser.
2. Refresh and reload the web page.
3. The customized branding style should be displayed.

If the new customization is not displayed, clear the browser cache.

To replace the styling and CSS rule of IA Web App:

1. Open the following directory in Windows Explorer: `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization/branding/css`.
2. Review the information about current CSS rules in the README file.
3. Open the `custom.css` file with a text editor and edit the styling rules using CSS syntax.

The above steps only work if the IA Web App is started using the `iawebapp` script that runs as a standalone Spring Boot application. For more information about this script, see Section 3.2 “Starting the components with a command prompt” in *OpenText Information Archive - Installation Guide (EARCORE-IGD)*.

If the IA Web App is deployed to an external Tomcat, the customization directory needs to be copied under the external Tomcat directory. For example, `<EXTERNAL_TOMCAT_ROOT>/webapps/infoarchive-webapp/config/iawebapp`, where `<EXTERNAL_TOMCAT_ROOT>` could be `C:\apache-tomcat-XX.X.XX` (where X is the version number). For more information, see Section 13.9 “Deploying IA Web App to Apache Tomcat” in *OpenText Information Archive - Installation Guide (EARCORE-IGD)*.

Use one of the following methods to configure Tomcat to find the directory:

Option 1

Set the customization location by editing the `application.yml` file located at: `<EXTERNAL_TOMCAT_ROOT>\webapps\infoarchive-webapp\WEB-INF\classes\application.yml`. Change the key:

```
webapp:
   customization:
      location: ”file:///C:/apache-tomcat-XX.X.XX/webapps/infoarchive-webapp/
config/iawebapp/customization/”
```

Option 2

Create a `web.xml` file at `C:\apache-tomcat\webapps\infoarchive-webapp\WEB-INF\web.xml`. To learn which versions of Apache Tomcat are compatible with OpenText Information Archive, see the *OpenText Information Archive Release Notes* on support.opentext.com. Set the contents to:

```
<web-app version="3.0"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">
  <context-param>
    <description>Information Archive Customization location</description>
    <param-name>infoarchive.webapp.customization.location</param-name>
    <param-value>file:///C:/apache-tomcat-XX.X.XX/webapps/infoarchive-
    webapp/config/iawebapp/customization/</param-value>
  </context-param>
</web-app>
```

The `param-name` value must match the location you copied the customization folder.

2.9 Adding custom logos

OpenText Information Archive uses Scalable Vector Graphics (SVG) logos:

- `opentext-infoarchive-logo.png` is replaced by `opentext-infoarchive-logo.svg`.
- `opentext-logo.png` is replaced by `opentext-logo.svg`.

This update affects branding customization. In the past, the above two logos could be customized by dropping customized PNG logos in the following `<IA_ROOT>/config/iawebapp/customization/branding/images` folder.

Currently, customers have the following choices:

- Create SVG logos and overwrite the following files:
  - `<IA_ROOT>/config/iawebapp/customization/branding/images/opentext-infoarchive-logo.svg`
  - `<IA_ROOT>/config/iawebapp/customization/branding/images/opentext-logo.svg`

This is strongly recommended. However, this requires the creation of new SVG logos.

- If you want to continue to use the PNG logos, add the following style rules to the `<IA_ROOT>/config/iawebapp/customization/branding/css/custom.css` file:

```
#ia-gloia-top-navbarbal-home > img

{ content: url('/images/opentext-infoarchive-logo.png'); }
.ia-about img

{ content: url('/images/opentext-logo.png'); }
.ia-splash img

{ content: url('/images/opentext-infoarchive-logo.png'); }
```

Customers may have to tweak width and height properties to get the logos sized properly. For more information, refer to the `README` file in the `<IA_ROOT>/first-time-setup/applications/Tenant/config/customization/branding/css` folder.

Chapter 3

Administering OpenText Information Archive

This chapter outlines the tasks that a user with the Administrator role or the Retention Manager role can perform.

As an Administrator, you can use the **Administration** tab in IA Web App to register data nodes, create databases, and add storage systems. Jobs can be run and edited by the Administrator. Furthermore, the Administrator can map groups to user roles via the Administration tab of the IA Web App.

As a Retention Manager you can use the **Applications Packages** tab to view a list of AIPs and perform related tasks.

3.1 User context

Many administration tasks will be performed using the OpenText Information Archive UI, which allows you to navigate, search, select, and edit administration objects, and invoke related administration functionality.

Before we start listing and explaining these functions and objects, here is a note on the UI's user context.

OpenText Information Archive remembers your selections for the secondary navigation toolbar, any filter values you have entered, as well as any column filter values you have entered. For example, you access the **Compliance > Holds** tab, set some filters and navigate to the Applications tab. When you return to the **Holds** tab, the filter values are preserved.

There is no way to deactivate this functionality. You can, however, click the following button to clear any filter information you have entered:

3.2 Storage abstraction: Configuring stores for an application

A previous section discussed how to set up storage. For an application to use this storage, there are two steps:

1. The Developer associates the storage with the space.
2. The Developer creates a store using the required type of storage.

Tip: Although it is possible to create multiple spaces for an application, it is simpler to just associate additional storage with one space.

3.2.1 Adding a store using the IA Web App

A store represents a place where you can store your data. Typically this maps to a File System Folder or Bucket. Stores are used for:

- Structured data for SIP applications
- Search results
- Granular retention
- Unstructured content
- Miscellaneous use

OpenText Information Archive recommends creating separate stores based on the intended use. Some types of data are restricted to what storage can be used. For example, cloud storage cannot be used for Structured data for SIP applications.

An application's **Stores** tab allows the Administrator to manage their stores. User's with the IT Owner role can access an application's **Stores** tab and view the information, regardless of the permissions assigned to the application. The IT Owner cannot, however, make any changes to the information.

Stores can be accessed from the application's **Stores** tab. Note that some store types require creating the storage in the other system. For example, Archive Center requires that the bucket be created first.

Existing stores are displayed in a table that contains the following information:

Store Name	Indicates the name of the store.

	Click to perform one of the following actions on the selected store: • **Edit**: Make changes to the selected store. • **Test Connection**: Once the **Test Connection** button is pressed, OpenText Information Archive tries to establish a connection with the bucket provided in the store. The system will inform you if the connection could be made or not. See Setting up storage for unstructured data to learn which storage types support this feature. If the connection is not successfully established, the error message indicates the reason why the connection failed. Make the necessary changes to the fields indicated in the error message and click the **Test Connection** button again. • **Offline**: Take the selected store offline. • **Online** – Read Only: Put the selected store online.
Space Name	Indicates the space that is connected to the store.
Status	Indicates whether the store is: • Online • Online – Read Only • Offline
Storage Type	Indicates the type of storage being used.
Type	Indicates whether the type is: • Regular • Result • Confirmation • Search Result • Library
In Use	Indicates whether the store is currently being used. If a store is not in use, user can edit or delete it. If a store is in use, user can only edit it.
Side panel	Indicates the details of the selected store.
Storage Class	Select the storage class when writing content into the store. The storage class set at the store will override the default storage class defined for the bucket. This field is only displayed for Google Cloud Storage.

1. Click '+' to add a store.
2. Enter the following information:

Store Name
Enter a name for the store being created.

Type
Select one of the store types:

- Confirmation
- Regular
- Export
- Search Result
- Library

Space
Select the space.

Space Root
Select the space root, which indicates the type of storage you want to use. Each type of storage has a distinct root.

Note: A Regular store type should be based on an object store not a file store.

File System Folder / Bucket
Select from the available values in the dropdown list.

You can also create a new file system folder or bucket. If adding a new file system, enter a name of the new object and click **Create**.

Follow these rules to create a Domain Name System (*DNS*)-compliant bucket name:

- Bucket names must be at least three and no more than 63 characters long.
- Bucket names must be a series of one or more labels. Adjacent labels are separated by a single period (.). Bucket names can contain lowercase letters, numbers and hyphens. Each label must start and end with a lowercase letter or a number.
- Bucket names must not be formatted as an IP address (for example, 192.168.5.4).

Status
Indicate the store status.

Do not push retention at the hardware level
Select this option if retention is to be managed at the software level. The storage device will not be aware of the retention level. The configuration impacts all unstructured data contained in the object storage. This field will only be displayed if the storage type supports pushing retention to the hardware. By default, this field is turned off. Furthermore, this functionality is only active going forward. In other words, if retention was pushed to the hardware in the past, activating this functionality will not change that fact.

3. Click **Create**.

3.2.2 Store configuration for Core Archive

When adding a store using the IA Web App, use the same logical archive that is pre-configured in Core Archive Business Administrator.

Prior to ingesting data for the first time:

1. Log into the Core Archive Connector Business Administrator and enable the data source.
2. Once selected, you should be able to view and accept the certificate.

3.3 SIP management

3.3.1 Using the Packages tab

The **Packages** tab allows Administrators and Retention Managers to perform the following actions:

- View a list of the available AIPs in an application
- Rebuild a SIP
- Reject or invalidate an AIP and rollback invalidation
- Apply a retention policy or hold to one or more AIPs. If you select a single AIP that does not have a commit date, the menu option to apply or remove retention is not displayed.

  If you select multiple AIPs and try to apply retention, but one of the packages does not have commit date, the background task will warn that retention was not applied to some packages. The IA Server logs will include details about each AIP with its name and externalId.

  You cannot apply retention or a hold to AIPs that could not be received (for example, because of a parsing error in the SIP descriptor). If a package is invalidated, however, it is possible to apply retention to the package, as it will have a commit date.
- Offline/Online
- Request closing of pooled library
- See the audits
- Display records

Each package is displayed in a table. The type of package is indicated by the following icons:

	An open aggregate type.

	A closed aggregate type.
	A standard AIP type.
	An open pooled library.
	Indicates a closed pooled library.

The table also contains the following information:

Name	The name of the AIP. Click the link to view the details of the package. If the package is an aggregate of multiple AIPs, click the link to view the AIPs that comprise the aggregate.
	A menu that allows a user to perform the following actions on a selected AIP: • Apply retention • Apply hold • Reject package • Invalidate package • Rebuild SIP • Ingest • Request the closing of the pooled library • Offline/Online • See the Audits • Display Records The actions displayed in the menu depend on your user role as well as the State of the package. For more information, see Applying actions to a package.
Phase	Displays the current phase of the AIP: • Reception • Waiting Ingestion • Ingestion • Waiting Commit • Completed • Reject • Invalid • Purge • Aggregate
Holding	The name of the holding the SIP was ingested into.

Reception Start Date	The reception date of the SIP.
Online	Indicates if the SIP is cached in or out.
Records	The number of records in the AIP. **Note:** There may be a discrepancy between the number of records listed on the **Packages** tab and the **Records Report**. The Packages tab does not count records in an open aggregate until it is closed. Conversely, the Records Report counts records in an open aggregate.
Return Code	Indicates the phase the AIP has reached: • OK: The AIP has been successfully processed. • Error: There is an error with the AIP.

View additional information by clicking on a package name. The **Package content** page contains the custom properties of the selected AIP, package content (`sip.xml`, logs files, `ci.container` file, etc.). The Retention Manager can see the **Storage Retention Date** for the selected package.

If the store of the content has content offline capacity, such as Glacier, then some content will not be available. A restore button will be displayed in order to restore the content. This may take time, depending on the setting of the store.

The Retention Manager also has the ability to select one, multiple or all packages in an application. A box appears in the column on the left side of the page for each package:

- Click Select all # to select all the packages listed in an application. Change the filter to not restrict to the last 7 days. The filter is used if you wish to only remove retention from a subset of the application.
- To remove retention from only the packages listed on the page, click the empty box in the column header to select all the packages that appear in the current page.
- Select a specific package's box or multiple boxes to only act on some of the packages listed.

Retention policies or holds can be removed from selected packages. As long as one package is selected, both functions (remove retention policy and remove hold) are enabled.

A panel on the right side of the page contains the custom properties of a selected package. The following tabs appear in the panel:

Summary
Contains general information about the selected package.

SIP
Contains information about the selected package's holding, DSS ID, PDI Schema, etc.

Metadata
Contains the metadata information for the selected package, including any customized data added to a package, as well as information related to partition keys.

Tip: The partition key information can be used to determine what data is included in the selected package. For example, if the **Metadata** tab for the selected package lists the following:

```
pkeys.dateTime01     Apr 21, 2011 12:44:27 AM
pkeys.dateTime02     Mar 3, 2015 12:19:42 PM
```

This indicates that the selected package only contains data between April 21, 2011 to March 3, 2015 for the times indicated.

Library
Contains the library information for the selected package, including its mode of ingestion, the Library PDI Schema, Library Name, etc.

Tracking
Contains information about the selected package that helps you track the various stages of reception, ingestion, when the package will or was committed, its disposition date, etc.

Confirmation
Contains information about confirmation options for the selected package.

Cyphering
Contains encryption information about the selected package.

Permissions
Indicates which user groups can access the data of the selected package in a search.

Compliance
Indicates whether a retention policy or hold has been applied to the AIP or any of its records.

3.3.1.1 Testing ingestion from the Packages tab

Caution

The **Packages** tab for a selected application allows you to ingest one or more SIPs. This feature is meant to test your system's configuration in a demo environment. In a production environment, however, you should continue to use the existing method of ingestion that is documented in Section 2.7.5 "ingest" in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*.

This feature allows you to ingest ten files, each with a maximum size of 2.048MB (maximum size of all 10 files would be 2GB). You can, however, configure the maximum size of all files via the **Global Settings** tab property `iawa.packages.ingestion.max.size`.

To test ingestion using the Packages tab:

1. In the Packages tab for the selected application, click the + icon.
2. Click the **Choose** button to select the package(s) or, alternately, drag the package into the field provided. Each package must contain the necessary manifest information.
3. Click **Ingest**.

   OpenText Information Archive processes ingestion requests made from the *Packages* tab asynchronously. Navigate to the **Background Requests** tab to see the results.

3.3.2 Using the Libraries tab

The **Libraries** tab shows the private and pooled libraries of the selected SIP application.

Each library is displayed in a table that indicates:

	An open pooled library.
	A closed pooled library.
	A private library.

The table also contains the following information:

Name	The name of the library. Click the link to view the packages in the library.

	A menu that allows the following user roles to perform specific actions for the selected library: • **Online/Offline**: Used by the Administrator to take libraries online and offline. • **Request the closing of the pooled library**: Used by the Business Owner to create a request to close pooled (open) libraries. • **See the Audits**: Used by the Business Owner and Developer to view the audits on the selected library. • **Post-Ingest**: Used by the Administrator to generate a background request to perform post-ingestion steps for a library associated with a package. For example, use this feature if you have to re-index a library because of a changed definition on a holding or enable text extraction. The request updates indexes, partition keys and sets the cache lock date. For more information on the three steps, see Post Ingest Processing job. The actions displayed in the menu depend on your user role and the state of the package. For more information, see Applying actions to a package.
Partition Key	The partition key created during ingestion and stored in the object.
Codec	The codec API used for the library.
Packages	The number of AIPs in the library.
Size	The size of the library.
Online	Indicates if the library is online.
Creation Date	The date and time the library was created.
Closed Date	The date and time the library was closed.

View additional information in the side panel by selecting a library in the table:

Library Mode (Type)
Private: Each SIP gets a dedicated library.

Pooled: Each SIP gets a dedicated library before being combined into one library after close.

Aggregated: Each SIP gets a dedicated library before being aggregated into one package with a dedicated library after close.

Library Name
The name of the library.

Sub Path
Location on the file system where the library is stored.

Library Size
The size of the library.

Online
Indicates if the library is online.

Stored AIP Count
The number of AIPs in the library. This is set to 1 when the library is **Private**.

Stored AIU Count
The number of records stored in the library.

Closed
Indicates if the library is closed.

Closed Date
The date and time the library was closed.

Closing Manually Requested
Indicates if the library was requested to be closed. When the value is true, the library is immediately eligible to be closed even if the eligible date or quota has been reached.

CI Text
Indicates if unstructured contents have been indexed.

CI Text Count
Indicates how many unstructured contents have been indexed.

CI Text Blank Count
Indicates how many unstructured contents have not been indexed due to a blank value.

CI Text Error Count
Indicates how many unstructured contents have not been indexed due to an error during extraction.

CI Text Char Count
Total number of indexed characters of unstructured contents.

View associated package information by clicking a library's name in the table. Different user roles can perform actions on the selected library's packages. Refer to Using the Packages tab for more information.

View additional information by clicking a package's name. The **Package content** page contains the custom properties of the selected AIP, package content (`sip.xml`, logs files, `ci.container` file, etc.). The Retention Manager can see the **Storage Retention Date** for the selected package.

3.3.3 Finding AIPs

The **Packages** tab allows you to find an AIP using the following filters displayed above the AIP list:

- **Find by ID**: The AIP's external ID
- **Phase**: The phase of the AIP's lifecycle
- **Packages**: Packages with errors or without errors
- **Reception**: When the package was received
- All filters specified go into effect immediately after choosing one.

To find a specific AIP:

1. Enter the AIP's external ID and press ENTER. This must be an exact match otherwise no packages will match the filter. Putting a value in this field disables the other filters.
2. To filter the list of AIPs:
   - Select a **Phase**: **Reception**, **Waiting ingestion**, **Ingestion**, **Waiting commit**, **Completed**, **Purge**, **Prune**, **Rejected**, **Invalidated**, **Aggregate**. The default is **All**.
3. Select Packages: Whether the AIP is with or without errors. The default is **All**.
4. Select when the AIP was received. **All**, **Today**, **Last 2 days**, **Last 7 days**, **Last 30 days**, **Custom**. To enter a date range, select **Custom**. The default is **Last 7 days**.

To reset all filters to the default:

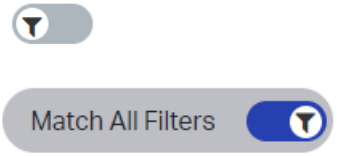
- Click the **Reset All Filters** icon to the right of the filters.

3.3.4 Applying column-based filters to the package list

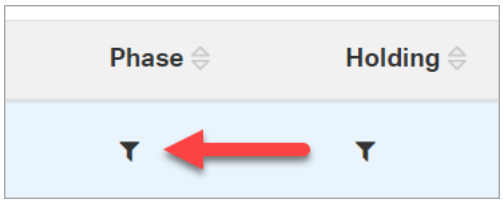
In the **Packages** tab, you can apply one or more column-based filters to the list of AIPs to narrow results.

To apply column-based filters to the package list:

1. Turn the switch on.



An additional row under the list header is displayed that includes at least one filter icon. The icon indicates that a column can be filtered.



2. Click the filter icon or anywhere in the cell in the list header.
3. In the popup window, select the operator to apply to the filter. These operators are available based on the data type associated with the column. For example, the **Phase** and **Holding** columns have the **Exact Match** operator available while the **Reception Start Date** column has the **Between** operator.

   Enter the filter criteria and click **GO** or press **ENTER**.
4. Repeat steps 2 and 3 if you want to apply multiple filters to the list.

Once a filter is applied, you can remove it by clicking **X**.

3.3.5 Rebuilding a SIP

When rebuilding a SIP, the export allows the Administrator and Developer to import this SIP into a different archive. The Business Owner can also rebuild a SIP but cannot download the rebuilt package.

To rebuild a SIP:

1. Select the application that contains the AIP to be exported and access the **Packages** tab.
2. Click the action button and **Rebuild SIP**.

   The request runs as a background request.
3. On the **Background Request** tab, once the request is completed, click **Download** to access the content of the exported SIP.

3.3.6 Rebuilding a library

Use the procedure in this section to rebuild an entire Lucene index library from the PDI XML documents stored in the corresponding packages.

Caution

Use the feature outlined in this section only when you experience issues:

- Running searches against the SIP
- Running the cache-in command against the SIP

- Running the Post Ingest Processing job against the SIP

Only if the Post Ingest Processing job fails, use this feature as a last resort before invalidating and re-ingesting the SIP.

You can only rebuild a library if the **Library Mode** field is set to Private or Pooled.

Rebuilding the library is not available for AIPs altered by granular disposition.

Rebuilding the library is performed through IA Shell. For more information, see the section for the `rebuild-library` command in the *OpenText Information Archive IA Shell Guide*.

To rebuild a library:

1. Connect to IA Shell and go to the library being rebuilt:

```
iashell> cd applications/PhoneCalls
iashell> cd libraries
iashell> cat
+-----------------------------------+--------+--------+------+--------+
|name                               |aipCount|aiuCount|closed|detached|
+-----------------------------------+--------+--------+------+--------+
|053849e1-925c-4af6-bf06-dda781b1f23c|1       |3       |true  |false   |
|06156b4b-0d7b-42c8-9c0e-5008ac75b197|1       |3       |true  |false   |
|08ddff12-2655-4d34-9c55-18f35cc8b8b1|1       |3       |true  |false   |
|1d3dbd3a-3e9b-4091-b34c-8277dc7f06f2|1       |10      |true  |true    |
|202e56a8-0e05-41fd-9d22-43a1224b7bfe|1       |10      |true  |false   |
|421070a1-1130-4168-95a8-a61be7495b8a|1       |3       |true  |false   |
|6974e58e-4115-41fd-b05d-9c0fa3bfec26|1       |3       |true  |false   |
|82212ee3-3f6c-4147-ad84-8bf0b4837834|1       |3       |true  |false   |
|a3dce469-ceb0-40d3-abcf-428b5fea3178|1       |3       |true  |false   |
|c9ff6ca3-61c7-4264-a437-062013e689ea|1       |4       |true  |false   |
|                                   |        |        |      |        |
|-------------------------------    |        |        |      |        |
|Elements count: 12. Page 1 of 2    |        |        |      |        |
|next ->                            |        |        |      |        |
+-----------------------------------+--------+--------+------+--------+
iashell> cd 053849e1-925c-4af6-bf06-dda781b1f23c
```

2. Run the `rebuild-library` command:

```
iashell> rebuild-library
Rebuild of library been submitted as background process.
Order Item: 053849e1-925c-4af6-bf06-dda781b1f23c_2025-02-13T17:25:08+01:00
OK
```

   If you use the `--overwrite-backup` option, the library backup generated by the rebuild overwrites the original backup. By default, the option is disabled and the system keeps the original backup as an archived format while the new backup takes its place for the backup format.

   IA Shell responds with the message, “Rebuild of library been submitted as background process” and provides the ID of the corresponding order item.

3. Go to the **Background Request** page and wait for the request to complete.

4. Once the rebuild is successful, if unstructured content full text indexation was enabled on the library, run the Post Ingest Processing job for the extraction done on the library.

3.3.7 Applying actions to a package

The actions that can be applied to an AIP depend on the State of the AIP:

Apply Retention
Displayed if the user can apply retention policies and the phase is not Aggregate and the package is not part of an aggregate. Retention cannot be applied to an aggregate until it is closed.

Apply Hold
Displayed if the user can apply holds and the phase is not Aggregate and the package is not part of an aggregate.

Reject Package
Displayed for the following phases:

- Waiting ingestion; or
- Waiting commit; or
- Completed; and

the sequence number in the DSS is null or not 1 or the package is not last.

Invalidate Package
Displayed for the following phases:

- Waiting ingestion; or
- Waiting commit; or
- Completed; or
- Reception; or
- Ingestion; and

the Return Code is With Errors for Reception and Ingestion phases (for the remaining phases, the Return Code does not matter).

Online
Displayed for the following phases:

- Waiting commit; or
- Completed; and

the package is not part of an aggregate and the library is closed, has cache support, and is detached.

Offline
Displayed for the following phases:

- Waiting commit; or
- Completed; and

the package is not part of an aggregate and the library is closed, has cache support, and is not detached.

Request the closing of the pooled library

Displayed if the ingestion mode is Pooled and the library:

- Is not closed; and
- Has no close request associated with it; and
- Has an effective close date set in the past.

Request the closing of the aggregate

Displayed if the ingestion mode is Aggregate and the library:

- Is not closed; and
- Has no close request associated with it; and
- Has an effective close date set in the past.

Rebuild SIP

Displayed if the phase is Completed and the package is not part of aggregate. This action restores the original SIP that was submitted. This action may fail if the PDI XML or SIP XML was not stored in the system or if the disposition policy is configured to delete the original XML files during disposition.

Ingest

Displayed for the following phases:

- Waiting ingestion; or
- Waiting commit; or
- Ingestion; and

the Return Code is without errors for the Ingestion phase (for the remaining phases, the Return Code does not matter).

See the Audits

Use this feature to run a search for audits for the selected AIP. For more information, see Seeing the audits.

Mark as Error

Displayed only for the following roles:

- Business Owner
- Developer

if the phase is Reception or Ingestion and the Return Code is without errors.

Mark as error offers a way to flag the package in error when the package stays in ING phase until the ingestion process is terminated (for example, after an unexpected error or after a server restart). After this action, ingestion can be

retried or the package can be invalidated. This action is not here to stop ingestion. Using this action during the ingestion can produce unexpected results.

Display Records
Displayed only for the following roles if the AIP contains records:

- Business Owner
- Developer

3.3.8 Rejecting or invalidating an AIP

When a Developer or Business Owner rejects or invalidates an AIP, the records that are contained in the AIP will no longer be returned in any new searches. A rejection can be applied to an AIP at any time, but the implications are different if these actions are performed before or after the AIP is committed:

- Reject an AIP when you want to invalidate all AIPs that belong to the same collection. When you reject an AIP, you cannot resubmit an AIP with the same DSS as long there is one or more rejected AIPs in the repository.
- Invalidate an AIP if the wrong SIP was submitted and you want to resubmit the correct SIP with the same identifier.

When you invalidate or reject an AIP, the AIP is immediately removed from a search's scope.

Once a completed AIP has been invalidated or rejected, it is possible that a user with Developer and Business Owner privileges may decide to cancel the “reject” or “invalidate” action and return the AIP to its initial state. For that purpose, the “Undo Invalidation” and “Undo Rejection” actions can be applied from the **Packages** tab in the IA Web App, based on the AIP state. The “Undo Invalidation” and “Undo Rejection” actions can only be used prior to executing the Invalidation job.

When the Invalidation job runs (manually or automatically), the invalidated/rejected AIPs are processed. If the AIP is part of a DSS with more than one AIP, invalidating or rejecting an AIP may impact the other AIPs:

- If you reject an AIP, other AIPs from the same DSS will be automatically rejected during the job execution.
- If you invalidate an AIP, other AIPs from the same DSS will be automatically demoted to the 'waiting to commit' phase during the job execution.

If the AIP has not been committed, the AIP is immediately destroyed. If the AIP has been committed and retention was applied (often through defining a retention class on the holding), then the AIP needs to be disposed regularly.

When the Confirmation job runs, it confirms that some events on packages has occurred (Receive, Available, Storage, Purge, Invalidation).

If you had rejected a package, and the Disposition job had marked some packages for purge, running the Confirmation job confirms everything outstanding (and that the next run of the Disposition job will cause the packages to be deleted).

You can reject an AIP if the AIP was part of a DSS with more than one other AIP. The AIP must also be in one of the following phases:

- Waiting Ingestion
- Waiting Commit
- Completed

You can only invalidate an AIP if the returnCode is 'OK'. The AIP must also be in one of the following phases:

- Waiting Ingestion
- Waiting Commit
- Completed

To reject or invalidate an AIP:

1. Select **Packages** tab.
2. Select the AIP that is being rejected or invalidated.
3. Click the context menu and select either:
   - **Reject Package** or
   - **Invalidate Package**
4. Select the Reason why the AIP is being rejected or invalidated.
5. If desired, enter any pertinent information in the Comment field.
6. Click **Reject** or **Invalidate**, depending on your selection in step 3.

3.3.8.1 Recovering when an AIP is stuck in ingestion or reception

If an AIP is stuck in the middle of either ingestion or reception, there are two ways to recover, using either IA Shell or IA Web App.

To recover a stuck AIP using IA Shell:

1. In IA Shell, navigate to the AIP. For example:

```
iashell>connect
Connected to "http://localhost:8765/services" as sue@iacustomer.com
iashell> cd applications
iashell> cd Trades
iashell> cd aips
iashell> ls
+------------------------------------+----------------------------+
|externalId                          |name                        |
+------------------------------------+----------------------------+
|e902Oeba-01ca-4508-b2ee-0179366fde93|Trades-TradeSystem-5000001-1|
|f152501f-aa61-4cc4-b5b3-a621edf5663d|Trades-TradeSystem-5000001-1|
```

```
|a8e719ec-80fa-41b7-9dd1-849359d740f0|Trades-TradeSystem-5000001-1|
|                                    |                            |
+------------------------------------+----------------------------+
iashell> cd a8e719ec-80fa-41b7-9dd1-849359d740f0
```

2. Run the `mark-as-error` command.

   IA Shell responds in one of the following two ways:

   - `AIP package has been marked as error`: This message means that the `mark-as-error` command was successful.
   - `The AIP package cannot be marked as error as corresponding link relation is not provided with REST`: This message means that the command is not allowed based on the state of the AIP. This could happen if you already marked the AIP as error or if the phase is not REC or ING.

3. Run the `invalidate` command.

To recover a stuck AIP using IA Web App:

1. In IA Web App, go to the **[APPLICATION NAME] > Packages** tab.
2. Click the context menu for the package that you want to recover. Choose **Invalidate Package** if it is available. If it is not available, choose **Mark as error**.
3. After a few seconds, force a hard refresh of the browser or navigate to another page and then come back.

   You should see on the side panel that the **Return Code** is set to **ERROR** and the **Return Message** is **Aip has been set as error manually**.
4. Click the context menu for the package that you want to recover and choose **Invalidation**.

3.3.9 Applying retention policies to AIPs

3.3.9.1 Applying a retention policy to a single AIP

To apply a retention policy to a single AIP:

1. For the AIP the retention policy is being applied to, click the context menu and select **Apply retention**.
2. Select the retention policy you want applied to the package and click **Next**.
3. Review the retention policy details to verify that it is the correct policy to apply to the package. The fields that are displayed depend on the Aging Strategy of the retention policy.
4. Select a **Base Date**. Options are **From Package** and **Specified Date (not from package)**. If you select **Specified Date**, select the date in the date picker.

5. Click **Next**.
6. Enter the following information and click **Next**:
   a. **Retention Set Name**: Enter a unique name for the policy.
   b. Optional Enter a **Description** for the retained set. The set can be seen on the **Retained Set** page.
7. Review the information you have entered. When satisfied that the information is correct, click **Finish**.

Note: The option to apply retention to a single package is not provided if the package does not have a commit date.

3.3.9.2 Applying a retention policy to multiple AIPs

To apply a retention policy to multiple AIPs:

1. Select the AIPs you want to apply the retention policy to and click **Apply Retention Policy**.
2. Select the retention policy that you want to apply to the packages and click **Next**.
3. Review the retention policy details to verify that it is the correct policy to apply to the application. The fields that are displayed depend on the Aging Strategy of the retention policy.
4. Select a **Base Date**. Options are **From Package** and **Use Same Base Date for All Packages**. If you select **Use Same Base Date for All Packages**, select the date in the date picker.
5. Enter the following information and click **Next**:
   a. **Retention Set Name**: Enter a unique name for the policy.
   b. Optional Enter a **Description** for the policy.
6. Review the information you have entered. When satisfied that the information is correct, click **Finish**. A background request is made, which you can view in the **Background Requests** tab.

Note: When you apply a retention to multiple packages, only packages that have a commit date will have the retention applied. If this is the case, a warning is given in the background task.

3.3.9.3 Removing one or more retention policies

To remove one or more retention policies:

1. Select the package or packages whose retention policy you want to remove.
2. Click the **Remove Retention Polices** button that appears above the list of packages.
3. In the confirmation dialog, click **Remove**.

Note: When removing retention from all packages (with no filter), when clicking the status in the background task, a new tab appears that allows you to view the order items that participated in the removal of retention.

3.3.9.4 Retention and ECS

Using Dell EMC Elastic Cloud Storage, if the store has been configured to push retention to the hardware, you are not allowed to apply second retention policy to an AIP until the existing retention period has expired, this is due to a known limitation of Dell EMC ECS.

3.3.10 Applying a hold to an AIP

1. For the AIP that requires the hold, click the context menu and select **Apply hold**.
2. Select the hold you want applied to the application and click **Next**.
3. Enter the following information and click **Next**:
   a. **Hold Set Name**: Enter a unique name for the hold.
   b. Optional Enter a **Description** for the hold.
4. Review the information you have entered. When satisfied that the information is correct, click **Finish**.

This is an asynchronous operation. The AIP will not indicate that there is a hold policy applied to it until the background request completes and the page is refreshed.

3.3.11 Canceling an apply hold or remove hold operation

If a hold is being applied or removed from data, The Retention Manager can cancel the operation from the Background Requests tab. An apply or remove hold operation may need to be canceled if, for instance, the Retention Manager neglected to add an additional filter when applying a hold to records, or the wrong hold was initially applied to an AIP and processing needs to stop.

When an apply hold or remove hold request runs, it runs in phases. The cancelation of the request is only allowed up to final phase of the request. Once the request reaches the final phase, it cannot be canceled.

To cancel an apply hold or remove hold operation, navigate to the Background Requests tab. Click the **Cancel** button for the desired request.

When an apply hold operation is canceled, it is possible some items may have been added to the set. In this case, either remove the set or resume the operation. When a remove hold operation is canceled, some items may have already been removed. No matter which operation was canceled, the Retention Manager can click the **Canceled** link in the **Status** column to view which batches were processed prior to the cancellation,

Once any issues have been resolved, and you want the operation to continue its run, click the **Retry** button.

Note: The log may not be available if the background task was canceled before the batch had a chance to run.

3.3.12 Display records

This allows you to display the records that are contained in the AIP. Select a search configuration to display the records. The search will run in background and will be available in the **Background Requests** tab. Note that the default search form settings defined in the selected search will not be considered to display the records.

3.4 Table management

3.4.1 Using the Tables tab

The **Tables** tab allows the user to perform the following actions:

- View a list of the available tables in an application. Use the **Find a Table** field to locate a specific table by entering a name and clicking Enter. You can also filter the tables by database or schema.
- The Retention Manager can apply a retention policy or hold to a table;
- Delete a table.

Each table is displayed and contains the following information:

Table Name	The name of the table. Click the link to view the details of the table.
	A menu that allows a user to perform the following actions on a selected table. These options may not be available based on your role. • Apply retention • Apply hold • See the Audits (for more information, see Seeing the audits) • Delete Table
Archived Records	Indicates the number of records that still exist after disposition has occurred at least once.
Reference Records	Indicates the number of records in the table that are defined in the table definition.
Last Ingestion Date	Indicates the date that data was ingested into the table.

A panel on the right side of the page contains the custom properties of a selected table. The following tabs appear in the panel:

Details
Indicates which columns appear in the selected table.

Compliance
Indicates whether retention or holds are inherited from the application, applied directly, or applied to at least one of the selected table's records.

Constraints
Provides information about any primary keys, foreign keys, or unique keys defined by the table. A separate sub-tab is shown for each of the three types of constraints.

3.4.2 Setting a table application online/offline

A table application can be set offline, which means that it is not stored in a PostgreSQL database. The primary advantage of this is that object storage is generally cheaper than the object/file storage that is required for a database.

An Administrator or Business Owner can use IA Web App to create the backup in the archive storage, as well as call for the deletion of the structured database. By doing so, it renders the table application to be effectively "offline". Users cannot access the application until the application is set online.

An application with an In Test status that has been set offline can still be deleted. Furthermore, it is possible to delete the ingested data of an In Test application that has been set offline.

Unless you are sure that you have the latest backup available, it is always recommended to enable backup creation via the IA Web App while taking an application offline. This is to ensure that you have an up-to-date backup in case the application is subsequently brought back online.

Setting an application offline or online is an asynchronous request.

Note: If you set the Reports application offline, any attempt to run the Refresh Metrics job will fail.

Before taking an application offline, you may want to disable any jobs that are scoped to it.

To set a table application offline:

Note: There is an additional step here that, if the system has been configured to not provide the ability for the data node user to create databases, prior to performing this procedure, a database administrator will need to create the database in PostGreSQL.

1. For the application being set offline, click **... > Set Application Offline**.
2. A pop-up is displayed. Adjust the following information, if required:
   a. In the **Request Name** field, a generic name is applied to the asynchronous request. Update this field, if desired.
   b. You have the option to back up all databases before the application is set offline. By default, the request will back up the databases. If you do not want the system to back up databases, uncheck this box.
   c. Click **Set Offline**. When viewing applications on the **Applications** page, users must uncheck the **Online only** filter box to view applications that have been set offline.

      Navigate to the **Background Requests** tab to review the status of the request.

To set a table application online:

1. For the application being set online, click **... > Set Application Online**. If you do not see the offline application, ensure that the **Online only** filter box is not checked.
2. A pop-up is displayed. Adjust the following information, if required:
   a. In the **Request Name** field, a generic name is applied to the asynchronous request. Update this field, if desired.
   b. Click **Set Online**.

      Navigate to the **Background Requests** tab to review the status of the request.

3.5 Seeing the audits

The Packages, Libraries, Tables tabs allow the Business Owner and Developer to view the audits associated with a particular AIP, library, or table. To view an AIP, library, or table audit, click  and select **See the Audits**.

Use this feature to run a search for audits for the selected AIP, library, or table. Instead of displaying the results in the Audit application, however, the results are displayed in the Packages, Libraries, or Tables tab for the currently selected application.

When selected, the system automatically uses the following criteria to run the search:

- The currently selected application
- The audited object ID of the selected package, library, or table
- The time period of 6 months

This feature can be configured in the Global Settings tab. For more information, see Configuring the See the Audits feature.

This feature is available for each application.

3.6 Running and viewing metrics with the Administration dashboard

Depending on your role, one of two possible dashboards is available to you:

- The administration dashboard
- The compliance dashboard

To view the administration dashboard, Administrators can click the **Dashboard** tab. When Retention Managers click the **Dashboard** tab, they see the compliance dashboard instead of the administration dashboard. Users with both the Retention Manager and Administrator roles see both dashboards.

When you view the administration dashboard, there is an **Applications** section at the top of the page that lists the number of total applications, active archive applications, and decommissioned applications. Underneath this section, you can toggle between either the:

- The **License** dashboard
- The **Storage** dashboard

Note: If you clean up a database, it keeps the maximum size that it has reached. For example, if you have a 1 GB database and dispose of all its data,

and then add 500 MB of data, the database size is still 1 GB. Also, when you dispose of data, the database might take up more space temporarily. Both points affect the storage calculations on both the license dashboard and storage dashboard.

3.6.1 The License dashboard

Pricing for OpenText Information Archive is based on how much application data is being managed (whether online or offline). The license dashboard helps you understand how the storage usage of your various applications is contributing to licensing costs. This dashboard does not show the total amount of storage used by OpenText Information Archive because some of the storage used does not affect licensing costs (for example, backups and log files).

The dashboard consists of the following sections:

- **Total Usage**: Shows the amount of content (**Unstructured Content**) and data (**Structured Data**, such as transactions) under management by OpenText Information Archive, for the purposes of licensing.
- **Application Breakdown**:
  - Shows the amount of space taken up by each application. Some applications tend to use more space for content, and some for data.
  - You can sort the list (default is **Total Size**) and narrow it to **Active Archiving** and **Application Decommissioning** applications. For decommissioned applications, typically the storage space needed does not change significantly. Active archiving applications typically have a growing need for storage space.
  - The index size for each application is for full-text searching.
- **Application Type**: Shows the amount of space taken up by **Active Archiving** and **Application Decommissioning** applications.

3.6.2 The Storage dashboard

The storage dashboard shows the total amount of storage used by your OpenText Information Archive deployment, including storage that is used but does not affect licensing costs (for example, backups and log files). You should refer to this dashboard when you want to understand the total storage requirements of OpenText Information Archive. The total amount of storage shown on the storage dashboard is greater than the amount shown on the license dashboard.

The dashboard consists of the following sections:

- **Total Usage**: Shows the total amount of storage used by OpenText Information Archive and breaks down that total storage into the following categories:
  - **Content**: Unstructured information, such as documents
  - **Structured Data**: Structured information, such as transactions

- **Search Results**: The cached results of user searches
- **Unarchived Audits**: Audits that have not yet been archived by the Audit application
- **System**: System data
- **Retention**: Granular retention for records
- **Export Content**: Exporting of searches/renditions and storing logs for jobs and background requests

- **Application Breakdown**: When the Export button is clicked from the **Application Breakdown** dashboard, only the following application-specific information is exported:

  When the dashboard is at the application-level, and the Export button is clicked, only data from the selected application is exported. The scope of the exported data includes:

  - Application-specific information
  - Database-related information (for table applications only)
  - Storage-related information for the selected application
  - Store-related information for the selected application

  Each scope maps to one of the charts shown on the UI. There is a line break between each scope in the exported data. The following is an example of data exported from the Invoices example application:

Scope	Name	Type	Storage Name	Storage Type	Structured	Regular Content	Search Results	Export Cor	Granular Retention	Total
Application	Invoices	ACTIVE_ARCHIVING			109918	366418	0	0	0	476336
Storage	defaultFileSystemRoot	FILESYSTEM			109918	366418	0	0	0	476336
Store	default-store	REGULAR	defaultFileSystemRoot	FILESYSTEM		366418				366418
Store	default-result-store	RESULT	defaultFileSystemRoot	FILESYSTEM				0		0
Store	default-search-result-s	SEARCH_RESULT	defaultFileSystemRoot	FILESYSTEM			0			0
Store	default-library-store	LIBRARY	defaultFileSystemRoot	FILESYSTEM	109918				0	109918

  In the above screenshot, note that the default-library-store is storing structured data for the packages.

  The following is an example of the exported data from a table application (in this case, the Baseball example application). Notice that a column is left blank when not appropriate for the store:

Scope	Name	Type	Storage Name	Storage Type	Structured Data	Regular Content	Search Results	Export Content	Granular Retention	Total
Application	Baseball	APP_DECOMM			337723392	1111932	7413	18360	0	3.39E+08
Database	Baseball-rdb				337723392					3.38E+08
Storage	defaultFileSystemRoot	FILESYSTEM			0	1111932	7413	18360	0	1137705
Store	default-store	REGULAR	defaultFileSystemRoot	FILESYSTEM		1111932				1111932
Store	default-result-store	RESULT	defaultFileSystemRoot	FILESYSTEM				18360		18360
Store	default-search-result-store	SEARCH_RESULT	defaultFileSystemRoot	FILESYSTEM			7413			7413
Store	granular-retention-store	LIBRARY	defaultFileSystemRoot	FILESYSTEM	0				0	0

  For the store types, you can cross-reference which one matches which based on the columns:

  - REGULAR = Regular Content

- RESULT = Export Content
- SEARCH_RESULT = Search Results
- LIBRARY = Can map to either structured data or granular retention, check the columns.

- Shows the amount of space taken up by each application. Some applications tend to use more space for content, and some for data.
- You can sort the list (default is **Total Size**) and narrow it to **Active Archiving** and **Application Decommissioning** applications. For decommissioned applications, typically the storage space needed does not change significantly. Active archiving applications typically have a growing need for storage space.
- You can click an application link to view application-level storage details. For example, Audit shows how much space is used by the Audit application, which stores the archived results:

> **Important**
> For the **Stores Breakdown** to be populated, the Refresh Metrics job must have been run with the latest version. If the **Stores Breakdown** still shows nothing, it means the selected application does not have any stores defined.

Dashboard widget	Description	SIP	Table
Total Usage	Overall breakdown	X	X
Table Database Online	Space used in the database for structured data		X
Storage System Breakdown	For each storage used, including unstructured content	X	X
Stores Breakdown	Indicates the store breakdown for the selected application	X	X

- **Storage System Breakdown**: Shows the amount of space taken up by unstructured content. The chart distinguishes between unstructured content stored in search results versus all other unstructured content. This can help you understand how much searching is being done on your system. If you use more than one type of storage, the chart will show more than one entry for **Content**.
- **System Databases Breakdown**: Shows the amount of space taken up by the system databases. You can click the name of each one for more information.

  - **rollForward**: Contains temporary data for recovery support, which is cleaned up every 24 hours
  - **synchronization**: Contains data for lock support
  - **system**: Contains the main system data (for example, order items and job instances)

- **Table Database Online**: Shows the amount of space taken up by structured data for applications that are online (as opposed to applications with offline data or applications that have been taken fully offline). You can sort the list (default is **Total Size**), and the index for each application is an index of structured information for full-text searching.

3.6.3 Exporting data from the Administration dashboard to a CSV file

You can export data from either the license dashboard or the storage dashboard to a CSV file (for example, for use in Microsoft Excel). The CSV file only contains data from the dashboard that you exported it from, so it does not contain both license and storage data. The numbers in the CSV file are in kilobytes.

The CSV file for the license dashboard contains the following columns:

Column	Value	Description
Scope	Application	The row contains data from the **Application Breakdown** dashboard widget
	Tenant	The row contains data from the **Application Type** or **Total Usage** dashboard widget
Name	<Application name>	The application name (for example, Audit)
	INFOARCHIVE	The tenant name
Type	ACTIVE_ARCHIVING	The application type is active archiving
	APP_DECOMM	The application type is application decommissioning
	<None>	The row contains data from the **Total Usage** dashboard widget
Archive Type (added for both License and Storage dashboards)	SIP	The archive type is SIP for Sip applications
	Table	The archive type is TABLE for table applications
Unstructured Content	<Number>	The size of content (unstructured information such as documents)
Structured Data	<Number>	The size of data (structured information such as transactions)
Total	<Number>	The total size of content and data

The CSV file for the storage dashboard contains the following columns:

Column	Value	Description
Scope	Application	The row contains data from the **Application Breakdown** dashboard widget
	Storage	The row contains data from the **Storage Type Breakdown** dashboard widget
	Database	The row contains data from the **Structured Data Online** dashboard widget
	System	The row contains data from the **System Databases Breakdown** dashboard widget
	Tenant	The row contains data from the **Total Usage** dashboard widget
Name	<Application name>	The application name (for example, Audit)
	<Storage system>	The type of storage system (for example, FILESYSTEM)
	<Database name>	The database name (for example, Tickets-rdb)
	<System collection>	This indicates the name of the configuration. For example, search result indicates how much space the search result configuration objects are using. Only the top consumers are used. Other represents the space used by the remaining system configuration objects.
	INFOARCHIVE	The tenant name
Type	ACTIVE_ARCHIVING	The application type is active archiving
	APP_DECOMM	The application type is application decommissioning
	<Storage Type>	The row contains data for each storage type being used, for example, FILESYSTEM, S3, corresponding to each entry in the **Storage Type Breakdown** dashboard widget.
	system	The row contains data from the **system** portion of the **System Database Breakdown** dashboard widget.

Column	Value	Description
	rollforward	The row contains data from the **rollforward** portion of the **System Database Breakdown** dashboard widget.
	synchronization	The row contains data from the **synchronization** portion of the **System Database Breakdown** dashboard widget.
	<None>	The row contains data from the **Total Usage** dashboard widget.
Archive Type	SIP	Indicates the archive type for the application. This is only set for rows representing applications
	Table	
Structured Data Only	<Number>	The size of data (structured information not including the index) at application database and tenant levels
Content	<Number>	The size of content (unstructured information such as documents) at the application and tenant levels
Search Result	<Number>	The size of the search results and the search result index at the application and storage levels
Export	<Number>	The size of unstructured content in the RESULT store at the application and storage levels
Unarchived Audit	<Number>	The size of the unarchived audits, provided only at the tenant level
System	<Number>	The size of system data
Granular Retention	<Number>	The size of retention objects provided only at the application and tenant levels

To export data from the administration dashboard to a CSV file:

- On the administration dashboard page, click **Export**. The ZIP file that is available to open or save contains the CSV file.

3.6.4 Updating the metrics for the dashboard

The Refresh Metrics job updates the metrics for the administration dashboard and compliance dashboard. The job must be scoped to the system and does not support application scoping.

Calculating the metrics information in the dashboard can take a significant amount of time. Therefore, the dashboard retrieves most of its information from pre-populated values, and the Refresh Metrics job scans the system and populates these values. You can decide how often the metrics information should be updated.

This job does not have any parameters.

There are three scenarios in which the information will not be available from the server and the UI will show an empty list of applications:

- If you upgrade the system but do not refresh the metrics
- If you did not install any applications and refresh the metrics
- The metrics were never refreshed

Also, if you take the Reports application offline, then you cannot run the Refresh Metrics job.

3.6.5 Auditing when dashboard data was exported to CSV format

Administrators can use audits to identify when data from the administration dashboard and compliance dashboard was exported to CSV format. First you must enable the audit. The audit is configured under **Application System**, **Event Category Other**, and the event name is called **Export**. Both audit event types are disabled by default. These are the separate Event Names:

- Compliance Metric
- Storage Metric

To see the audit, you must do the export from a dashboard and then archive the audits.

Once the audits are archived, you can find them by performing a Tenant Audit search in the Audit application. These audits are not enabled by default.

In the search results, the following elements are key:

Created Date
When the user performed the export.

Created By
The user who performed the export.

Event Type
The type of export performed:

- **compliance_metric**: An export of the compliance dashboard
- **storage_metric**: An export of the administration dashboard

Event Name
The event name is **export** for any dashboard exports.

As Of
The most recent date when the Refresh Metrics job was run before the export was performed. To learn which date formats are accepted, see the note in Apply Retention Rule to Records job.

type
The type of administration dashboard export:

- **license-dashboard**: An export of the administration dashboard's license tab
- **storage-dashboard**: An export of the administration dashboard's storage tab

If a specific application was exported, the application name is appended to the type.

No type appears for exports of the compliance dashboard.

For more information about the Audit Application, see The Audit application.

To enable audits for when dashboard data is exported to CSV format:

1. In IA Web App, go to the **Administration > Audit** page. For the **Application** filter, select **System** and for the **Event Category** filter, select **Other**.
2. Do any of the following:
   a. To enable audits for when the administration dashboard is exported, in the **Storage Metric** row, select the **Export** check box.
   b. To enable audits for when the storage dashboard is exported, in the **Compliance Metric** row, select the **Export** check box.
3. Click **Save**.

   When the Archive Audits job runs, it generates the audits that you specified.

To search for audits that identify dashboard exports:

1. In IA Web App, go to **Applications > Audit > Tenant Audit**, and then perform a search.
2. In the search results, look for results with the event type **compliance_metric** (for exports of the compliance dashboard) and **storage_metric** (for exports of the administration dashboard).

3.6.6 Calculating pricing

For XML metadata of both SIP and table files, the volume size for pricing is calculated by counting all characters of an element and attribute content. Characters used for XML formatting purposes are not counted. For example, in the following XML, only the characters in bold are used for pricing. Each character is counted as 1 byte, even though a UTF-8 character can consist of multiple bytes. This character count runs during ingestion and before encryption.

```
<BASEBALL>
  <ALLSTARFULL>
    <ROW>
      <PLAYERID>aaronha01</PLAYERID>
      <YEARID>1955</YEARID>
      <GAMENUM>0</GAMENUM>
      <GAMEID>NLS195507120</GAMEID>
      <TEAMID>ML1</TEAMID>
      <LGID>NL</LGID>
      <GP>1</GP>
    </ROW>
```

For SIP archiving:

- The characters in PDI XML files are calculated.
- The size of the additional content files contained in a `sip.zip` file are calculated.

For table archiving:

- The characters in table XML files are calculated.
- The size of the attached content files is calculated.

The size of the original content file is measured before ingestion.

3.6.6.1 Performing a byte count on application data

If you want to estimate the license cost of SIP ZIP files or tables before ingestion, you can use a standalone Java tool called iametrics to calculate file sizes and character counts. The tool automatically chooses a unit (bytes, KB, MB, and so on) in which the smallest file has a value greater than 1. All files have the same unit for easy comparison. Formatting data in a file is not included in the final calculation.

You can access the iametrics tool in the `<IA_ROOT>/bin` directory. There is a version for Windows (`iametrics.bat`) and Linux (`iametrics`).

The following parameters are required to perform a byte count:

`-xml / -sip / -table`
: The `-xml` parameter is used to calculate the byte size of either an XML file or table.

  The `-sip` parameter is used to calculate the byte size of a SIP ZIP file.

  The `-table` parameter is used to calculate the byte size of the XML and the size of the attached content.

You must enter one of the above parameters.

`<location>`
You must specify a directory or file that the byte count is performed on.

`-csv / <csv-location>`
While it is not mandatory, you can generate a report that contains the information returned in a byte count. The report is in CSV format. The `-csv` parameter allows you to specify the directory and filename of the report.

When you perform a byte count using the `-xml` parameter, the following information is returned:

File size
The compiled data size of the table or XML file.

Data size
The character count of the table or XML file.

When you perform a byte count using the `-sip` parameter, the following information is returned:

File size
The size of the `eas.pdi.xml` file.

Data size
The character count of the `eas.pdi.xml` file.

Content size
The byte count of any additional files (for example, images) stored in the `sip.zip` file.

SIP size
The compiled data size of the `sip.zip` file.

3.6.6.2 Examples of byte counts

The following examples illustrate how to perform different kinds of byte counts. These examples are all done in Windows, using the command prompt, in the `<IA_ROOT>` directory.

To perform a byte count on...	Run the following command...
A table named `BASEBALL-MASTER-00001.xml` that is stored in the Baseball example application	`bin\iametrics -xml examples\applications \Baseball\data\BASEBALL\BASEBALL-MASTER-00001.xml`
All tables that are stored in the Baseball application	`bin\iametrics -xml examples\applications \Baseball\data\BASEBALL`
All tables that are stored in the Baseball application, and generate a report called `baseball.csv`	`bin\iametrics -xml examples\applications \Baseball\data\BASEBALL -csv baseball.csv`

To perform a byte count on...	Run the following command...
The SIP ZIP files stored in the Trades application	`bin\iametrics -sip examples\applications \Trades\data`

3.6.6.3 Questions and answers about pricing

After ingestion how is volume calculation kept in check?
All information is computed during the ingestion and saved into the system repository to be used later.

How does encrypted information impact licensing volume calculations?
Encryption does not affect licensing volume considerations, but it can increase the storage footprint.

How does AIU or record disposition impact the volume calculation?
When the AIUs or records are disposed of, the licensing metrics are updated to reflect the disposition. The storage footprint is not impacted.

How is cache-in/cache-out considered in the licensing volume calculation?
Cache-out/cache-in does not have an impact on the licensing.

Is the storage of auditing information, the system data repository and the managed item data repository included in the calculation of the storage footprint?
No, the dashboard does not expose these repositories.

Does the system keep track of the size of all other unstructured information (for example, raw XML, renditions, database back up)?
Yes, the other unstructured information is included in calculating the content size. It is reflected in the storage footprint.

What jobs are required to ensure that the correct data is displayed on the dashboard?
The Administrator needs to manually run or schedule the Refresh Metrics job to view the most recent information.

If a table-based application is offline, how does that affect pricing?
You are only charged for data that is online.

3.7 Jobs

Jobs are units of archiving functionality that can be invoked to run once (manual) or on a schedule. They can be found under the **Administration** tab under **Jobs**.

Most jobs can be configured to apply to a single application or a group of applications. Many run a sequence of subtasks (order items), where order items may, in turn, run a collection of parallel batches.

The **Jobs** interface lets you navigate into executions of jobs, order items and batches to find out about their results, progress and logs.

Note: All jobs that are configured to run on a schedule will not run automatically until the schedule is started either using the IA Shell or IA Web App.

3.7.1 User roles and jobs

Each job falls into one of the following categories:

- Administrative
- Compliance
- Data
- Upgrade

While the Administrator role can run each job, the Retention Manager is only able to run Compliance jobs and the Developer can only run Data jobs. The following table outlines each job, its respective category, and which role can run it. A checkmark indicates that the user role can run the job while an empty cell indicates that the user role cannot run the job:

Job	Category	Administrator	Developer	Retention Manager
Apply Hold Rule to Records	Compliance	✔		✔
Apply Retention Policy To Records	Compliance	✔		✔
Apply Retention Rule to Records	Compliance	✔		✔
Archive Audits	Administrative	✔		
CacheOut	Data	✔	✔	
Clean	Administrative	✔		
Clean up Purge Candidate List and Applications	Administrative	✔		
Close	Data	✔	✔	
Commit	Data	✔	✔	
Confirmation	Data	✔	✔	
Consistency Checker	Administrative	✔		
Dispose Purge Candidate List	Compliance	✔		✔

Job	Category	Administrator	Developer	Retention Manager
Disposition RollForward Recovery	Compliance	✔		✔
Generate Purge Candidate List	Compliance	✔		✔
Invalidation	Data	✔	✔	
Migrate Compliance Data	Administrative	✔		
Post Ingest Processing	Data	✔	✔	
Process Retention Events	Compliance	✔		✔
Refresh Metrics	Administrative	✔		
Remove Policy	Compliance	✔		✔
Requalification	Administrative	✔		
Table Indexing	Data	✔	✔	
Trigger Event Policy	Compliance	✔		✔
Trigger Event Rule	Compliance	✔		✔
UPGRADE: Execute Asynchronous Upgrade Tasks	Upgrade	✔		
UPGRADE: Execute Synchronous Upgrade Tasks	Upgrade	✔		

Developers cannot run a system-scoped job if there is one application they do not have permission to access. Furthermore, Developers are also not able to edit or clone a job that can only be configured to run against the system and not any applications.

3.7.2 Apply Hold Rule to Records job

Note: This job is only available if rule-based processing has been enabled for your system.

Provides the ability to apply holds to records using rules. Rules for holds can be uploaded using DC and some sample applications include examples. When creating the rule, use the type `APPLY_HOLD` instead of `APPLY_RETENTION`.

You can use the job's properties to specify a search criteria to restrict which records are evaluated. If this is not specified, the system evaluates all records.

This job can be canceled and retried. For more information, see Canceling a job.

This job is manually scheduled. It is recommend that you scope it to only one application, since searches are specific to applications.

Caution

When specifying the search criteria, ensure the criteria is associated with a partition key for SIP searches. Failure to do this could cause the search to cache-in every package because of running this job.

The following outlines the properties that can be configured within the job:

`Search Name`
: Specify the name of search in the application. The search must be set to ready. For table applications, the search must be associated with a schema and table.

`Search Set`
: Specifies the search set within the search to use.

`Search Criteria File`
: Define the search criteria in XML that outlines how you want to narrow the search results and enter the file path here. The file path can also accessible by IA Server that contains the search criteria in XML format. This XML is same payload if looking at a REST call when doing the search from the IA Web App client. For an example of how to use this property, see searchCriteriaFile property).

`Max Server Time Drift In Minutes`
: Applies to SIP applications only. If you are running with multiple servers, this value is used to ensure packages that were ingested on servers that may be lagging behind are still considered. It is recommended to set this value to `0` if there is only a single IA Server. This value is ignored if the job is scoped to a table application.

`Developer Mode`
: If selected, developer mode runs the rule, but does not apply the hold. This property is meant for testing.

`Process New Records Only`
Applies to SIP applications only. If selected, only new records ingested since the last time the job successfully ran for the application are processed.

`Rule Name`
Use this property to limit using rules matching the specified name only. If not specified, all rules of type `APPLY_HOLD` are evaluated.

3.7.3 Apply Retention Policy To Records job

The Apply Retention Policy to Records job applies a retention policy to AIUs or table rows. The job uses a pre-configured search to determine which AIUs or table rows need to have retention applied. The job allows for criteria to be specified to narrow down the results. Each result will have the configured retention policy applied to each AIU/table row.

This job can be canceled and retried. Refer to Canceling a job for further information.

This job is manually scheduled and is applied to applications.

The job also requires an XML file to be created on the IA Server to indicate the search criteria. Records that already have the policy will not have the policy applied again.

When using this job to apply event or mixed retention policies, the name of the set will be `<Retention Policy Name>:<Condition>:<date that the job started>`.

For example, the retained set would be called the following if the retention policy (Policy A) was applied:

```
Policy A:

NoLongerNeeded:2018-07-

27T10:17:44.668-07:00
```

Select the application to run the job against. Searches are specific to applications so you can only select one application at a time. This job has to be application scoped.

For a table search, specify a particular table to be protected. If a search's results combine multiple tables, only records from a single table can be protected.

The following outlines the properties that can be configured within the job:

`Search Name`
Name of the search used to find the records. The search must be defined for the application and the status must be set to ready.

`Search Set`
Specifies the search set within the search to use.

`Search Criteria File`
Define the search criteria in XML that outlines how you want to narrow the search results and enter the file path here. The file path can also accessible by IA

Server that contains the search criteria in XML format. For an example of how to use this property, see searchCriteriaFile property).

`Max Server Time Drift In Minutes`
For SIP applications only. If you are running with multiple servers, this value is used to ensure packages that were ingested on servers lagging behind are still considered. It is recommended that this value be set to `0` if there is only a single IA Server. This value is ignored if the job is scoped to a table application.

`Retention Policy Name`
All four types of retention policies are supported. Depending on the type of policy, other properties are required to be set.

`Retention Date Attribute`
This value is used for Duration and Duration portion of mixed retention policies. Duration policies that are applied to records need to include a date to calculate the age of the record. The date will be taken from the attribute specified in this field. If the attribute does not have a date, the record will be skipped, meaning a policy will not be applied

`Context Type`
Used for Event policies. There are two possible values:

- Attribute: The job gets the context from an attribute in the data. The attribute is set in the context field.
- Fixed: The job uses the value in this field as the context for the events.

For example, to have all records for an employee age for five years after leaving the company, make the context the employee ID, since it will be the same for all documents. OpenText Information Archive groups all documents with the same context together and you would trigger the event using this context. All records associated with this context (for example, employee #) will be eligible for disposition in five years.

`Context`
The value is dependent on the `ContextType`. Either:

- The attribute (its value) to be used as the context, or
- A value entered in this field will be used as a context for all records that are returned by the search.

`Trigger Check Attribute`
The job can use an attribute to determine if the trigger should be performed. For example, if the data has a field that indicated if the employee has left the company (for example, `hasLeft = true`). The attribute should be set to `hasLeft` and the value should be set to `true`. The `TriggerCheckValue` would be to set the value to check, in this case `true`. If the value in the attribute is anything but the trigger check value, then the event would not be triggered.

`Trigger Check Value`
This is value to check against to determine whether the event should be triggered.

`Trigger Date Attribute`
The job requires a date to trigger the event. This property is an attribute name where the job will get the trigger date. For example, if the event is to keep all employee records five years after the employee leaves the company. There would be an attribute called `terminationDate` that contains the date the employee left the company. Putting `terminationDate` in the `TriggerDateAttribute` property would instruct the job to fetch the trigger date from the `terminationDate` attribute.

`Trigger Date`
Use Trigger Date if you want the same date for all the records that the policy is being applied to.

`Trigger`
Only used for Event policies. If selected, the job attempts to conditionally trigger the event policy that was applied to the records. The other trigger job properties are used to determine if the event should be triggered and the date.

`Process New Records Only`
Applies to SIP applications only. If selected, only new records ingested since the last time the job successfully ran for the application are processed.

Example 3-1: searchCriteriaFile property

This is a path to a criteria file that contains what you want to narrow the results to. The following is an example of a SIP search:

```
<data><criterion><name>CustomerID</name>
<operator>EQUAL</operator><value>000103</value>
<value>391</value></criterion></data>Example of a table search:<data><customerID>16</
customerID></data>
```

The location is relative to where the server is deployed. If multiple IA Servers are installed, it is recommended that you use a network location (versus a local path).

It is also possible to provide the necessary contents of this file in-line into the property instead. This may be a more convenient option because you may not have the ability to modify the server's files system in all deployments.

3.7.4 Apply Retention Rule to Records job

Note: This job is only available if rule-based processing has been enabled for your system.

Allows you to run previously uploaded rules that apply retention policies to records. Job properties allow the ability to specify search criteria that limit which records are evaluated.

This job is manually scheduled and is applied to applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

For a table search, specify a particular table to be protected. If a search's results combine multiple tables, only records from a single table can be protected.

The difference between this job and the Apply Retention Policy To Records job is that this job allows more flexibility, as different retention policies can be applied. The rule criteria could decide to only protect a subset of the results from the search. For example, the rule could decide not to apply retention to some records that are from a particular region.

Caution

When specifying the search criteria, ensure the criteria is associated with a partition key for SIP searches Failure to do this could cause the search to cache-in every package as a result of running this job.

When specifying dates in the retention rules, the following formats are accepted:

- YYYY-MM-dd (for example, 2001-01-20)
- YYYYMMdd (for example, 20010120)
- YYYY-MM-ddTHHmm:ss (for example, 2016-01-21T00:00:00)
- YYYY-MM-ddTHH:mm:ss.SSS (for example, 2016-01-21T00:00:00.000)
- YYYY-MM-ddTHH:mm:ssX (for example, 2016-01-21T00:00:00Z, or 2016-01-21T00:00:00-05:00)
- YYYY-MM-ddTHH:mm:ss.SSSX (for example, 2016-01-21T00:00:00.000Z, or 2016-01-21T00:00:00.000-05:00)

If the time is not provided, the values will be interpreted as midnight UTC time. If a time is provided and no offset, the value will be interpreted as UTC time.

Two digit dates are not supported.

The following outlines the properties that can be configured within the job:

`Search Name`
Specify the name of search in the application. The search must be set to ready. For table applications, the search must be associated with a schema and table.

`Search Set`
Specifies the search set within the search to use.

`Search Criteria File`
Define the search criteria in XML that outlines how you want to narrow the search results and enter the file path here. If this is a file path, it must be accessible by the IA Server and contain the search criteria in XML format. This XML is the same payload as if looking at a REST call when doing the search from the IA Web App client. For an example of how to use this property, see searchCriteriaFile property.

`Max Server Time Drift In Minutes`
For SIP applications only. If you are running with multiple servers, this value is used to ensure packages that were ingested on servers lagging behind are still considered. It is recommended that this value be set to `0` if there is only a single IA Server. This value is ignored if the job is scoped to a table application.

`Rule Name`
Use this property to limit using rules matching the specified name only. If not specified, all rules of type `APPLY_RETENTION` are evaluated.

`Process New Records Only`
Applies to SIP applications only. If selected, only new records ingested since the last time the job successfully ran for the application are processed.

`Developer Mode`
If selected, developer mode runs the rule, but does not apply the retention policy. This property is meant for testing.

`Best Effort`
If selected and there are issues applying either retention or a hold, the job applies retention or a hold to as many records as possible. If not selected, processing stops when it encounters an issue.

3.7.5 Archive Audits job

When audits are generated, they are stored in a PostgreSQL database and you cannot run a search against it. The objects are collected in a temporary storage until they are archived. The Archive Audits job archives these audits so that they can be searched via the Audit application, which is installed when setting up first time applications.

The default scope of this job is at the system–level, which ensures all audits are archived. This job also supports to application scoping, which can be used to limit the scope of audits being archived.

Important

To prevent the unarchived audit database from growing indefinitely, start the schedule and review the frequency depending on how many audits you are expecting.

Note: Archived audits are only removed after the default roll forward interval has been met and the job runs again.

When running the Archive Audits job, order items are made to archive the audits. Depending on how the job is configured, the job may create an order item for the:

- System audits,
- Tenant audits, and
- For each installed application.

The following properties can be configured:

`Holding`
: This is the holding that corresponds to the IA App name. Typically, the application name and holding name are the same. The default is Audit.

`Application Name`
: The name of the Audit application (normally does not need to be changed).

`Restrict To System And Tenant`
: If the job is system-scoped, setting this to true indicates that the job should only process the system and tenant audits, excluding the application audits. If the job is scoped to any specific application, the job will fail.

`Archive Audit With Missing Content`
: If selected and content is missing, the audit is archived without content. The logs will indicate that at least one audit was archived with missing content.

  If not selected, audits with missing content and audits belonging to the same audit package will not be archived, and a warning is applied to the order item. The warnings alert you that audits were not archived during the job's run. The logs indicate which audits contain missing content.

  If the Archive Audit job failed because content for an audit is missing, on a later run of the job, setting this to `true` archives the audit without the content; otherwise, the job will fail.

`Preferred Number Of Records Per SIP`
: Preferred number of audits in an individual SIP.

  In case multiple SIPs need to be created, and if the number of audits in last SIP would contain less 20% of preferred number of audits, then these audits are assigned to another SIP. In this case, the SIP handling the reassigned audits will have an audit count greater than the preferred number of audits. The minimum value is . The max

  - Minimum value: `10000`

- Maximum value: `10000000`

`Hash Encoding`
Select the code used to validate the integrity of the data string.

`Hash Algorithm`
Select the algorithm used to calculate the hash codes.

`Minimum Unarchived Audits (*)`
Minimum number of unarchived audits required to start audit archiving.

`Maximum Aging Of Unarchived Audits (*)`
Maximum aging in hours of unarchived audits.

The existence of at least one unarchived audit older than this aging is required to start audit archiving.

This property is ignored when its value is empty or set to `0`.

(*) The default value for `Minimum Unarchived Audits` and `Maximum Aging of Unarchived Audits` is 10,000 audits and 6 hours, respectively. This means that archiving per order item will be skipped until the number of audits is greater than or equal to 10000 or at least one audit is aged more than 6 hours.

In addition to the properties above, the following Global Settings impact the behavior of this job:

- The global setting `retention.rollForwardObject` indicates, in hours, how long the customer requires for replication between the servers to complete. The default is 24 hours. For more information about replication, see Section 12 "Disaster recovery" in *OpenText Information Archive - Installation Guide (EARCORE-IGD)*s. You are encouraged to review and update this setting, as required.
- The global setting `audit.chaining.enabled` can be used to create chaining (see Chaining audits). Audit SIPs are archived in parallel when chaining is disabled. Conversely, SIPs are sequentially archived if chaining is enabled.

3.7.6 CacheOut job

Caches out AIPs, if necessary, to stay below the threshold defined for the SIP application. Scoping this job to a table application has no effect. Provides the ability to reduce the metadata footprint by removing unused AIP libraries.

This job is manually scheduled and is applied to the system and applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

Tip: It is important to have this job scheduled if the limit data in cache option is enabled on one or more SIP applications. The recommendation is to schedule this job every 5-15 minutes. The default interval value for the CacheOut job is 15 minutes.

The CacheOut job caches out AIPs, if necessary, to stay below the threshold defined for the SIP application. Scoping this job to a table application has no effect.

The following property can be configured:

`Max Library Per Application`
The property is used limit the Cache-Out job from processing too many libraries.

3.7.7 Check package retention job

This job is now deprecated and will be removed in a future release. The job is only seen if your original version of OpenText Information Archive was 21.4 or earlier.

For more information, see the technical paper *Ensuring retention applied correctly (KB0816795)* on My Support (https://support.opentext.com/csm?id=kb_article_view&sysparm_article=KB0816795).

3.7.8 Clean job

The Clean job frees up resources, such as orders, search results and AIPs. The job removes events from the system if no managed items are referring to the event. It is important to have this job scheduled.

This job runs every 15 minutes and is applied to the system and applications.

You may not want to enable the cleaning of events if you plan on setting event dates before retention is applied to records.

The clean search results phase of the Clean job can be quite expensive. Depending on the number of search results to clean up, it will be batched to improve throughput at the cost of higher system load. This can be configured using the Global Settings tab. When there are not enough search results to be cleaned up to create more than one batch, the operation will not be batched; otherwise, the operation creates up to the specified number of batches that run in parallel, if possible.

The minimum batch size is by default `1000`, but can be changed from the **Global Settings** tab with the `batch.chunk.cleanSearchResults` property.

The maximum number of batches is, by default, `10`, but can be changed from the **Global Settings** tab with the `batch.nrOfBatches.cleanSearchResults`.

For more information, see Global Settings.

It is possible to trigger events if the event used to be associated before retention was removed. If the Clean job runs with the `Empty Event Phase` property set, all events that are not associated with a record will be removed.

The following properties can be configured:

`All Phases`
Value to use for all phases. If selected, the other properties are ignored, and the Clean job evaluates all phases of work To disable certain phases, do not select this , and then ensure the phase to be skipped is not selected.

`Empty Event Phase`
If selected, the system deletes all events that are no longer referenced (retention removed).

If setting an event fulfillment (using the job or via REST) before records have been ingested, it is recommended that you disable the `emptyEventPhase` property.

`Reclaim Empty System Data Space`
If selected, PostgreSQL will reuse empty space in the datafile after the deletion/update of the records, which is done by running the `vacuum` command (refer to PostgreSQL documentation) on the OpenText Information Archive system databases.

During this phase, there is an optional step to truncate free pages at the end of tables (enabled/true by default). This can, however, trigger locking issues. To avoid these issues, disable the option in under the **Global Settings** tab's `cleanup` category. For more information, see Global Settings.

`AIP Phase`
If selected, the system deletes all PRUNE AIPs + rejected and invalidated AIP without commit date.

`Clean Library Phase`
If selected, the system deletes all orphaned libraries using POOLED mode.

`Compliance Partition Key Phase`
If selected, the system cleans up compliance partition keys used by the Refresh Metrics job.

`Empty Retained Sets Phase`
If selected, the system cleans up any empty retained sets.

`Order Item Phase`
If selected, the system deletes all expired order items.

`AIU Content Phase`
If selected, the system cleans the AIU content to free up space when pruning AIPs.

`Clean Keystore From Search Results Keys`
If selected, the system removes keys from search results from the keystore.

`Search Result Phase`
If selected, the system cleans up search results.

`Empty Hold Sets Phase`
If selected, the system cleans up any empty hold sets.

`Table Content Phase`
If selected, the system cleans up table content.

`Content Phase`
If selected, the system deletes all orphaned content.

`Clean Non Detachable Segments`
If selected, the system cleans up non detachable segments (SIP applications).

The Clean job is run by multiple (batch) order items that run in parallel. During or after Clean job execution, you can navigate to the order item. For each order item, view the log by clicking on the **Logs** tab. By default, only log messages of following types are displayed: error, warning and info. These messages will show per application if something is actually cleaned. Access the **Debug** tab to see which order items are skipped because there is nothing to clean.

3.7.9 Clean up Purge Candidate List and Applications job

Deletes purge candidate lists that have been canceled or disposed. The job also removes disposed applications after the Clean job has run.

This job runs weekly and is applied to the system only.

The following is the process to dispose of an application, assuming nothing in the application is under hold or under longer retention:

1. Run the Generate Purge Candidate Lists job.
2. Approve the purge candidates list for the application.
3. Run the Dispose Purge Candidate List job.
4. Run the Clean job.
5. Run the Clean Up Purge Candidate List and Applications job.

This job cleans up purge lists for applications, so it is recommended that purge lists processed by the Dispose Purge Candidate List job be reviewed before running this job.

3.7.10 Close job

Closes eligible libraries and aggregates for SIP applications. It is important to have this job scheduled only if the AGGREGATE or POOLED LIBRARY SIP ingestion modes are used. The recommendation is to schedule this job every 15 minutes. This job is meant for SIP applications only.

This job runs every 15 minutes and is applied to the system and applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

The Close job is responsible for:

- Closing any AIPs using AGGREGATE mode that which meet the conditions to be closed (according to the manually requested configuration). The job aggregates all its AIP child objects into a single AIP object, and attaches the children to the prune lifecycle.
- Closing AIPs that are using POOLED mode libraries that meet the conditions to be closed (according to the manually requested configuration).

The Close job takes a backup of any libraries for packages for holdings that use either pooled or aggregate modes.

Running the Close job requires disk space. Before running this job, ensure that you have enough disk space to properly store the individual libraries being closed.

To determine the amount of data, view the package you want to close:

Aggregate application

Access the application's **Packages** tab and select the package you want to close (for more information, see "Using the Packages tab" on page 79). Select **Library** in the panel on the right side of the page and note the **Stored AIU Count**.

Open the package to display the list of children that will be merged.

Assuming that all children hold the same amount of data, note the **CI Size** value for one of the children.

The disk should have at least **Stored AIU count** multiplied by **CI Size** in bytes available to properly store the closed libraries.

If the children's data varies and if the SIP is still available, the amount of data required on disk should be at least the size of all the SIP zip files.

Pooled application

Access the application's **Packages** tab and select the package you want to close. Select **Summary** in the panel on the right side of the page. The disk should have at least the **CI Size** value of available space.

If multiple packages are to be closed, add the sizes for all the packages. The sum is the minimum amount of disk space required.

The following properties can be configured:

`Phase To Process`

Controls what is closed.

- `POOLED_ONLY`: Only closes AIPs using pooled mode.
- `AGGREGATE_ONLY`: Only closing AIPs using aggregate mode
- `ALL_PHASES`: Closes all eligible AIPs in both modes.

`Clear Locks`

Force cleaning of Close job applicative locks

Note: If the Close job consistently fails with the error "CloseJob for application <applicationId> cannot run because lock is already taken by another job instance" and no other instance of the job is currently running, a cleaning of the applicative locks might be necessary. Launch the job once with the Clear Locks option is selected. After successful execution, it is advised to ensure this property is not selected.

3.7.11 Commit job

The commit process, in asynchronous mode, is invoked by running the Commit job. It will make available the package of the same DSS. When all SIPs pertaining to a DSS have been ingested (lifecycle state = Waiting Commit), executing the Commit job performs the following actions on every SIP in the batch:

- Attaches a commit log file content to the AIP for traceability purposes.
- Pushes the retention date, as well as content attributes, to the contents stored in hardware based retention, such as Dell EMC CAS Elastic Cloud Storage. For more information, see Section 9.2.2.2 "When are dates pushed to the hardware?" in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*
- Promoted the AIP to the completed state.

If retention is set for the archives and, if during the commit (asynchronous or synchronous), the retention could not be applied, the AIP is set in Waiting retention commit state. In this case, the Commit job handles such AIPs and applies retention to them.

This job is manually scheduled and is applied to the system and applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

Tip: It is important to have this job scheduled only if SIPs of the same DSS (seqno > 1) are ingested. The recommendation is to schedule this job every 15 minutes.

3.7.12 Confirmation job

Generates confirmations for a package. Which confirmations are generated are configured on the holding, and can include received, available, stored, or purged. Refer to step 5 of the procedure found in *Creating a holding with the holding wizard* in the *Configuration Guide* for more information.

This job can be canceled and retried. Refer to Canceling a job for further information.

By default, this job is scoped to the system and scheduled to run every 15 minutes. This job supports scoping to applications and the system.

Confirmation processing is very CPU-intensive and I/O-intensive due to the frequent queries on AIPs, and frequent writes to the working directory configured for the Confirmation job. To minimize its impact on system performance during

business-critical hours, schedule the Confirmation job to run in periods that are less resource-demanding.

Confirmation is a batch job and runs multiple batch order items in parallel. There are two types of batches:

- `CONFIRM_AIP`: Runs the confirmation for all AIPs that have a library set.
- `CONFIRM_AIP_WITHOUT_LIB`: Runs the confirmation for all AIPs without a library (in prune, invalidation, rejection state, etc.).

The following property can be configured:

`Cut Off Days`
: Optional argument ensuring a fast scan, even if many AIPs are stored in the repository. If this argument is present, then the date of the confirmation event must be greater than the (current date – the number of days (cutoff days)). For example, for the reception confirmation event, the AIPs taken into account for confirmation must have: reception date >= (current date - cutoff days).

3.7.13 Consistency Checker job

The Consistency Checker job performs various consistency checks for ingested packages, AIPs, and table documents. The job provides warnings for possible inconsistencies in packages and table documents and is, therefore, helpful for debugging purposes.

The Consistency Checker job starts with the reporting phase, which provides global information and configuration details of the system and the various applications deployed.

The next phase checks your system's upgrade readiness by performing the following:

- For a SIP application, capture the count of AIPs, saved searches, compliance objects (retention data), and search results (synchronous, asynchronous, and saved search results) on different Lucene codec versions.

  For a table application, capture the count of saved searches, compliance objects (retention data), and search results (synchronous, asynchronous, and saved search results) on different Lucene codec versions.

  If the count of any of the above components (AIPs, saved searches, search results, compliance objects) is found to be more than 0 on Lucene 9 (Lucene codec 90 and Lucene codec 95), the system issues a warning message in the order items logs. Based on these warnings, customers should follow these recommendations to migrate their Lucene data to the latest Lucene version:

Type of data	Recommended action
SIP structure data	Run the Post Ingest Processing job.
Compliance data	Run the Migrate Compliance Data job.

Type of data	Recommended action
Saved search	Run the Migrate Compliance Data job.
Background Searches	Re-run the searches.
Job and batch items	Delete them.

- A number of OTDS groups with the `ROLE_` prefix in the name were used to represent OpenText Information Archive roles. OTDS application roles have been introduced to represent the OpenText Information Archive roles. In a future release, the plan is to convert these legacy role groups to actual application roles in OTDS and migrate the group members. The system will remove any other members along with the legacy role groups themselves.

  As part of Consistency Checker product readiness option, the job logs a warning if any other type of members (users, roles, user partitions, organization units, *etc.*) are assigned to any the OTDS legacy groups used by OpenText Information Archive. The customer needs to migrate such members before upgrading to OpenText Information Archive 26.x.

After the job runs, click the link in the **Status** column and, to view the reporting results, access the **Report** tab. Click **Download report**. The report is downloaded as a GZIP file with the reporting results represented using the Markdown format.

If one or more order items are available, access the **Logs** tab and click **Download full diagnostic logs**.

This job is manually scheduled and is applied to the system and applications.

Note: Dashboard data will be available in the Consistency checker report only if the Refresh Metrics job has already been run at least once.

When scoped to the system, which is the default setting, the job performs consistency checks for each package in a completed state for SIP applications or table documents for table applications. The job can be configured to only run against specific applications. The default scheduling mode is Manual, but Interval and Expression modes are also available.

Tip: The Consistency Checker job is labor-intensive so it is recommended that it be run manually and only be scoped to certain applications. You can also disable one or more of the properties listed below.

The following properties can be configured:

`Random Mode (%)`
: Enter an integer between 1-99 (inclusive). Based on the user's input, the specified percentage of AIPs/table documents are randomly picked per application for the Consistency Checker job. Also, for each randomly selected AIP or table document, the specified percentage of CIs are randomly picked for the Check Unstructured Contents (CIs) check. If an incorrect value is entered, the base mode (default mode) is used to select AIPs or table documents for the job.

`Start Date (YYYY-MM-DD)`
Enter a date. All AIPs ingested after the specified Start Date, and are in Completed phase, are selected for the Consistency Checker job. This property is applicable only for SIP applications.

`End Date (YYYY-MM-DD)`
Enter a date. All AIPs ingested before the specified End Date, and are in Completed phase, are selected for the Consistency Checker job. This property is applicable only for SIP applications.

`Check Contents`
Checks consistency of contents and warns about potential orphans and missing content, if any. Validates content hashes and Lucene backup content, if applicable.

`Check Online Structured Data`
For a SIP application, checks that Lucene libraries are present in the database and validates the presence of Lucene indexes. Validates the AIU count for each online package and checks if duplicated AIU IDs exist.

For a table application, if the application is online, checks the existence and connectivity to the RDB of that particular application. No validation is performed if the application is offline.

`Check Offline Structured Data`
Checks that Lucene libraries are present in the database and validates the presence of Lucene indexes. Validates the AIU count for each offline package. Check if duplicated AIU IDs exist. As a part of this phase, the backup library of an offline AIP is temporarily restored except when it has been archived (moved to Glacier/Deep Archive storage class in S3 or Archive access tier in Azure). This property is applicable only for SIP applications.

`Check Unstructured Contents (CIs)`
Checks hashes of unstructured content and attachments that are part of packages, AIPs, and table contents. Running this phase can take a lot of time, as each CI is checked. For SIP applications, it is recommended to run in combination with `Random Mode (%)` (see above for more information).

`Check Retention Data`
Checks consistency of content object for granular hold/granular retention and warns about potential orphans and missing content, if any. It also checks whether the retention backup store is online and validates content hashes, if applicable. It also captures the count of retention objects on different Lucene codec versions

`Check Upgrade Readiness`
Described earlier in this section.

`Check Orphaned Lucene Indexes`
Checks and reports which orphaned Lucene indexes are on the file system for SIP and table applications. If enabled, this property detects the following types of orphans:

- STRUCTURED_DATA
- COMPLIANCE
- SAVED_SEARCH
- FIND_BY_ID
- SEARCH_RESULT

`Configuration Check`

Helps ensure the integrity and correctness of your library store configuration by performing automated checks and providing warnings for potential issues.

The system checks for unwanted files or folders in the library store's filesystem root.

All checks are based on the current store configuration. Warnings are provided to help identify and resolve configuration or data integrity issues promptly.

The following checks are performed:

1. **Detecting shared filesystem folders**: The system checks if two or more different library stores are configured to use the same filesystem folder. For example, StoreA and StoreB is pointing to same filesystem folder. If such a mis-configuration is detected, a warning is logged indicating a configuration issue.
2. **Validating store folder contents**: The scope of this check is the library store. If unwanted files or folders are found (for example, `test.txt` or `testfolder1`) and the first check (Detecting shared filesystem folders) succeeded without any warnings, a warning is logged for each unwanted file or folder.

You can enter a set of combinations for the `Start Date`, `End Date`, and `Random Mode (%)` properties. For example, if you enter both start and end dates, all AIPs ingested between the specified dates are selected by the Consistency Checker job.

Along with Start Date and End Date, if you enter a `Random Mode (%)` value, the specified percentage of AIPs that were ingested between the specified start and end dates are randomly selected by the Consistency Checker job.

You can also provide a combination of `Start Date` with `Random Mode (%)` or, conversely, `End Date` with `Random Mode (%)`.

For each AIP containing duplicated AIU IDs, the job will log the following message: "`Aip '<aip_id>' has duplicated Aiu Ids.`"

In this case, AIPs must be corrected using the `fix-duplicated-aiu-ids` command. This can be done through IA Shell, either for the whole application:

```
cd applications/<Application_Name>
> fix-duplicated-aiu-ids (--apply-ingest-granular-retention)
```

Or for one specific AIP:

```
cd applications/<Application_Name>/aips
> fix-duplicated-aiu-ids <aip_id> (--apply-ingest-granular-retention)
```

With or without the option to re-apply the granular retention as per ingestion configuration:

- If the option `--apply-ingest-granular-retention` is specified, all existing granular retentions on the package is removed and ingestion configuration for granular retention is re-applied.
- If the option `--apply-ingest-granular-retention` is not specified, granular retentions set on a record whose ID was duplicated will be duplicated on records with the new ID.
- In both cases, holds are always duplicated and package–level retention is kept.

Calling this action creates a background request which can be followed from the IA Web App.

3.7.14 Dispose Purge Candidate List job

Disposes items in approved purge candidate lists.

This job can be canceled and retried. Refer to Canceling a job for further information.

The default schedule is weekly. The job can be scoped to one or more applications and does not support system scoping.

When this job runs, it creates order items that can viewed from the job history. The following outlines the steps that occur during processing:

1. Determine what needs to be processed per approved purge list. The system issues a warning if any item in a purge list is no longer eligible for disposition, which occurs if a hold or longer retention policy was applied to the item.
2. Create roll forward objects (for disaster recovery).
3. Dispose. The system issues a warning if an item in a purge list could not be disposed because of an error, whether it is a package, table, or record.

   If there is a problem disposing a record during step 3 (the actual disposition step), the current batch and the job will not fail directly, but always try to dispose as many items as possible. In that case, the batch logs will include warnings about what happened to the problematic record, and the batch, order item, and the job will succeed with warnings.
4. Backup (backup is done per processed AIP or schema).
5. Dispose cleanup to clean up any intermediate objects used by disposition processing.

The job's logs include a summary of how many items were disposed. This could be a mix of packages and records, depending on the purge lists that were processed. For example, if two purge lists were processed, one containing five packages and other

containing 50 records, the logs will indicate that 55 items were processed. This is in addition to the summary reporting for each purge list that already was available.

Once an application is updated from active from in-test, this job becomes the only means to remove packages or records from the application, or remove the application. For more information, see Section 2.4 “Setting an application’s status to active” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

3.7.14.1 Canceling the Dispose Purge Candidates List job

The Dispose Purge Candidates List job can run for days, depending on the number of items contained in the list. OpenText Information Archive allows the job to be canceled. This may be required if, for example, the Retention Manager has neglected to place certain items in the disposition list on hold. The job can be canceled and, if those items have not already been disposed, a hold can then be placed on them.

When a job runs, it runs in phases. The cancellation of the Dispose Purge Candidates List job is only allowed up to and including the third phase of the job’s run, which is when the disposition process commences. Once disposition is done, the job cannot be canceled while the system cleans up the retention information and creates any necessary backups.

To cancel the job, click the **DisposePurgeCandidatesList** link and then click the **Cancel** button. If the button is not displayed, it is either too early in the job’s run to cancel the job or it is too late and the option to cancel the job is no longer available.

When requesting the job to cancel, click the **DisposePurgeCandidatesList** job link to view the order items and watch the progress as the order and batch items respond to the request.

If the Dispose Purge Candidates List job is canceled, the resulting purge list will be set back to the approved state. If some items had already been disposed, the purge list will show the remaining items left in the purge list. The Retention Manager can remove the approval from the purge list to prevent any of the remaining items from being disposed.

Caution

If you have canceled the job and want to prevent items from being disposed, and the Dispose Purge Candidates List job is not running on a schedule, do not click the **Retry** button if the job has entered the third phase (dispose). Instead, you have the following options:

- Revoke approval of the purge list.
- Apply holds to the records you want to retain.
- Disable all disposition processing for the retention policy.

If, however, you have canceled the job and want to prevent items from being disposed, and the Dispose Purge Candidates List job is running on a schedule, you have the following options:

- Suspend the schedule.
- Revoke approval of the purge list.
- Apply holds to records.
- Disable all disposition processing for the retention policy.

If you do not need to prevent items from being disposed, and the issue is resolved, you have the following options:

- Manually run the job again or wait for the schedule to commence.
- Click the Retry button.

Click the **Canceled** link in the **Status** column to review the job's log. Here, you can review which order items were processed prior to the job's cancellation.

Note: The log may not be available if the job was canceled before the batch had a chance to run.

3.7.15 Disposition RollForward Recovery

Runs the necessary disposition logic after a rollforward recovery. This job runs automatically, when required and cannot be scheduled or run automatically. If this job runs, it was because of a recovery operation for disposition. There are no properties for this job.

3.7.16 Generate Purge Candidate List job

Generates a purge candidate list for items that are eligible for disposition. It is important to ensure that the Dispose Purge Candidate List job runs on a similar schedule. The Generate Purge Candidate List job is the first step in the disposition process. The next steps are to approve the list for disposition and then run the Dispose Purge Candidate List job. The Dispose Purge Candidate List job will only dispose approved purge lists. If the Generate Purge Candidate List job is run before the Dispose Purge Candidate List job has run, any purge lists in the 'In Review' state will be canceled. Use the job properties to configure whether approved purge lists should be canceled.

This job runs weekly and is applied to the system and applications.

The following properties can be configured:

`Cancel Approved`
: If selected, the job cancels an approved purge list that was not disposed. For upgraded systems, this property is selected to match original behavior.

`Auto Approve`
: If selected, the generated purge lists should be automatically approved. If this job property is set, the next time the Dispose Purge Candidate List job runs, it will dispose the purge lists for the scoped application.

`Restrict Purge List Size ***DEPRECATED`
This property is deprecated and will be removed in a future release.

If selected, restricts the content of the purge list to a single partition key. For package retention, the partition key used is controlled by the holding wizard whereby a partition key can be created twice a day or for a year. For SIP records, the partition key is based on the parent. For table records, the partition key is based on how the records were in ingested. The default value is `FALSE`.

Tip: To have fewer generated purge lists, do not select this property.

3.7.17 Migrate Compliance Data job

Compliance data for granular retention and record ids of saved searches are stored in Lucene indexes. After upgrade to a newer version of OpenText Information Archive, these Lucene indexes may use an older Lucene codec than the current codec. In that case, you should use this job to upgrade the Lucene indexes to the current Lucene codec.

This job is manually scheduled and is applied to one or more applications.

This job can be canceled and retried. For more information, see Canceling a job.

The following properties can be configured:

`Maximum Number of Libraries`
Specify a maximum number of libraries to migrate per application.

`Migrate Compliance Data`
Check if you want compliance data to be migrated.

`Migrate Saved Searches`
Check if you want saved search record IDs to be migrated. For more information, see Section 4.3.1 “Search results and saved searches” in *OpenText Information Archive - Fundamentals Guide (EARCORE-ACS)*.

3.7.18 Post Ingest Processing job

After data has been ingested and application configuration has changed, the Post Ingest Processing job recomputes partition keys, indexes, library cache lock dates, and/or permissions.

This job is manually scheduled and is applied to SIP-based applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

The following properties can be configured:

`Holding Name`
The property specifies the holding name, for which the update job is run. If not specified, the job executes on all the holdings of the selected application(s).

`PDI Schema Name`
Specify a PDI schema name to reduce the number of AIPs/libraries to update during the job.

`Maximum Number of Libraries`
Specify a maximum number of libraries to update during the job. This number is applied to each selected phase.

`Update Library`
Updates indexes on already ingested AIPs, based on the actual PDI configuration.

`Force Library Codec Upgrade`
Upgrades Lucene Codec of indexes on ingested AIPs to the current Lucene Codec used by the IA Server during the update library phase, even if PDI configuration has not been upgraded.

`Update Partition Keys`
Updates partition keys on already ingested AIPs, based on the actual PDI configuration.

`Update Cache Lock Date`
Updates library cache lock date on already ingested AIPs based on the actual library-policy configuration.

`Update Permissions`
Updates permissions on already ingested AIPs based on the actual holding configuration.

`Force CI Text Indexation`
Re-index unstructured contents even if they are already indexed. Available only if the `ingestion.ci.text.enable` property in the **Global Settings** tab is set to true. See the Global Settings tab section for more information. Once set, this will be performed during the update library phase. The `Maximum Number Of Libraries` property cannot be used when this property is enabled.

There are four phases in which you would run the job:

1. **Update Library * * *** phase updates libraries to adjust search fields based on the latest PDI configuration. If you have updated the configuration to include a new search field, change a field type and/or full-text option, it is necessary to use this phase. The job can detect the changes and process only outdated libraries. This phase also updates the unstructured content extraction. If the content extraction is enabled or disabled, or if the configuration has been modified, it is necessary to use this phase to align the unstructured content extraction.

   When a library is updated during this phase, it is always upgraded to the current Lucene Codec used by the IA Server, if necessary. If the `Force Library Codec Upgrade` option is checked, all libraries that are not already in the current Lucene Codec are upgraded whether the PDI configuration has changed or not.

2. **Update Partition Keys * * *** phase updates AIPs to adjust partition keys based on the latest PDI configuration. If you have updated the configuration to edit

partition keys settings for one or more search field, it's necessary to use this phase. The job can detect the changes and to process only outdated AIPs.

3. **Update Cache Lock Date** * * * phase updates libraries to adjust cache lock date based on the latest library-policy configuration. If you have updated the configuration to update cache lock period, it could be necessary to use this phase.
4. **Update Permissions** phase updates AIPs to adjust permissions based on the latest holding configuration. If you have updated the configuration to change the permissions, it could be necessary to use this phase. It is not necessary to run this phase if you have edited an existing permission to add/remove some groups.

Tip: * * * After changing the configuration on the PDI or updating the cache lock date on the library policy, test your change by using the **Post-ingest** feature to a library object on the **Libraries** tab (for more information, see Using the Libraries tab). When the post-ingest feature is used, the system applies the update library, update partition keys, and update cache lock date phases on the library.

Reports Only mode: If none of the four phase options are checked, the job runs in "Reports Only" mode, meaning that no library or AIP is upgraded during the execution. The log will indicate which objects are eligible for upgrade. If the `Force Library Codec Upgrade` is checked, the job also reports if any library needs to have its codec upgraded.

Note: When a search field is not indexed, the record is not reachable. When a search field does not have full-text option, it will not be returned if using full text operators. If you plan to introduce a new search field or want to use a full-text operator, you should update the ingestion configuration and the search configuration to expose the new search field and/or full-text operators, then:

- You can run the Post Ingest job to re-index all libraries.
- Until all libraries are reindexed, the libraries that are not up to date with the ingestion configuration will be considered as out-of-sync by the system. As for cached-out libraries, if the user tries to launch a synchronous search that targets such libraries, the system will propose to launch a background search instead. Doing so will re-index the library in advance during the background search run to provide the results to the user as soon as possible.

If you plan to introduce a new partition key, it is also important to first update all AIPs before updating the search configuration with the new partition key.

If, during the Update Library phase, content is extracted for indexing, this text extraction does not happen within the IA Sever's JVM, but rather outside in separate processes called kvoop. We typically get two per attached piece of unstructured content for text extraction. These processes, albeit briefly, take some memory (typically up to 170 MB per pair), and if they are run in parallel, this can theoretically add up to problematic amounts.

For the Post Ingest Processing job, there are two ways to limit the number concurrently running kvoop processes:

- The global setting `batch.nrOfBatches.postIngestByLibraries` determines the number of batches (per application) of AIPs with associated unstructured content to extract text from (default: 10). As batches are the unit of concurrency for an order item, this setting implicitly limits the number of concurrently running kvoop processes.
- The YML setting `background.numberOfThreads.batchItems` determines the number of threads an IA Server has available for batch processing altogether.

If you are experiencing memory issues coming from text extraction on behalf of the Post Ingest Processing job, you can use the above two settings to reduce its memory consumption.

If the Post Ingest Processing job fails to run, there are two places to view logs that will identify the reason why the job failed, as well as identify which properties need to be updated prior to running the job again:

- Clicking the **Post Ingest Processing** job link in the **Job Name** column.
- Click the information button. The reasons why the job failed are displayed.
- Review the `<IA_ROOT>\iaserver\logs\*.log` files that contain information about updated AIPs, page handle number and AIP date ranges.

3.7.19 Process Retention Events job

Processes triggered events to update the qualification date for policy applications and the estimated disposition date for retained sets. Run this job after running the Apply retention rule to records job, if the rules could trigger events. The batch setting is only used for the order item that processes policy applications. The process retention events order item always uses a batch size of 1.

This job is manually scheduled and is applied to applications.

The following property can be configured:

`Retention Policy Name`
: To limit event processing to only one retention policy, enter the name of the event or mixed retention policy that was previously applied by one of the apply retention jobs. By default, calculates the projected disposition for all events that were triggered.

3.7.20 Refresh Metrics job

Updates the metrics for the dashboards and updates statistics for the Reports application. The calculation is based on the total number of records in the system. If any table applications are installed, including the Reporting application, every row in every table across all schemas and databases is considered a record. Because the sum is calculated based on the total number of records, the percentage of records under retention is zero, as a large number of rows in tables are usually not under retention.

Tip: If you have made retention–related changes to your data, wait approximately 10 minutes before running the Refresh Metrics job so the changes can be adequately captured in the Audit application.

Depending on how many items in the archive are under retention, it is recommended to run this job at off-peak times and run weekly if applying retention directly to records.

This job runs daily and is applied to the system only.

When running the Refresh Metrics job, any compliance changes (adding or removing retention or holds) within five minutes before starting the job, may not be reflected in the dashboard reporting.

This job can be canceled and retried. Refer to Canceling a job for further information.

There are three scenarios in which the information will not be available from the server and the UI will show an empty list of applications:

- If the customer upgrades the system but does not refresh the metrics.
- If the customer did not install the first time applications.
- The metrics were never refreshed.

Calculating the metrics information in the OpenText Information Archive dashboard can take a significant amount of time. Therefore, the dashboard retrieves most of its information from pre-populated values and the Refresh Metrics job populates these values. The job will scan the system and populate the metrics information. You can decide how often the metrics information should be updated, as it depends on the individual use cases.

Run the Refresh Metrics job to view the number of archived records versus the number of reference records, which counts the number of records that were initially ingested in a table application.

The storage metrics now include `FindByResults` in the Search Results category, which makes the calculation more accurate. It is important to note, however, that the search result size may be affected by this change.

When run, the Refresh Metrics job determines the compliance partition key information, which stores metrics per partition (for granular retention, there is a

partition per package or table). Next, the job determines the forecasts by only reviewing the partition keys that have disposition scheduled within the next seven months. Finally, the job updates the metrics and reports,

Applying a hold to records impacts the disposition projections.

This job is also responsible for updating the data for the report application.

The Refresh Metrics job contains the following metrics to the Reports schema:

- The number of containers and records were added for each application.
- The number of containers (packages) and records were added for each holding. Only AIPs that have been committed are included in the pricing metrics.
- The number of containers (tables) and records were added for each schema.

The (expected) counts of records was set 0 for the tables.

For SIP applications, the calculation for packages and number of records has been changed to only include packages in the Completed phase. Packages in the Prune and Aggregate phases are no longer counted. In the Reports tables, additional counts have been provided for SIP-specific counts.

If multiple applications are using granular retention, the Refresh Metrics job must complete a lot of calculations. Consequently, the job can take a long time to complete its run. Partition keys are used to streamline the job's run time. For each compliance-related partition key, the following changes are tracked:

- When a retention policy or hold has been applied to data.
- When a retention policy or hold has been removed from data.
- When data is disposed.

If no changes are made to a compliance partition key, the original calculated dates are used during the job's run. For disposition forecasts, only partition keys that contain qualification dates within the next six months are processed during the job's run. While this calculation is expensive because it is relative to the day the job is run, it ultimately serves to limit the run time.

If you are not interested in disposition forecasts, this option can be disabled in the job properties to improve performance.

The following property can be configured:

`Calculate Upcoming For Disposal`
: If selected, updates the Compliance dashboard with the latest data. If not selected, job execution time is quicker because it skips compliance disposition projection calculations.

3.7.21 Remove Policy job

Removes the named retention policy from items. Can be limited to a type of retention policy. Only one retention policy can be specified.

This job is manually scheduled and is applied to the system and applications.

This job can be canceled and retried. Refer to Canceling a job for further information.

The following properties can be configured:

`Retention Policy Name`
: Name of retention policy that will be removed from items. Retention policy must be defined and the name is case-sensitive.

`Type`
: Optional property that allows you to restrict which type of object to remove the policy from. If left blank, the policy is removed from the following types:

- `application`
- `aip`
- `aiu`
- `table`
- `row` (refers to a table row)

3.7.22 Requalification job

If retention policies are changed and the changes are retroactive, this job updates the retention information for all managed objects. Some storage systems may not support updating qualification dates. If multiple retention policies are changed, the job needs to run for each changed retention policy.

This job is manually scheduled and is applied to the system.

This job is batched and creates the following order items:

- `REQUALIFY_SET`: Deals with sets that age together.
- `PREPARE_REQUALIFY`: Prepares the system to determine which policy applications need to be requalified.
- `REQUALIFY_POLICY_APPLICATIONS`: Updates the dates.
- `REQUALIFY_MANAGED_ITEMS`: Updates the managed items, including the projected disposition date.

The following property can be configured:

`Retention Policy Name`
: Indicates the name of retention policy that changed.

3.7.23 Table Indexing job

Performs table indexing when requested via IA Shell commands and provides visibility on indexing efforts whenever table indexing operations are requested.

This job, which is applied to the system only, runs automatically and cannot be scheduled or run manually. If you attempt to schedule or run the job manually, the job will fail. Instead, trigger the job through IA Shell.

Table indexing can be stopped using the `index-stop` IA Shell command, but not through the job instance REST API.

This job is primarily available to track progress of ongoing and completed table indexing activities, but there is no need or ability to run this job directly either manually or on a schedule. This job only runs as a result of performing one of the table indexing commands via the IA Shell.

The following property can be configured:

`Batches To Define (Deprecated)`
: This property has been deprecated and should not be updated, as it will be ignored by the Table Indexing job. Instead, update the `batch.nrOfBatches.tableIndexing` in the **Global Settings** tab.

Note: If you previously updated this property, on upgrade, the new global setting will inherit the value you entered.

Note: One of the sub–tasks (order items) of the Table Indexing job is to extract text from attachments (unstructured content) linked from ingested records, if your application is configured to do that (for more information, see the *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*). This text extraction does not happen within the IA Sever's JVM, but rather outside in separate processes called kvoop. We typically get two per attached piece of unstructured content for text extraction. These processes, albeit briefly, take some memory (typically up to 170 MB per pair), and if they are run in parallel, this can theoretically add up to problematic amounts. For the Table Indexing job, there are two ways to limit the number concurrently running kvoop processes:

- The global setting `batch.nrOfBatches.extractTextFromTableUnstructuredContent` determines the number of batches (per application) of record attachments (unstructured content) to extract text from (default: 10).

  As batches are the unit of concurrency for an order item, this setting implicitly limits the number of concurrently running kvoop processes.

- The YML setting `background.numberOfThreads.batchItems` determines the number of threads an IA Server has available for batch processing altogether.

If you are experiencing memory issues coming from the text extraction on behalf of the Table Indexing job, use the above two settings to reduce its memory consumption.

3.7.24 Trigger Event Policy job

Triggers events using an XML file for records using event or mixed retention policies. Requires an XML file to be placed on IA Server to indicate the event, event context, and date that the event happened (or will happen).

Usually this is a product of another system that generates the event (for example, HR system would indicate when the employee left the company). The values of the trigger file contain the context that groups the records together. For example, if you are keeping employee records until 5 years after the employee leaves the company, then you would want to group the records around a common field (for example, context). The context in this case would be the employee number. When the event policy is applied, a context would have been specified. When the event needs to be triggered, a context and a trigger date need to be specified.

It is required that you run the Process Retention Events job to force requalification after this job has finished.

This job is manually scheduled and is applied to applications only.

The following property can be configured:

`Trigger File`
: This is a path to a trigger file that contains a list of triggers (context, trigger date and condition).

  The location is relative to where the server is deployed. If multiple IA Servers are installed, it is recommended that you use a network location (versus a local path).

Example 3-2: Example of a triggerFile file:

```
<?xml version ="1.0?"
<triggers>
  <event>
     <context>00457</context>
     <triggerdate>2010-01-31</triggerdate>
     <condition>Condition></condition>
  </event>
  <event>
     <context>00345</context>
     <triggerdate>2014-02-28</triggerdate>
     <condition>Condition</condition>
  </event>
<triggers>
```

3.7.25 Populating event dates for the Trigger Event Policy job

An XML file is used to populate event dates for the Trigger Event Policy job:

`context`
Enter the context for the event (for example, employee number).

`triggerdate`
Enter the date when an event happened or is planned to happen. To learn which date formats are accepted, see the note in Apply Retention Rule to Records job. A date must be within the following range: 1000-01-01 – 2999-12-31.

`condition`
Enter the name of the condition, which must match the condition specified on the retention policy. The value is case sensitive.

The following illustrates how to populate the XML file:

```
<?xml version="1.0"?>
<triggers>
<event>
<context>89</context>
<triggerdate>2016-02-28</triggerdate>
<condition>tradeversion</condition>
</event>
<event>
<context>77</context>
<triggerdate>2016-02-28</triggerdate>
<condition>tradeversion</condition>
</event>
</triggers>
```

3.7.26 Trigger Event Rule job

Note: This job is only available if rule-based processing has been enabled for your system.

Provides the ability to use rules previously uploaded for trigger event rules to start aging for records under an event or mixed retention policy. Run the Process Retention Events job to force re-qualification after this job has finished.

This job is manually scheduled and is applied to applications.

The following properties can be configured:

`Rule Name`
Can be used to limit the job to only use the one name. If not specified, all rules of type `TRIGGER_EVENT` are evaluated.

`Developer Mode`
If selected, developer mode runs the rule, but does not apply the rule. This property is meant for testing.

To learn which date formats are accepted, see the note in Apply Retention Rule to Records job.

3.7.27 Upgrade jobs

There are two upgrade jobs that impact both asynchronous and synchronous tasks. The tasks the job updates change from release to release.

UPGRADE: Execute Asynchronous Upgrade Tasks

The previous upgrade jobs have been converted to asynchronous upgrade tasks and automatically run by this job. These asynchronous upgrade tasks typically perform cleanup that are not strictly necessary for the current version of OpenText Information Archive to function properly.

Some of this work might be optional and purely for optimization purposes, although most of the work is expected to be complete before upgrading to the next version. This job automatically starts after the upgrade process, when startup is no longer blocked, and runs in the background on an IA Server node configured to run jobs.

If any of these tasks should fail, this job fails, although it performs a best effort to complete as many tasks as possible. Any failing upgrade tasks can be retried by running the job again after addressing the underlying problem(s). While this job runs, or if not, all tasks completed successfully, you might experience reduced system performance, but this is not expected to cause functional problems.

Each task can be run in isolation or in combination with others by using the corresponding property checkboxes in the IA Web App. This can be done as many times as necessary. In case any task has failed during the original run, however, it is required to, after addressing all problems, run the job once more with all tasks enabled to properly register that it has completed successfully, which allows you to upgrade to the next version.

This job has one special option called *`Ignore Optional Task Failures`*. If checked, failures of optional tasks will not block upgrading to the next version. This should only be relied on as a workaround if truly necessary to avoid skipping potentially important optimizations that might need additional attention to be applied properly.

UPGRADE: Execute Synchronous Upgrade Tasks

This job automatically runs the first time the first IA Server node starts after installing a new major version of OpenText Information Archive. All available tasks run this first time, although which specific tasks run depend on the version of OpenText Information Archive being started and the version of being upgraded from.

IA Server cannot start until this job is successfully finished. If this job fails during this automatic run, the problems need to be resolved either outside of OpenText Information Archive, or in the previous version.

Under normal circumstances, there will be no need to run this job manually, but customer support might identify a need run it to resolve specific problems. In the event of this scenario, use the available properties in the IA Web App to control which tasks are run. If you need to rerun the job manually, run it in a period of low activity to minimize the risk of lock timeout during ingestion procedures, searches, or job executions.

3.7.28 Canceling a job

If any of the following jobs are taking too long to complete or were started in error, they can be canceled:

- Apply Hold Rule to Records
- Apply Retention Policy To Records
- Apply Retention Rule to Records
- CacheOut
- Apply Retention Rule to Records
- Clean
- Close
- Commit
- Confirmation
- Dispose Purge Candidate List (refer to Canceling the Dispose Purge Candidates List job for more information)
- Invalidation
- Migrate Compliance Data
- Post Ingest Processing
- Refresh Metrics
- Remove Policy

When a job runs, it runs in phases. The cancellation of any of these jobs is only allowed up to final phase of the job's run. Once the job reaches the final phase, it cannot be canceled while the system processes the compliance data.

To cancel a job, click the job link and then click the **Cancel** button. To ensure you do not cancel the wrong job instance, you will be prompted to verify that the selected job should be canceled.

If the **Cancel** button is not displayed, it is either too early in the job's run to cancel the job or it is too late and the option to cancel the job is no longer available.

When requesting the job to cancel, click the job's link to view the order items and watch the progress as the order and batch items respond to the request.

Click the **Canceled** link in the **Status** column to review the job's log. Here, you can review which order items were processed prior to the job's cancellation.

Note: The log may not be available if the job was canceled before the batch had a chance to run.

Once any issues have been resolved, and you want any of the following compliance-related jobs to continue its run, click the **Retry** button:

- Apply Hold Rule to Records
- Apply Retention Policy To Records
- Apply Retention Rule to Records
- Dispose Purge Candidate List
- Refresh Metrics
- Remove Policy

3.7.29 Using the Jobs tab

Jobs are displayed in a table that contains the following information:

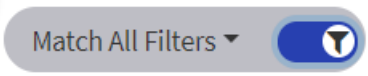
Job Name	Indicates the name of the job. Click the name of a job to view its run history. For more information, see Viewing a job's run history.
	A menu that allows you to: • Run a job or starting a schedule. • Edit a job. • Create a duplicate of the job. • Suspend a job. • Inactivate a job.
Applied To	Indicates the scope of the job and whether it is applied to: • System • Specific Applications: Where relevant, the list of applications is filtered by archive type to ensure no applications can be specified to which the job is not applicable.
Next Run	Indicates the next scheduled run of the job.
Last Run Start	Indicates the date and time the job was last run.
Status	Indicates the status of the last execution of the job. Possible values include: • Canceled: The last job instance was canceled • Scheduled: The job is scheduled to run. • Running: The job is currently running. • Success: The last job instance that was successful. • Failure: The last job instance that failed. • Skipped: The job was skipped.

History	If a job instance encountered any errors, the following is displayed:  If a job instance encountered any warnings, the following is displayed:  If the job instance encountered both warnings and errors, the error icon is displayed.

An **Information** tab contains the custom properties of a selected job.

To locate a specific job, use the **Find a Job** field to search for a job. Narrow the search even further by selecting one of the values in the **Job Scope** field (**All**, **Application**, or **System**).

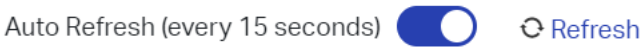
Furthermore, activate the column filtering for the **Jobs** tab:



You have the following options:

- **Match All Filters**: Enter criteria in each of the filters available on the **Jobs** tab: **Job Name**, **Description**, and **Handler**.
- **Match Any Filter**: Enter criteria in any of the filters available on the **Jobs** tab.

By default, the **Jobs** tab is refreshed every 15 seconds. To disable auto refresh, turn off the **Auto Refresh** switch. To manually refresh the tab, click the **Refresh** link.



3.7.29.1 Viewing a job's run history

If a job has run recently, you can click the link in the **Status** column to view the run history. A job's run history is typically cleaned up after 48 hours, although this time is configurable. A job's run history contains the following information:

Scheduled Date
Indicates the date of the job schedule.

Scheduled By
Indicates the name of the person who initiated the job run.

Start Time
Indicates the time the start time of the job run. If a job has expired, the time the job was expired is displayed here.

End Time
Indicates the end time of the job run. If a job has expired, the time the job was expired is displayed here.

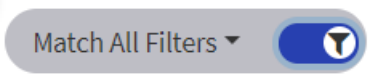
Status
Indicates the status of the last execution of the job. Refer to the different status levels in Using the Jobs tab.

Below the status, a message may be shown depending on the type of job and runtime execution. In that case, the message itself or the job logs may provide additional details regarding the end result status of the job.

Application Name
If the job was applied to an application, the name of the application is indicated.

You can delete a selected job instance or all job instances. You cannot delete a job instance if the job is currently running.

Activate column filtering for the job selected:



You have the following options:

- **Match All Filters**: Enter criteria in each of the filters available in the job's run history: **Scheduled Date**, **Scheduled By**, **Start Time**, **End Time**, and **Application Name**.
- **Match Any Filter**: Enter criteria in any of the filters available in the job's run history.

Click **X** to delete a particular job instance. You will be prompted to confirm that you want to delete the selected job instance.

Click **X Clear All Completed** to delete all completed job instances (which are jobs that finished running). You will be prompted to confirm this action.

It is important to note that, when you click **X Clear All Completed**, the system does not stop scheduled job instances. It is not possible to delete a scheduled job instance. You can, however, stop the schedule.

3.7.29.2 Cloning a job

Cloning a job is useful when you want to schedule the job to run for different applications using different properties.

Administrators can clone any job definition. Retention Managers can only clone Compliance job definitions and if there are any applications that they do not have access to or do not have Retention manager access, they will not be able clone jobs that only support system scoping. This is similar for a Developer except Developer can only clone Data Jobs.

Note: If enabled, The `permission.restrictRolesToApplicationGroups` global setting to restrict group permissions impacts what the Developer and Retention Manager can do with jobs.

To clone a job:

1. On the **Jobs** tab, click **+**.
2. Select the job type you want to act as the template for the job being created. The new job will inherit all configuration and property values from this existing job.
3. Click **Next**.
4. Enter the following information:

   Job Name
   Enter a unique name for the job.

   Log Level
   Set one of the 7 log levels for the job, which controls the threshold level for messages that are written in the diagnostics log files.

   Applied To
   Specify the scope of the job and whether it is applied to:

   - **System**: If selected, the job will not be dependent on any application and the JobHandler will not receive the application during execution.

     If prompted, redefine values of the job properties.

   - **Specific Applications**: If selected, select the applications the job can be applied to. Click **Select all** to select all of the applications. The selected applications appear in a list below the **Applications** window.

     – **Select All** selects all the items in the current list. If a user creates a new application and wants the job to run for this new application, open this page and select the new application.

     – **De-select All** removes all previously selected applications.

   This field may be read-only if either the job does not support system or application scoping, or you do not have access to all applications (for a Developer or Retention Manager).

   For any role other than an Administrator, the list of applications may be limited based on your permissions.

5. Further configure the job by updating the information in the **Properties** section. Refer to each specific job and its available properties in Jobs.
6. Click **Next**.
7. Enter the following information:

Scheduling Mode
Indicate the if job is to run:

- Manually
- Interval: If selected, enter
  - Interval: Specify the number of minutes that must pass before the job can be repeated. If you configure a job to run every minute on an interval, it does not mean the job will run every minute, It means it will run a minute after the last job finishes.
  - Expiration Interval
  - Retry Interval: Indicate the number of minutes the job will try to start in regard to its schedule. If it is not able to start in this interval, it will stop to try and wait for the next schedule. -1 means that it will retry indefinitely. If the execution instance, indicate the number of minutes the server should wait to reschedule the job.
- Expression: If selected:
  - Enter the expression that defines the job schedule by specifying when the job will run by the second, minute, hour, day of the week, day of the month, month or any combination of these options. The expression supports “Cron Expression” syntax.

    Once the expression has been entered, the system indicates the job's next run date and time.
  - Indicate the maximum number of attempts the job instance will be rescheduled after failing to run successfully.
  - Expiration Interval: Indicate the number of minutes the job will run before logging a failure.
  - Retry Interval: After an instance execution fails, indicate the number of minutes the server should wait to reschedule the job.

Priority
Define the priority of the job to be run in the job executor. By default, the priority is `0`. It should be set accordingly to the order item server priority.

8. Click **Next**.
9. Review the information you have entered. When satisfied that the information is correct, click **Finish**.

The job now appears in the table on the **Jobs** tab.

3.7.29.3 Editing a job

Caution

If you schedule a job, and then subsequently change the job's duration, you must schedule the job again.

When you edit a job's properties, OpenText Information Archive leverages server information about the type of property for a job. For example, if the property requires a Boolean value to be entered, click the check box to set the value to True. When specifying an interval, it must have a minimum value of 1. If the job property has a fixed set of possible values, a selector is provided to allow choosing only a valid value.

If you modify a job to run on an interval and set the duration to '0', the system will treat the job as if it has been set to manual.

When determining how often a job runs, consider the fact that running a job too often can impact system performance.

Note: If enabled, The `permission.restrictRolesToApplicationGroups` global setting to restrict group permissions impacts what the Developer and Retention Manager can do with jobs.

To edit a job:

1. Select **Edit Job** from the job's menu on the **Jobs** tab.
2. Edit the following information:

Log Level
Set the log level for the job, which controls the type of messages that are written to the various log files.

Scheduling Mode
Refer to the options in the Cloning a job section.

Apply To
Specify the scope of the job and whether it is applied to:

- **System**: If selected, redefine values of the job properties.
- **Specific Applications**: If selected, select the applications the job can be applied to. Click **Select all** to select all of the applications. The selected applications appear in a list below the **Applications** window.
  - **Select All** selects all of the items in the current list. If a new application is created, and you want the job to run for this new application, open this page and select the new application.
  - **De-select All** removes all previously selected applications.

3. Further configure the job by updating the information in the **Properties** section. Refer to each specific job and its available properties in Jobs.

4. Click **Update**.

3.7.29.4 Duplicating a job

If enabled, The `permission.restrictRolesToApplicationGroups` global setting to restrict group permissions impacts what the Developer and Retention Manager can do with jobs.

To create a duplicate instance of an existing job:

1. For the job being duplicated, click the **down arrow > Create Duplicate**.

2. Edit the following information:

   Job Name
   Enter a unique name for the job.

   Description
   Enter a description of the job being created.

   Log Level
   Set the log level for the job, which controls the type of messages that are written to the various log files.

   Apply To
   Specify the scope of the job and whether it is applied to:

   - **System**: If selected, redefine values of the job properties.
   - **Specific Applications**: If selected, select the applications the job can be applied to. Click **Select all** to select all the applications. The selected applications appear in a list below the **Applications** window.
     - **Select All** selects all the items in the current list. If a user creates a new application and wants the job to run for this new application, open this page and select the new application.
     - **De-select All** removes all previously selected applications.

3. Further configure the job by updating the information in the **Properties** section. Refer to each specific job and its available properties in Jobs.

4. Click **Next**.

5. In the Set Schedule step, indicate the if job is to be run:

   - Manually
   - Interval: If selected, enter

  - Interval: Specify the number of minutes that must pass before the job can be repeated. If you configure a job to run every minute on an interval, it does not mean the job will run every minute, It means it will run a minute after the last job finishes.
  - Indicate the maximum number of attempts the JobInstance will be rescheduled after failing to run successfully.
  - Expiration Interval: Indicate the number of minutes the job will run before logging a failure.
  - Retry Interval: After an instance execution fails, indicate the number of minutes the server should wait to reschedule the job.

- Expression: If selected:

  - Enter the expression that defines the job schedule by specifying when the job will run by the second, minute, hour, day of the week, day of the month, month, or any combination of these options. The expression supports “Cron Expression” syntax.

    Once the expression has been entered, the system indicates the job's next run date and time.
  - Indicate the maximum number of attempts the JobInstance will be rescheduled after failing to run successfully.
  - Expiration Interval: Indicate the number of minutes the job will run before logging a failure.
  - Retry Interval: After an instance execution fails, indicate the number of minutes the server should wait to reschedule the job.

6. Click **Next**.
7. Review the information that you have entered and click **Finish**.

3.7.29.5 Running the job ad hoc

It is possible to run a job without starting the schedule.

- When you select **Run** for a particular job, you are provided with a screen to review and update the job scoping and job properties.

  Note: These changes are for this run only and do not affect any scheduled jobs.

- When you select **Run now** for a particular job, the job runs based on the information defined in the job definition and ignores the scheduling mode.

  This option is only available if you are able to run the job with the current settings. For example, you are a Retention Manager and want to run the Apply Retention Policy To Records job. The Administrator, however, last configured the job to an application that you cannot access. The Run Now option will not be available in this instance. Instead, you can configure the job to an application you

can access and select Run or duplicate the job and configure it for an application you can access.

Note: If a cron expression is specified, the **Run now** command schedules the job according to the specified cron expression but will only run once. If you want to run the job immediately, use the **Run** command instead of the **Run Now**.

If the job is scheduled, running a job with the same scope may cause the schedule job to be skipped if the schedule indicates the job definition should run. Note that the schedule will continue based on the original scheduling. For example, suppose the Clean job was scheduled to run each hour and a user selects **Run Now** (using the same scope). If the job was supposed to run at 7PM, but the ad hoc instance is still running, the job will be skipped and will be scheduled to start at 8PM.

3.7.29.6 Inactivating a job

When a job definition is inactive, job execution is disabled. This means that the moment a job run starts, it skips the actual execution and the job's state is updated to 'skipped'. To perform this action, select **Inactivate** from the job's menu.

This can be useful to prevent specific jobs from running, regardless of who or what asks them to run (for example, to prepare for potentially disruptive changes, upgrade procedures, *etc.*).

To have such jobs run normally again, select **Activate** from the job's menu to reactivate the job.

3.7.29.7 Stopping and suspending a job schedule

Suspending and resuming a job is like the Inactivate/Activate feature except that it only affects scheduled job instances. Unlike inactivate/activate, manual requests to run a job are unaffected by this operation

This is to be used when you want to effectively pause the schedule without resetting it. This is mostly useful for interval-based schedules, but it also applies to cron-based ones.

Any job instances that are part of a schedule will skip execution if the job definition is 'suspended' and will start running again when the job definition schedule is 'resumed'.

Manual job instances are not part of a schedule and, therefore, do not skip execution, even if a job definition is running on a schedule and has been suspended.

Starting or stopping the schedule clears the flag on the job definition if it was suspended.

Access the menu for a particular job to stop, suspend or start the job's schedule:

- **Stop Schedule**: Stops and clears the schedule for a job definition. This function is not available for jobs where the scheduling mode is manual.
- **Suspend Schedule**: Suspends the job's schedule. This function is not available for jobs where the scheduling mode is manual. Click **Resume Schedule** to remove the suspension. This function does not prevent manual runs of the job. Instead, use the Inactivate function to prevent the job from running (refer to Inactivating a job).
- **Start Schedule**: Starts the schedule for a job definition. This function is not available for jobs where the scheduling mode is manual. If the schedule has already started, the schedule will be reset and start on the new schedule.

3.7.29.8 Deleting a job

You are only able to delete a job definition that you either duplicated or clone and only if it the job is not running or scheduled. While on the **Administration > Jobs** tab, click **X** to delete a particular job definition. When prompted to confirm the deletion, click **Delete**.

The job definition no longer appears in the **Administration > Jobs** tab.

3.7.30 Job scoping

Jobs can be system-scoped or scoped to one or more applications. Depending on the job, it may support one or the other or both.

3.7.30.1 Application scoping

If a job definition is application-scoped, when the job is run or scheduled, a job instance is created for each application that was specified.

If the job was scheduled, the job will normally automatically create new instances for the next run (for each scoped application), depending on the status of the job, the job may not be scheduled.

The UI no longer supports the deletion of scheduled job instances. Instead, stop the schedule, as this will remove all the scheduled job instances for each of the scoped applications.

If an application is deleted or disposed, for any job definitions that were scoped to the application, the job definition will no longer be scoped to that application. If this was last application-scoped to the job, the job cannot be run.

3.7.30.2 System scoping

If a job is system-scoped, only one job instance will be created when the job is started or scheduled.

3.7.31 Reviewing the logging information for jobs

When a job is started, the following information may be available in the first log entry for the job instance:

`name`
Indicates the name of the job. This field is always present.

`handler`
Indicates the handler for the job. This field is always present. For jobs that were cloned, this field helps identify what the job does.

`properties`
Indicates the properties that the job execution used. Job properties are shown in curly brackets (for example: `type =aip, retentionPolicy=Policy A`).

`scheduled by`
Indicates the user who ran job. This field is not shown for scheduled jobs.

`application`
Indicates the application that the job instance was scoped to.

`attempt`
Indicates the attempt for running the job. This field is always present. Value is always one unless the job has been configured for auto retry on failure. Does not increment when doing a manual retry.

The following is an example of the log entry after the execution of the Remove Policy job:

```
[Jan 3, 2018 4:48:17 PM]: INFO

Starting job: name=Remove Policy, handler=RemovePolicyJob, properties={type=,
retentionPolicyName=Policy A}, application=PhoneCallsGranular, scheduled by
sue@iacustomer.com, scheduled at 2018-01-03T21:48:17.804Z, attempt=1
```

3.7.32 Troubleshooting issues with jobs

Why is the option to download logs for jobs not provided?
Even if the job is scoped to an application, the logs for jobs are stored in the System application. If the System application is not installed, the option is not provided. If this does happen, those logs can be accessed on the machine in which IA Server resides under the `logs > iaserver > context-sifting`.

Why are the Generate Purge List job or Clean Up Purge Candidate List and Applications job failing?
The Generate Purge List job and Clean Up Purge Candidate List and Applications job require that the System application is installed.

Why is the Dispose Purge Candidate List job failing?

The Dispose Purge Candidate List job is now application-scoped and requires at least one application to be configured for it. If the Dispose Purge Candidate List was scoped to an application and the application is disposed, the Dispose Purge Candidate List job will not run unless it is associated to another application.

Why is my application still being shown after running the Dispose Purge Candidate List job twice?

The process for disposing an application has changed. Instead of running the Dispose Purge Candidate List job twice, the Clean Up Purge Candidate List and Applications' job is responsible for finishing disposition after the Clean job is run.

A second reason is that application had items under hold or longer retention and marking the content in the application was not done.

Why does the Generate Purge List job fail to run and, instead, results in an exception error?

This issue occurs if a large operation is being performed (for example, a hold is being applied to a large set of records).

If you receive the exception, try running the Generate Purge List job after the large operation is complete.

It is recommended that, when doing operations that will affect many records under retention or hold, to do this during off-times to avoid contention for locking objects. If the apply hold operation fails due to a LOCK_NOT_GRANTED exception, try the operation later.

Why do I receive an error message when I click Retry to rerun a job?

Refresh your browser to see if the job is already running.

Why do I have several of the same job handlers?

Any time a job is duplicated, there will be multiple jobs using the same handler.

Why does my job fail with a locking issue during what is clearly a read-only operation?

This may be related to the Clean job running at the same time, with the truncation option for system data cleanup enabled. To learn how to disable this, see the Clean job's `Reclaim Empty System Data Space` property.

3.8 Background requests

This section discusses the **Background Requests** tab. For more information, see Section 3 “Working with background requests” in *OpenText Information Archive - End User Guide (EARCORE-UGD)* and Section 10.2 “Improving the response time for a background request” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

When the following types of tasks are initiated, OpenText Information Archive can process them asynchronously as background requests, and the results are displayed on the **Background Requests** tab.

These are the types of background requests:

Aviator Indexing
If a user engages with Aviator, OpenText Information Archive must perform indexing on the search results as a background request.

Backup
Back up a resource. You can invoke this action from IA Shell using the `backup-resource` command. For more information, see Section 2.7.2 “backup-resource” in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*.

Cache In
An action on an AIP object to restore the object in a library from backup.

Cache Out
An action on an AIP object to detach the object from a library.

Create Cross-Application-Matter
An action that can be done on a delegate search to create a matter for all of the delegates in a cross-application search.

Custom Indexes
All custom indexes, except the custom indexes on unstructured content, are created. These custom indexes are defined in the table metadata.

Custom Indexes Extracted Text
The custom indexes on extracted text, defined in the table metadata, are created.

Delete Application Data
Delete all the data from an application.

Delete Table
Remove a table from a table-based application database. For more information, see Table management.

Export
Exports search results. This background request is not generated if the selected export configuration accepts instant export and number the number of records is less than the threshold. For more information, see Section 6.6.12 “Configuring search result export” in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

Export Matter
Exports a matter, including all saved searches.

Ingest AIP
Asynchronously ingest an AIP package. You can invoke this action from IA Shell using the `ingest` command. For more information, see Section 2.7.5 "ingest" in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*.

Rebuild SIP
Convert an ingested package back to the original SIP.

Recovery
Recover a resource. You can invoke this action from IA Shell using the `recover-resource` command. For more information, see Section 2.7.12 "recover-resource" in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*.

Refresh Compliance Summary
Request the summary of the compliance information for an application.

Refresh Holds of Matter
Refreshes hold of matter without rerunning the matter.

Remove items from Saved Search
Removes items from a saved search.

Rerun Matter
Reruns the saved searches defined in the matter.

Request Closing
When a group of libraries are selected and a close requested (for SIP applications that are pooled or aggregate).

Set Mek Alias
Updates the Mek alias initiated by an IA Shell command. Meant for Key Mediator integration.

Set Application Offline
Reduce the database footprint by taking an application offline. Available only for table-based applications. Can be initiated by either the UI or IA Shell.

Set Application Online
Brings a table application back online.

Text Extraction/Ingestion
The text from unstructured content is extracted and stored.

Update Holds on Matter
Ensures that all the records in saved searches have the holds defined on the matter.

Update Holds on Saved Searches
Ensures that all the records in the saved search have the holds applied to directly on the saved search.

By default, the **Background Requests** tab is refreshed every 15 seconds. To disable auto refresh, turn off the **Auto Refresh** switch. To manually refresh the page, click the **Refresh** link.

In addition to the functionality that is available to End Users when finding background requests, Administrators can find background requests that have been initiated by other users. A **Find by user** box is available, and you can use it with the other elements on the **Background Requests** page to find a particular request.

For more information about working with background requests, including canceling or deleting background requests, see Section 3 "Working with background requests" in *OpenText Information Archive - End User Guide (EARCORE-UGD)*.

To find a background request by user name:

1. On the **Background Requests** tab, select the **All users** check box to show background requests that have been initiated by all users. The **Find by user** box also appears.
2. Click the **Contains** list and choose the type of match that you want:
   - **Contains**
   - **Exact Match**
   - **Begins with**
   - **Ends with**
3. In the **Find by user** box, type the full or partial name of the user whose request you want to find and press **ENTER**. This box is only case sensitive if you have chosen to search for an exact match.

3.9 OpenText Information Archive batch framework

Many background operations in OpenText Information Archive are using batch processing to improve performance and scalability.

A batch operation performs the business logic using one or more batches that can potentially run in parallel. During a batch operation, the progress indicated in the IA Web App is based on the number of items that have been processed.

The batch framework can be configured using the global settings for batch, which support the following subsections:

`interval`
: The frequency in milliseconds at which to poll/check for progress/status updates of the overall operation.

`nrOfBatches`
Specifies the desired number of batches to create.

Tip: Creating more batches than what can be run in parallel in the entire environment is not useful. Depending on the expected load and total number of background processing IA Server nodes, it might be better to create fewer batches to support multiple simultaneous operations or reduce the overall load on the system. Fewer batches per operation might still be created for small data sets, depending on the configured chunk size.

`chunk`
Generally, controls the transaction size within each batch, if applicable. Each batch generally commits and starts a new transaction after processing this many items. The exact behavior/meaning of changing this setting may differ per operation type.

The minimum batch size is implicitly equal to the chunk size.

Each of these subsections supports optional configuration per type of batch operation.

All items processed within the same chunk are committed automatically. This allows for more graceful recovery in the event of interruption, cancellation, or failure.

When a batch operation fails or is cancelled, it can typically be retried, in which case, only the failed or canceled batches are rerun. Furthermore, if for whatever reason, the IA Server node running a certain batch operation and/or an individual batch goes down, other IA Server nodes eventually resume the operation and/or restart the ended batches automatically.

A fair number of jobs use one or more batch operations internally. In these cases, because it is the job that manages the batch operations, the individual batch operations cannot be accessed individually, nor be retried explicitly. Usually, such jobs can be manually retried after a failure though, in which case they internally restart the failed batch operations which, in turn, retry only the failed batches.

3.9.1 Viewing batch and log information

The following example demonstrates how you can view the batch information and logs for a particular job instance or order item. This scenario involves the successful run of the Apply Hold Rule to Records job:

1. When the **Last Run Status** column states Success, click the **Apply Hold Rule to Records** link.
2. In the **Status** column, click **Success**. A new page is displayed.
3. In the **Status** column, click **Complete**.

Now you can toggle between the batch information and log for this particular job instance. You can also download the diagnostics log.

Tip: For each order item, the **Batches** tab contains **Duration** time and **Total Execution Time**.

Whether additional system resources might help improve performance depends on the difference between the **Duration** time and **Total Execution Time**. When these two values are close together, it could be an indication of a system resource bottleneck and additional resources could help improve throughput. A high duration or high total execution time alone is not necessarily an indication of anything other than it simply being an expensive operation.

When large datasets that result in multiple batches are run in parallel, the total execution time will usually be considerably higher than the duration. If the total execution time is not considerably higher than the duration, it indicates that there is not much work being performed in parallel, which is not necessarily indicative of a problem. This could be because of a lack of batches (not enough data for the operation to be performed in parallel), or due to system load preventing batches being run in parallel. Sometimes total execution time could be less than duration, because this only represents the sum of individual batch execution, but it does not include preparation, cleanup and other batch framework overhead. This would usually be the case for small datasets and a few, small batches.

3.10 User accounts and permissions

3.10.1 Mapping groups to OpenText Information Archive roles

The **Groups** tab allows you to configure which groups can access specific OpenText Information Archive functionality. For example, you can permit a user in the Administrator group to perform a compliance task, such as creating a retention policy.

Note: In previous OpenText Information Archive releases, the Developer could configure the group to role mapping using the **Groups** tab. While the Developer can still access the Groups tab, they are not able to update group and role mappings.

Click the icon to learn what actions can be performed by each user role.

A check mark specifies whether a group can perform the actions of a specific user role.

A drop-down list allows the Developer or Administrator to toggle between the following:

- Show all groups
- Show only groups with assigned roles
- Show only groups without assigned roles

- Show only obsolete groups with assigned roles (if the OTDS SSO profile is enabled, this option will not be displayed).

3.10.2 Managing permissions

OpenText Information Archive provides the ability to set permissions on the following objects:

- Applications: Application permissions are described in the following section.
- Search sets: Searches provide the ability to create search sets in which a search designer creates different views of the search (showing different columns based on the group).
- Retention Policies: You can set permissions on retention policies only by declarative configuration (DC). For more information, see Section 8 "Declarative configuration" in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)* and the CreditTresor example application.

3.10.2.1 Using the Permissions tab

The **Permissions** tab allows you to restrict access to applications or retention policies.

- Application
- Retention Policy
- User Group

The **Application** category is the default and this view allows you to indicate which groups have access to specific applications.

The **Retention Policy** category allows you to indicate which groups have access to specific retention policies.

The **User Group** category allows you to view which applications a group has access to (if the OTDS SSO profile is enabled, obsolete groups are displayed but cannot be selected).

3.10.2.2 Restricting access to an application, or retention policy

The group permission role allows finer control over what kind of access a user has in applications, and retention policies. With the feature disabled (default for upgrading customers), if a group is mapped to a role, there are two options:

- You put a group permission on the application to prevent access, unless the user is in one the groups.
- The user gets access to the application, retention policy or search with all the role permissions. For example, if the user is included in a role that grants DEVELOPER access, as long as the user is in a group defined on the application, they will have DEVELOPER access.

With the new mode, the Administrator can allow that user access, but reduce their access to a lesser role. For example, a user could be in two groups, one mapped to the Retention Manager group and the other mapped to the End User group. An Administrator can set a group permission on the application so that users in the second group get End User access. If the application only specifies the one group associated to an End User, all access to the application will be restricted to the End User.

Note: Administrators always have the ability to change the group permissions for an application, even if they are not listed in any of the groups specified for the application.

In previous OpenText Information Archive releases, the Developer was able to access the **Permissions** tab and configure group accessibility. This is no longer the case, and the Developer is not able to access the **Permissions** tab.

By default, all users will be able to access an application or retention policy. You might want to restrict access to an application or retention policy to protect sensitive data. For example, you might want to restrict access to the Audit application because audits often provide information about applications that End Users should not be able to access. You can restrict access to an application and all its searches, or you can restrict access to just a particular search. For more information, refer to the Audit application.

Tip: It is recommended that you limit the groups that can access the Audit and Reports sample applications.

If you do not want auditors to run searches in certain applications, restrict the groups for those applications to only specific groups.

To restrict access to an application or retention policy:

1. For the lists in the **Administration** > **Permissions** tab, configure the following:
   - **Category**: Select **Application** or **Retention Policy**.
   - **Application**: Select the application you want to configure.
   - or **Retention Policy**: Select the retention policy yo want to configure.
2. Select the **Grant access to specific groups** check box. The list of groups available is displayed.

   Note: Groups are only shown if they have been mapped to a role.
3. For the groups that require access, select the **Grant Access To** check boxes for those rows.

   Note: You must select at least one group; otherwise, you will not be able to save your changes.
4. Click **Save**.

If you receive an error message that indicates **Version mismatch**, it means another user attempted to update the application's permissions at the same time. Discard your changes and the permissions will refresh to the current configuration.

Tip: By default, all groups are displayed (**Show all groups**). After saving your changes, you may want to change the filter to view fewer groups by using the **Show only groups with access** filter.

To view permissions across multiple applications for a particular group:

1. Select **User Group** from the **Category** list.
2. Choose the user group. A strikethrough through a group name indicates that the group no longer exists, but is still mapped to one or more roles. Only groups that are mapped to OpenText Information Archive roles in the **Administration** > **Groups** tab appear on this list.

   By default, all applications are shown. It is possible to filter to only show applications that have access or only applications that match a pattern using the **Find an Application** field. For example, enter `one` in the **Find an Application** field. Assuming all of the sample applications are installed, only the PhoneCalls and PhoneCallsGranular applications are displayed.

   For each application, two columns are displayed (read only):

   - **Has Access**: Indicates if the group has access to the application.
   - **Group Permission Set**: Indicates if any group permissions were set for the application. If group permissions were not set, the group has access.

   Note: This view only provides information to user access on the application level, restrictions set to on the search set level cannot be seen. For information on how to manage permissions on a specific search set, see Section 6.3.2 "Managing permissions" in *OpenText Information Archive - Configuration Guide (EARCORE-CGD)*.

3.10.2.3 Managing application permissions

The **Permissions** tab is used to specify the groups that are able to access an application.

OpenText Information Archive provides two modes for determining which roles a user has for an application. Independent of the mode, if a user is not in any of the groups, they will not be able to access the application:

- Legacy mode: In this case, in every application, the user has the effective roles for any group they are a member of. Customers upgrading to the latest version of OpenText Information Archive will use this mode.

  For example, consider the scenario in which a user is in two groups: GROUP_END_USER (mapped to the End User role) and GROUP_RETENTION_MANAGER (mapped to Retention Manager role).

However, for application A, only the GROUP_END_USER group is added to the application permissions. In legacy-mode, the user will *still* have the Retention Manager role for application A, meaning they can apply retention and/or holds to records in A.

- Restrict role access based on groups mode: In this case, a user only gets the roles for the application based on the groups specified for the application. In this mode, with the groups and permissions configured as in the above scenario, the user would *not* be able to apply retention and/or holds to application A's records.

  In this mode, certain roles can be completely excluded for selected applications. In the above scenario, with only GROUP_END_USER set on it, there is no user who can apply retention and/or holds to application A's records.

Note that in both modes, if an application specifies *no* group permissions, the user in the scenario above *would* be able to apply retention and or holds to that application's records.

Toggling between which mode can be done either by the IA Shell (see Section 3.9 "Configuring the restrictRolesToApplicationGroups feature" in *OpenText Information Archive - IA Shell Guide (EARCORE-ARE)*) or via the Global Settings tab in the IA Web App.

3.10.2.4 Enabling the group permissions restriction feature

The group permissions restriction feature can be enabled from the **Administration** > **Global Settings** tab:

1. On the **Global Settings** tab, select **permission** from the **Categories** list. For the `permission.restrictRolesToApplicationGroups` parameter, click  and select **Edit**.
2. Set the **Value** field to **True**.
3. Click **Update**.

   Restart the IA Server after enabling the group permissions restriction feature, as it might take time to update the cache if you have more than one IA Server.

3.10.2.5 Enabling server-side logging for a user for the group permissions restriction feature

This procedure can be used to help debug why a particular user can no longer perform actions in applications after the group permissions restriction feature has been enabled.

1. On the **Global Settings** tab, select **permission** from the **Categories** list.
2. Select the **Display all settings** check box.
3. For the `permission.enableLoggingForUser` parameter, click  and select **Edit**.
4. Enter the applicable user name that logging should be enabled for. The value is case-sensitive.
5. Click **Update**.

3.10.3 Things to consider when manually creating data node and database users

If you plan to create users in PostgreSQL manually, and decide to have a separate user for data node and database, that the data node user has to be a member of database user's role. Consider the following potential scenarios:

1. If you are creating only a data node user, the user requires CREATEDB and CREATEROLE privileges. Then the IA Server will create database users, if required, grant role membership, create databases, *etc*.
2. If you are creating a DataNode user and Database user(s), ensure the data node user is a member of database user's role. If that is the case, the IA Server will create databases, *etc*. However, if the membership role/access is not created, IA Server will fail to create databases.
3. If you are creating users and databases (by hand or via some other means), ensure that the database user is the owner of the databases created. If that is the case, IA Server will use pre-created databases to populate them with schema, *etc*.

For the above scenarios, it is possible to only create single user and use it for the dual purpose. The user is essentially a data node and database user. This may not be the preferred method, but it should work. The purpose of dual roles is to ensure that each user has only as much permissions as required and not more.

3.11 Changing passwords and other secrets

These topics explain how to change passwords safely for organizations that have requirements to change passwords regularly as part of their security policies.

The following sensitive information will be discussed:

- Passwords, including those for data nodes and databases
- Secret keys for communication between components
- Information used to secure encryption, including keystore passwords

Before you begin:

1. Make sure that all jobs have finished running, and stop the job schedule for each job. Do not ingest any data.
2. Stop all instances of the IA Server and IA Web App.
3. Make sure you back up any configuration files that you are changing, in case the changes need to be reverted. You should back up the *<IA_ROOT>*/config directory.
4. Consider using a password management tool to store the original passwords, as it not possible to retrieve the original unencrypted passwords.

It is not necessary to do a full backup of PostgreSQL or your unstructured data before changing passwords.

When you are changing passwords in a configuration file, the passwordEncryption.enabled parameter indicates that all passwords in the file are expected to be encrypted. You must take this into account when you change values in IA Server's <IA_ROOT>/config/iaserver/application.yml file. The following topics describe the passwords that are stored in this file, as well as recommendations for changing them.

For more information about encrypting passwords, see Section 7.2 "Encrypting passwords manually" in *OpenText Information Archive - Encryption Guide (EARCORE-AGE)*.

To change RDB data node passwords, use the PG Admin Client, and then update the <IA_ROOT>/config/iaserver/application.yml file. The file must be changed for each instance of IA Server.

- systemData.psql.databaseCluster.superuser.password

To change the system RDB Database passwords, use the PG Admin Client, and then update the <IA_ROOT>/config/iaserver/application.yml file.

- systemData.psql.database.admin.password
- rollForwardData.psql.database.admin.password

- `synchronizationData.psql.database.admin.password`
- `auditData.psql.database.admin.password`

For TLS/SSL settings:

- `systemData.psql.databaseCluster.connectionProperties.ssl.keystorePassword`
- `systemData.psql.databaseCluster.connectionProperties.ssl`

For the IA Web App's `application.yml` file, the first parameter in the following table is the key for communication:

- `infoarchive.gateway.token.secret`: Must match the corresponding parameter in IA Server's `<IA_ROOT>/config/iaserver/application.yml` file.
- `infoarchive.gateway.client.ssl.key-store-password`
- `infoarchive.gateway.client.ssl.trust-store-password`: Only necessary if you are using TLS/SSL with the IA Server.

The `<IA_ROOT>/config/iawebapp/application-CLIENTS.yml` file contains keys for communicating with other clients.

You can change the `clientSecret` for clients that you are using. OpenText recommends that you do not change the `clientSecrets` unless necessary, especially if using OTDS SSO. When using OTDS SSO, the client secrets are provided during setup. You can change the section `clientId: "infoarchive.cli"`. CLI is officially known as IA Shell.

If you change the `clientId` for IA Shell, then you also need to update the `connection.clientSecret` parameter in the *`<IA_ROOT>`*`/config/iashell/application.yml` file.

If an instance of IA Shell is running, you must close it and start a new instance for the configuration changes to take effect.

Passwords for structured data, including data nodes and databases, can only be changed through PG Admin. IA Web App should be used afterwards to update the corresponding passwords in IA to ensure IA continues to be able to connect accordingly, but doing so will not change the actual PG Admin passwords.

Both IA Server and IA Web App must be running. When changing the passwords, you do not need to stop IA Server and IA Web App.

1. Using PG Admin Client, change the passwords.
2. Using IA Web App, go to **Administration**, click the context menu for the data node or database, and then click **Edit**.
3. Update the password, and then click **Save**.

   IA Web App checks that the password that you specified matches the password that PostgreSQL Server is using.

If you are not using OpenText Directory Services (OTDS), then you can skip this section.

OTDS has the following configuration file for IA Web App: *<IA_ROOT>*/config/iawebapp/application-infoarchive.gateway.profile.AUTHENTICATION_OTDS.yml.

- OTDS.password: Must match the value on OTDS.

There are client secrets in the *<IA_ROOT>*/config/iawebapp/application-infoarchive.gateway.profile.OTDS.yml file. However, it is recommended not to change the client secrets.

- OTDS.password: Must match the value in OTDS.
- OTDS.infoarchive.clients.gateway.clientSecret and OTDS.infoarchive.clients.iawa.clientSecret: This is the key that you obtained when you ran the otds-ia-init utility. OTDS does not allow you to change this key after the client is created, at least using the Admin Web Client.
- OTDS.password: Must match the value in OTDS.

You change the following parameter in the <IA_ROOT>/config/iashell/application.yml file.

- connection.clientSecret: This is the key that you obtained when you ran the otds-ia-init utility. OTDS does not allow you to change this key after the client is created, at least using the Admin Web Client.

Note that secret keys might need to be encrypted based on a setting in the configuration file.

If you are not using the Active Directory profile, then you can skip this section.

You change the following parameter in the *<IA_ROOT>*/config/iawebapp/application-infoarchive.gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORY.properties file.

- AUTHENTICATION_ACTIVE_DIRECTORY.managerPassword

This value might need to be encrypted, based on the passwordEncryption.enabled parameter that is defined in the *<IA_ROOT>*/config/iawebapp/application.yml file.

If you want to install more applications using a declarative configuration, then you need to update the *<IA_ROOT>*/congif/iashell/default.properties file to specify the correct data node password.

It is also possible to specify per application which username and password to use for the database in the application's configuration.properties file (for example, *<IA_ROOT>*/examples/applications/Baseball/config/default.properties). The parameters to add are rdbDatabase.username and rdbDatabase.password.

If your example applications are using different data nodes, also specify the `rdbDataNode.username` and `rdbDataNode.password` parameters per application.

3.12 Global Settings

The IA Server supports two different types of configuration settings:

- Settings defined in the IA Server's `application.yml` file, and
- Global settings that can be updated using the **Global Settings** tab in the IA Web App or IA Shell.

Settings managed with the `application.yml` file can differ per IA Server node and require a restart to apply changes. Settings defined here typically either cannot be managed dynamically or need to potentially be different per IA Server node. Global settings, however, are implicitly shared between all IA Server nodes of the same environment and changes are applied dynamically without necessitating a restart.

Note: For performance reasons, global settings are cached in memory, and only refreshed periodically, so changes may take up to a configurable amount of time (5 minutes, by default) before they become visible and active in each IA Server.

Name	The name of the property.
	Click to edit the setting or reset the default value. For more information, see the procedure below.
Description	A brief description of the parameter.
Category	A functional reference to the business logic using the settings.
Type	The type of the parameter's value (boolean, positive integer, or string).
Default Value	Indicates the original, default value of the setting if it has not been updated. If the setting has been updated, this column indicates what the value will be if you reset the value.
Value	The value that is persisted in the database for the setting.

Use the filters to locate a specific parameter:

- Type keywords into the **Find Global Settings** field.
- Use the **Categories** list to filter the list of parameters to one of the following:
  - `audit`
  - `background`
  - `batch`
  - `compliance`
  - `ingestion`
  - `export`

  - `jdbc`
  - `kafka`
  - `permission`
  - `rest`
  - `retention`
  - `search`
  - `cleanup`
- Select the **Display all settings** to view the global setting parameters, including those that currently have no value set.

To update a global setting parameter:

- On the **Global Settings** tab, click  and select one of the following options:
  - **Edit**: Allows you to update the setting. How you update the value depends on the type of parameter:
    - For a positive integer setting, enter or select the desired value of the parameter.
    - For a Boolean setting, select whether to set the parameter to true or false.
    - For a string setting, enter the desired value of the parameter.

      Once the setting has been updated, click **Update** to save the change.
  - **Reset to default value**: Select to reset the setting to its default value.

3.12.1 Enabling Content Aviator

OpenText™ Content Aviator is an intelligent assistant powered by AI that allows users to ask natural language questions about or summarize search results in OpenText Information Archive. Content Aviator is referred to as Aviator for the remainder of the section, and is available only for the Cloud Edition of OpenText Information Archive.

Content Aviator uses Google Vertex as its AI engine. By design, your data is protected and will not be used to train large language models (LLM).

Before users can use this feature, it must be enabled via the **Global Settings** tab. The following properties can be configured:

`aviator.enabled`
: Set to `true` to enable Content Aviator for all applications.

`aviator.result.size.max`
: Configure the maximum number of results allowed to chat with Content Aviator. When a question is asked, Content Aviator performs a “similarity”

check to determine the appropriate results. If the value of this property is set too high, the results may deviate from the user's initial context and, therefore, impact the integrity of the results.

`aviator.timeout`
Configure the timeout in seconds to contact Content Aviator.

`aviator.url`
Set the Content Aviator URL endpoint.

3.12.2 Enabling and disabling the message tray

When messages are generated by the system, they are logged in the message tray for users to review in the IA Web App. For more information, see Section 1.7 "Messages" in *OpenText Information Archive - End User Guide (EARCORE-UGD)*.

The `iawa.messagetray.enabled.default` property in the Global Settings tab allows you to enable and disable the message tray. By default, the message tray is enabled for all users. Edit the property and set the Value to `true` (enabled) or `false` (disabled).

Changes made from the Global Settings tab do not override the options set when users configure their general preferences. For more information, see Section 1.2.2 "Configuring general preferences" in *OpenText Information Archive - End User Guide (EARCORE-UGD)*.

3.12.3 Configuring the See the Audits feature

Use the following settings in the Global Settings tab to configure the See the Audits feature (see Applying actions to a package):

`audit.applicationName`
The application name of the nested search . By default, this property is set to `Audit` and it is not recommended that this value is changed.

`audit.search.searchName`
The search name. By default, this property is set to `Application Audit` and it is not recommended that this value is changed.

`audit.search.searchCompositionName`
The search composition name. By default, this property is set to `Set 1` and it is not recommended that this value is changed.

`audit.search.lastNumberOfDays`
The last search period in days. By default, this value is set to `180` (the number of days, which is equal to the last 6 months).

3.13 Logging

OpenText Information Archive log files will be created in the `logs` directory and rollover into `.gzip` archives in the `archives` subdirectory once a log file reaches 10 megabytes in size. There is also another subdirectory called `context-sifting` that is used by the OpenText Information Archive logging functionality related to job and order item logs:

- Startup logging, as well as most errors, will be printed on console.
- Errors will be logged to the `errors.log` file.
- The `recovery.log` file contains vital information about the disaster recovery process. For more information, see Section 12.5 “Disaster recovery logging” in *OpenText Information Archive - Installation Guide (EARCORE-IGD)*.
- Context (request, job, order item, scheduled task, etc.) information is logged in the `context-mapping.log` file.

  This information includes an ID, type correlation and user, as well as optional context-specific fields (for example, the full request URI for requests or job details for jobs, etc.).

  Only a single message per context will be logged here.

  Each log message in any of the other log files typically includes the context-id, which can be used to lookup the additional context information in this file. The same context-id can be used to correlate multiple log messages when analyzing the logs.
- Various other log files are maintained to distribute logging based on log message origin or context, primarily to ease troubleshooting.

  For IA Server, logging configuration cannot/should not be modified by editing the logback configuration directly. There are a few system properties that can be set, however, to control some of the logging behavior.

  `max.size=10MB`
  Limits the size of individual IA Server log files before they rollover.

  `max.history=15`
  Controls how many days archived IA Server log files are kept.

  `logsdir=logs/iaserver`
  Controls the directory in which IA Server log files are persisted.

! Important

While some jobs, such as the Disposition job, create order items and update the logs while as they progress, not all jobs refresh the logs while running. For both background requests and jobs, it is important to manually refresh the logs. Furthermore, the system’s ability to automatically refresh the log information is suspended if the state of the job/background request's status is Completed, Success, Failed, or Canceled.

Also remember to view the logs and note the timestamp for when a log was last refreshed.

The **Refresh** button will not work if the state of the job/background request's status is Completed or Success.

There are at least three ways these system properties can be set:

1. The startup script can be modified by adding the following line in an appropriate place:
   - For Windows:
     ```
     set IASERVER_OPTS=-Dmax.size=10MB -Dmax.history=15 -Dlogsdir=logs/iaserver
     ```
   - For Linux:
     ```
     set IASERVER_OPTS='-Dmax.size=1MB -Dmax.history=10 -Dlogsdir=logs/iaserver'
     ```
2. The <*IASERVER_OPTS*> environment variable can be set through other means.
3. Create the file `config/iaserver/config-override.properties` and populate it with the before-mentioned properties.

The last option is preferred because this is least error prone and the file can simply be copied over into each new release without having to make any changes to scripts or other files in the distribution. This mechanism of setting system properties, however, only works for IA Server.

3.13.1 Enabling dynamic logging level changes for the IA Server

While the `log.level.threshold` property in the IA Server's `application.yml` file applies the loglevel threshold for the regular logging performed by IA Server, the **Global Settings** tab allows you to dynamically override the following logging-related thresholds:

`loglevel.threshold.background`
The global loglevel threshold for background processing.

`loglevel.threshold.chainOfCustody`
The global loglevel threshold for chain of custody processing.

`loglevel.threshold.rest`
The global loglevel threshold for REST request processing.

`loglevel.threshold.scheduled`
The global loglevel threshold for scheduled task processing.

To override the default setting, enter a new value for any of the above loglevels. If a new value is not entered, the loglevel automatically reverts to the setting in the IA Server's `application.yml` file.

To learn how to use the Global Settings tab, see Global Settings.

3.13.2 Enabling logging for LDAP

For LDAP/AD-based authentication, OpenText Information Archive basically configures the Spring Security LDAP implementation. The actual authentication is implemented by the Spring Security LDAP library. Refer to the following for the details of the configuration parameters:

- https://docs.spring.io/spring-security/site/docs/4.2.3.RELEASE/reference/htmlsingle/#nsa-ldap-authentication-provider-attributes
- https://docs.spring.io/spring-security/site/docs/4.2.3.RELEASE/reference/htmlsingle/#ldap-authentication

You can configure the logging for Spring Security LDAP using the `org.springframeworksecurity.ldap` key in the `<IA_ROOT>/config/iawebapp/application.yml` file.

3.13.3 Enabling logging for embedded Tomcat for IA Web App as a standalone Spring boot application

To enable embedded Tomcat access logs when running in standalone mode, ensure the `<IA_ROOT>/config/iawebapp/application.yml` file includes the following:

```
server:
  # Uncomment the following if you want the standalone. Gateway to listen to a specific
  # network or loopback interface address. The default is to listen on
    all network interface
  # addresses (e.g. 0.0.0.0).
  # address: ${infoarchive.gateway.host}
  port: ${infoarchive.gateway.port}
  servlet:
    contextPath: ${infoarchive.gateway.contextPath}
  tomcat:
    accesslog:
      directory: "c:/tomcat/accesslog
      enabled: true
```

3.13.4 Using a preexisting open source logging solution

In a multi-server deployment, logs need to be stored within a unified location to easily:

- Track and manage logs across several servers; and
- Allow for querying to better understand issues.

When a job is started, the following information may be available in the first log entry:

Field	Always present	Description
Name	Yes	Indicates the name of the job.

Field	Always present	Description
Handler	Yes	Indicates the handler for the job. For jobs that were cloned, this field helps identify what the job does.
Properties	No	Indicates the properties for the job. Job parameters are shown in curly brackets. For example: type `=aip, retentionPolicy=Policy A`.
Scheduled By	No	Indicates the user that ran the job. This field is not shown for scheduled jobs.
Application	No	Indicates the application the job instance was scoped to. If a job definition is scoped for multiple applications, when the job runs, each job instance will be scoped to an application. If the job was scoped to run for two applications, two job instances for each application are created.
Attempt	Yes	Indicates the attempt for running the job. Value is always one unless the job has been configured to auto retry upon failure. Does not increment when doing a manual retry.

The following example illustrates how to integrate a preexisting logging solution, in this case, Graylog2, with a multi-server logging deployment.

1. Create a multi-server OpenText Information Archive deployment. Ensure that each IA Server machine has a different host name.
2. To consume the OpenText Information Archive logs, install and configure `syslog-ng` by running the following command:

   ```
   sudo apt-get install syslog-ng
   ```

   a. Add the log files to the `etc/syslog-ng/syslog-ng.conf` file.
   b. Start `syslog-ng` by executing:

   ```
   syslog-ng
   ```

3. Install Graylog2.

   - Download the latest image from the Graylog website and deploy it within a virtual machine.

4. Configure Graylog to consume IA Server logs via `syslog-ng`:

   a. Add the following lines to the `etc` > `syslog-ng` > `syslog-ng.conf` file:

```
# Define graylog TCP syslog destination.
destination d_graylog {
    syslog("graylog location" port(5140));
};

# Tell syslog-ng to send data from Information Archive sources to the
newly defined syslog destination.
log { source(s_ia_log); destination(d_graylog); };
log { source(s_ia_error); destination(d_graylog); };
```

b. Update the Graylog server:

   i. Log into the Graylog web application.

   ii. Configure input using the process described in the Graylog documentation. Make sure to use the Syslog TCP input and bind the address to 0.0.0.0.

5. Start up IA Server servers. The logs should appear in the configured input after a few minutes.

3.13.5 Job instance and order item logs

Whenever a job or order item is run:

- Activity is logged in the regular log files; and
- Activity is also captured in a separate log file dedicated to that specific job instance or order item.

In the `application.yml` file, the `log.level.threshold` property specifies the global log level threshold. This log level controls the type of messages that are written to the various log files.

The log messages in the dedicated log files, however, are not affected by the global log level threshold, and will instead always be written to the dedicated log files, regardless of log level.

Logs are archived into the RESULT content store defined by the associated application. While order items are generally application-specific, jobs run system-wide and are, therefore, not associated with an application defining the content store to persist such logs.

The OpenText Information Archive System application defines the necessary content store to persist job logs. You install the System application as part of installing the core OpenText Information Archive applications. For more information, see Section 7.9 “Installing the first-time setup applications” in *OpenText Information Archive - Installation Guide (EARCORE-IGD)*.

If the System application is not installed, the dedicated logs will be stored in the `<IA_ROOT>/logs/context-sifting` directory.

3.13.6 Downloading diagnostics logs

The ability to download job instance and order item logs is restricted to users who have permission to download content and are allowed to view the job or order item the logs are associated with.

To download the dedicated log file for a specific job instance:

1. In the **Jobs** tab, click the link in the **Status** column to access the job's log files.
2. Click the **Status** link for an individual order item or job instance.

   Tip: You can determine which job instance's log files you want to access by reviewing the **Scheduled Date** column.

3. Click **Download diagnostics logs**.

3.13.7 Downloading composite logs

The composite logs contain the logging information for order items that have batches and other levels, such as a job that has been run.

When a job runs, it essentially creates three different log levels:

1. Logs for the job instance,
2. Logs for every order item and job instance, and
3. Logs for every batch of each order item.

If you click **Download diagnostics logs**, as described in Downloading diagnostics logs, the downloaded log only covers the logging information for the selected order item or job instance.

To download the log information for the three levels outlined above, click **Download composite logs**. The downloaded `ZIP` file will contain the logging information for every job instance, order item, and every batch of each order item.

Tip: If you tried downloading the composite log, but the download timed–out, click **Prepare composite logs**. Wait a few minutes and click **Download composite logs** again.

3.13.8 Log level configuration

The setting for the `log.level.threshold` can be updated.

```
log.level.threshold: 'WARN'
```

By default, the server is configured to use the WARN level, as that limits the amount of information logged in the system (saves disk space, helps performance, *etc*.). If there are issues in the system, the logging level should be raised (for example, to DEBUG level). IA Server needs to be restarted for this change to take effect.

The following log levels are supported:

- ERROR
- WARN
- INFO
- DEBUG

3.13.9 Enabling REST logging

OpenText Information Archive allows for granular configuration for REST logging of the REST API resources that support being logged. The `request` and `response` bodies can be configured to be logged via the Global Settings tab's `rest.logging` parameter. By default, a number of commonly used and useful headers are logged, while others are suppressed.

The fact that the REST logging can be configured allows more control over this type of logging. You can:

- Customize for which headers the value will be logged.
- Control whether to enable request and/or response logging at all.
- Control whether to enable the logging of headers for requests and/or responses.
- Control whether to enable the logging of the body for requests and/or responses.
- Control whether to enable logging of the request query string. as it could potentially contain sensitive information.

Even if these parameters are enabled, only requests and responses that are cached are logged.

Regarding headers, only a specific subset of headers will have their values logged. Others will be suppressed, and only the fact that they contained a value is logged.

3.13.10 Authorization log

The Authorization log file for IA Server helps to debug issues when users are unable to perform actions in an application. The Authorization log becomes especially useful if errors are occurring after application permissions have been changed, as the log indicates which specific user is unable to perform which actions in a specific application.

The following is an example of an Authorization log file:

```
2023-08-17 06:10:58.778 [25] [bdc58345-73c7-4c71-af98-a12ab31e14e4] INFO
c.e.i.a.core.AuthorizationImpl
- User sue@iacustomer.com has these roles: [IT_OWNER, DEVELOPER, RETENTION_MANAGER] but
does not have these roles for application CreditTresor
2023-08-17 06:10:58.779 [25] [bdc58345-73c7-4c71-af98-a12ab31e14e4] DEBUG
c.e.i.a.core.AuthorizationImpl
- User sue@iacustomer.com cannot do action: VIEW_HOLD_SET for application CreditTresor
due to restricted group permissions
```

If the restrict role access based on groups mode feature is enabled, users may no longer have the access they were accustomed to. OpenText Information Archive provides a `authorization.log` file for the IA Server that can help determine if a user was expected access but was denied. The IA Server logs at INFO or DEBUG level, with the DEBUG level providing more low-level information. Logging per user can be enabled via **Global Settings**.

3.14 Using the Audit application to monitor activity

The Audit application is one of the applications provided by OpenText Information Archive that must be installed when setting up OpenText Information Archive for the first time. The Audit application is a SIP application to archive the audit records of OpenText Information Archive as packages, making them searchable and managed in the sense that they can be retained and disposed.

You can archive audits to the Audits application through the Archive Audits job. Until you run this job, and between runs of this job, any audits will be temporarily stored in the audit database (most likely in the data node where the system data lives).

> **Important**
> You must run the Archive Audits application regularly to free up space in the temporary location.

When using a search in the Audit application to find a specific application, application names are case-sensitive. You can search for multiple applications by using a comma as a separator: `Baseball,PhoneCalls`.

3.14.1 Searches in the Audit sample application

If the first-time setup applications have been installed, the Audit application contains six searches.

- **Activity Reports** provide access related to ingestion, searches and downloads. You can limit the search to a particular event's date and by a specific application.
- **Application Audit** provides access to only the application audits. This includes events such as ingestion and applying retention or holds to items in applications. This search has been improved to automatically determine which applications are installed so there is no need to modify the search. This search will no longer return the system or tenant audits.
- **Aviator Activity Report** provides information about Content Aviator usage per day. Limit the search to a particular date or by a specific application.
- **Storage Reports** provide a footprint for system, audit, retention, content stores and online data. The Storage Reports will not return information unless the Refresh Metrics job is run followed by the Archive Audits job. Once these jobs run, there will be entries in the report based on the time when the Refresh Metrics job was run. The following information is presented:
  - **System Repositories**: Total size of the segment data files of the databases:main, synchronization and rollforward.
  - **Audit Repository**: Total size of the segment data files of the unarchived audits (auditDatabase).
  - **Content Stores**: Total size of the content saved in the stores (file system, ECS, S3, etc.).
  - **Online Data**: Total size of the segment data files of structured data online (cache-in).

    Note: Segment data files will not, typically, decrease in size. Segment data files are, essentially, allocated space and may not be completely utilized.

    A nested search brings details by application (system repositories, content stores, online data, and licensed volume), from the **Details** button in the Storage Reports result page.

  To ensure the Storage Report contains the latest information, complete the following procedure:

  To run the Storage Report:

  1. On the **Audit** page, ensure that the **Create** audit is enabled for the **Application > System > Storage Metric** type.
  2. On the **Jobs** page, run the **Refresh Metrics** job.
  3. Once the Refresh Metrics job has completed its run, run the **Archive Audits** job.

The up-to-date data is now available.

- **System Audit** provides access to only the system audits. This includes events such as login, jobs, and modifying system-wide configuration objects, such as endpoints for Dell EMC Elastic Cloud Storage or OpenText Archive Center. This search will no longer return the tenant or application audits.
- **Tenant Audit** provides access to only the tenant audits. This includes events such as provision events on retention policies and holds (configuration objects shared among applications). This search will no longer return the system or application audits.

3.14.2 Reviewing the audit history of an AIP, library, or table

OpenText Information Archive allows the Developer and Business Owner to use the IA Web App to conveniently search the audit history of a specific AIP or table.

To have access to this functionality, the first-time applications must be installed and ensure that the user is in the appropriate group (if group permissions are set on the Audit application).

1. Run the Archive Audits job prior to using this functionality.
2. Once the job has completed, the Developer and Business Owner can access an application and navigate to the **Packages**, **Libraries**, or **Tables** page, depending on the application selected. Click  and select **See the Audits** for a specific package, library, or table.

   If the See the Audits option is not available:

   - Ensure that the configured search is not in draft mode.
   - You meet any group permission restrictions on the Audit application. For more information, see Configuring the See the Audits feature.

This automatically runs the Application Audit search for the selected package, library, or table. Instead of displaying the results in the Audit application, however, the results are displayed in the Packages, Libraries, or Tables tab for the currently selected application.

Click **Prepare Export** to initiate a background request to review the exported audit information in greater detail.

3.14.3 Audit troubleshooting

I have turned off an audit for an event type at the application level but the audit is still being generated.

Check to see if the audit is enabled at the tenant level. If it is, disable it at the tenant level and enable the audit for applications that still want that audit event type to be generated. The setting to disable an audit for the application is ignored if set at the tenant level.

When I search the audits, I do not get any results.

Run the Archive Audits job periodically so that the audits are archived.

An audit for the rollback event for the AIP will be created and in the supplemental data, the `stateCode` and `phaseCode` can be used to determine that the invalidation was rolled back (both values would be set to `COM`).

3.15 Health monitoring

For customers who operate OpenText Information Archive in a lights out model, which allows the Administrator to monitor servers by remote, OpenText Information Archive allows you to conduct periodic health checks to ensure that the PostgreSQL, OpenText Information Archive and web application servers are running. The health check allows you to check a few vitals and parameters.

The following servers allow you to conduct these types of checks, each with a default `/health` endpoint:

If the server is the IA Server

The health endpoint is `http(s)://<IAS_HOST_NAME>:8765/health`.

If the server is the IA Web App

The health endpoint is `http(s)://<IAWA_HOST_NAME>:8080/health`.

These endpoints can be used to ensure that the servers are up and running. In this initial implementation, HTTP 200 is returned if the server is, in fact, up. This functionality can be used with automated tools to check the status and availability of the related services.

3.16 Changing the default ports for OpenText Information Archive components

This section illustrates where to update the default ports for the PostgreSQL server, IA Server and the IA Web App:

- For the PostgreSQL server, update the port property in `<IA-ROOT>/psql/conf/postgresql.conf`
- For IA Server, update the `server.port` property in the `<IA_ROOT>/config/iaserver/application.yml` file.
- For IA Shell, update the `rdbDataNode.bootstrap` property in the `<IA_ROOT>/config/iashell/default.properties` file.
- For the applications, update the `connection.gatewayUrl` and `connection.restApiUrl` properties in the `<IA_ROOT>/config/iashell/application.yml` file.
- The IA Web App can be deployed using two different methods:
  - It can run as a standalone when the Spring Boot application is run as a process or as a service.
  - It can also be deployed to Tomcat.

Update the `infoarchive.gateway.host` and `infoarchive.gateway.port` properties in the `<IA_ROOT>/config/iawebapp/application.yml` file.

This is later used as:

```
server:
  host: ${infoarchive.gateway.host}
  port: ${infoarchive.gateway.port}
```

When deployed as standalone the `server:*` properties are used by the embedded Tomcat container. When the web application is deployed to Tomcat, however, the host and port are really controlled by the Tomcat configuration. The web application implementation still needs to know about these values and that is why it is better to use `infoarchive:*` properties. Also, if there is a load balancer in front of the web application, these `infoarchive:*` properties need to point to it.

3.17 Working with gateway and IA Web App configuration files

A version of the configuration files is bundled in the `infoarchive-webapp.(jar|war)` file. The files are also extracted into the `config/iawebapp` folder. They are picked up because of the `-Dspring.config.location=file:config/iawebapp/` property and sent to the IA Web App start-up script via one of the following:

```
<IA_ROOT>/bin/ iawebapp.bat (windows).
```

```
<IA_ROOT>/bin/ iawebapp (linux).
```

The `<IA_ROOT>/config/iawebapp/application.yml` file is the main configuration file of IA Web App. The configuration in this file further impact and use other configuration files.

Profiles activate and deactivate specific functionality of the IA Web App. The following are authentication related profiles:

- `infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY`: This is out-of-box active profile and should come later than the other profiles below.
- `infoarchive.gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORY:` Use this profile to activate connection with Active Directory.
- `infoarchive.gateway.profile.AUTHENTICATION_EXTERNAL_LDAP`: Use this profile to activate connection with external LDAP.
- `application-infoarchive.gateway.profile.AUTHENTICATION_OTDS`: Use this profile to activate simple integration with OTDS.
- `application-infoarchive.gateway.profile.OTDS`: Use this profile to activate SSO integration with OTDS.

The HTTPS (SSL) Related profile activates the `HTTPS(SSL)` mode of Gateway/web application. To activate this profile, add it to the `spring:application:profiles:include` property using a list of profile names on each line prefixed with a – (dash). Once this profile is activated, the clients must access the Gateway/web application using the `https ://... url` (note the 's' in https).

The `<IA_ROOT>/config/iawebapp/application-infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY.properties` file configures the out-of-box in-memory users. This file is active out-of-box for easy setup, demo-ability and bootstrapping of the customer installation. This file is picked up when the `spring:application:profiles:include` property contains the profile name `infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY`:

```
spring:
  application:
    ...
    profiles:
     include:
       - infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY
```

The following illustrates the out-of-box content of the file:

```
AUTHENTICATION_IN_MEMORY.group.groups[0]=GROUP_ADMINISTRATOR
AUTHENTICATION_IN_MEMORY.group.groups[1]=GROUP_BUSINESS_OWNER
AUTHENTICATION_IN_MEMORY.group.groups[2]=GROUP_DEVELOPER
AUTHENTICATION_IN_MEMORY.group.groups[3]=GROUP_END_USER
AUTHENTICATION_IN_MEMORY.group.groups[4]=GROUP_IT_OWNER
AUTHENTICATION_IN_MEMORY.group.groups[5]=GROUP_RETENTION_MANAGER
AUTHENTICATION_IN_MEMORY.group.groups[6]=GROUP_AUDITOR

# Example of how to add additional roles if needed. This can be done in external
application.properties file.
# AUTHENTICATION_IN_MEMORY.group.additionalGroups[0]=GROUP_COMPLIANCE_DEPARTMENT

AUTHENTICATION_IN_MEMORY.user.users[0]=adam@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[0]}
AUTHENTICATION_IN_MEMORY.user.users[1]=bob@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[1]}
AUTHENTICATION_IN_MEMORY.user.users[2]=connie@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[2]}
AUTHENTICATION_IN_MEMORY.user.users[3]=emma@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[3]}
AUTHENTICATION_IN_MEMORY.user.users[4]=imran@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[4]}
AUTHENTICATION_IN_MEMORY.user.users[5]=rita@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[5]}
AUTHENTICATION_IN_MEMORY.user.users[6]=sue@iacustomer.com,password,GROUP_ADMINISTRATOR|
GROUP_BUSINESS_OWNER|GROUP_DEVELOPER|GROUP_END_USER|GROUP_IT_OWNER|
GROUP_RETENTION_MANAGER
AUTHENTICATION_IN_MEMORY.user.users[7]=audrey@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[6]}

# Example of how to add additional users if needed. This can be done in external
application.properties file.
# AUTHENTICATION_IN_MEMORY.user.additionalUsers[0]=dave@iacustomer.com,password,$
{AUTHENTICATION_IN_MEMORY.group.groups[2]}
```

The `<IA_ROOT>/config/iawebapp/application-infoarchive.gateway.profile.AUTHENTICATION_EXTERNAL_LDAP.properties` file configures the external LDAP systems such as Apache DS. This file is picked up when the `spring:application:profiles:include` Spring property contains the profile name `infoarchive.gateway.profile.AUTHENTICATION_EXTERNAL_LDAP`:

```
spring:
  application:
    ...
    profiles:
      include:
      - infoarchive.gateway.profile.AUTHENTICATION_EXTERNAL_LDAP
```

The following illustrates the out-of-box content of the file:

Property	Description
`# External LDAP`	This is just an example for connecting to Apache DS service running on local host. You will have to change the settings for your LDAP.
# The following two are default Administrator and password for Apache `DS#AUTHENTICATION_EXTERNAL_LDAP.managerDn=uid=admin,ou=system`	Supports non-personal account NPA.

Property	Description
`#AUTHENTICATION_EXTERNAL_LDAP.managerPassword=secret`	Supports non-personal account NPA.
`AUTHENTICATION_EXTERNAL_LDAP.userDnPatterns=uid=\{0\},ou\=people`	None
`AUTHENTICATION_EXTERNAL_LDAP.userSearchFilter=uid=\{0\}`	None
`AUTHENTICATION_EXTERNAL_LDAP.userSearchBase=ou\=people`	None
`AUTHENTICATION_EXTERNAL_LDAP.groupSearchFilter=(member\=\{0\})`	None
`AUTHENTICATION_EXTERNAL_LDAP.groupSearchBase=ou\=groups`	None
`AUTHENTICATION_EXTERNAL_LDAP.url=ldap://localhost:10389/dc\=infoarchive,dc\=emc,dc\=com`	URL + context of local Apache DS LDAP service.
`AUTHENTICATION_EXTERNAL_LDAP.url=ldaps://localhost:10636/dc\=infoarchive,dc\=emc,dc\=com`	Now the secure ldaps:// protocol is supported. For that you have to: • Generate or obtain a certificate for LDAP. Populate it in the Keystore and configure the Keystore for LDAP server. This may have been for the LDAP server. Import the certificate for LDAP server into the Gateway+IAWA truststore.
# Optionally specify `objectClass` for groups - defaults to `groupOfNames` `# AUTHENTICATION_EXTERNAL_LDAP.groupObjectClass=groupOfNames`	Optionally specify `objectClass` for groups - defaults to `groupOfNames` Sometimes the name of `objectClass` is different than the default
# This will be banded with (`objectClass=”groupOfNames” &(...)`) `# AUTHENTICATION_EXTERNAL_LDAP.groupFilter=`	This will be banded with (`objectClass=”groupOfNames” &(...)`) For example, (`cn=*INFOARCHIVE*`) will only match groups that have `INFOARCHIVE` in the name.

This profile can be used in conjunction with other profiles. For example:

```
spring:
  application:
    ...
    profiles:
      include:
      - infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY,
      - infoarchive.gateway.profile.AUTHENTICATION_EXTERNAL_LDAP
```

The <IA_ROOT>/config/iawebapp/application-infoarchive.gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORY.properties file configures the connection to an external Active Directory server, such as the Microsoft Active Directory Server. This file is picked up when the spring:application:profiles:active Spring property contains the profile name infoarchive.gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORY:

```
spring:
  application:
    ...
    profiles:
      include:
      - infoarchive.gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORY
```

The following illustrates the out-of-box content of the file:

Property	Description
# External ACTIVE DIRECTORY	This is just an example for connecting to an Active Directory server. You will have to change the settings for your LDAP.
AUTHENTICATION_ACTIVE_DIRECTORY.managerDn=cn\=Administrator,cn\=users,dc\=iigads,dc\=com	None
AUTHENTICATION_ACTIVE_DIRECTORY.managerPassword=Password@123	The Password@123 is the just an example. Change it to suit your AD access. Needs to be changed to encrypted form if the secret and password encryption is turned on in application.yml file above. For more information, see Section 7 “Encrypting component passwords and secret tokens in configuration files” in *OpenText Information Archive - Encryption Guide (EARCORE-AGE).*
AUTHENTICATION_ACTIVE_DIRECTORY.userDnPatterns=cn=\{0\},ou\=Users,ou\=infoarchive,dc\=iigads,dc\=com	None
AUTHENTICATION_ACTIVE_DIRECTORY.userSearchFilter=sAMAccountName\=\{0\}	None
AUTHENTICATION_ACTIVE_DIRECTORY.userSearchBase=ou\=Users,ou\=infoarchive,dc\=iigads,dc\=com	None
AUTHENTICATION_ACTIVE_DIRECTORY.groupSearchFilter=(member\=\{0\})	None
AUTHENTICATION_ACTIVE_DIRECTORY.groupSearchBase=ou\=Groups,ou\=infoarchive,dc\=iigads,dc\=com	None
AUTHENTICATION_ACTIVE_DIRECTORY.url=ldap://10.31.70.140:389/	URL Location of AD server.

Property	Description
# Optionally specify objectClass for groups - defaults to group # AUTHENTICATION_ACTIVE_DIRECTORY.groupObjectClass=group	Optionally specify objectClass for groups - defaults to group Sometimes the name of objectClass is different than the default
# Optional filter for groups. This will be banded with (objectClass="group" &(...)) # AUTHENTICATION_ACTIVE_DIRECTORY.groupFilter=	Optional filter for groups. This will be banded with (objectClass="group" &(...)) For example, (cn=*INFOARCHIVE*) will only match groups that have INFOARCHIVE in the name.

This profile can be used in conjunction with other profiles. For example:

```
spring:
  application:
    ...
    profiles:
      include:
      - infoarchive.gateway.profile.AUTHENTICATION_IN_MEMORY
      -infoarchive. gateway.profile.AUTHENTICATION_ACTIVE_DIRECTORYgateway.
       profile.AUTHENTICATION_ACTIVE_DIRECTORY
```

OpenText Information Archive uses OAuth2 for authentication. The <IA_ROOT>/config-default/iawa/application-CLIENTS.yml file configures all the out-of-box Oauth2 clients of OpenText Information Archive. Customers may add additional clients to integrate applications with OpenText Information Archive (for example, via REST APIs).

The following illustrates the out-of-box content of the file:

`clientId`
ID of client. OpenText Information Archive supports the following values:

- infoarchive.gateway
- infoarchive.iawa
- infoarchive.jdbc
- infoarchive.cli

`authorizedGrantTypes`
Specifies which operations are authorized. Do not change the values for each client.

`authorities`
The value should be role.Administrator for the infoarchive.gateway. Otherwise, the value must be ROLE_TRUSTED_CLIENT.

`scopes`
Do not change these values because they are specific for each client.

`clientSecret`
Only set for the infoarchive.jdbc and the infoarchive.cli client.

Needs to be changed to encrypted form if the secret and password encryption is turned on. For more information, see Section 7.2 “Encrypting passwords manually” in *OpenText Information Archive - Encryption Guide (EARCORE-AGE)*.

`accessTokenValiditySeconds`
Configures how long the access token is valid for. It is recommended that you do not change these values.

`refreshTokenValiditySeconds`
Configure how long the refresh token is valid for. It is recommended that you do not change these values.

3.17.1 Increasing the timeout of the IA Web App

To configure how long it takes for the IA Web App to timeout in seconds, update the `sessionTimeout` parameter in the `<IA_ROOT>/config/iawebapp/application.yml` file. As seen in the following example, the default value is 1,730 seconds:

```
infoarchive:
  gateway:
    host: localhost
    port: 8080
    contextPath: /
    token:
      secret: #SECRET_gateway_token_secret_DEFINITION
    client:
      ssl:
        trust-store:
        trust-store-password:
        trust-store-type:
        # Set following entries to enforce two-way TLS authentication
        key-store:
        key-store-password:
        key-store-type:
  webapp:
    enableDateAndNumberLocalization: true
    sessionTimeout: 1730
    customization:
      location: "file:config/iawebapp/customization/"
```

Important

The `refreshTokenValiditySeconds` parameter of the IA Web App Oauth2 client configuration in the `<IA_ROOT>/config/iawebapp/application-CLIENTS.yml` file must at least exceed the `sessionTimeout` by one minute.

3.18 Enabling full-text search in unstructured content

OpenText Information Archive supports the extraction, storage, indexing, and querying of text from unstructured content, both for SIP and table applications. Although further configuration work must be done, the feature is enabled (set to `true`), by default. This enables you to opt in through the `ingestion.ci.text.enabled` property in the **Global Settings** tab. For more information, see "Global Settings" on page 170. Ensure that the feature is enabled before ingesting the data, since the extraction of text is an ingestion phase.

3.19 Miscellaneous configuration via the IA Server's application.yml file

The `config/iaserver/application.yml` file is generated when the install program is run for a new installation or an upgrade. The supported values are defined in the `config-templates/application.yml` file, which also includes comments. In general, the rules for modifying YAML file must be followed (the section can only be specified once).

This section explains the properties that can be overwritten in the IA Server's `<IA_ROOT>/config/iaserver/application.yml` file.

Caution

If you are changing a setting, and you have multiple IA Servers, depending on the change, you may want to change to all of the IA Servers.

3.19.1 infoarchive section

The data for the `infoarchive` section can be updated.

The value of `infoarchive.gateway.token.secret` should be exactly the same as it is set in the IA Web App `application.yml` file. Both the IA Server and the IA Web App client need to have this value synchronized to the same value to ensure that JWT tokens can be decrypted correctly.

The collections section controls the `defaultPage`. When displaying collections through pagination, you typically start with first page, which is called the `defaultPage`. Collections are zero-indexed, meaning that the first page is at index '0'. If, for some reason, you want to always start at the second page, set the `defaultPage` to '1' and so on. For most environments, however. this value should be set to '0'.

Collections are paginated. The value for `defaultPageSize` controls the size of the page. The following example indicates pages of 10 items (each). You can adjust this value, as needed.

Note: Modifying the `defaultPageSize` value impacts the real estate of the pages in the IA Web App. This should be properly tuned to ensure all clients can handle a given page size.

The `search.defaultTimeOutMs` indicates the amount of time a synchronous search is allowed to run before the server will interrupt it and return an error. The value is in milliseconds (MS). In the following example, the value of `8000` indicates a default time out of eight seconds. If the search does not complete, it will be interrupted in eight seconds and the server will return an error. To handle this, the user should try re-running the search in asynchronous mode as a background search.

3.19.1.1 Crypto keystore section

The IA Server's `application.yml` file includes attributes related to the key store under the `crypto.keyStore.control` JSON property:

```
# Crypto related settings
crypto:
  # KeyStore configuration
  keyStore:
    keyStoreType: jceks
    keyStorePass:
```

There was one (control) that already had cache-related attributes, but related to caching of the key store. This set of attributes has been enhanced with the following attributes, specific to the cache of secret keys:

`keyStore` section

Attribute	Type	Description
`keyStoreType`	String	This value is only required if the `control.type` is set to `DATABASE`. For more information, see Section 7.2.3 “Using the password encryption utility to manually encrypt passwords and tokens” in *OpenText Information Archive - Encryption Guide (EARCORE-AGE)*.
`keyStorePass`	String	This is the keystore password. This value is only required if the `control.type` is set to `DATABASE`.

`control` section

```
control:
     # false in case of new deployments, otherwise should always be set to true
     mustExist: false
     # whether to cache the keystore itself
     cacheKeyStore: true
     # allowed values are DATABASE, KEY_MEDIATOR
     type: DATABASE
   # global key cache settings
```

Attribute	Type	Description
`mustExist`	Boolean	This must be set to true if OpenText Information Archive has ever been started to avoid new keys from being generated.
`cacheKeyStore`	Boolean	Indicates whether to cache the keystore or not.
`type`	String	Must be set to `DATABASE` or `KEY_MEDIATOR`. If set to Key Mediator, the `mediator` section must be completed.

`cache` section

```
cache:
      enabled: true
      capacity: 100
      refreshPeriod: 3600
      type: LFU
      # unit is in seconds
      keyExpiry: 3600
```

Attribute	Type	Description
`enabled`	Boolean	Controls whether cross-transactional keys cache is enabled. When disabled, which is by default, the system does not cache any keys between transactions, and the current transactional key cache continues to be used. When this property is set to true, cross-transactional key cache is enabled, and the system does not use transactional key cache for keys during searches.
`capacity`	Integer	When cross-transactional keys cache is enabled, this attribute specifies the maximum capacity of keys in the cache. For example, setting it to 10 allows up to 10 secret keys to be cached. The default capacity is internally set to 20. Update the value of this attribute to overwrite the internal default value.

Attribute	Type	Description
`type`	LRU or LFU	This attribute specifies what cache type is used. The values are either: • LRU (Least Recently Used): The proprietary cache algorithm that always arranges the keys in the cache by modelling how the keys are used. New keys are added to the top of the list. Each time a key is used from the cache, it is also placed at the top of the list. This way, the keys that are used the least are pushed to the end and, when capacity is overflowing, the last key from the queue will be purged from the cache. Also, whenever the cache is refreshed, any expired keys will be removed from the cache. • LFU (Least Frequently Used): A Caffeine-based algorithm similar to LRU, but not as predictable. The LFU algorithm keeps track every time a key is used and purges the least-used keys. LFU is the default algorithm.
`refreshPeriod`	Integer	When enabled and the cache type is LRU, this controls the frequency (in seconds) the system attempts to purge the cache-off of expired keys. In case of Caffeine-based algorithms, the system allows Caffeine to control a refresh internally. The default value is 3600 seconds.
`keyExpiry`	Integer	The number of seconds after which a given secret key in the cache expires. The key expires after its last usage. This value is per key. Each time the key is used, expiry is extended by another full duration. The default value is 3600 seconds.

`mediator` section

Refer to Key Mediator documentation to learn how to gather the required information.

The following is the section related to Key Mediator:

```
mediator:
      host:
```

```
      port: 8443
      ssl:
      # only used for initialization of first time for Key Mediator
      initialMekAlias:
      tenantId:
      # valid values are OTDS or API_KEY
      authenticationStrategy: API_KEY
      apiKey:
      otds:
        host:
        port: 8443
        ssl:
        # this is configured by Key Mediator admin tool
        clientId:
        # this is registered with OTDS for getting credentials to call the Key Mediator
web service
        clientSecret:
```

Attribute	Type	Description
`host`	String	The name of the host for the Key Mediator service.
`port`	Integer	The port number for the Key Mediator service.
`ssl`	Boolean	It is recommended to set this value to true.
`initialMekAlias`	String	This is the MEK defined in Key Mediator, which is to be used for encrypting and decrypting. This value is only read once when either initializing a new system or migrating from the database strategy.
`tenantId`	String	Used for debugging REST calls made to Key Mediator.
`authenticationStrategy`	String	Must be set to either `OTDS` or `API_KEY`. If set to OTDS, the `otds` section must be completed.
`apiKey`	String	Provided by Key Mediator services for authentication using the `API_KEY` strategy.

`otds` section

```
otds:
        host:
        port: 8443
        ssl:
        # this is configured by Key Mediator admin tool
        clientId:
        # this is registered with OTDS for getting credentials to call the Key Mediator
web service
        clientSecret:
```

Attribute	Type	Description
host	String	The name of the host for the OTDS service.
port	Integer	The port number for OTDS.
ssl	Boolean	It is recommended to set this value to true.
clientId	String	This is the client ID defined in OTDS for Key Mediator, which is to be used for authentication.
clientSecret	String	This is the client secret defined in OTDS for Key Mediator, which is to be used for authentication.

3.19.2 Updating the working directory

An IA Server requires a working directory for the storage of temporary files (for example, during the processing of ingested SIPs).

It could be necessary to update corresponding entry in the `application.yml` file if, for example, you are ingesting a lot of data so, consequently, the working directory requires more space.

Define the properties of the working directory in the `localStorage` entry of the `application.yml` file, as follows:

1. Before updating the working directory, stop the IA Server for each configuration being updated.
2. Access the `<IA_ROOT>/config-templates/iaserver/application.yml` file.

   The following information is displayed:

```
# Locations on the IA Server's local file system
localStorage:
  # Folder for temporary files during reception/ingestion etc.
  tempFolder: data/temp
```

3. Make the necessary changes for the desired settings. Save your changes.
4. Restart the IA Server for each configuration that was updated.

Note: Should the IA Server's Java Virtual Machine crash during the execution of a heavy space-consuming process (for example, ingestion of a large AIP or the run of the Close job), the temporary working directory of the process is not deleted and could result in the loss of a lot of space in the working directory.

Administrators should check those folders and clean them accordingly before restarting a crashed IA Server.

3.19.3 System and audit database

The system database is where all system objects live (tenants, applications, spaces, searches, holdings, tables, etc.).

Audit data is where audit records are stored until they are archived.

The system database and audit data are both configured in the .yml file.

The following is from the `application.yml` file for a system database:

```
systemData:
  enableEntityDebug: false
  enableQueryDebug: false
  objectCacheSize: 1000
  psql:
    database:
      admin:
        password:
        username:
      connectionProperties:
        sslcert:
        sslkey:
        sslpassword:
      name: system
    databaseCluster:
      connectionProperties:
        ssl: false
        sslcert:
        sslkey:
        sslmode:
        sslpassword:
        sslrootcert:
      superuser:
        password:
        username:
      url: jdbc:postgresql://<hostname>:<port>
    enableTransactionPooling: false
    pool:
      initialSize: 4
      maxActive: 100
      maxIdle: 10
  reindex: true
  userName: system
```

The following section is from the IA Server's `application.yml` that is specific for the storing the unarchived audits in PostgreSQL:

```
auditData:
  # number of audits to process in a transaction
  cleanupChunkSize: 1000

  # Maximum number of objects held in cache
  objectCacheSize: 1000
  psql:
    # Set this to true when using pgBouncer in transaction pooling mode
    enableTransactionPooling: false
    databaseCluster:
      url:
    database:
      name: audit
      connectionProperties:
        sslcert:
        sslkey:
        sslpassword:
      admin:
```

```
        username:
        password:
```

You can leave the following empty to inherit the values from the `systemData` database:

- `psql.database.admin.password`
- `psql.database.admin.username`
- `psql.databaseCluster.superuser.password`
- `psql.databaseCluster.superuser.username`
- `psql.databaseCluster.url`

The `connectionProperties` for the database and `databaseCluster` can also be set for a database, if necessary.

The database sections are:

- `auditData`
- `rollForwardData`
- `synchronizationData`
- `systemData`

3.19.4 Configuring the number of items listed in the search results

The `defaultPageSize` value in the `<IA_ROOT>/config/iaserver/application.yml` file indicates the number of result items that appear in the **Record Search** tab. The default value is 10.

In the following example, the user is changing the default value to 5.

1. Before updating the `defaultPageSize`, stop the IA Server for each configuration being updated.
2. Copy the following section of the `<IA_ROOT>/config/iaserver/application.yml` file.

```
infoarchive:
  gateway:
    token:
      secret: XXXXXXXXXX
```

3. Paste the information into the following section of the `<IA_ROOT>/config/iaserver/application.yml` file. The modified file should appear like the following:

```
infoarchive:
  gateway:
    token:
      secret: XXXXXXXXXX
```

```
  rest:
    collections:
      defaultPage: 0
      defaultPageSize: 10
      defaultMaxPageSize: 2000
```

4. Change the `defaultPageSize` value to the desired value (in the following example, the desired number of result items is 5):

```
infoarchive:
  gateway:
    token:
      secret: XXXXXXXXXX

  rest:
    collections:
      defaultPage: 0
      defaultPageSize: 5
      defaultMaxPageSize: 2000
```

5. Save the change.
6. Restart the IA Server for configuration changes to take affect.

Tip: If you have multiple IA Servers, ensure that they all use the same values to avoid confusion.

3.19.5 Configuring the time limit for a background search

If a search exceeds the time limit specified in the **search.defaultSearchTimeOut** parameter listed in the **Global Settings** tab, it automatically becomes a background search. The default time limit set is set to 8,000 milliseconds (8 seconds). You are, however, able to change the time limit.

3.19.6 Limiting the size of files transferred through REST

The `<IA_ROOT>/config/iaserver/application.yml` file allows you to limit the size of files that are transferred through REST during reception.

The following is the section in the `<IA_ROOT>/config/iaserver/application.yml` that can be updated:

```
spring:
  # settings for Spring multipart requests
  servlet:
    multipart:
      enabled: true
      maxFileSize: 2048MB
      maxRequestSize: 2048MB
      fileSizeThreshold: 0
```

The above values must match the equivalent values in the `<IA_ROOT>/config/iawebapp/application.yml` file.

The following explains the Spring Boot properties:

`multipart.maxFileSize`
: Specifies the maximum size permitted for uploaded files.

`multipart.maxRequestSize`
Specifies the maximum size allowed for multipart/form-data requests.

`multipart.enabled`
Enables the support of multipart uploads.

Specify the values using long values or using more readable variants that accept KB, MB, or GB suffixes.

Note: The `multipart.maxFileSize` and `multipart.maxRequestSize` default values can be increased. Prior to doing so, however, consider the architectural impact of the increase, particularly in terms of networking, monitoring, partitioning and performance. For example, are the packages pure XML or do they contain files that will increase the package sizes in relation to the number of records/AIUs?

Chapter 4

Appendix

4.1 Supplemental data

Audits contain supplemental data that includes additional information. More specific information may have been generated, depending on the audit event or type. The supplemental data can be found for the following event categories:

- Background events
- Compliance events
- Content events
- Content provisioning events
- Ingestion events
- Package events
- Provisioning events
- Search events
- Other

While the following subsections do not list all the supplemental data, they represent some scenarios you are likely to encounter.

4.1.1 Provisioning events

The following supplemental information can be provided. For create audit events, depending on the type, the supplemental data may contain information about the initial values.

updatedFields
Which values were updated when the job was edited. For certain types like retention policies, holds, and holdings, the supplemental data also includes the new values for each field that changed.

4.1.2 Compliance events

The following supplemental information is provided by an audit after a retention policy is applied to a package:

baseDate
Indicates whether the base date for aging is set from the package or another date was specified that differs from the retention policy.

The system allows you to define a base date as a possible supplemental data field. The base date relates to which date should be used to start aging for retention policies. This field may be included in the following audit events:

- Apply > Retention Policy
- AIP > Apply Retention
- Table > Apply Retention

description
The description of the retained set.

retainedSet
The name of the retained set.

agingStrategy
The aging strategy of the retention policy applied to the retained set.

objectProtected
The name of the object included in the retained set.

retentionPolicy
The name of the retention policy applied to the retained set.

objectProtectedType
The type of object included in the retained set.

4.1.3 Ingestion events

This section illustrates the supplemental information that is provided when table and SIP information is ingested.

The following audit supplemental information is provided for a table ingestion event:

ciSize
The size in bytes of the content information (CI) for the table.

ciCount
The number of unstructured objects that were ingested.

charCount
For structured data, the character count of the ingested table data. This value is used license storage calculations.

tableName
Name of the table that the data was ingested into.

updateType
There are three possible values:

- `INGEST_TABLE`: When data in ingested, this value appears. In this scenario, this is the value that is displayed.
- `ADD_TABLE`: When a schema is defined for a table, this value appears.
- `SET_TABLE_ALLOCATION_STRATEGY`: When the allocation strategy is changed, this value appears.

recordCount
The number of records that were ingested.

executionTime
The amount of time in milliseconds it took to ingest the table data.

tableDocumentId
The ID of the table document that was created during ingestion. Each time an ingestion is performed, a new table document is created.

tableDocumentSize
The size in bytes of the individual XML document being ingested.

The following audit supplemental information is provided for a SIP ingestion event:

name
Name of the SIP that was ingested.

dssId
The data submission session (DSS) ID that is specified in the SIP descriptor.

ciSize
The size in bytes of the content information (CI) for the SIP.

ciFormat
The format of the content information files.

priority
Ingestion sub-priority of the SIP.

sipSeqno
Sequence number of the SIP within the DSS that it belongs to.

dssEntity
The entity specified in the SIP descriptor.

phaseCode
If a SIP is successfully ingested, one of the following values is displayed:

- `COM`: Completed
- `PRUNE`: Used when the ingestion mode was configured to use AGGREGATION mode.

returnMsg
The message that was returned when the SIP was ingested. This value is often blank if there was no issue.

sipIsLast
Indicates if this was the last SIP in the ingested package.

stateCode
Indicates the state of the AIP.

commitDate
The date the package was successfully committed.

commitSync
A Boolean to indicate whether or not you performed a synchronous commit.

externalId
The external ID is an optional custom identifier assigned by the business application producing the SIP. It is used to identify the AIP after ingestion.

returnCode
The phase the AIP has reached. If an AIP is successfully ingested, the value would be `OK`.

dssPriority
Ingestion sub-priority of the SIP.

dssProducer
Code of the business application producing the SIP.

libraryMode
Indicates whether the ingestion mode was PRIVATE, POOLED, or AGGREGATED.

pdiFileSize
The size in bytes of the Preservation Description Information (PDI) file, an XML document containing structured data

sipAiuCount
Number of AIUs contained in the PDI file.

sipFileHash
Specifies the encoded hash value of the PDI file.

sipFileSize
The size in bytes of the SIP.

dssPdiSchema
URN of the schema applied by the PDI file.

executionTime
The amount of time in milliseconds it took to ingest the SIP.

pdiConfigName
The name of the PDI configuration object.

sipFileFormat
The format of the SIP file. Usually, the value is `sip_zip`.

dssApplication
The application that is specified in the `<dss>` section in the SIP descriptor.

dssHoldingName
Name of the holding where the SIP is archived, as specified in the `<dss>` section in the SIP descriptor.

ingestNodeName
The ingest node name.

ingestStartDate
The date ingestion started for the SIP.

partOfAggregate
A Boolean indicating whether the SIP was part of an aggregate.

aggregateCiSeqno
The content information sequence number. Only applicable in aggregate mode.

libraryPdiSchema
The name of the schema that defines the format of the structured data.

receiveStartDate
The reception date of the SIP.

receiverNodeName
The name of the data node that received the SIP.

aggregateAiuSeqno
The sequence number for the AIUs. Only applicable in aggregate mode.

dssProductionDate
The production date specified in the `<dss>` section in the SIP descriptor.

sipProductionDate
The production date specified outside of the `<dss>` section in the SIP descriptor.

pdiValuesCharCount
The number of characters in the unstructured data. This value is used for licensing usage calculations.

commitWaitStartDate
The start date for when the system was waiting to commit.

executionTime
There are many values to indicate how much time was spent during the processing. The values are in milliseconds.

ingestWaitStartDate
The start date for when the system waited for ingestion.

dssBaseRetentionDate
This date is used to determine the base date when applying package based retention during ingestion.

sipFileHashAlgorithm
The hash algorithm used to calculate the hash for the SIP. The default is XXH64.

aggregateLastAiuSeqno
The last AIU sequence number. Only applicable for aggregate mode.

4.1.4 Content events

The following supplemental information is provided by an audit after content is downloaded from an application:

cid
Identifier for the content.

rowId
ID for the AIU or table row.

seqno
The sequence number.

fileName
Name of the downloaded file. If the name of the file was not specified, this is the name used internally storing the content.

fileSize
Size of the downloaded file in bytes.

mimeType
MIME type of the downloaded file.

externalId
The external ID of the package.

executionTime
There are many values to indicate how much time was spent during the processing. The values are in milliseconds.

searchResultId
The ID for the search result.

4.2 Editing and deleting storage objects

This section includes a list of objects along with their respective and its properties.

When OpenText Information Archive is running (for example, ingesting, searching, etc.), it is very dangerous to update certain fields of existing objects. It is recommended that you test what you are going to do in a development environment before moving to production.

To ensure that references are not broken, objects can be deleted only when “Not In Use”.

Whether you can edit a property depends on whether the object is in use or not. After updating a property, OpenText Information Archive provides the ability to test the connection with the new configuration prior to saving your changes.

There may be an instance when you want to delete a storage system. For instance, a system was added for testing purposes and now it should be removed. For Dell EMC Elastic Cloud Storage (ECS), the system considers it is in use, even if it was just created and data nodes, PostgreSQL databases, or applications are accessing it.

Once created, the system considers Dell EMC Elastic Cloud Storage (ECS) storage to be in use and, therefore, it cannot be deleted. This is because credentials are associated with ECS storage that prohibit its deletion.

The following table outlines which properties allow or prohibit the edit and delete functions:

Legend:

- \+ – Object is allowed to be edited or deleted without any issues. Typically, this can be done for unused objects.
- +/- – Object is allowed to be edited or deleted with some restrictions or concerns. For example, incorrect configuration may cause a “data unavailability” exception to be issued.
- \- – Object is not allowed to be edited or deleted in order to not:
  - Break references between objects
  - Lose data availability
  - Move data

Object	Field	Delete		Edit		Notes
		Not In Use	In Use	Not In Use	In Use	
DataNodes						
DataNode		+	-	+	+	

	DataNode Name	Not applicable		+	+	
	Superuser Password	Not applicable		+	+	
	Connection URL	Not applicable		+	+	
Databases						
Database		+	-	+	+	
	Database Name	Not applicable		+	+	
	Admin Password	Not applicable		+	+/- (test connection)	Change of the password does not change physical PostgreSQ Lpassword. To achieve this, use PG Admin.
	DataNode	Not applicable		+	- (read-only property)	To change the dataNode, remove the current database from current dataNode and create another database for the required data node.
Storages						
PowerScale		+ (only when no SpaceRootFolders are associated)	-	+	+	
	Storage Name	Not applicable		+	+	
	Description	Not applicable		+	+	

	Folder Path	Not applicable		+	+/- (allowed when there is no data)	Remove the data first.
Local File System		+ (only when no SpaceRoot-Folders are associated)	-	+	+	
	Storage Name	Not applicable		+	+	
	Description	Not applicable		+	+	
	Folder Path	Not applicable		+	+/- (allowed when there is no data)	Remove the data first.
Dell EMC Elastic Cloud Storage		+ (only when no credentials are associated)	-	+	+	
	Storage Name	Not applicable		+	+	
	Description	Not applicable		+	+	
	URL	Not applicable		+	+/- (test connection)	
	Enable Proxy	Not applicable		-	-	Required only when proxy enabled
	Proxy URL (required only when proxy enabled)	Not applicable		+	+/- (test connection)	
	Credentials (Multiple)	+ (only when no Space is associated)	-	+	+/- (test connection)	
	Credential Name	Not applicable		+	+/- (test connection)	
	Credential Description	Not applicable		+	+	
	Access Key	Not applicable		+	+/- (test connection)	

	Secret Key	Not applicable	+	+/- (test connection)	
Dell EMC Elastic Cloud Storage (continued)	Proxy (required only when proxy enabled)	Not applicable	+	+/- (test connection)	Required only when proxy enabled
	Proxy User Name (required only when proxy enabled)	Not applicable	+	+/- (test connection)	Required only when proxy enabled
	Proxy User Password (required only when proxy enabled)	Not applicable	+	+/- (test connection)	Required only when proxy enabled
Amazon S3 Storage		+ (only when no credentials are associated) -	+	+	
	Storage Name	Not applicable	+	+	
	Description	Not applicable	+	+	
	URL	Not applicable	+	+/- (test connection)	
	Enable Glacier	Not applicable	-	-	
	Enable Proxy	Not applicable	-	-	
	Proxy URL	Not applicable	+	+/- (test connection)	Required only when proxy enabled
	Credentials (Multiple)	+ (when no Space uses Credential) -	+	+/- (test connection)	
	Credential Name	Not applicable	+	+/- (test connection)	
	Credential Description	Not applicable	+	+/- (test connection)	

	Access Key	Not applicable	+	+/- (test connection)	
	Secret Key	Not applicable	+	+/- (test connection)	
Amazon S3 Storage (continued)	Proxy	Not applicable	+	+/- (test connection)	Required only when proxy enabled
	Proxy User Name	Not applicable	+	+/- (test connection)	Required only when proxy enabled
	Proxy User Password	Not applicable	+	+/- (test connection)	Required only when proxy enabled
Custom Storage		+ (when no space is associated) +	+ (only when not in Use)	+	
	Storage Name	Not applicable	+	+	
	Description	Not applicable	+	+	
	Content Store Factory Service	Not applicable	+	+/- (test connection)	
	Properties (Multiple)	Possible adding and deleting of properties	+	+/- (test connection)	
	Name	Not applicable	+	+/- (test connection)	
	Value	Not applicable	+	+/- (test connection)	
Dell EMC CAS Elastic Cloud Storage		+ (only when not in use) +	+	+	
	Storage Name	Not applicable	+	+	
	Description	Not applicable	+	+	
	Connection String	Not applicable	+	+/- (test connection)	
	PEA (multiple)	Adding and deleting of property	+	+/- (test connection)	

	Variable	Not applicable		+	+/- (test connection)	
	Content	Not applicable		+	+/- (test connection)	
Spaces						
Space		+ (only when none of the following objects refer to it: SpaceRoot-RdbLibrary, SpaceRoot-Folder, or SpaceRoot-Object	-	+	+	Role: Admin., Developer Space can be removed as long as the following objects do not refer it: • SpaceRootRdb Database, • SpacesRoot Folder, • SpaceRoot Object
	Application	Not Applicable		-	-	Role: Admin., Developer
	Space Name	Not Applicable		+	+	Role: Admin., Developer Application for the space cannot be changed. To change the application, remove the space and create a new space for the required application.

	Database Library	Not Applicable	+	+	Role: Admin., Developer To change the database, remove the attached database, then add the new one. Attached database can be removed only when SpaceRoot Rdb-Library object that represents connection is not in use.

	Storage System	Not Applicable	+	+	Role: Admin., Developer Connection with additional storage systems can be added at any time. Connection with existing storage systems can be removed only when the object that presents connection is not in use. The following two types of objects are used, depending on the storage type: SpaceRootFolder and SpaceRoot Object.
Store		+ (only when no objects use the store) -	+	+	Role: Admin., Developer
	Store name	Not Applicable	+	+	Role: Admin., Developer
	Application	Not Applicable	-	-	Role: Admin., Developer
	Space	Not Applicable	+	-	Role: Admin., Developer

	Space Root	Not Applicable	+	-	Role: Admin., Developer
	File System Folder/ Bucket	Not Applicable	+	-	Role: Admin., Developer
	Status	Not Applicable	+	+	Role: Admin., Developer
	S3 to Glacier Settings, if applicable	Not Applicable	+	+	Role: Admin., Developer
Encryption					
CryptoObject		+ -	+	-	Role: Admin., Developer Crypto objects can be removed and edited from the user interface only when not in use. It is risky to change any crypto configuration when the object is in use.
	Name	Not Applicable	+	-	Role: Admin., Developer
	Service Provider	Not Applicable	+	-	Role: Admin., Developer
	Key Size	Not Applicable	+	-	Role: Admin., Developer
	Encryption Algorithm	Not Applicable	+	-	Role: Admin., Developer

	Encryption Mode	Not Applicable	+	-	Role: Admin., Developer
	Padding	Not Applicable	+	-	Role: Admin., Developer
	Service provider = Germalto	Not Applicable	+	-	Role: Admin., Developer
	NAE Name	Not Applicable	+	-	Role: Admin., Developer
	NAE Username	Not Applicable	+	-	Role: Admin., Developer
	NAE Password	Not Applicable	+	-	Role: Admin., Developer
	NAE Group	Not Applicable	+	-	Role: Admin., Developer
	NAE Alias	Not Applicable	+	-	Role: Admin., Developer
	NAE Keystore Password	Not Applicable	+	-	Role: Admin., Developer

4.3 IA JDBC driver

The IA JDBC driver supports PostgreSQL SELECT queries to reach archived table data.

Supported SQL feature	Example
Direct column selection	select a,b from c
Column Aliases	select a as b from c
Like	select a from b where a like ‘c%’
In	select a from b where a in (‘1’, ‘2’)
between	select a from b where c is between d AND e
is null	select a from b where a is null
order by	select a from b order by a
not (can be combined with not, in, between, is null, like)	select a from b where a not in (‘1’, ‘2’)

Supported SQL feature	Example
All columns (select *)	select * from b
Table name alias	select a.b from c a
Theta Style inner join	select ta.b,tc.d from ta,tc where ta.b = ta.d
Standard Math operations	select a from b where c*2 > 5 select a*2 from b **Note:** Standard math syntax is supported.
Schema Selection	select a from schema1.table1 **Note:** If no schema is defined, then default schema will be used as specified in database metadata.
ANSII Style inner join	select ta.b,tc.d from ta join tc where ta.b = ta.d
ANSII Style left outer join	select a from b left outer join c on b.d = c.d
ANSII Style right outer join	select a from b
Group by	select a, count(*) from b group by a
Aggregate functions	select count(*),max(*),min(*),max(*),sum(*)
Fetching first n rows	select a from b limit 2
Fetching rows starting at specific point in query	select a from b limit 2 offset 5
Case insensitive column/table selection	select AnyCaseColumn from AnyCaseTable **Note:** Specific column/table will be determined at runtime based on metadata.
select on in clause	select a from b where c in (select d from e)
union all	select a from b union allselect c from d
distinct	select distinct a from b select distinct a,b from c select distinct * from a select a, count(distinct b) from c group by a
having clause	select a, count(*) from b group by a having count(*) > 2

Supported SQL feature	Example
Column names referencing tables is not required	Select a, b from c join d on c.key=d.key **Note:** If the column name appears in both tables, you must reference a specific table.
column alias use in group by, order by, and having clause	select a c, d from d group by corder by c having c > 2
sub select keywords – any/all/some	SELECT * FROM AlbumSales WHERE album_gross > ALL (SELECT album_costs FROM AlbumProduction);
exist	SELECT * FROM suppliers WHERE EXISTS (select * from orders where suppliers.supplier_id = orders.supplier_id);
Select from other select	select a,b from (select a,b from c where c > 2)
Left/right outer join	select a from b join c on a.b = c.d right join d on a.c = a.d

Unsupported SQL feature	Example
Updating existing content	update …
Deleting existing content	delete ...
Inserting new content	insert into …
Multiple selections	only a single statement will be processed select a,b from c; select c,d from e is not valid

OpenText Information Archive transfers SQL statements to the PostgreSQL JDBC driver.

Supported data types from java.sql.Types: BIT (of length 1), SMALLINT, INTEGER, BIGINT, FLOAT, REAL, DOUBLE, NUMERIC, DECIMAL, CHAR, VARCHAR, DATE, TIMESTAMP, BOOLEAN, NCHAR, NVARCHAR.

Unsupported data types from java.sql.Types: TINYINT, LONGVARCHAR, TIME, BINARY, VARBINARY, NULL, OTHER, JAVA_OBJECT, DISTINCT, STRUCT, ARRAY, BLOB, CLOB, REF, DATALINK, ROWID, LONGNVARCHAR, NCLOB, SQLXML, REF_CURSOR, TIME_WITH_TIMEZONE, TIMESTAMP_WITH_TIMEZONE.

Chapter 5

Glossary and acronyms

The following table contains short definitions for OpenText Information Archive resources, processes, and mechanisms:

Name	Acronyms and Related Terms	Description
Active Archiving	None	An archiving process whereby data is ingested into an archive on a time-related basis (for example, per hour, per day, per week, etc.).
Active Directory	AD	One of the authentication mechanisms supported by OpenText Information Archive.
Application	None	A logical configuration object in an archive system that presents the customer business item for preserving and storing the data. The following is the logical archive object hierarchy: Tenant -> Application -> Holding. An application can be one of the following types: SIP or table.
Application Decommissioning	None	An archiving process whereby data is extracted from a legacy application and ingested into an archive system to reduce the cost of supporting the legacy application.
Archival Information Package	AIP (Package)	An archive resource that represents the package of information inside OpenText Information Archive. It may contain from zero to multiple structured data elements, which are represented as XML, and/or zero to multiple unstructured data elements. AIP is a shorter name for a package.
Archival Information Unit	AIU (Record)	An information atom. The smallest archival unit of an information package. An AIP contains AIUs.
Archive Information Collection	AIC	Search configuration resource that contains a set of criteria to be used during a search. It organizes a set of AIPs that support flexible and efficient data access.

Name	Acronyms and Related Terms	Description
Audit	None	An audit indicates that a particular action has occurred. A particular audit is associated to an event type that defines the action that occurred (for example AIP / DISPOSE indicates that an AIP was disposed). Audits can be searched from the audit application after running the Ingest Audits job. Audits can be associated to the system, a tenant, or a particular application,
Audit Event Type	None	An audit event type is a pairing of a type and name that act as a mechanism for enabling an audit. For example, it is possible to enable a DISPOSE event for a TABLE and an AIP, as each one is an audit event type.
Background Ingestion	None	Ingestion performed in the background. After the ingestion command call, the server creates a background request and returns immediately a response without having to wait for the end of ingestion. See also Direct Ingestion on page 223 and Batch Ingestion on page 220.
Background Search	None	A background search is a search that runs asynchronously and is associated to a background task. A search may need to run in the background if the number of results returned is large, or if the content is offline. A user can request that a search be run in the background.
Backup	None	Copying the data in a recovery mechanism that allows data to be restored in the event of data loss or corruption. This copy of the data is also called as backup.
Batch Ingestion	None	A SIP ingestion approach that consists of three steps: Reception, Enumeration, and Ingestion. When ingesting SIPs in a batch, it is required to first receive all SIP packages and then ingest them into the application. Use the batch ingestion approach when there is a large set of SIPs that need to be archived. See also Direct Ingestion on page 223.

Name	Acronyms and Related Terms	Description
Batch Processing	None	An approach used to improve the performance of the long-running operations and jobs, such as applying/removing holds and retention policies. With batch functionality, an operation is broken into smaller chunks. Even if there is a small number of items to process, at least one batch is created.
Bucket	None	A storage configuration resource used within Dell EMC Elastic Cloud Storage, Microsoft Azure, Archive Center, and Amazon S3 Storage, and PowerScale OneFS S3 storage systems.
Cache-In	None	An action on an AIP object to restore the object in a library from backup. The opposite action is referred to as a cache-out. Cache-in and cache-out provide the ability for SIP archiving applications to improve performance by reducing the number of libraries in Lucene.
Cache-Out	None	An action on an AIP object to detach the object from a library. The opposite action is referred to as a cache-in. Cache-in and cache-out provide the ability for SIP archiving applications to improve performance by reducing the number of libraries in Lucene.
Chain of Custody	None	A set of tests that check the integrity of the ingested tables in OpenText Information Archive.
Content Identifier	CID	An internal identifier for content information that can be associated to one or more records.
Confirmation	None	A message generated in reaction to an AIP event (lifecycle transition).
Content Information	CI	A piece of unstructured content that is associated with a record. For each CI, there is corresponding RI entry in the table of contents (`RI.xml`).
Content Item	None	Internally, unstructured content that is related to an OpenText Information Archive configuration object is represented as a Content Item.

Name	Acronyms and Related Terms	Description
Crypto Object	None	The general encryption configuration object that contains settings for encryption/ decryption, such as the security provider name, encryption algorithm, padding, key size, *etc*. All crypto objects can be seen in the IA Web App on the **Administration > Encryption** page.
Custom Search	None	Allows a Search Designer more control on how to show the results of a search.
Data Submission Session	DSS	A delivery of media or a single telecommunications session that provides data to a consumer.
Database	None	Archive information resource that presents a database for table-based applications.
Database Crypto	None	A configuration object specific to table-based applications only. The object contains a reference to `crypto object` with settings for a security provider name, encryption algorithm, padding, key size *etc*. The object is used for configuring encryption/decryption of structured and unstructured data for table-based applications.
Declarative Configuration	DC	The way to describe a set of configuration objects in YAML format. Declarative configuration is used to define entire system information, an application, a holding, or a search.
Default Crypto Object	None	OpenText Information Archive uses this object to encrypt and decrypt sensitive data, such as passwords for RDB databases, nodes and credentials for remote storage systems, in the System Data and Search Results databases. It is prohibited to modify or delete the default crypto object because, once the object is in-use and password encrypted with its settings, after the change of object properties, the existing encrypted passwords with the old settings can no longer be decrypted with the new properties' values. OpenText Information Archive blocks change of properties of default crypto object, as it is always "in-use".
Delivery Channel	None	The configuration resource that defines a destination where to send search results.

Name	Acronyms and Related Terms	Description
Direct Ingestion	None	A SIP ingestion process that allows the archival of a single SIP in one request, simultaneously, avoiding "receive-enumerate-ingest" steps. Direct ingestion is used in case a single SIP is not archived frequently. Selection of the ingestion approach is done based on performance requirements. See also Batch Ingestion on page 220.
Disposition	None	The controlled process of removing data from the archive after the required aging period has elapsed (defined by the retention polices applied). Only items that are under retention go through the disposition process. The disposition process has the following steps: 1. Put information into purge lists. 2. Obtain approval to dispose list. 3. Run the Disposition job. Depending on the resource being disposed, additional steps may be required (for example, disposing AIPs require that the Confirmation job is run).
Dissemination Information Package	DIP	An archive resource that presents an information package that is returned to the user via search or some other retrieving operation.
Export Pipeline	None	An xProc pipeline that is specifically intended to export OpenText Information Archive search results.
File System Folder	None	The storage configuration object that represents a file system folder in which unstructured content is stored.
File System Root	None	Storage configuration object that represents the storage of a Local File System or a PowerScale type and indicates a root location on a disk.
Group	None	A group defines a set of users. Groups are used to restrict access to applications and searches.
Hold	None	A compliance configuration object. A hold is applied to an object to block deletion or disposition, either temporarily or indefinitely.

Name	Acronyms and Related Terms	Description
Holding	None	A logical configuration destination archive in which to ingest and store data that shares common characteristics used for SIP applications. For example, you can create a holding to archive data from the same source application (such as ERP data), or of the same format (such as audio recordings), or belonging to the same business entity.
Holding Composition	None	A configuration object used by the Holding Wizard to define a new holding.
Holding Crypto	None	A holding configuration object specific to SIP-based applications only. The object contains references to `crypto` objects with settings for a security provider name, encryption algorithm, padding, key size, *etc*. The holding object is used for configuring the encryption for structured, unstructured data of the SIP packages, as well as the SIP descriptor. See also crypto object on page 222, database crypto on page 222, PDI crypto on page 226, and default crypto object on page 222.
Hold Set	None	A logical container that references items with a hold applied against it. A hold set is created when a hold is applied to one or more items.
IA Shell	CLI	The Command Line Interface for OpenText Information Archive. It is a tool that provides the set of commands for the Administrator to manage the product and its resources.
OpenText Information Archive	IA	OpenText Information Archive is an integrated product suite designed for application agnostic information management and archiving. It is an information management system that preserves, maintains, and controls continuing access to valuable enterprise information assets.
Ingest	None	Ingestion is the process that registers an AIP (for SIP-based applications) or table data (for table-based applications) into an archive so the data can become searchable.

Name	Acronyms and Related Terms	Description
Ingestion Mode	mode	Defined on a holding for SIP applications. The way business data is preserved. There are three types of ingestion modes supported by OpenText Information Archive: PRIVATE, POOLED and AGGREGATE.
Ingestion Node	None	An archive process configuration resource that defines the parameters for the ingestion and enumeration processes.
Invalidation	None	“Invalidate” is an action on an AIP package. Invalidation is required in case an incorrect SIP was submitted, and the correct SIP must be submitted with the same identifier. After invalidating an AIP, it is immediately removed from a search's scope.
Job	None	A job defines a type of operation that can be done asynchronously. Jobs can be configured to either run on a schedule or run manually. Examples of jobs include the Clean job, Disposition job, and the Ingest Audits job.
Job Instance	None	Represents either a scheduled or running instance of a job definition.
Library	None	A storage configuration resource that represents a container stored within a database. Libraries can be taken offline (cached out).
Library Policy	None	An archive process configuration object that contains a set of properties that define when a library can be closed when ingestion in POOLED and AGGREGATE modes.
Lightweight Directory Access Protocol	LDAP	One of the authentication mechanisms supported by OpenText Information Archive.
Nested Search	None	A search within a search that allow more flexibility for creating searches.
Open Archival Information System	OAIS (ISO 14721)	A reference model that a wide variety of organizations use for archiving digital information for long-term preservation. It specifies the format that data is ingested into, stored in, and retrieved from OpenText Information Archive throughout the information's lifecycle.
OpenText Directory Services	OTDS	One of the authentication mechanisms supported by OpenText Information Archive.

Name	Acronyms and Related Terms	Description
Order	None	A configuration object used by SIP applications to set permissions and retention duration when running an Order Item (search).
Order Item	Background task	An order item represents an operation that is being processed asynchronously. Order items can be initiated by a user and can be viewed from the Background Tasks tab. Order items can also be created by the system for jobs. Order items may also be split into chunks called batch items (for very large operations, such as applying hold to search results).
Partitioning Keys	PKEY	Keys defined on a holding in a SIP application to improve search performance. A partition key is used in the first tier of the query process to limit the data returned when a search is run. Partition keys are created during ingestion and are stored in the AIP object.
PDI Crypto	None	A configuration object specific to SIP-based applications only. It holds cryptography related information, such as indexes, partitioning, etc., for ingestion processors in its XML content file. It fully replaces indexes and pkeys from main PDI configuration object.
PDI Schema	None	A holding configuration resource that contains a reference to the XML schema to be used for validation of the PDI XML file during the ingestion.
Permission	None	Indicates which groups have access to either an application, search set, or an AIP.
Preservation Description Information	PDI	A OAIS standard and related to the mandatory SIP package resource with business data to be put into archive. There is `eas_pdi.xml` file in the SIP package, which is a PDI descriptor with business data.
Purge Candidate List	None	A list that contains items that qualify for disposition. Purge candidate lists need to be approved before the items can be disposed. Purge candidate lists have a state that controls what actions can be taken. The Clean Up Purge Candidate Lists and Applications job can be used to remove canceled or disposed purge lists.

Name	Acronyms and Related Terms	Description
Query	None	A search configuration resource that defines options for building a query to retrieve records.
Query Quota	None	A search configuration resource that defines the maximum number of records to be returned in a search. Query quotas are specific to searches in SIP-based applications.
RDB Database	None	A storage configuration resource that represents a database in Postgres. It contains a set of properties to access physical database.
RDB Data Node	None	A storage configuration resource that contains a set of properties to establish a connection with a physical Postgres data node (formerly referred to as a federation). A data node is a container for databases.
Reception	None	The process of transferring SIP packages to IA Server with preparation of SIP packages for ingestion. Usually in part of ingestion process with sequence `receive`, `enumerate`, `ingest`. The reception stage is used in "batch" ingestion process.
Reception Node	None	An archive process configuration resource that defines the parameters for the receiver process.
Record	None	A record is either a row in a table or an AIU in an AIP.
Reference Information	RI	Identifies and, if necessary, describes one or more mechanisms used to provide assigned identifiers for the CI. It also provides those identifiers that allow outside systems to refer to this CI.
Rejection	None	Rejection is one of the AIPs lifecycle states that can happen if the data did not match the schema. "Reject" is an action on the AIP package. Rejection and Invalidation are similar. The difference is at what state the AIP was in.
Result Configuration Helper	None	A search configuration resource that defines a set of columns that can be used by the Result Master to help configuring the search result page.

Name	Acronyms and Related Terms	Description
Result Master	None	A search configuration resource that represents the result search page columns and tabs (in-line panels and detail panels). Result masters are part of a search set.
Retention	None	A general term that indicates how long content should be kept for compliance. Retention is associated to items via applying a retention policy and for many types of the retention policy, a base date is specified.
Retention Class	None	An alias that can be used to associate a retention policy that will be applied to the SIP on ingestion. The retention class overrides any default that is specified by the holding. The holding must specify the name of the retention class and map to zero or more retention policies that would be applied.
Retention Policy	None	A compliance configuration object that specifies the rules for how long to retain the data.
Retention Set	None	A logical container that references data under retention, including whether items in the set are aging together and the type of items in set (for example, application, package, table or record).
Role	None	A role is mapped to set of actions that can be done in OpenText Information Archive, such as run search, apply retention, or ingest content. Groups are mapped to roles. Roles are fixed by OpenText Information Archive. Some examples include End User, Retention Manager, Administrator, Developer, and E-Discovery Administrator.
Rule	None	Rules can be defined to apply retention or holds to records or packages. Rules can be evaluated by running the associated jobs (Apply Retention via Rule, Apply Hold via Rule). A rule contains one or more individual rules using DROOLS. When defining a rule, the type of action the rule is for is defined. Rules are defined per application, and can be defined using declarative configuration.
Schema	None	An archive information resource that presents a schema for table-based applications. Schemas are associated with a database and contain one or more tables.

Name	Acronyms and Related Terms	Description
Search	None	The primary method used to access data that has been ingested by OpenText Information Archive. The process of searching an archive to retrieve the set of AIUs that satisfy the search criteria. At the same time there is a search configuration object in OpenText Information Archive that is used to represent the search item for either a SIP or table application. A search contains one or more search compositions.
Search Composition	Search Set	A search configuration resource that manages the search components (XForm, result master, search, permissions) and is used to run a search.
Search Quota	AIU quota per parent	Maximum number of AIUs per AIP parent. The value zero indicates an unlimited number.
Space	None	A storage configuration object that represents the relation between storage and application. It is used by: Space Root Folder, Space Root Object and Space Root RDB Library.
Space Root Folder	None	A storage configuration object that represents the relation between a Space and a Local File System/PowerScale storage systems.
Space Root Object	None	A storage configuration object that represents the relation between a Space and Dell EMC Elastic Cloud Storage/Amazon S3 storage systems.
Space Root RDB Library	None	A storage configuration object that represents the relation between a Space and an RDB database.
SQL Query	None	A search configuration resource that contains the query to be used during the search for table-based applications.
Storage	None	A storage configuration object that contains a list of properties for target storage configuration. A storage system holds data, such as unstructured content for records, library backups, raw XMLs, ingestion logs, *etc*.

Name	Acronyms and Related Terms	Description
Storage End Point Credentials	None	A storage configuration object that represents the user credentials used when establishing a connection to the target storage system. The object is used by storages of following types: Dell EMC Elastic Cloud Storage, Amazon S3 Storage, and PowerScale OneFS S3.
Store	None	A storage configuration object that contains properties for linking a space with a File System Folder or Bucket. Stores holds records in a context of an application.
Submission Information Package	SIP	An archive resource that presents an information package before the ingestion process. It is a term from OAIS standard for input archive package.
Table	None	An archive information resource that presents a table for table-based applications. Tables are contained within schemas.
Tenant	None	A logical configuration object in an archive system that presents a customer business item for preserving and storing the data. Tenants store zero or more applications.
Transformation	None	A configuration resource that defines the XQuery/XSLT to use to perform a transformation. Transformations are associated with holdings.
Universally Unique Identifier	UUID	An identifier that is used to identify OpenText Information Archive resources. Every resource has its own and unique UUID.
Value List	None	A search configuration resource that identifies the list of possible values in an XML document. Value lists are used by search forms to externalize the values to avoid the search forms storing the information. Value lists are per application and can be used by multiple search forms.
xForm	None	The search resource that represents a user form that contains search criteria on the UI level.
xProc	W3C Recommendation	A W3C Recommendation that defines an XML transformation language to define XML Pipelines. For information on xproc see the W3C website