

Configure Archive

Last Modified on 02/05/2024 7:04 am EST

Starting with V10.0, Cora SeQuence has been renamed to Cora Orchestration.

V9.9 and later

Overview

Setting up the archiving mechanism involves configuring:

- Archive policies: Each policy determines the data to be archived based on your organization's data retention needs and business and legal requirements for retrieving data.
- Archive jobs: Data archiving is performed by JES jobs that you need to set up on the Administration site.

Job	Description
Archiving Data Collector	Marks the data that needs to be archived based on a JSON file.
Archiving Data Worker	Runs right after the Archiving Data Collector job. Collects the marked data, transforms it from SQL to CSV format, and moves it to the archiving location.
Archiving File Worker	Moves the case files (attachments) to the archiving location.
Archiving Data Cleaner	Deletes the archived data from the operational database.
Archiving Data Retriever	Creates a ZIP file with the archived data for download.

Configure archive

All the archiving configuration steps are defined in the Administration site.

	Step	Performed by
1	Define archive data	<ul style="list-style-type: none">• Architect• Tech Lead• Developer
2	Set up archiving jobs	<ul style="list-style-type: none">• Architect• Tech Lead• Developer

Prerequisites

Before you configure the archiving policies and jobs, make sure that:

- The application administrator or the DBA has deployed an archive database.
- You have the connection string to the archive database.
- You have the connection string to the operational database.

1. Define archive data

Archive policies determine which data from the operational database will be moved to the archiving database. Archive policies can also determine if the data will be deleted without archiving or kept in the operational database (not deleted and not archived). You configure archive policy requirements in a JSON file.

The JSON file includes two main objects:

- "collector": Here you mainly define which workflow's data you want to archive and from when should it be archived.
- "archive": Here you mainly define how to archive and retrieve the collected data (system tables, custom tables, archive filters).

JSON parameters

IMPORTANT

The JSON parameters are case sensitive.

Parameter	Description
<code>collector\workflowSpaceId</code>	<p>Determines which workflow configuration will be considered for archiving. To use system archiving configuration, delete this parameter. (Mandatory)</p> <p>Value: Workflow space template GUID</p> <p>The workflow instance determines the archive or delete date.</p>

Parameter	Description
<code>collector\variables</code>	<p>Array of variables which determine the criteria for archiving the completed instances of the specified workflow space.</p> <p>If not set, then the collector job will archive all the completed instances of that workflow space.</p> <p>For each variable, set these parameters:</p> <ul style="list-style-type: none"> • "name": Name of variable in the workflow. • "parameterName": By default, same value as name. • "dbType": Variable datatype for example, "dbType": "String", • "value": Permanent value of variable to transfer • "comparisonOperator": Comparison operator to check the variable value <p>Expected values:</p> <ul style="list-style-type: none"> ◦ eq ◦ gt ◦ lt ◦ gte ◦ lte ◦ isnull ◦ isnotnull • "logicalOperator": Expected values: <ul style="list-style-type: none"> ◦ - OR ◦ & - AND ◦ NOT
<code>collector\completedAfter</code>	<p>Sets the date after which data is archived or deleted.</p> <p>Value: Date (yyyy-mm-dd)</p> <p>Example: 2015-12-23</p>
<code>collector\olderThan</code>	<p>Sets the upper limit date for archiving data. All the data created before this date is archived.</p> <p>Set these parameters within:</p> <ul style="list-style-type: none"> • amount: number • timeUnit: date unit like day (d), month (m), or year (y) (Mandatory) <p>For example, amount set as 1, and timeUnit set as d archives all the data that is 1 day older than the current date.</p> <div data-bbox="603 1727 1417 1883" style="background-color: #e1f5fe; padding: 10px; border: 1px solid #cfcfcf;"> <p>NOTE</p> <p>If <code>completedAfter</code> is set, then the policy archives the completed instances that meet both the <code>olderThan</code> and the <code>completedAfter</code> values.</p> </div>

Parameter	Description																
<code>collector\archiveWhenOpenChildrenExists</code>	<p>Determines whether open workflows are archived or not.</p> <p>Value:</p> <ul style="list-style-type: none"> • True • False (default) <p>If set to <code>false</code>, the master workflow is archived only after all subworkflows are closed.</p> <p>If set to <code>true</code>, the master workflow is archived with closed subworkflows only, and does not check for open subworkflows. The open subworkflows remain in the operational database.</p>																
<code>collector\archiveAsStandAlone</code>	<p>Determines which configuration is used for archiving. If set to <code>true</code>, each completed workflow instance is archived separately.</p> <p>Value:</p> <ul style="list-style-type: none"> • True • False (default) <div data-bbox="603 862 1414 1756" style="border: 1px solid #add8e6; padding: 10px; margin-top: 10px;"> <p>Correlation between <code>collector\archiveAsStandAlone</code> and <code>collector\archiveWhenOpenChildrenExists</code></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="border: none;"></th> <th colspan="2" style="border: none;"><code>collector\archiveWhenOpenChildrenExists</code></th> </tr> <tr> <th colspan="2" style="border: none;"></th> <th style="border: none;">True</th> <th style="border: none;">False (default)</th> </tr> </thead> <tbody> <tr> <th style="border: none;"><code>collector\archiveAsStandAlone</code></th> <th style="border: none;">True</th> <td style="border: none;">All completed instances of the space guid are being archived.</td> <td style="border: none;">All completed instances of the space guid are being archived.</td> </tr> <tr> <th style="border: none;"></th> <th style="border: none;">False</th> <td style="border: none;">The master workflow is archived with closed subworkflows only. The open subworkflows remain in the operational database and are not archived.</td> <td style="border: none;">The master workflow is archived only after all subworkflows are closed.</td> </tr> </tbody> </table> </div>			<code>collector\archiveWhenOpenChildrenExists</code>				True	False (default)	<code>collector\archiveAsStandAlone</code>	True	All completed instances of the space guid are being archived.	All completed instances of the space guid are being archived.		False	The master workflow is archived with closed subworkflows only. The open subworkflows remain in the operational database and are not archived.	The master workflow is archived only after all subworkflows are closed.
		<code>collector\archiveWhenOpenChildrenExists</code>															
		True	False (default)														
<code>collector\archiveAsStandAlone</code>	True	All completed instances of the space guid are being archived.	All completed instances of the space guid are being archived.														
	False	The master workflow is archived with closed subworkflows only. The open subworkflows remain in the operational database and are not archived.	The master workflow is archived only after all subworkflows are closed.														

Parameter	Description
<code>collector\childWorkflowSpaceIdsToExclude</code>	Determines which subworkflows should be excluded from archiving. For example: ["182b2d2b-ebb2-45c2-a94f-ab276912c22a"]
<code>archive\defaultArchiveGroup</code>	Determines which tables to archive. Value: <ul style="list-style-type: none"> • All: The system archives all the default system instance tables, the UACT tables, and the UWF tables. • Explicit: The system archives only the tables that you specify in the tables array.
<code>archive\tables</code>	Array of tables, and specific actions for each table. Define this parameter to include custom tables or exclude/delete unnecessary tables. The action depends on the setting of the parameter <code>defaultArchiveGroup</code> . <ul style="list-style-type: none"> • "schema": The SQL schema where this table exists (Mandatory) • "name": The SQL table name (Mandatory) • "alias": The alias for this table to be used in the retrieved data file. • "masterInstanceColumnName": Value that connects between the specified table and the master instance workflow Id (Mandatory) • "columns": List of the columns to be archived. You set the name of the columns and you can set the alias to be used in the retrieval. • "indexColumns": List of the columns to be indexed. The columns you specify are displayed in the portal and used as filters for retrieving the archived data. • "action": What to do with the defined data (Mandatory) <ul style="list-style-type: none"> ◦ Delete: Data is removed from the operational database without archiving. ◦ Archive: Data is archived and removed from operational database. ◦ Exclude: Data is not archived and not removed from the operational database. Excluded data is not indexed. This action cannot be applied on <code>tblInstanceWorkflowsClosed</code> .

Parameter	Description
<code>archive\simulationMode</code>	<p>Determines whether or not, after archiving, the data is removed from the Cora SeQuence database. (Mandatory)</p> <p>Value:</p> <ul style="list-style-type: none"> • True • False (default) <p>In Simulation Mode, the data is not written to the archiving tables like <code>arc.tblInstanceWorkflowsClosed</code>. Instead, the data is written to from the operational database. In this mode, the data is not deleted</p> <div style="background-color: #e1f5fe; padding: 5px; border: 1px solid #ccc;"> <p>Tip Use the simulation mode to check if the archiving settings are properly configured.</p> </div>
<code>moveFilesToFilesArchiveStorage</code>	<p>Determines whether archived files will be moved to archive file storage.</p> <p>Value:</p> <ul style="list-style-type: none"> • True (default) • False <p>If set to <code>true</code>, the actual files (blobs) move to the archive file storage.</p>

[Download the JSON template](#) .

Create an archive policy

When you create a policy you define which data needs to be archived, deleted, or left in the operational database. You also set time parameters, such as "older than" or "completed after," and grant permissions to individuals or groups to access the archived data.

1. In the Administration site, go to **Administration>Archiving>Policy Management**.
2. Click **Add New record**.

The screenshot shows the 'Add Record to: Policies' interface in the COQA SeQUENCE application. The left sidebar contains a navigation menu with 'Administration' expanded, and 'Policy Management' highlighted with a red box. The main form area contains the following fields and controls:

- Policy Name ***: Insurance_North
- Policy Alias ***: Northinsurance
- Database Connection Name ***: NIDB1
- Archive File Connection Name ***: NIFC1
- Configuration ***: A JSON configuration block with line numbers 1-18. The configuration defines a collector with a workflow space ID, completion time, and amount, and an archive with a default group, simulation mode, and file storage settings.
- Policy Enabled**
- Set Permissions** button
- Validate** button
- Add** button (highlighted with a red box)
- Cancel** button

3. Enter the following details:

- Policy Name: A unique policy name. It is used as the root of the archive folder structure.
- Policy Alias: An alias for the policy. The alias you enter here displays on the archive page in the portal, when the user selects a policy.
- Database Connection Name: Name of the operational database connection: *From where the data is taken.*
- Archive File Connection Name: Name of the archive database connection: *Where the data will be archived.*
The supported storage types are **AWS S3**, **Azure File Storage** and **Network Storage**.
- Configuration: JSON parameters that determine which data to archive.
- Policy Enabled: Select the checkbox to activate the policy.
- Set Permissions: Specify the users who can access the archived data in the portal.

4. Click **Add**.

The **Validate** button checks the JSON syntax and prompts any issues.

2. Set up the archiving jobs

The configuration of most archiving jobs is similar. You basically set up the job's name, batch size, and scheduler. For the Archiving Data Retriever job, you only need to set up its timeout.

The archiving jobs are interdependent and work in a linear mode. For example, the Data Worker job only runs after the Data Collector job.

Order	Job	Parameters
-------	-----	------------

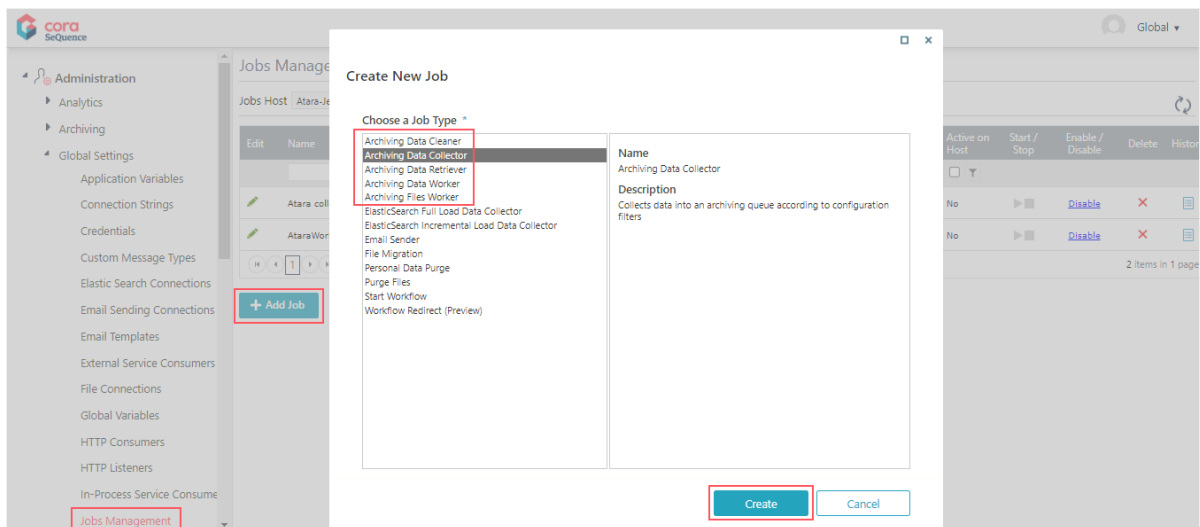
Order	Job	Parameters
1	Archiving Data Collector	Name
2	Archiving Data Worker	Work Batch Size
3	Archiving Data Cleaner	Batch Iteration Delay (seconds)
4	Archiving Files Worker	Job Execution Scheduler
		Job Start and End (optional)
		Delay job start
5	Archiving Data Retriever	Stop Job if it executes more than
		Retrieval Data Timeout (minutes)

NOTE

The archiving jobs do not support Scaling.

Configuring the archiving jobs

1. Go to **Administration>Global Settings>Jobs Management**, and click **Add Job**.



2. Select an archiving job, and then click **Create**.
3. Configure the archiving job:
 - a. On the **Job** tab, define the following:
 - **Name:** Enter a meaningful name for the job.
 - **Job is enabled:** Select the checkbox only if you want to run the job for testing purposes.
 - Click **Next**.
 - b. On the **Command** tab, set the following:
 - **Work Batch Size:** Batch Size is *the number of records* that are synced at one time. Depending on the complexity of the cases, you may consider to set a smaller batch size. The default is set to 100. You can enter up to 10000.
 - **Batch Iteration Delay (seconds):** Time in seconds, after which the next batch will be synced. Default is 5 seconds.
 - Click **Next**.
 - c. On the **Advanced Options** tab, set the following:
 - **Job Execution Schedule:** Set the job to execute once a day.

- Click **Finish**.

Configuring the Archiving Data Retriever job

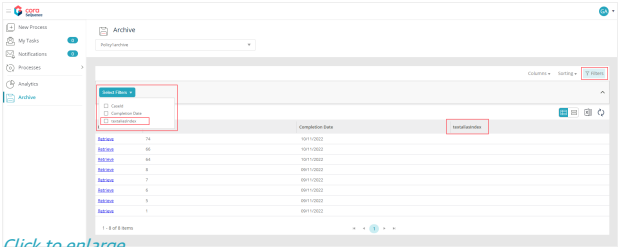
1. Go to **Administration>Global Settings>Jobs Management**, and click **Add Job**.
2. On the **Job** tab, define the following:
 - **Name**: Enter a meaningful name for the job.
 - **Job is enabled**: Select the checkbox only if you want to run the job for testing purposes.
 - Click **Next**.
3. On the **Command** tab, enter a retrieval timeout value in minutes.

After you set up all the archive jobs and policies, enable the jobs on the Jobs Management page.

Archive index

The `indexColumns` array defines how the data is indexed and stored. When you set up this parameter, you define the columns that are displayed on the Archive page and by which the end user can filter the retrieved data.

When the Archiving Data Worker job runs, it populates the archive database with the columns that you specify in the `indexColumns` array and the Case ID and Completion Date columns, which are created by default.

Policy configuration	Portal
<pre>"indexColumns": [{ "name": "text", "indexColumnName": "textaliasInde }], "action": "Archive" },</pre>	 <p><i>Click to enlarge.</i></p>

Use cases

In general, you define a policy to archive data, but if required, you can exclude specific tables from archiving or completely delete the data from the operational database.

- Exclude specific tables:
 - Set the `defaultArchiveGroup` parameter to "all", and under "Tables", specify the tables that you want to exclude with the "Exclude" action.
- Delete specific tables from the operational database without archiving them.
 - Set the `defaultArchiveGroup` parameter to "all", and under "Tables", specify the tables that you want to exclude with the "Delete" action.

When you set the "exclude" or "delete" actions, there's no need to define "columns" or "indexColumns".