



HIPS

Data Archiving

18 April 2020 7.3

Approved for external use

Acknowledgements

Council of Australian Governments

The Australian Digital Health Agency is jointly funded by the Australian Government and all state and territory governments.

HL7 International

This document includes excerpts of HL7™ International standards and other HL7 International material. HL7 International is the publisher and holder of copyright in the excerpts. The publication, reproduction and use of such excerpts is governed by the [HL7 IP Policy](#) and the HL7 International License Agreement. HL7 and CDA are trademarks of Health Level Seven International and are registered with the United States Patent and Trademark Office.

7-Zip

The HIPS Data Archiving makes use of the 7-Zip program for compressing archived files. 7-Zip is licensed under the GNU LGPL License. www.7-zip.org

Disclaimer

The Australian Digital Health Agency ("the Agency") makes the information and other material ("Information") in this document available in good faith but without any representation or warranty as to its accuracy or completeness. The Agency cannot accept any responsibility for the consequences of any use of the Information. As the Information is of a general nature only, it is up to any person using or relying on the Information to ensure that it is accurate, complete and suitable for the circumstances of its use.

Document control

This document is maintained in electronic form and is uncontrolled in printed form. It is the responsibility of the user to verify that this copy is the latest revision.

Copyright © 2020 Australian Digital Health Agency

This document contains information which is protected by copyright. All Rights Reserved. No part of this work may be reproduced or used in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems – without the permission of the Australian Digital Health Agency. All copies of this document must include the copyright and other information contained on this page.

OFFICIAL

Document information

Key information

Owner	Executive General Manager Innovation and Development
Contact for enquiries	Australian Digital Health Agency Help Centre
Phone	1300 901 001
Email	help@digitalhealth.gov.au

Table of contents

1	Summary.....	5
2	Prerequisites	6
2.1	Install SQL Server Agent.....	6
2.2	Configure SQL Server Agent.....	6
2.3	DB Upgrade	6
2.4	Configuration	7
3	Data Archiving.....	8
3.1	On Demand Execution	8
3.2	Setup SQL Jobs	9
3.3	Restore Data	13
Appendix A	ArchiveSettings.psd1	14
Appendix B	Archive Settings in Database	15
Appendix C	Archiving Steps and Conditions	20

1 Summary

The purpose of the HIPS Database Archiving script is to maintain the HIPS-Core database over time to ensure HIPS performance is not impacted due to excessive database growth.

The archiving job will move the accumulated data and logs found in several database tables into flat files before clearing the records out of the HIPS-Core database tables. For most tables, rows are deleted after being archived, however for the ClinicalDocumentVersion table, rows are retained with the Package column set to null. Documents whose CDA package is removed in this way can be superseded or removed from the My Health Record but cannot be viewed from the local database.

Sites implementing this archiving job will need to devise their own strategy beyond the basic move that this script performs. They may wish to move these files to a different file system or some form of long-term storage.

Tables containing HI Service audit logs are archived into “Non-Purgeable” folders as there is a complex retention requirement that often in practice results in the audit logs being kept for an extended period.

The remainder of the tables are archived into “Purgeable” folders as the site may purge these files when the site no longer requires the information. Each table archived is compressed to a zip file using the 7-Zip program, which is included in the package.

This archiving job only moves the records to a file system accessible from the database server. This will free up space in the HIPS-Core database, but the archived files will continue to grow and consume storage space. Sites will need to address this and act upon this issue by devising their own archiving strategy.

This archiving job will be bundled in HIPS version 7.3 onward and is available as an individual package for v7.x versions prior to 7.3.

2 Prerequisites

This section outlines the major prerequisites that an implementer will need to set up to run the data archiving job.

2.1 Install SQL Server Agent

If not already installed, install SQL Server Agent on the database server and configure it to start up automatically.

2.2 Configure SQL Server Agent

- 1 Create a new role called 'RunArchiving' to the HIPS Core database if it is not created yet. This role is created for archiving purpose only.
- 2 Add the <HIPS_ARCHIVING_SERVICE_ACCOUNT> as a login to the SQL Server and make sure <HIPS_ARCHIVING_SERVICE_ACCOUNT> is assigned RunArchiving, db_datareader and db_datawriter roles in the HIPS Core database.

2.3 DB Upgrade

It is important to make sure that the HIPS Core database has the latest database changes provided in this package.

- 1 Locate the folder <DATA_ARCHIVING_INSTALL_SOURCE>\HIPS-AppServer\database and application named HIPS.PcehrDataStore.DBUpgrade.exe. This application will install or upgrade the HIPS Core database archiving objects and data into an existing HIPS Core database.

Example:

```
.\HIPS.PcehrDataStore.DBUpgrade.exe upgrade-db "Data  
Source=<SQL Server Name>;Initial Catalog=<HIPS CORE Database  
NAME>;Integrated Security=SSPI;Connect Timeout=15;"
```

IMPORTANT

Ideally the login used to connect to the specified SQL Server instance must be a member of the *sysadm* fixed server role. If using integrated security, this will be the domain account of the user executing the preceding command. Alternatively, modify the connection string in the preceding command to specify the username and password for a SQL login with the appropriate membership.

- 2 DB Upgrade will populate default settings in **dbo.ArchiveSetting** table. However, *ArchiveDirectory* is empty and it must be updated with the correct value. *ArchiveDirectory* is the path to the folder where the archive files will be stored.

IMPORTANT

- Do not add a backslash at the end of the path.

- Two folders will be created based on the ArchiveDirectory setting, <ArchiveDirectory>-Purgeable-Archive and <ArchiveDirectory>-NonPurgeable-Archive

2.4 Configuration

The following two elements must be configured before running the Archiving:

- 3 ArchiveSettings.psd1 (refer to [Appendix A](#)).
- 4 **dbo.ArchiveSetting** table in HIPS-Core database (refer to [Appendix B](#)).

3 Data Archiving

3.1 On Demand Execution

Users can execute the scripts On Demand by following the steps listed below:

- 5 Open a Windows PowerShell console. (Standard mode is OK)
- 6 Execute the following command to change directory to the location where the Data Archiving scripts are located:

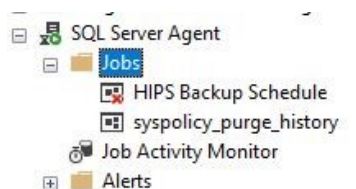
```
cd <DATAARCHIVINGSCRIPT_PATH>
```

- 7 Execute the following command to execute archiving process:

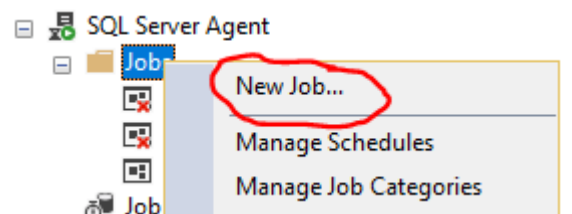
```
.\ExecuteArchivingProcess.ps1 -ConfigurationDataFile  
ArchiveSettings.psdl -Verbose
```


3.2 Setup SQL Jobs

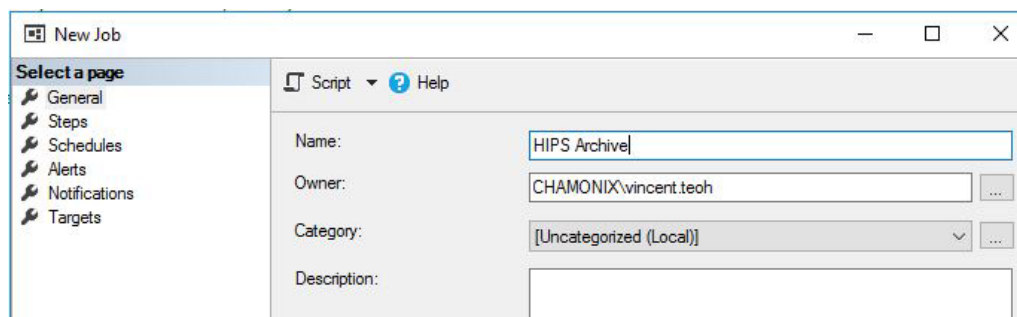
1. Open SQL Server Management Studio and connect to the SQL Server instance that hosts the HIPS-Core database.
2. Expand Security, right-click on Credentials and select New Credential. Provide a name to the credential. Enter the identity and password of the HIPS Archiving Service Account. Click OK.
3. Look for SQL Server Agent and expand it.



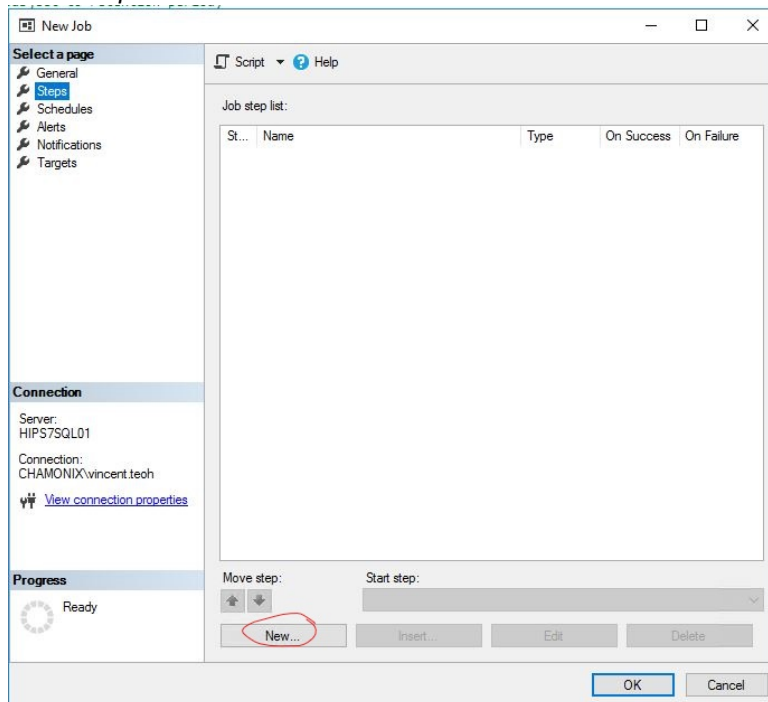
4. Expand Proxies, right click on PowerShell and select New Proxy. Provide a name to the proxy and the name of the credential for the HIPS Archiving Service Account. Click OK.
5. Right click on *Jobs* folder and select **New Job**.



6. Provide a name to the job.



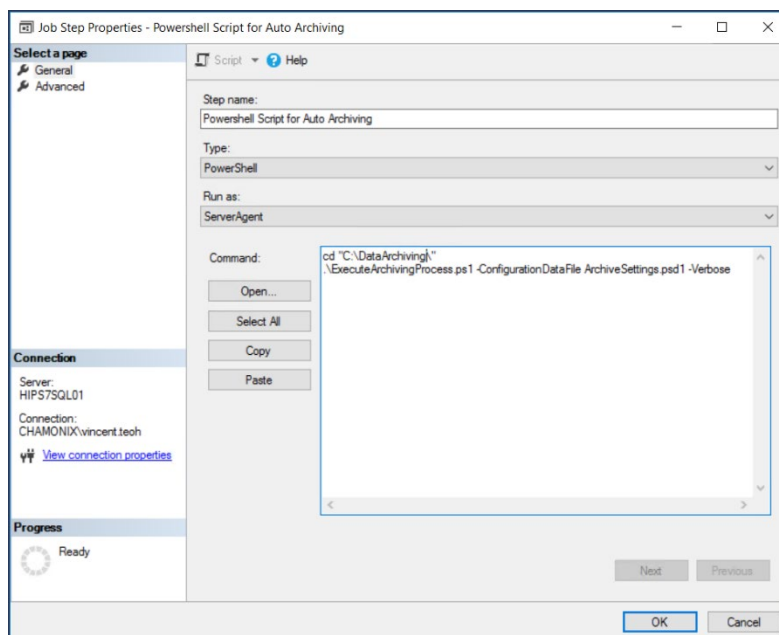
7. Go to *Steps* and select **New**.



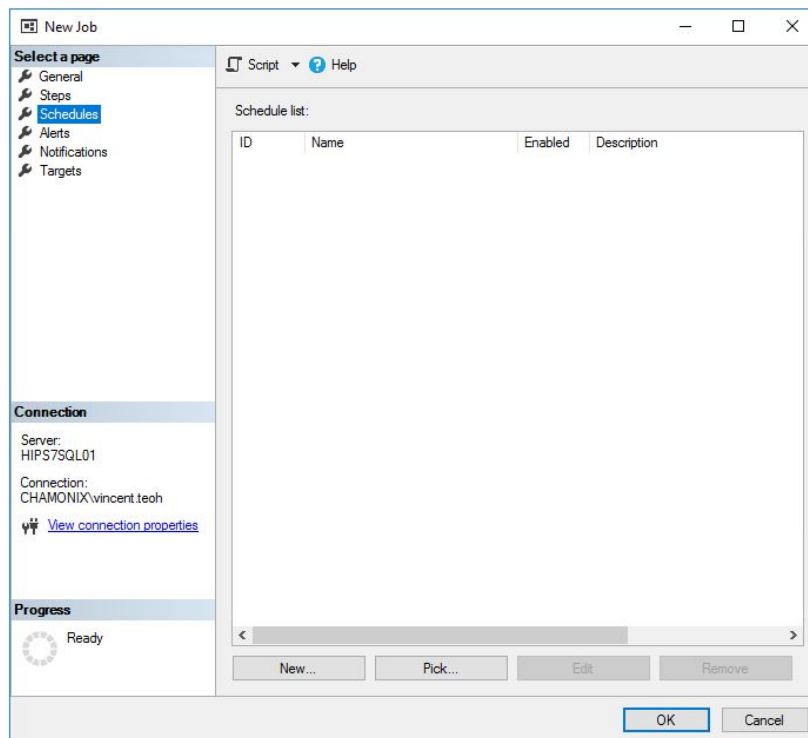
8. Provide a *Step name*, select “PowerShell” as *Type* and input

```
cd "<DATAARCHIVINGSCRIPT_PATH>\\"
.\ExecuteArchivingProcess.ps1 -ConfigurationDataFile
ArchiveSettings.psd1 -Verbose
```

in the *Command* textbox. In the “Run as” drop-down, select the proxy that was created to refer to the credential for the HIPS Archiving Service Account then select **OK**.



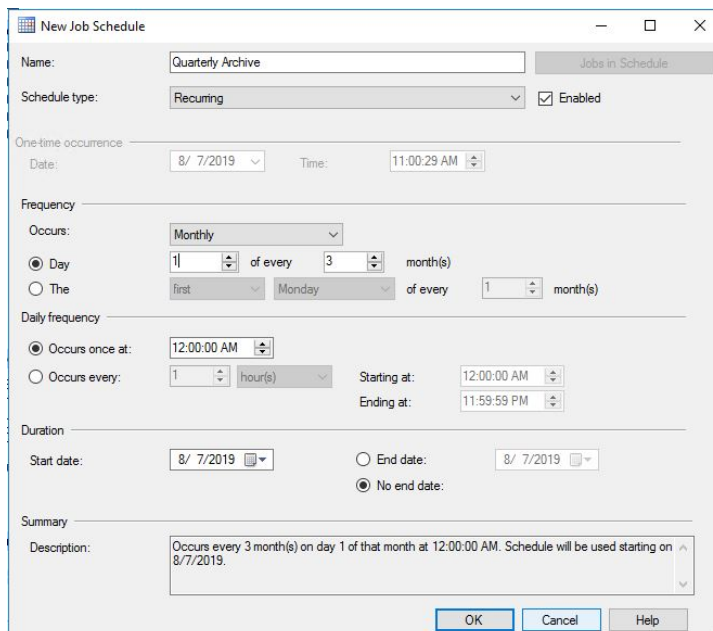
9. Go to *Schedules* and select **New**.



10. Provide a *Name* and configure the *Frequency* as below:

- Occurs: **Monthly**
- Select Day: **1 of every 3 month(s)**

Under *Daily frequency*, select **Occurs once at** and set the *preferred time* then select **OK**.



11. At the “*New Job*” window, select **OK** to close the window and the job is created ready to run.

Note: The execution policies in place in your environment may prevent the archive scripts from running as a SQL Server Agent task, and result in error messages that include text such as "AuthorizationManager check failed" or "ExecuteArchivingProcess.ps1 is not digitally signed. You cannot run this script on the current system." To enable execution of the scripts, start a Windows PowerShell as Administrator, navigate to the folder containing the scripts and execute the Unblock-File cmdlet for each script:

```
Unblock-File .\ExecuteArchivingProcess.ps1
```

```
Unblock-File .\Common.ps1
```

3.3 Restore Data

1. Archived data can be restored by following the steps listed below:
2. For restoring purgeable tables, go to <DATAARCHIVING_PATH>-Purgeable-Archive\yyyy-MM-dd-HH-mm. Or go to <DATAARCHIVING_PATH>-NonPurgeable-Archive\yyyy-MM-dd-HH-mm for restoring non-purgeable tables.
3. Select the table you want to restore and extract the zip file (yyyy-MM-dd-HH-mm-TableName.zip).
4. Open a Windows PowerShell console. (Administrator mode if required)
5. Execute the following command to restore the flat file data into table:

```
BCP <TABLE NAME> in "<BCP FILE PATH>" -f <FORMAT FILE PATH> -S <DATABASE SERVER> -d  
<DATABASE NAME> -T -E -q
```

Example

```
BCP hips.HL7MessageLog in "C:\TestData\2019-08-12-16-43- HL7MessageLog.bcp" -f  
C:\TestData\2019-08-12-16-43-HL7MessageLog.fmt -S . -d <HIPS_CORE_DB> -T -E -q
```

REMARK

-E option is to specify that identity value or values in the imported data file are to be used for the identity column. This will restore data with its original ID value. If -E is not given, the identity values for this column in the data file being imported are ignored, and SQL Server automatically assigns unique values based on the seed and increment values specified during table creation.

Appendix A ArchiveSettings.psd1

ArchiveSettings.psd1 contains the database configurations to connect to HIPS Core database. This is in the same folder where the other PowerShell scripts are.

Name	Suggested Value	Setting Description
SourceSqlServer	.	This is the database server name or IP address. The suggested value "." refers to the default SQL Server instance on the local host. If the HIPS-Core database is hosted on a named instance, include the instance name after a backslash: ".\instancename"
SourceDatabase	<HIPS Core database name>	This is the HIPS-Core database name. Do not include the angle brackets.
CommandTimeout	3600	This is the maximum number of seconds the command is given to execute the SQL command. The value can be changed to suit the conditions in each environment.

IMPORTANT

The archiving scripts are relying on this setting to connect to the correct HIPS Core database and retrieve other settings required to execute the archive process.

Appendix B Archive Settings in Database

The dbo.ArchiveSetting table in the HIPS Core database contains the settings required when executing the archive process. DB Upgrade will populate default settings in dbo.ArchiveSetting table. However, the ArchiveDirectory setting is empty and it must be updated with the correct value before the archive process is run.

Code	Suggested Value	Setting Description
ArchiveDirectory	C:\HIPS\DbArchive	Folder path prefix for archive output. <i>Backslash must not be included at the end of the path.</i> Two folders will be created using this path by appending -Purgeable-Archive and -Non-Purgeable-Archive onto the supplied value. The suggested value will result in the following two folders being created: C:\HIPS\DbArchive-Purgeable-Archive C:\HIPS\DbArchive-NonPurgeable-Archive

MaxDataSizeKB	1000000	<p>Maximum data size in kilobytes to be extracted from each table each time the archiving job runs. The number of rows to extract is calculated using this value as a proportion of the size of the data in the database table, so the configured value will only approximate the actual size of the exported file. The default is 1 GB.</p> <p>For a HIPS-Core database with Recovery Model set to Full, this setting must be managed with respect to the amount of space available for the database log, because each row that is archived takes up space in the log until the next full backup.</p> <p><u>IMPORTANT</u></p> <p>The archive process will attempt to calculate the amount of data to archive for each table using the average size of all rows in the table to determine the number of rows to archive. This is not always accurate due to varying sizes of rows and can result in archive files being significantly less or, at times, more than double the requested size. This can be most noticeable for the ClinicalDocumentVersion, HL7MessageLog, MessageQueueContent and SystemInteractionLog tables, and is worthy of note when calculating target storage requirements.</p>
---------------	---------	--

PurgeableTables	<p>SystemInteractionLog,SystemErrorLog,SystemEventLog,HL7MessageLog,ClinicalDocumentVersion,PcehrAudit,IhiLookupAlert,MessageQueue</p> <p>List of purgeable tables separated by commas. Sites may purge the archived files from these tables when the information is no longer required by the site.</p> <p>IMPORTANT</p> <p>MessageQueue</p> <ul style="list-style-type: none"> The default setting includes the entry MessageQueueContent. This entry instructs the script to perform archiving for both the MessageQueue and MessageQueueContent tables, with special logic to only archive items with status 'Success' and 'Failed'. This ensures that items with status 'Pending' are not archived. Do not add a separate entry for MessageQueue. If a separate entry MessageQueue is added, the script will archive data in the MessageQueue table without the status filter and so 'Pending' items will also be archived. <p>HL7MessageLog</p> <ul style="list-style-type: none"> The default setting includes the entry HL7MessageLog. This entry instructs the script to perform archiving for the HL7MessageLog table, with special logic to only archive items with status 'Success' and 'Failed'. This ensures that items with status 'Pending' are not archived. <p>ClinicalDocumentVersion</p> <ul style="list-style-type: none"> The default setting includes the entry ClinicalDocumentVersion. This entry instructs the script to perform archiving for the ClinicalDocumentVersion table, with special logic to only archive items where Package is not null. Archiving for this table does not delete the row, but sets Package to null. The special logic ensures that items previously archived are not archived again on the next run.
-----------------	--

NonPurgeableTables	IhiLookupAudit,HpiiLookupAudit	<p>List of non-purgeable tables separated by commas. Prior to HIPS 7.2.2, sites were required to ensure they retain the archived files from these tables to comply with the relevant Healthcare Identifiers legislation, rules, regulations and conformance requirements in <i>Use of Healthcare Identifiers in Health Software Systems</i>.</p> <p>The completion of ADO 10796 – Add user’s name to ID field in HI Service requests for auditing in HIPS 7.2.2 means sufficient information will be provided in a request to identify a user which means the site is not required to maintain audit data for 7 years after employee ceases to be authorised. Due to the difficulty of identifying when each employee ceases to be authorised, sites have been erring on the side of caution and requiring retention of 60 years for all HI audit data (this allows for an employee who is hired and performs their first IHI lookup when 18 years old and retires at the age of 78).</p> <p>For archived data files that only contain records from the tables IhiLookupAudit and HpiiLookupAudit which were created after and upgrade to HIPS 7.2.2 or higher, the archived data in the Non-Purgeable folder can be treated as for archived data in the Purgeable folder, that is, the retention period for the archived files is determined by the site's local policy rather than the Healthcare Identifiers legislation.</p> <p>The average size of an IhiLookupAudit record is 346 bytes. For a large HIPS site that writes 200,000 audit records into IhiLookupAudit per day, the expected savings is approximately 25 GB retained data per year, which is 1,500 GB over 60 years.</p> <p>The IhiCreationAudit table will be added by default if the HIPS version is 7.2 or higher. When a site is upgraded from HIPS 7.1 to 7.2 or higher, this value should be updated manually to <i>“IhiLookupAudit,HpiiLookupAudit,IhiCreationAudit”</i> in order to start archiving from the IhiCreationAudit table.</p>
RetentionPeriod_SystemInteractionLog	3	Retention period for SystemInteractionLog (in months)
RetentionPeriod_SystemErrorLog	3	Retention period for SystemErrorLog (in months)

RetentionPeriod_SystemEventLog	3	Retention period for SystemEventLog (in months)
RetentionPeriod_HL7MessageLog	3	Retention period for HL7MessageLog (in months)
RetentionPeriod_ClinicalDocumentVersion	3	Retention period for ClinicalDocumentVersion (in months)
RetentionPeriod_PcehrAudit	3	Retention period for PcehrAudit (in months)
RetentionPeriod_IhiLookupAlert	3	Retention period for IhiLookupAlert (in months)
RetentionPeriod_MessageQueueContent	3	Retention period for MessageQueue (in months)
RetentionPeriod_IhiLookupAudit	3	Retention period for IhiLookupAudit (in months)
RetentionPeriod_HpiiLookupAudit	3	Retention period for HpiiLookupAudit (in months)
RetentionPeriod_IhiCreationAudit	3	Retention period for IhiCreationAudit (in months)

Appendix C Archiving Steps and Conditions

The HIPS data archiving process performs the following steps for each table configured in the PurgeableTables and NonPurgeableTables settings in the ArchiveSetting table.

Step 1 - Count

Calculate the maximum number of rows to extract by dividing the configured data size (MaxDataSizeKB) by the average size of rows in the specified table.

Retrieve the maximum *ID* to extract from the table, from rows where the Date Created is before the retention period configured for the table, ordered by Date Created ascending, and limited to the above calculated maximum number of rows to extract.

- For the ClinicalDocumentVersion table, exclude rows whose Package is NULL, since they have already been archived.
- For the MessageQueue and MessageQueueContent tables, calculate the maximum MessageQueueID using the average size of rows for the MessageQueueContent table because this table is vastly larger than MessageQueue.

Step 2 - Extract

Filter the records in the table where the ID is less than or equal to the maximum ID calculated above, and:

- For the HL7MessageLog table, extract rows where the row has completed processing, i.e. the QueueStatusID is 2 (Successful) or 3 (Failed).
- For the MessageQueue and MessageQueueContent tables, extract rows where the MessageQueue row has completed processing, i.e. the MessageQueueStateID is 2 (Successful) or 3 (Failed).
- For the ClinicalDocumentVersion table, extract rows where the Package is not NULL.
- For other tables, extract all rows where the ID is less than or equal to the maximum ID calculated above.

Step 3 - Archive

Archive the matching records in the table according to the following rules:

- For the HL7MessageLog table, delete rows:
 - Where the ID is less than or equal to the maximum ID calculated above, and
 - That have completed processing, i.e. QueueStatusId is 2 (Successful) or 3 (Failed), and
 - That have not been modified after the process started.
- For the MessageQueue and MessageQueueContent tables, delete rows:
 - Where the ID is less than or equal to the maximum ID calculated above, and
 - That have completed processing, i.e. MessageQueueStateID is 2 (Successful) or 3 (Failed), and
 - That have not been modified after the process started.
- For the ClinicalDocumentVersion table, update Package to NULL for rows where the ID is less than or equal to the maximum ID calculated above.
- For other tables, delete all rows where the ID is less than or equal to the maximum ID calculated above.