Content Archiver

Purpose

The Archiver can be installed to service a primary upload directory or an individual workstation. In either case, it will take selected directories or files and load them into the Repository, index and OCR as specified within the Archiver setup.

NOTE: The Archiver can and will upload any file into the repository. We have experienced in the past clients using the Content Repository as a data warehouse. Please be advised, ECM will do this in that entire systems, operating systems, applications, or application development environments have been loaded into the repository. ECM can and will store such items as a secondary function but ECM is designed to manage content, files, graphics and when used to manage a warehouse, the repository may grow too much greater than expected sizes. ECM will also load and index several type of zipped content. The zipped files and its content are indexed so if a document exists within the zip file that is being sought, it will say that the zip file contains a document meeting you search criteria. ECM allows zipped files to be expanded as well within the repository. This allows a specific document to be found within a specific zip file. The drawback, the repository will grow accordingly and in large amounts.

OCR

Optical character recognition is performed as a scheduled task and runs separately of the Archiver. The OCR Engine is installed, generally, on the same server as the repository but that is not a requirement. Clients, for performance reasons, elect to run the OCR engine on its own machine at times. When the Archiver loads a file that has been defined to contain graphics or is a graphic, it is flagged as requiring OCR. When the OCR engine runs, it executes OCR against the file and marks the file as OCR'd.

Auto-Scheduling OCR

IN order to run the OCR Engine unattended a simple Scheduler Task can be setup on the server or the desk top. This allows the OCR to be executed on a schedule. Most clients elect to use a 15 or 30 minute execution schedule.

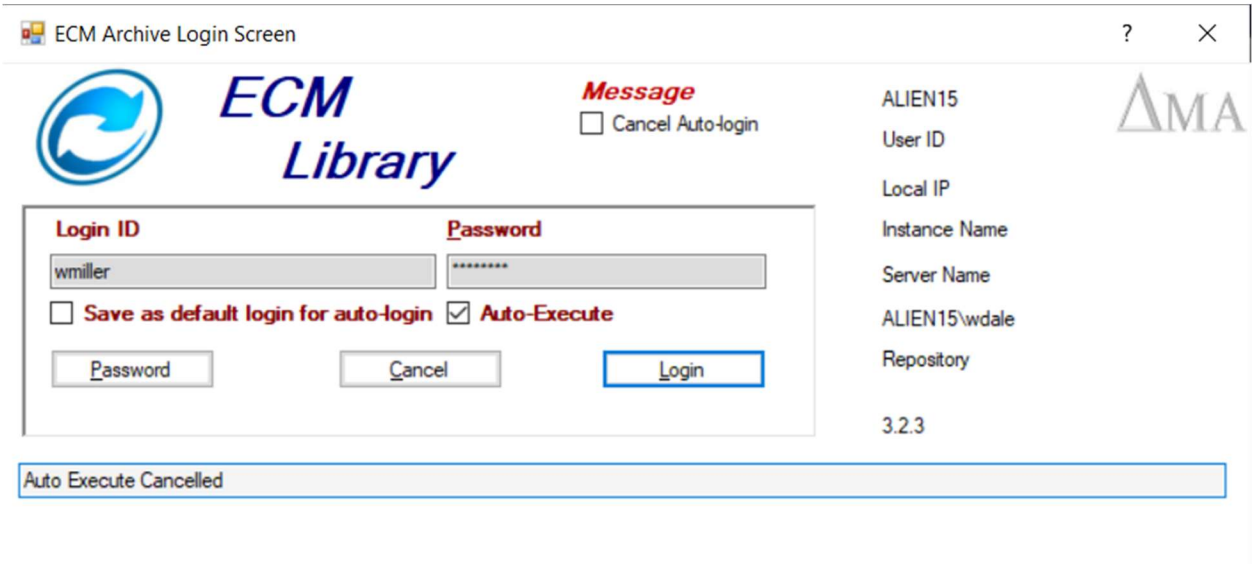
Task Scheduler for Developers

Directory Listener

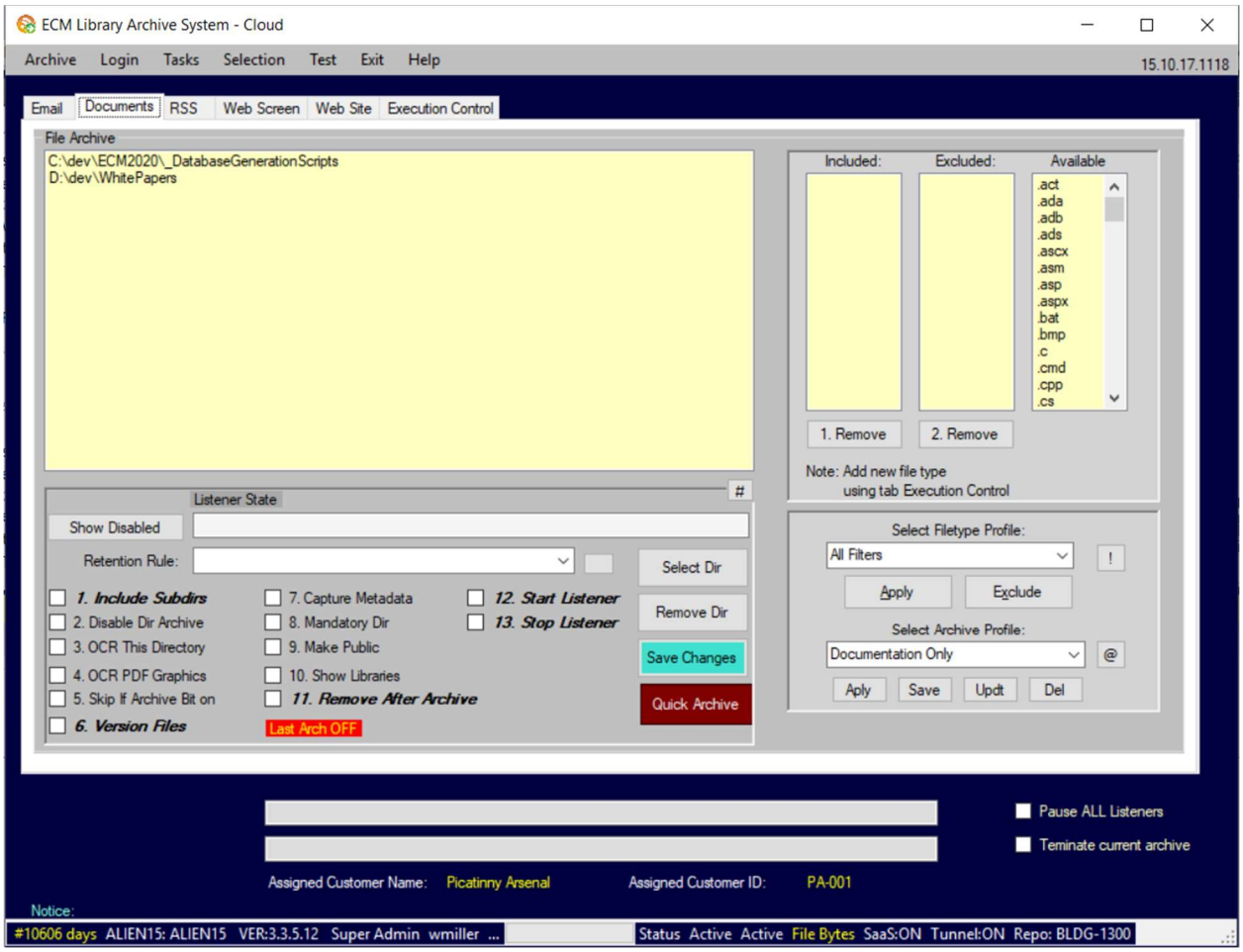
One of the abilities of the Archiver is to interface with the ECM Directory Listener. The Listener is designed so that as files change within a targeted directory and if specified its subdirectories, the newly added or modified file is registered within the Archiver to be

archived or updated in the repository. Some companies have hundreds of thousands and in a few cases, missions of files contained within primary or shared drives. The time to scan and determine if a file needs to be archived can be substantial, in the order of hours. The listener eliminates this requirement and focuses only on the files that have changed or have been added. The Listener runs as a Service and is therefore running 7 X 24 and starts automatically when the hosting machine starts.

The Archiver



The above is the Archiver login screen. The Archiver can be set to auto-start so that it can be executed using the task scheduler (check the auto-execute) or as above, used to open the Archiver for setup, modification, or to run contained utility programs.



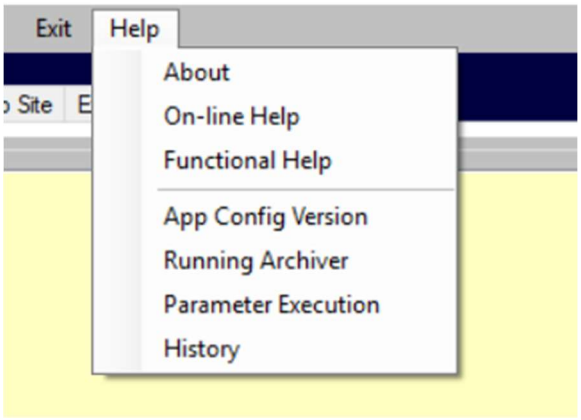
Main Archiver Screen

The above screen is the primary archiver screen. Showing is the “Documents” tab. This is where documents, graphics, and primary content and target load directories are defined.

For full details on setup and use, see:

Archiver Online Help

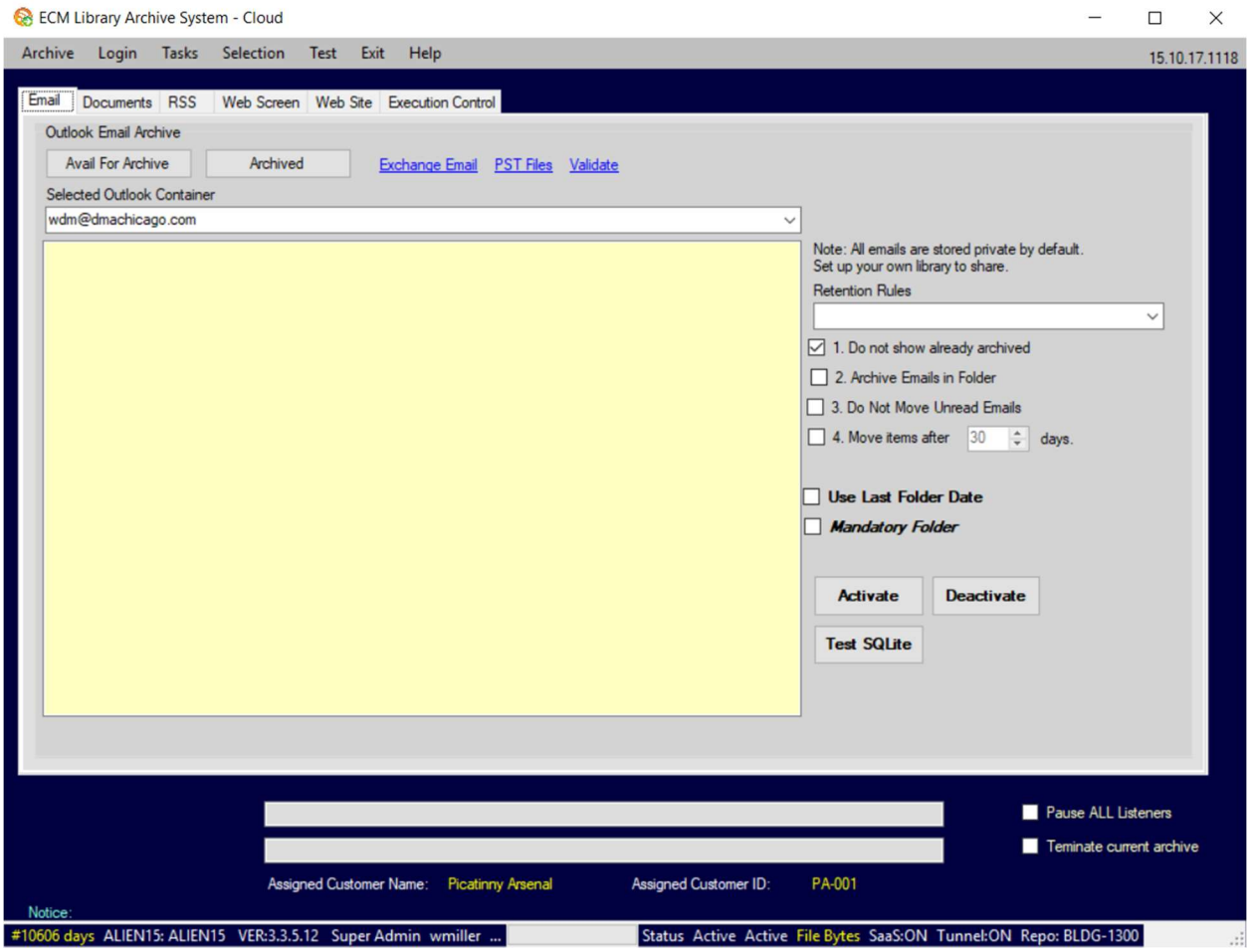
(See below)



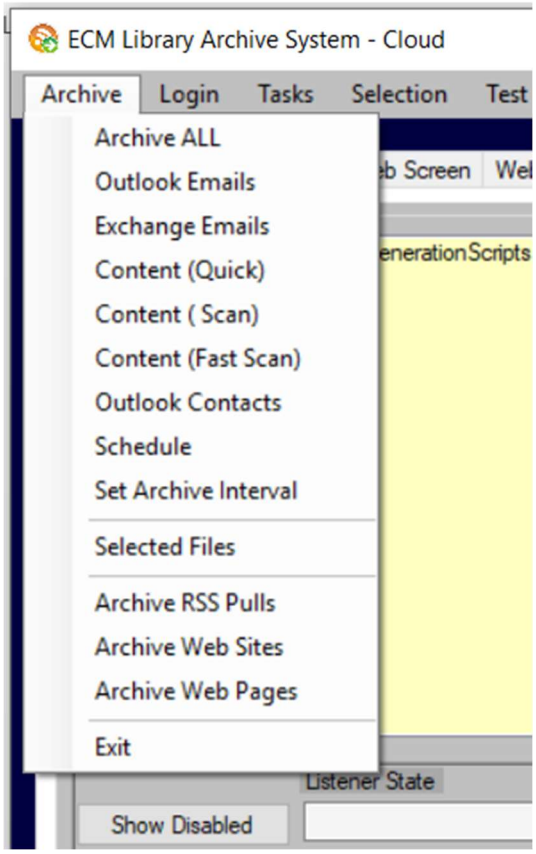
It is important to point out that User and Setup help is different than Functional Help. The latter provides the technical description of the calls, subroutines, functions and parameters used internally to the Archiver. This documentation is ***Classified and Confidential***. It is not to be shared outside of the Archiver Technical Support team.

Email Archiver

The below, the Email Archive tab, is used to setup email archives. The email archiver operates with Outlook, Exchange, and can be setup to access and archive IMAP or POP3 email servers.



Archive Drop down Menu



Archive All:

Runs the archiver against all of the setup archivers, e.g. documents, emails, rss feeds, web sites, ets.

Outlook Emails:

Since Outlook is the dominant email in Windows environments, this will archive the specified Outlook folders immediately.

Exchange Emails

Exchange Servers can be setup as an archive source for emails and this option executes the same immediately.

Content (Quick)

A quick scan uses listener files and is significantly faster than a content scan and somewhat faster than a content fast scan. However, this does not take a full re-inventory of the target directories, it uses the SQLite tables and determines if a file has changed or been inserted accordingly.

Content (Scan)

This is a slow and full re-inventory of the target directories. This archive should be executed once per week and can be accomplished using passed in parameters (documented later) and a scheduled task.

Content (Fast Scan)

Use this archive to perform an inventory of the targeted directories against the SQLite inventory and archive accordingly. The Fast Scan is considerably faster than the Content Scan and slower than the Content Quick archive.

Outlook Contacts

Click this to backup all Outlook contacts into the repository.

Schedule

Use the task scheduler fro much better results than this, the built-in scheduler. This item will be deprecated in 2021.

Set Archive Interval

Use the task scheduler fro much better results than this, the built-in scheduler. This item will be deprecated in 2021.

Selected Files

Show the list of selected files that will be inventoried and assessed for archive.

Archive RSS Pulls

Execute the archive of all defined RSS interfaces.

Archive Web Sites

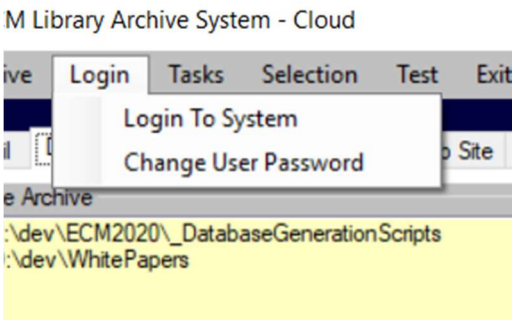
Execute the archive of all defined Web Sites.

Archive Web Pages

Execute the archive of all defined Web Pages.

Exit

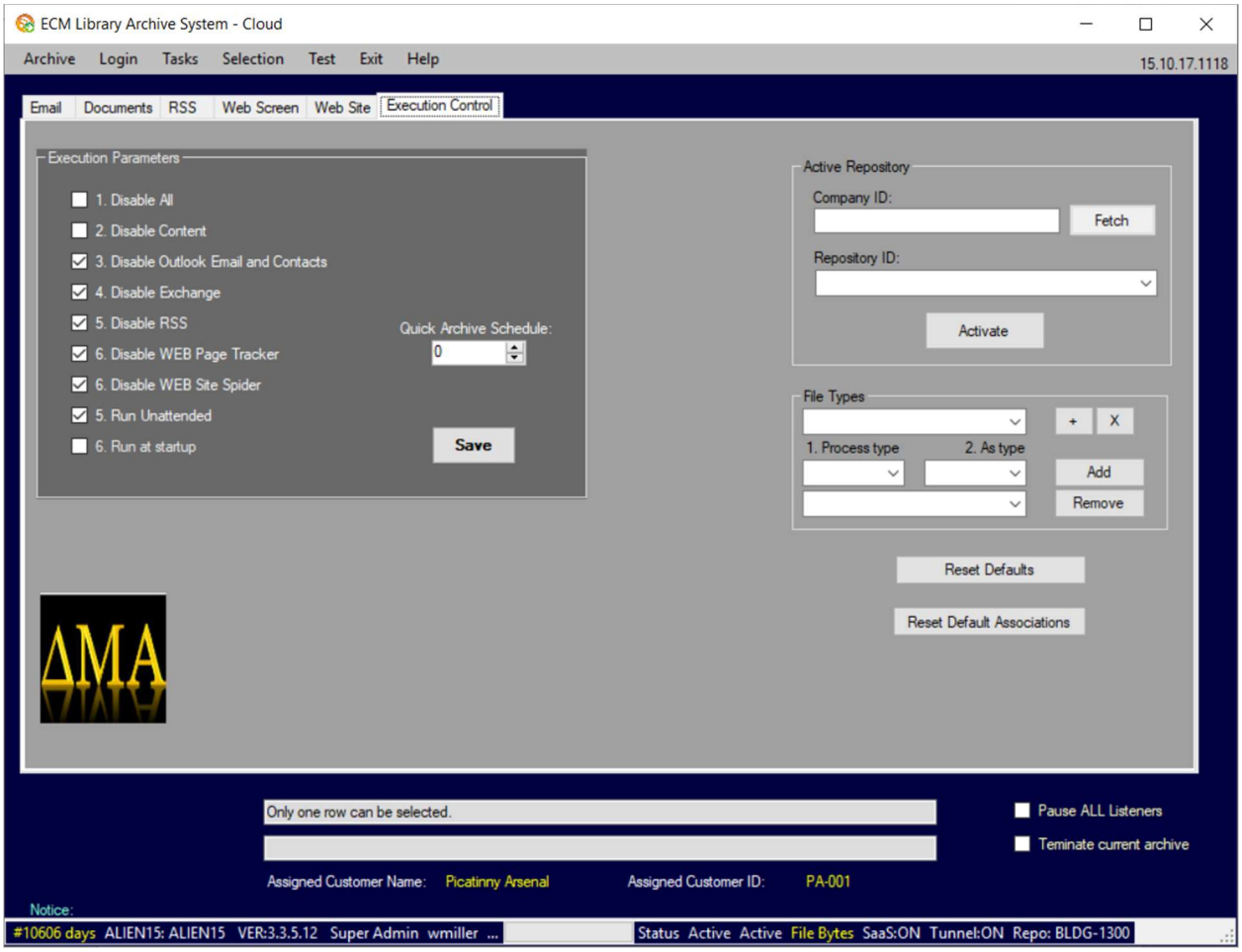
Close the Archiver.



Login Drop down Menu

- Login to the Archiver as a different user
- Change your password

Execution Control

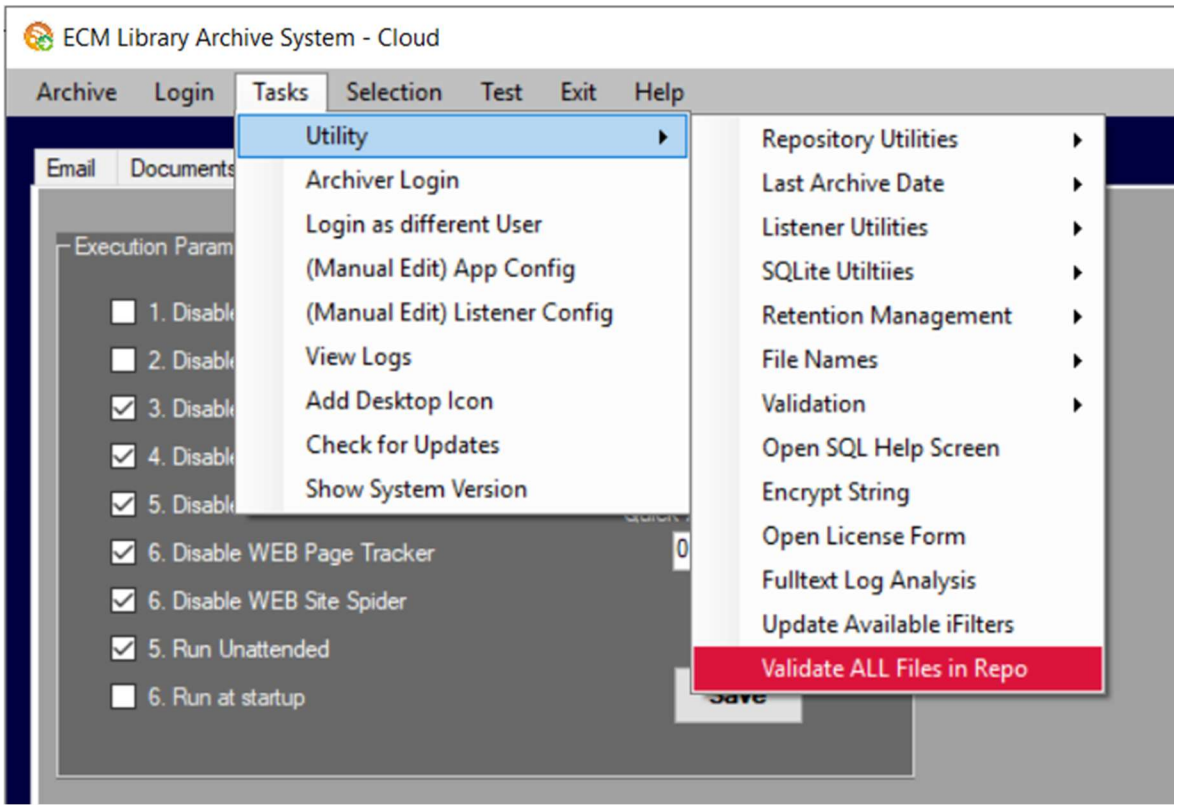


The Execution Parameters (upper left) are used to enable/disable certain archives and to establish Run Unattended and Run at startup.

Active Repository: used in an environment that is running a HIVE configuration.

File Types: Used to define a file, by extension, which can be processed by ECM but may not have an associated iFilter. For example, a .bat file can be processed as a .txt file and ECM will archive the .bat and use the .txt iFilter to index the file making it fully searchable.

Tasks Dropdown Menu



The utilities showing a small arrow to their right will be explained in detail later in the document.

Open SQL Help Screen

There are commands and processes that can be executed using SSMS and run against the repository as well as commands that execute SQL utilities to perform cleanup and tuning of the repository. Open this screen and it will show you predefined commands and utilities to aid the administrator in this effort.

Encrypt String

Use this to encrypt strings that might be required, for example, in the config file.

Open License Form

This form is used to define and apply the ECM License.

Full text Log Analysis

The logs of the full text indexers concerning failures and successes of indexing specific documents are, on their best day, complex and confusing. This utility will provide a simplified way to examine and search these logs for anomalies and errors.

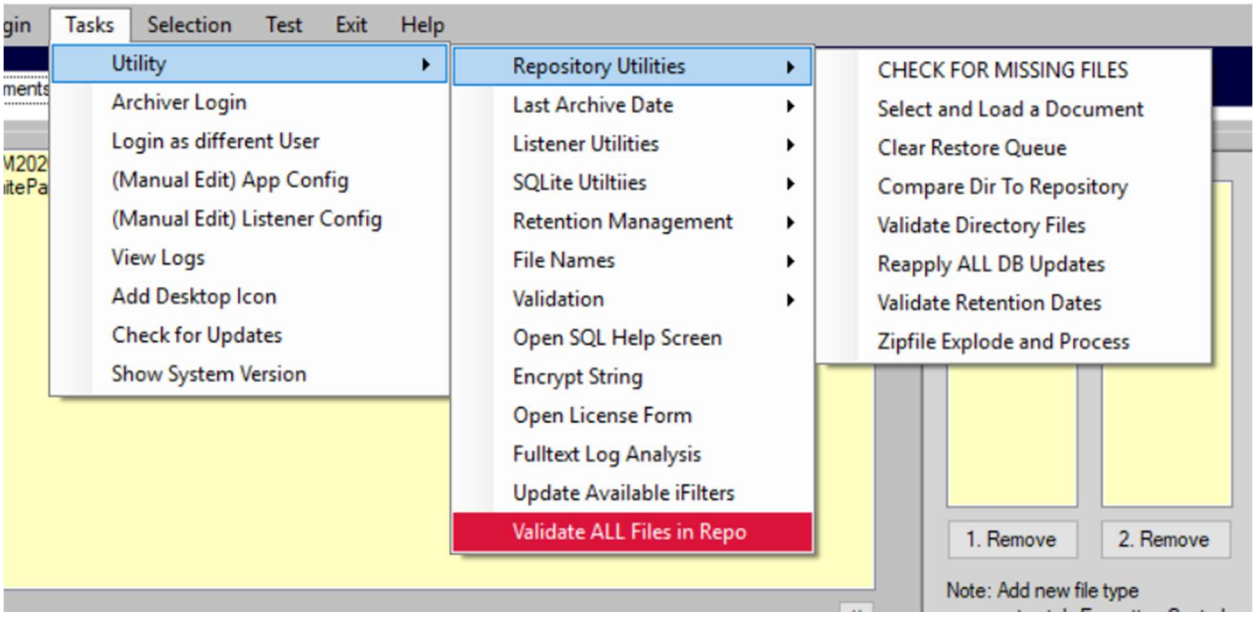
Update Available iFilters

Click this to update existing and available iFilters.

Validate ALL Files in Repo

This is used to scan targeted directories and examine each file contained within to insure it exists within the repository. If a file is found to exist within the repository and not on the specified target drive and directory, it is reported as “stand alone”.

Repository Utilities



Check for missing files:

Select and load a Document:

Clear Restore Queue:

Inventory target directories and flag missing files for archive.

Compare Dir to Repository:

Verify a specific directory to existing repository content.

Validate Directory Files:

Look at all files contained with a specified directory and determine if it needs to be inserted into or updated with the repository.

Reapply all DB updates:

Database updates are supplied with ECM updates as needed. In some cases, these updates need to be reapplied (e.g. a repository is restored from a previous version. This will reapply needed updates.

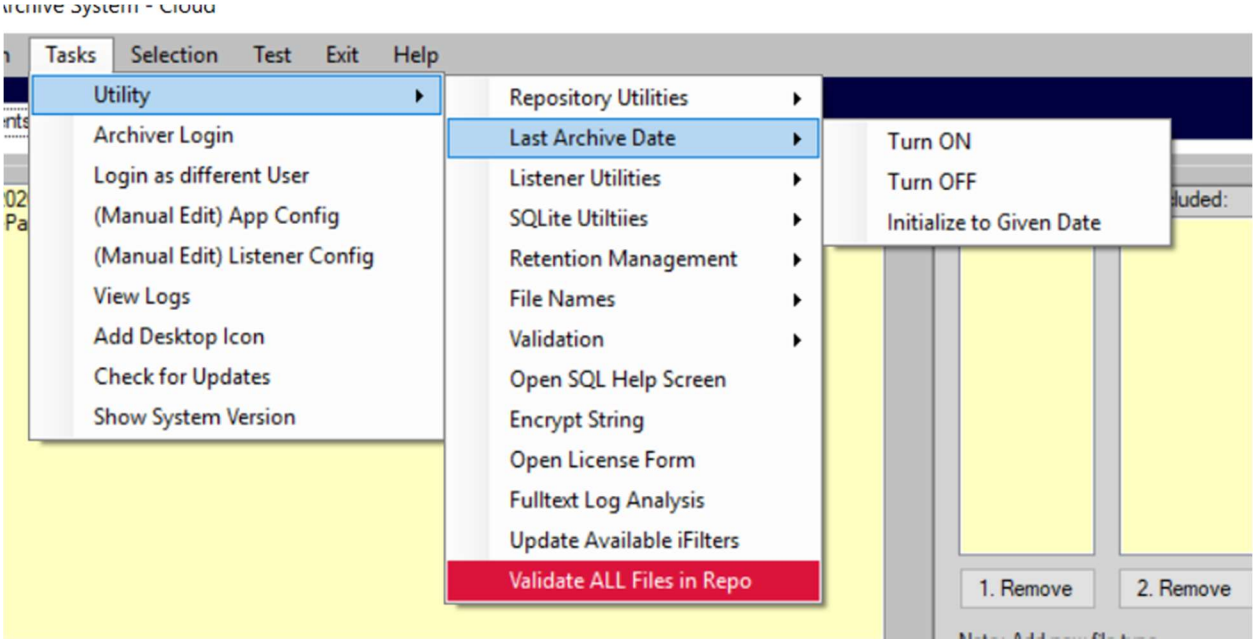
Validate retention dates:

At times, a file may be archived and the retention date was not established. This will find and fix.

Zip file explode and process:

There are times when zip files are loaded into the repository and not exploded in order to save time or space or whatever the reason. If that decision is reversed for whatever reason, rather than having to find and do a delete and reload, this utility will extract, explode and reload zip files that are not so.

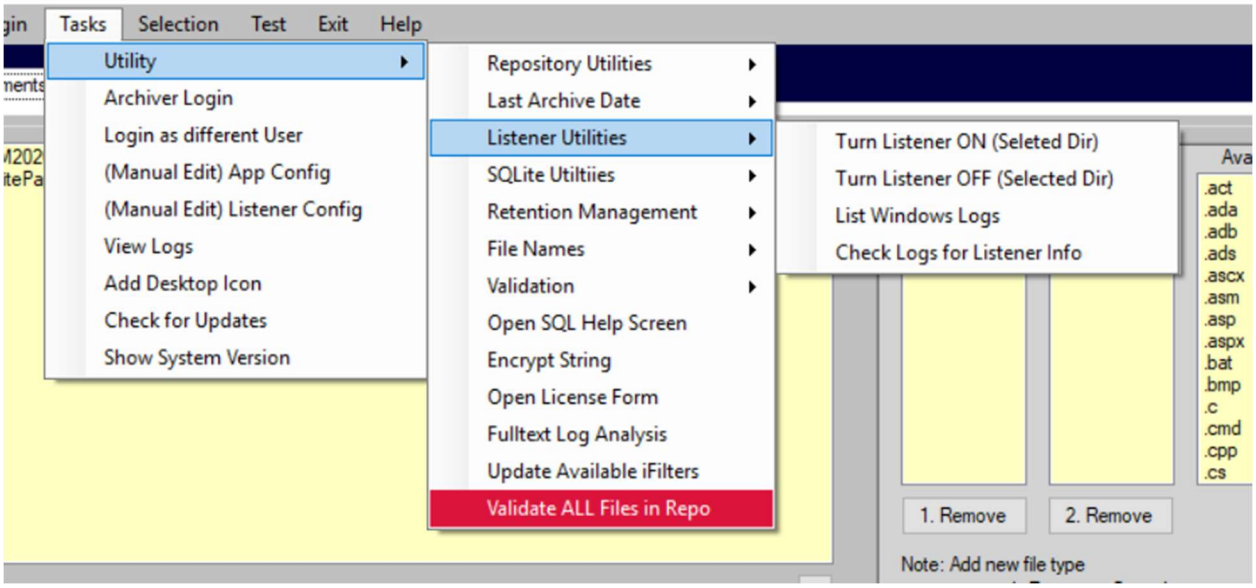
Last Archive Date



When the last archive date is turned on, the archiver will only have to look at a files last modified date and compare it to the last time the archiver ran. If last modified is greater than last archive, then the file is processed. This is extraordinarily fast, but has a drawback – if a file is copied from one directory into a target directory, the last modified date will be retained and may be prior to the last archive date causing a file to be missed.

- Turn ON: Turn on the Last Archive Date function
- Turn OFF: Turn off the Last Archive Date function
- Initialize to Given Date: Set the last archive date to a specific date.

Listener Utilities drop down menu



Turn the Archiver listener on:

As stated, turns the Archiver Listener ON

Turn the Archiver listener off:

As stated, turns the Archiver Listener OFF

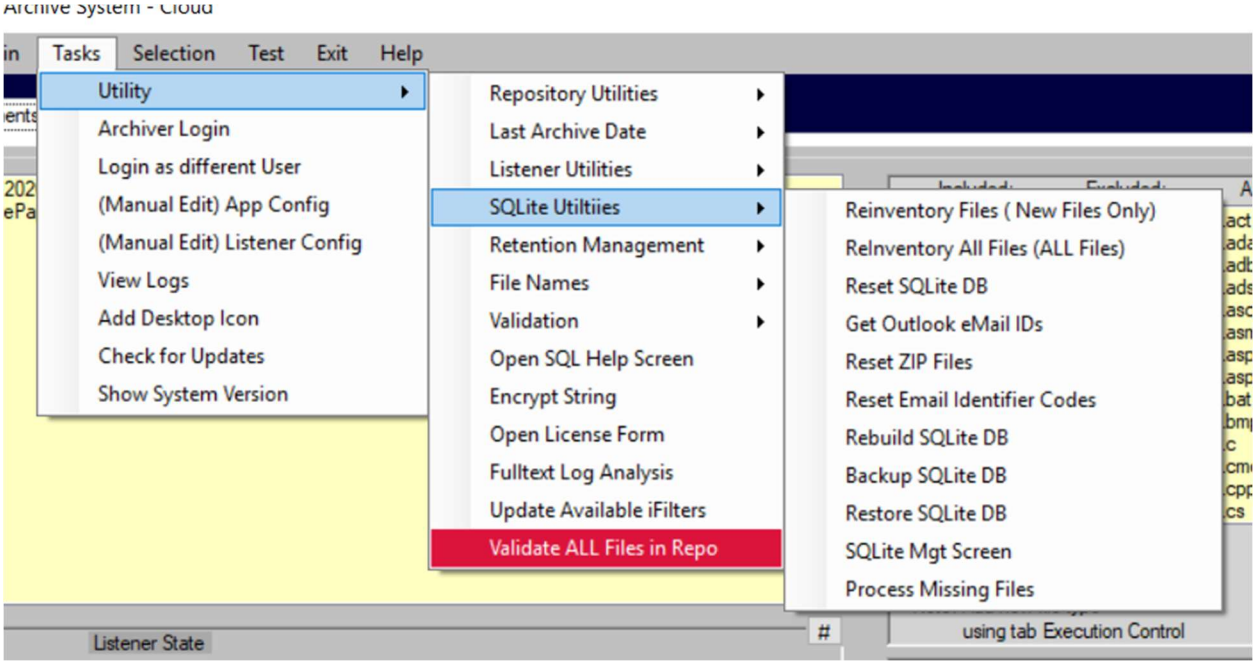
List Windows Logs:

List the location and availability of windows' logs

Check Logs for Listener Info:

Shows how long log will be retained, how big as log can be, nbr of rows in the log, log meta-data in general.

SQLite Utilities drop down



Re-inventory Files (new): Find all new files that are not entered into and managed through the SQLite interface and add them.

Re-inventory ALL: Process all target directories and update the SQLite inventory with the actual and current metadata of existing and new files.

Reset SQLite: Zeroes all data from within the SQLite tables forcing a full re-inventory during the next archive.

Get Outlook email ID: Outlook uses an email id to create and identify a unique email. This processes all target folders and re-inserts all email Ids.

Reset ZIP Files: Flags all zip files as needing to be re-entered into the SQLite tracking system.

Reset Email Identifier: Fetches all available emails and re-enters their ID into the SQLite tracking system.

Rebuild SQLite: Rebuilds the SQLite database.

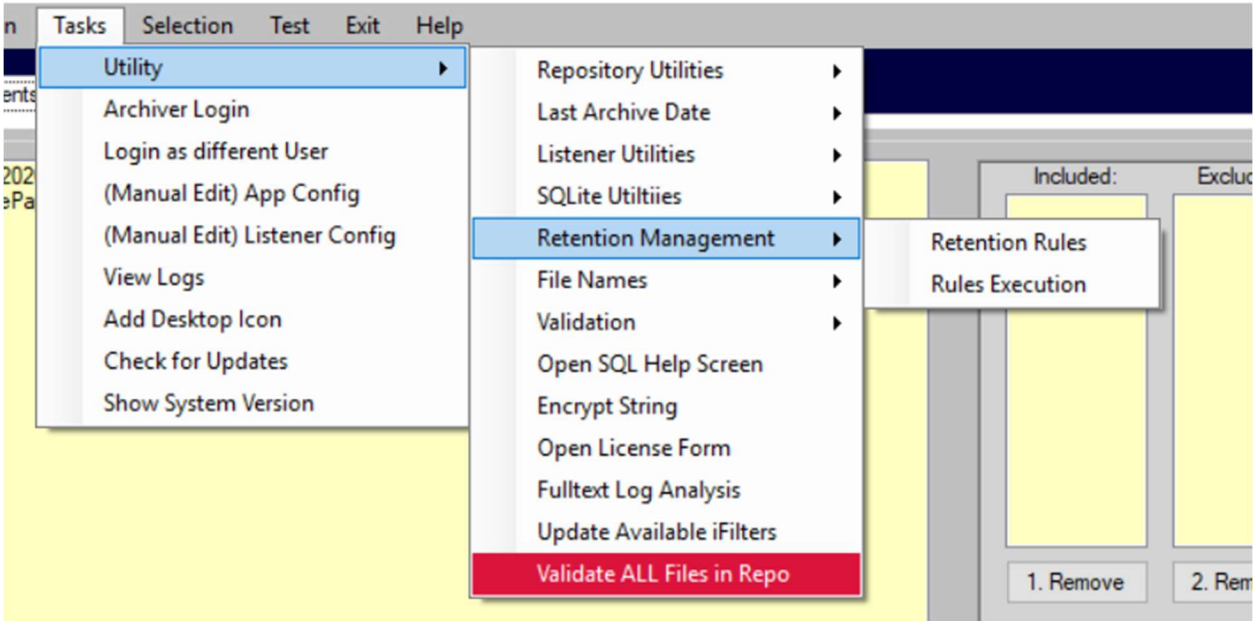
Backup SQLite: Generates a SQLite database backup.

Restore SQLite: Restores from a SQLite database backup.

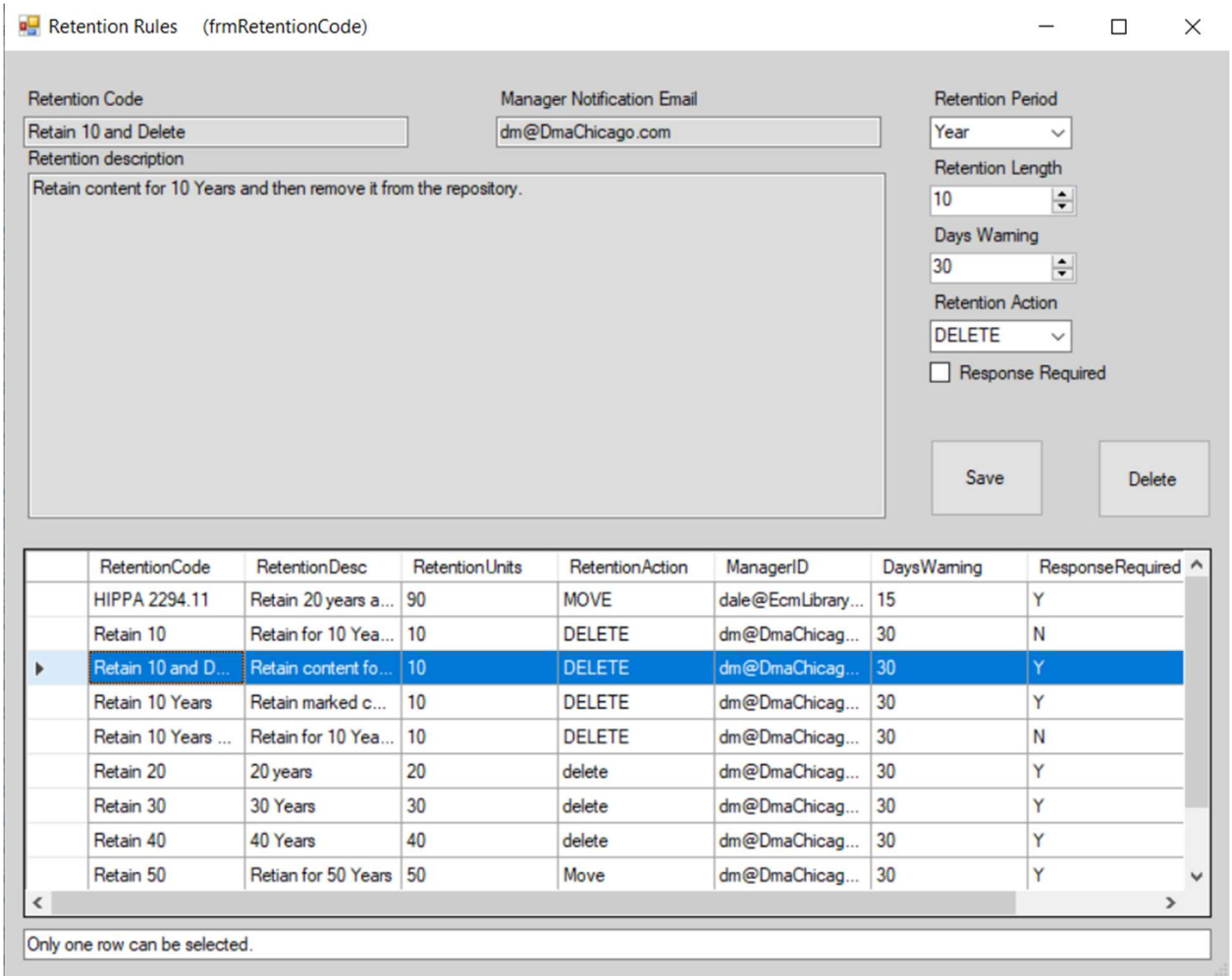
SQLite Mgt Screen: This is a utility screen that allows remedial SQL statements to be executed against the SQLite database and also allow data to be reviewed that is contained within the database.

Process Missing Files: Simply, finds files that are in target directories and not contained within the SQLite tracking system and adds them to SQLite.

Retention Rules drop down

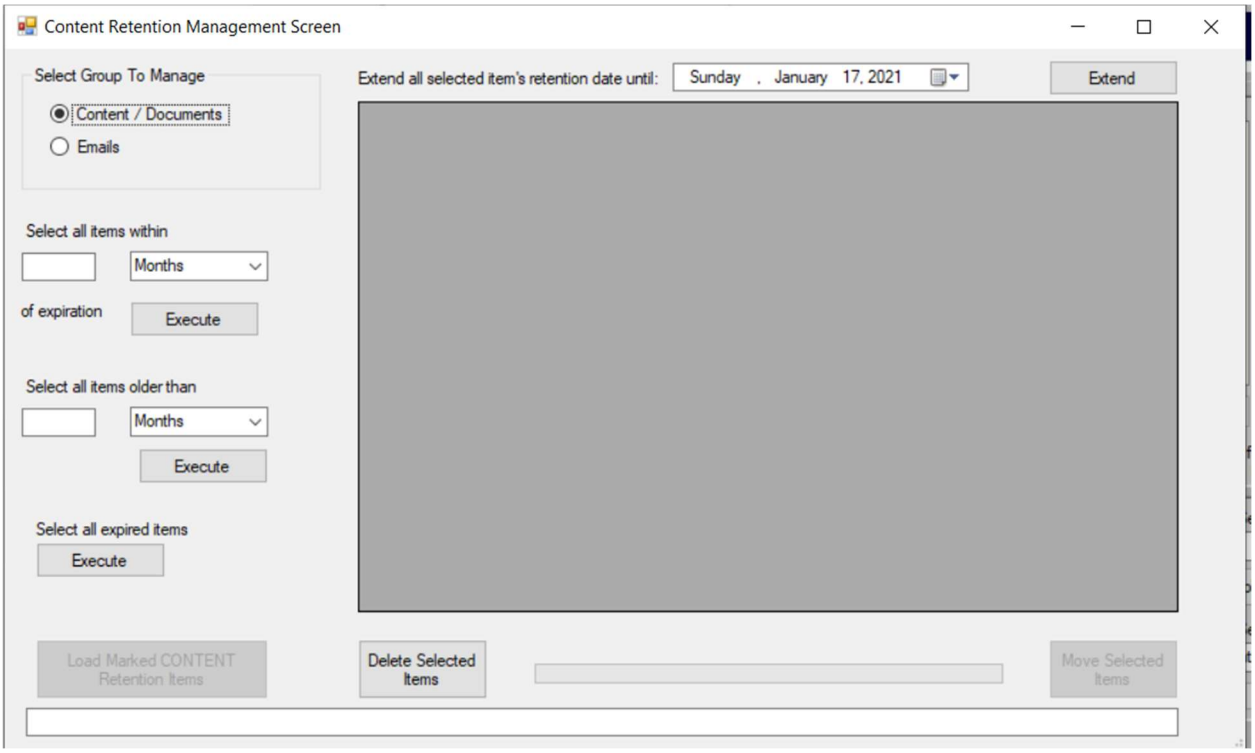


Retention Rules Screen



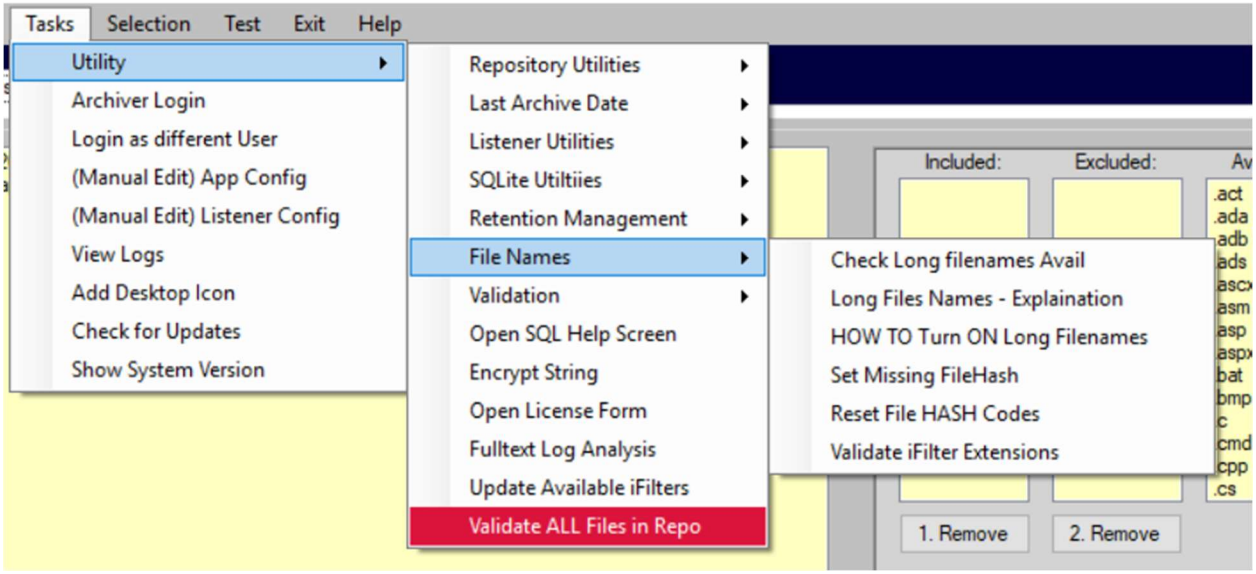
In this screen, repository retention rules can be created, deleted, or modified.

Rules Execution Management Screen



This screen allows bulk management of files' retention rules. A set of files can be selected and the entire set of selected files can have their retention rule(s) reset very quickly.

File Names drop down



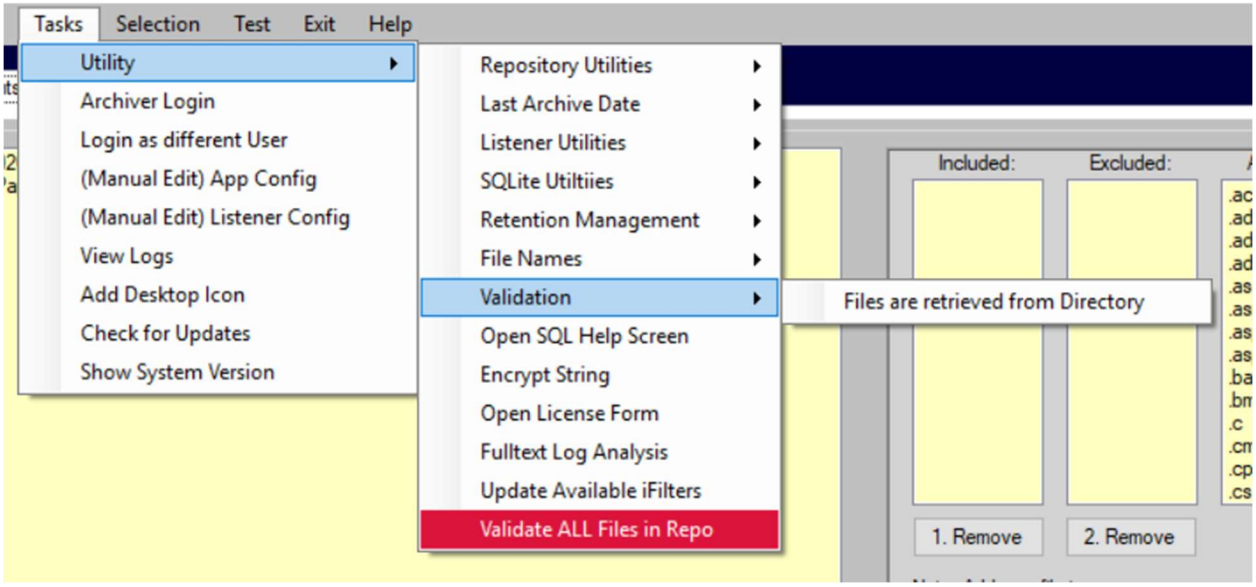
Check long file names available: Long file names, either directory, file, or combined name longer than 260 bytes), create a unique problem in Windows environments. The fix for this is now available in Windows 10 and servers later than 2012. This utility checks to see if the hosting platform is compatible with long file names.

Set missing file hash: Due to the inability of most systems to process long file names, ECM turns file names into a HASH. This hash is then used for lookups. This will set all missing file name hashes.

Reset File Hash: recalculates and applies hash codes to existing files.

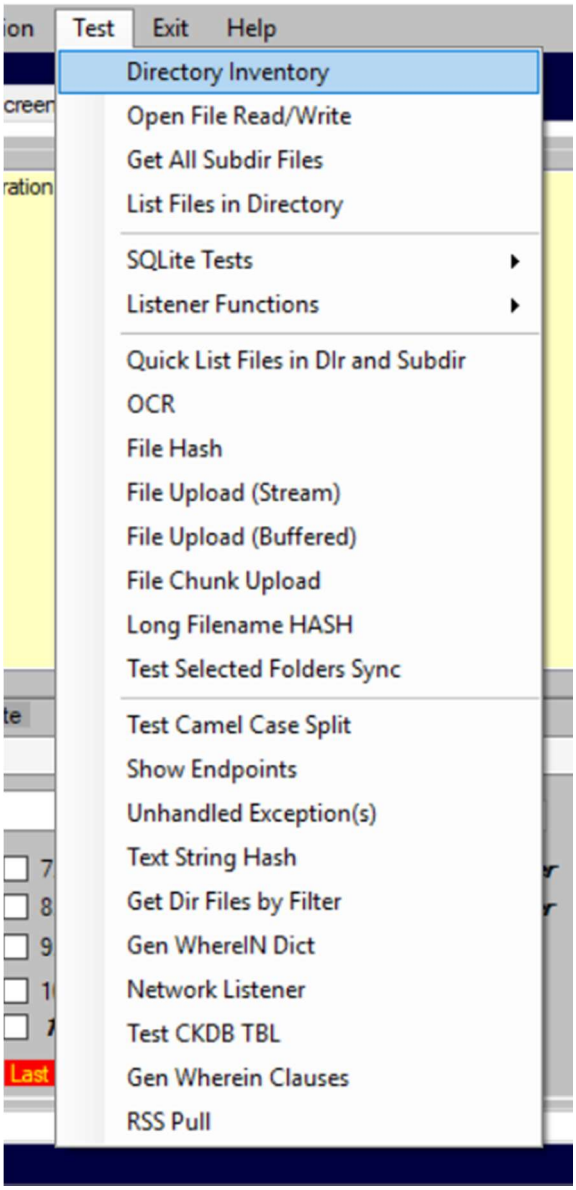
Validate iFilter Extensions: Run this to make certain files have a valid iFilter or an “index as” extension associated with them. If a file is in the repository and is not associated to an iFilter, it will not have its content indexed. However, the creator, directory, and filename will still be searchable.

Validation drop down



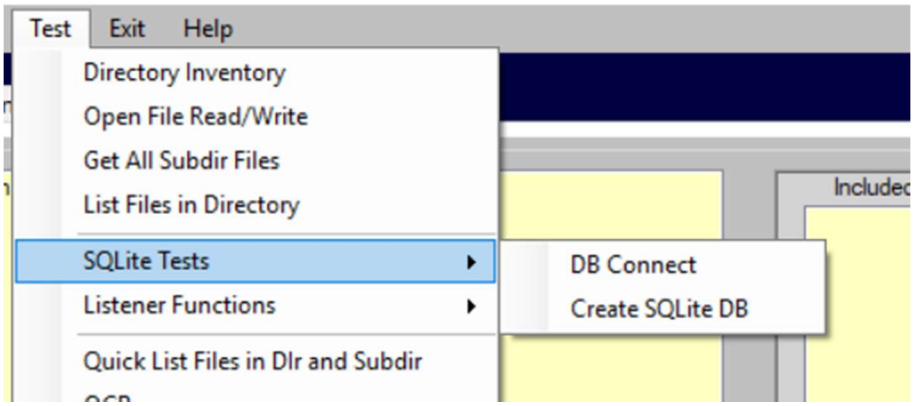
In this, each file in a selected directory and sub-directories, if desired, is presented on the screen. This is so that the admin can get a quick view of all files contained.

Test drop down



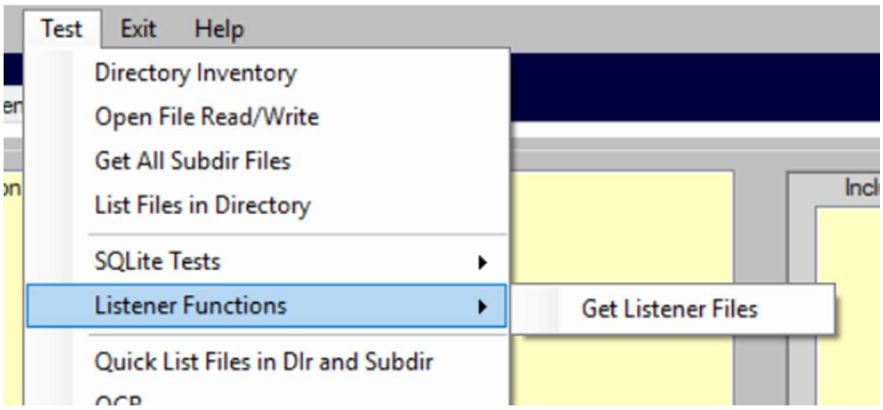
Many of these tests are designed to be executed by ECM developers and none are designed to be executed by anyone other than an administrator. These tests allow ECM developers to work with ECM Admins and find problems that potentially exist at the operating systems level.

SQLite Test drop down



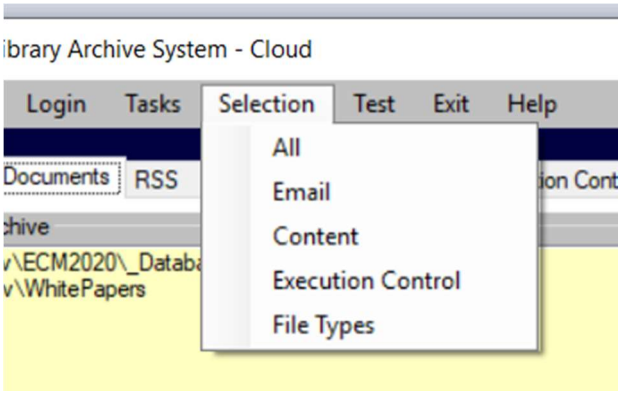
- Test the configured SQLite connection to the database.
- If the SQLite database is missing, create it.

Get Listener Files drop down



A quick check to validate that the listener is working and that files are being captured.

Selection drop down



Use this to open a specific tab. Clicking the tab does exactly the same thing.

iFilters

PDF iFilter Installation

W. Dale Miller (11-05-2020)

- Make sure Adobe PDF iFilter is installed. Today, 11-05-2020, the current iFilter is Adobe PDF iFilter 11.
- Make sure that PATH in environment variable(s) is set to the bin folder where you have installed iFilter in the previous step.
- By default, the path should be “C:\Program Files\Adobe\Adobe PDF iFilter 11 for 64-bit platforms\bin\”. If you have selected custom path, then we need to provide a path to the bin folder.
- Check the path of iFilter in SQL Server’s catalog view also.

```
SELECT *
FROM sys.fulltext_document_types
WHERE document_type LIKE '%pdf%'
```

Sometimes, we need to run below command so that iFilter is loaded correctly.

```
EXEC sp_fulltext_service 'load_os_resources', 1
GO
```

EXEC sp_fulltext_service 'verify_signature', 0

GO

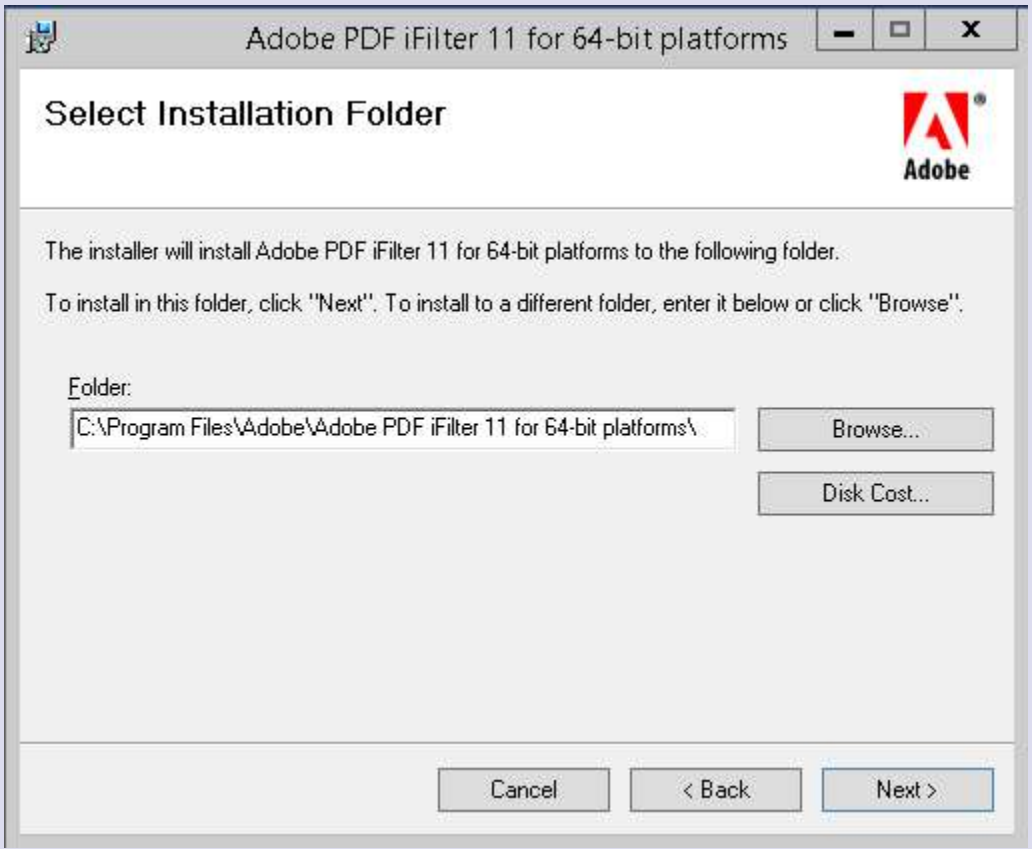
Rebuild the INDEX

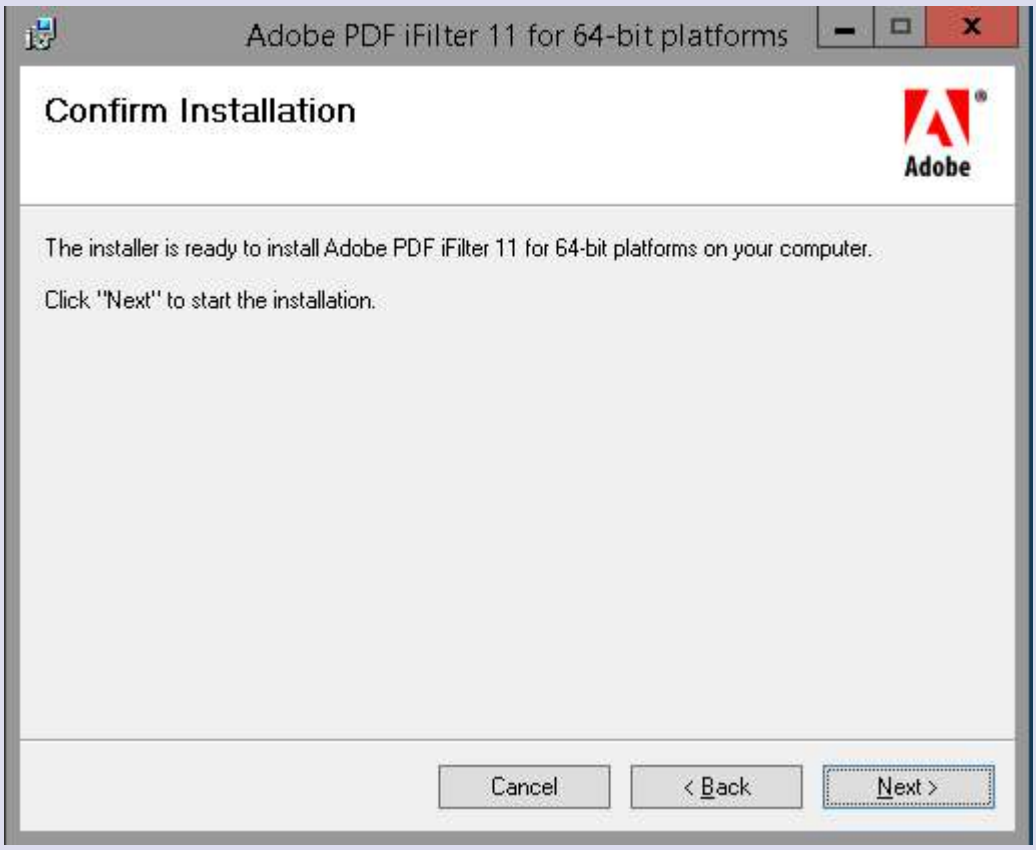
Now, if you can search few documents but not all, then you should check catalog population.

The PDF filter is not installed by default. So, for starters, we need to go download and install the PDF ifilter (PDFFilter64Setup.msi).

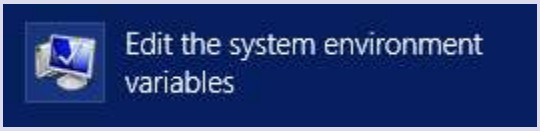
This is a next, next, finish install:


Adobe PDF iFilter 11 for 64-bit platforms
License Agreement
Adobe
Please take a moment to read the license agreement now. If you accept the terms below, click "I Agree", then "Next". Otherwise click "Cancel".
ADOBE
Adobe PDF iFilter
Software License Agreement
PLEASE READ THIS AGREEMENT CAREFULLY. BY COPYING, INSTALLING, OR USING ALL OR ANY PORTION OF THIS SOFTWARE, YOU (HEREINAFTER "CUSTOMER") ACCEPT ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT, INCLUDING, WITHOUT LIMITATION,
I Do Not Agree
I Agree
Cancel
< Back
Next >

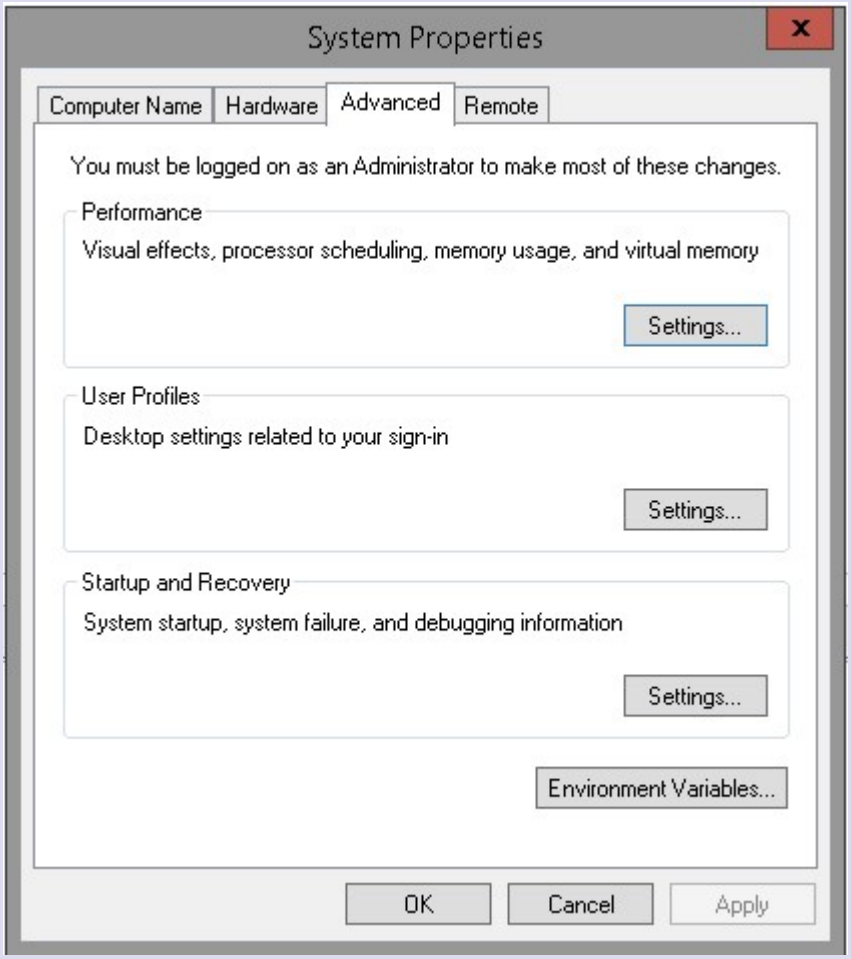

Adobe PDF iFilter 11 for 64-bit platforms
Select Installation Folder
Adobe
The installer will install Adobe PDF iFilter 11 for 64-bit platforms to the following folder.
To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".
Folder:
C:\Program Files\Adobe\Adobe PDF iFilter 11 for 64-bit platforms\
Browse...
Disk Cost...
Cancel
< Back
Next >



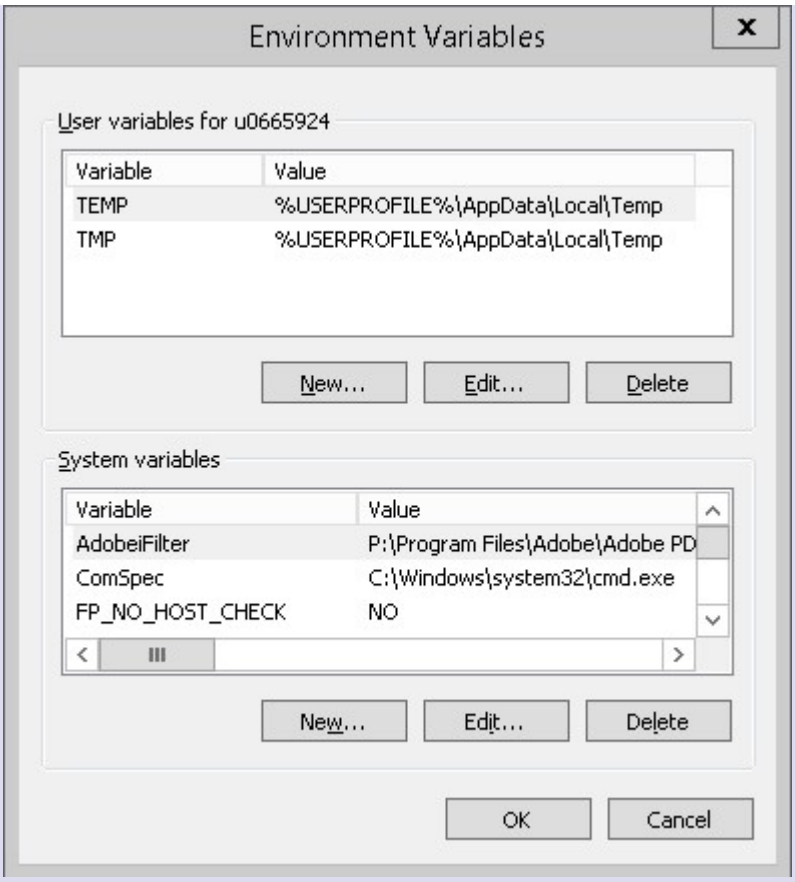
There is a trick with getting the installation setup correctly; you have to setup a system variable. Here's how I did it.

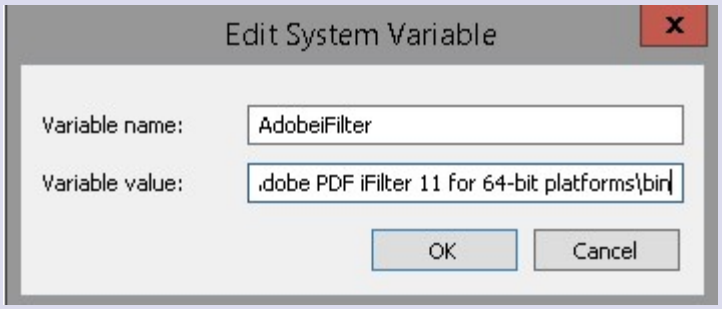


Environment Variables:

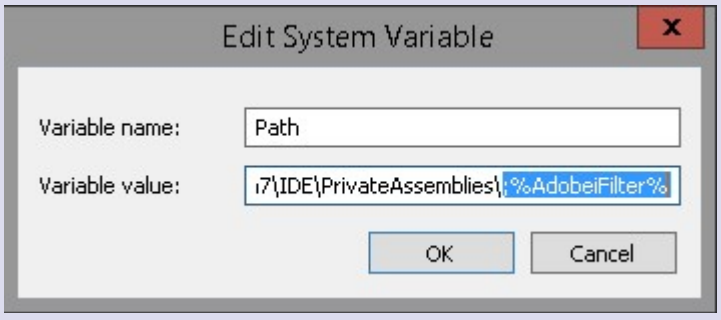


From Environment Variables I first created a new Variable "AdobeiFilter" and pointed it to the bin directory associated with the install:

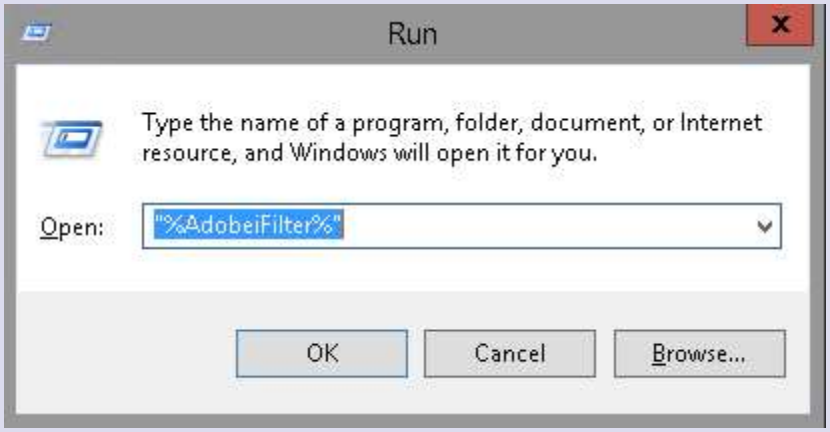




I then clicked Edit on the Path Variable, in environment Variables, and appended %AdobeiFilter% to the end:



I can validate this by opening run and typing "%AdobeiFilter%" which drops me to my bin directory here: "C:\Program Files\Adobe\Adobe PDF iFilter 11 for 64-bit platforms\bin"

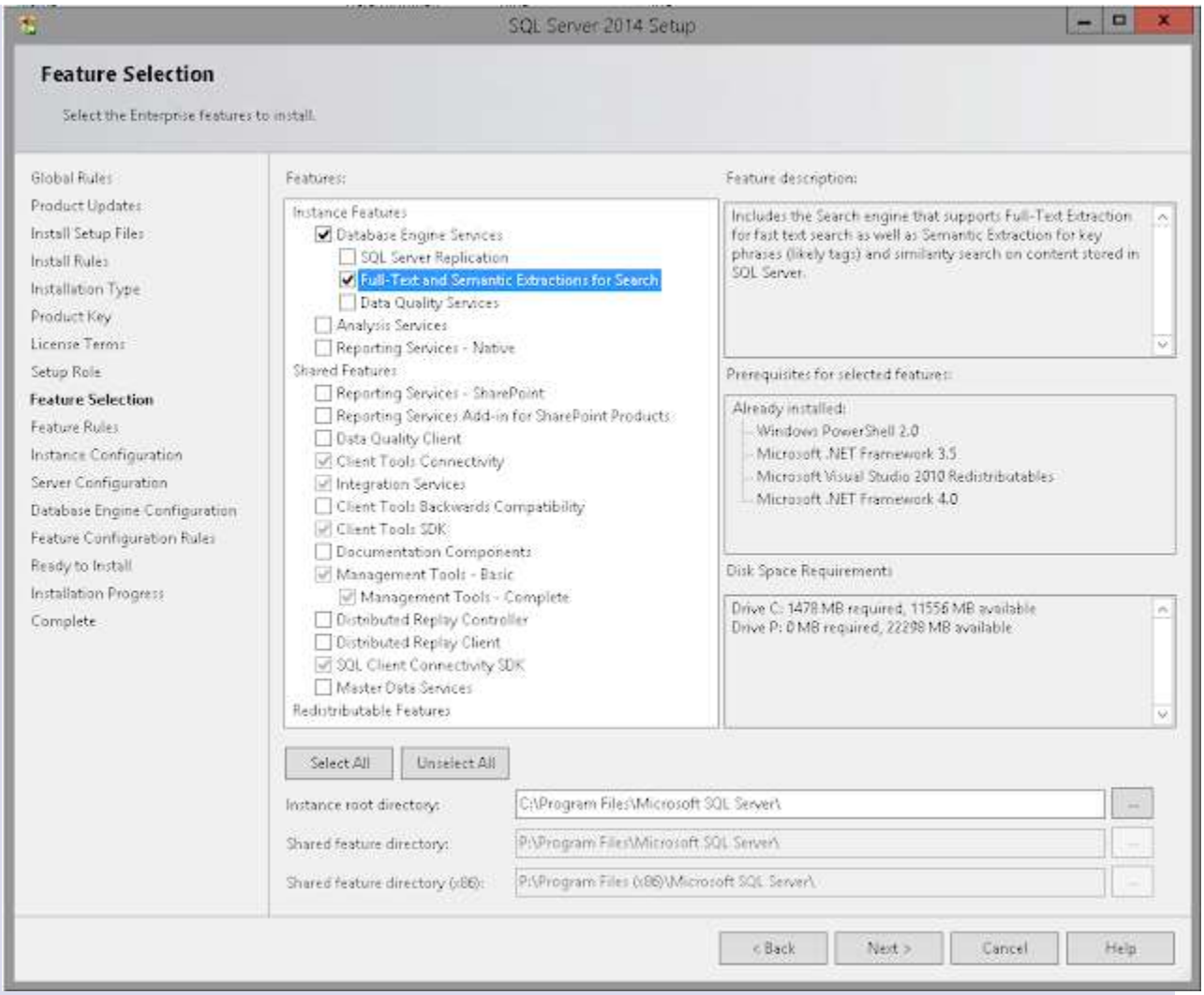


You may also need to grant your SQL Server Service account access to this folder.

Next we'll configure SQL Server Full-Text. You can check to see if it's already installed by running the following:
SELECT Serverproperty('IsFullTextInstalled')

If it's not installed you can run through the standard SQL Installation choosing "New SQL Server stand-alone installation or add features to an existing installation":

The feature is listed as "Full-Text and Semantic Extractions for search":



Next we run a couple of sp_fulltext_service commands:

```
--Load operating system filters and word breakers
EXEC Sp_fulltext_service
  @action='load_os_resources',
  @value=1;

--Do not verify whether binaries are signed
EXEC Sp_fulltext_service
  @action='verify_signature',
  @value=0;
```

Validate SQL is associating Full-Text and the PDF filter:

```
SELECT document_type,
       path
FROM   sys.fulltext_document_types
WHERE  document_type = '.pdf'
```

You may need to restart the SQL Service, possibly the box, at this point if the association isn't there.

SQLite Database Interface

SQLLite is used to track an inventory of files and content contained with directories that are to be archived. This eliminates a significant amount of network traffic and reduces archive time more than 50%.

Requirements

In order to avoid incorrect integration with MS SSMS, the working environment must meet the following conditions:

- The data source must be a configured system DSN. Refer to the Driver Configuration article to learn how to configure a System DSN
- The driver, studio, and SQL Server must be of the same bitness. For example, if you are using 64-bit SQL Server Management Studio on 64-bit Windows platform, then configure the 64-bit version of the driver using ODBC Administrator launched from %windir%\system32\odbcad32.exe. Otherwise, configure the driver using the 32-bit version of ODBC Administrator - launch it from %windir%\SysWOW64\odbcad32.exe.
- ODBC Driver for SQLite and SQL Server must be installed on the same computer.
- .NET Framework 4.5 must be installed on the computer.

Connecting to SQLite from SQL Server Management Studio using ODBC Driver for SQLite

You can use the Microsoft SQL Server Management Studio to connect your SQLite data to an SQL Server instance. Linked Server is a tool of MS SQL Server that allows to execute distributed queries to refer tables stored on non-SQL Server datbase in a single query. With linked servers, you can execute commands against different data sources such as SQLite and merge them with your SQL Server database. You can create a linked server with one of these methods: by using the options in the Object Explorer or by executing stored procedures.

Below are major advantages of using SQL Server Linked Servers to connect to SQLite:

1. The ability to connect other database instances on the same or remote server.
2. The ability to run distributed queries on heterogeneous data sources across the organization.
3. The ability to work with diverse data sources in the same way.

How to configure a SQL Server Linked Server to connect to SQLite

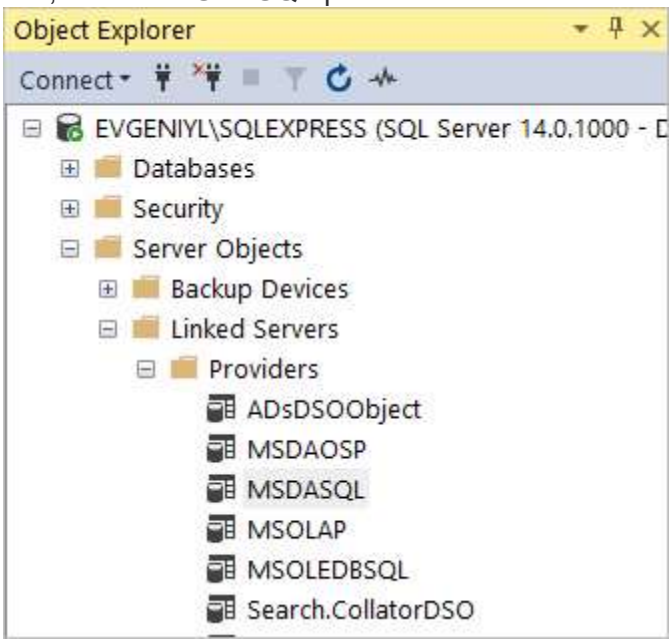
You can follow the steps to create a linked server for SQLite in SQL Server Management Studio by using Object Explorer:

1. Start your Management Studio and choose your SQL Server instance.
2. In the **Object Explorer pane**, expand the **Server Objects**, right-click on **Linked Servers** and then click on **New Linked Server**.
3. Configure your linked server in the dialog box:
   - Give a name for your server in the **Linked server** field.
   - Under **Server type**, select **Other data source** .
   - Choose **Microsoft OLE DB Provider for ODBC Drivers** in the **Provider** drop-down list.
   - In the **Data source** field, enter the name of your DSN, e.g. Devart ODBC Driver for SQLite . Alternatively, you can input the ODBC Driver connection string in the **Provider** field.

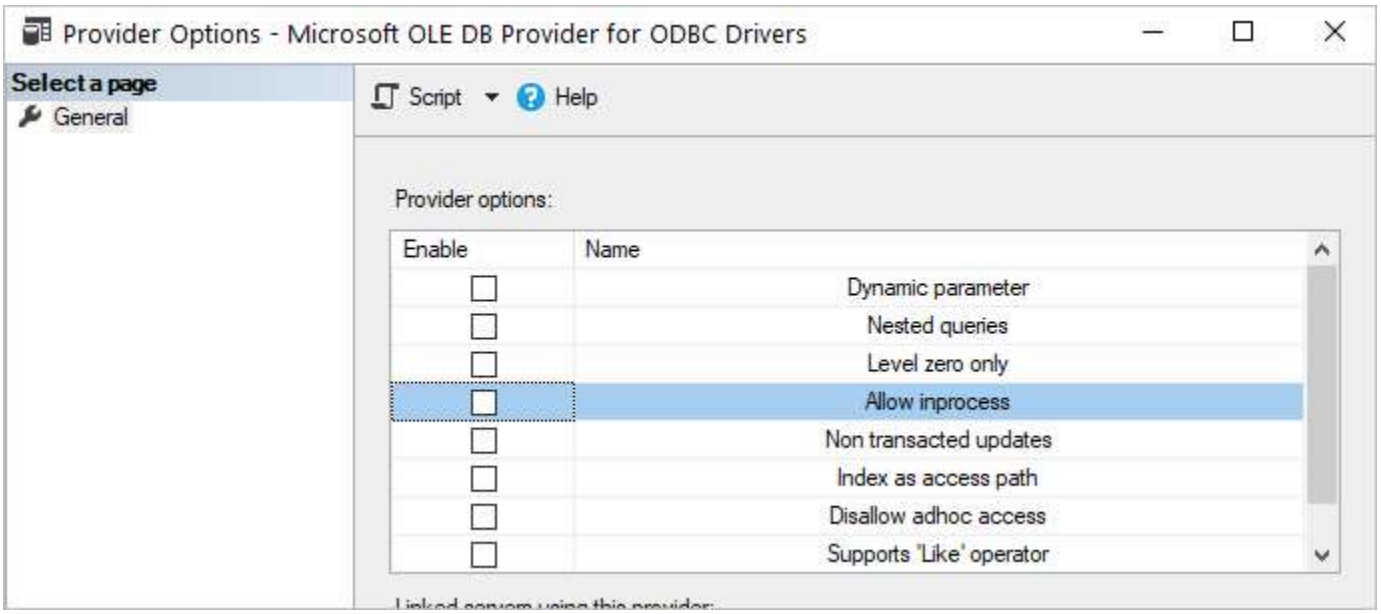
The linked server will appear under the Linked Servers in the Object Explorer Pane. You can now issue distributed queries and access SQLite databases through SQL Server.

Retrieving Data From SQLite

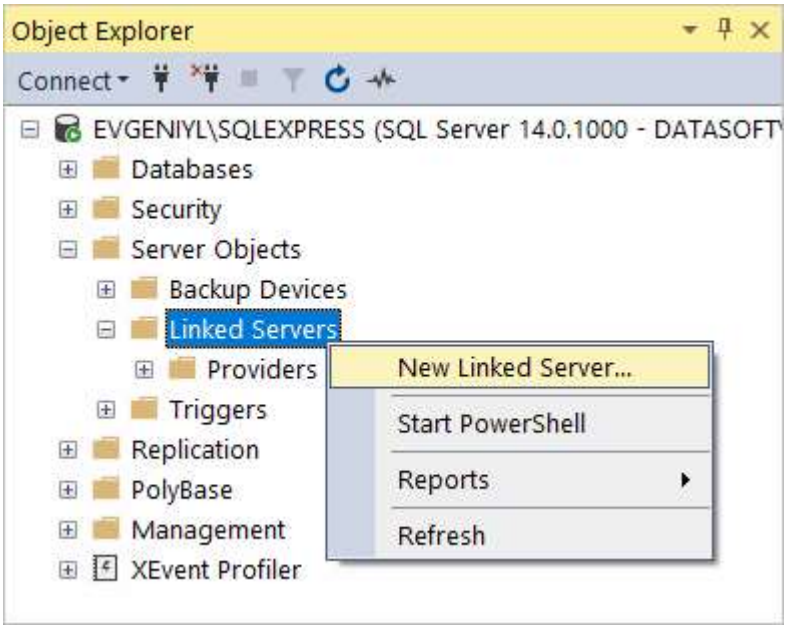
Disable the **Allow inprocess option** of MSDASQL OLE DB Provider for ODBC Drivers. For this, find the **MSDASQL** provider in the list of Linked Servers and double-click on it



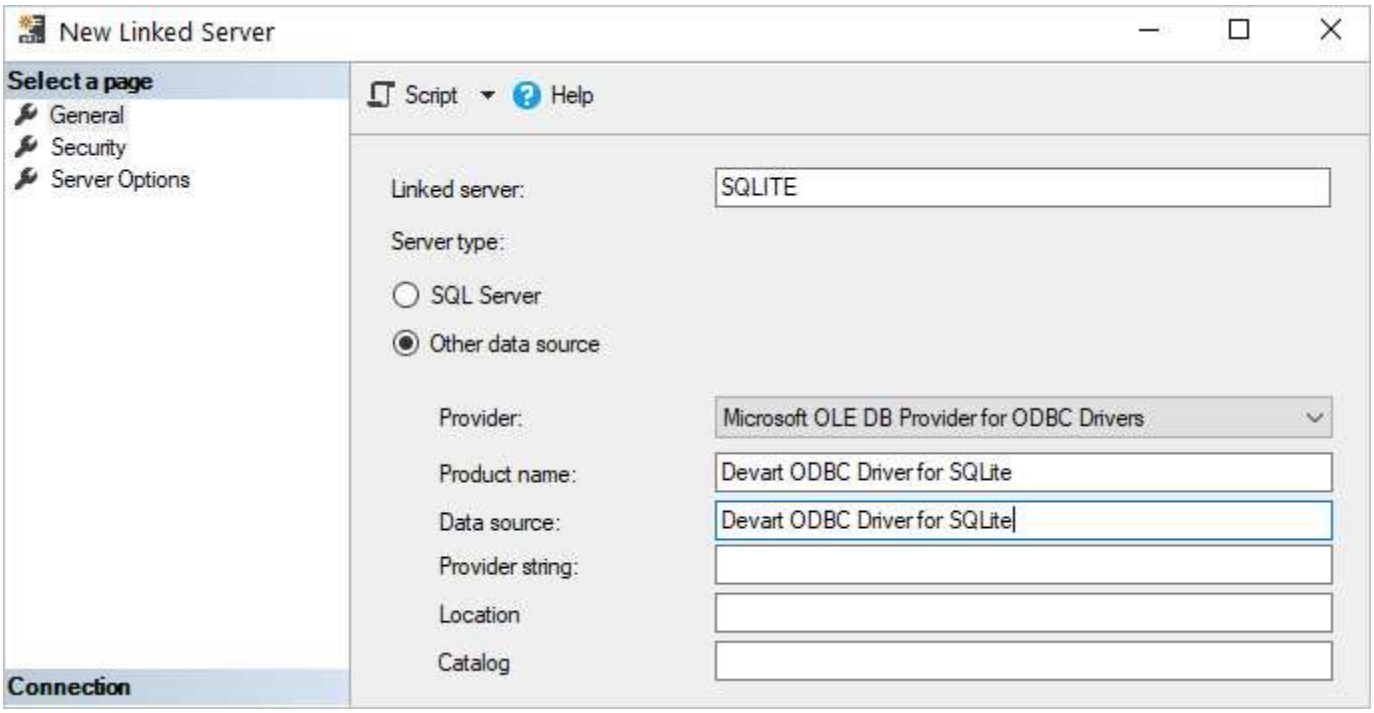
In the appeared **Provider Options** window, clear the **Allow inprocess** checkbox:



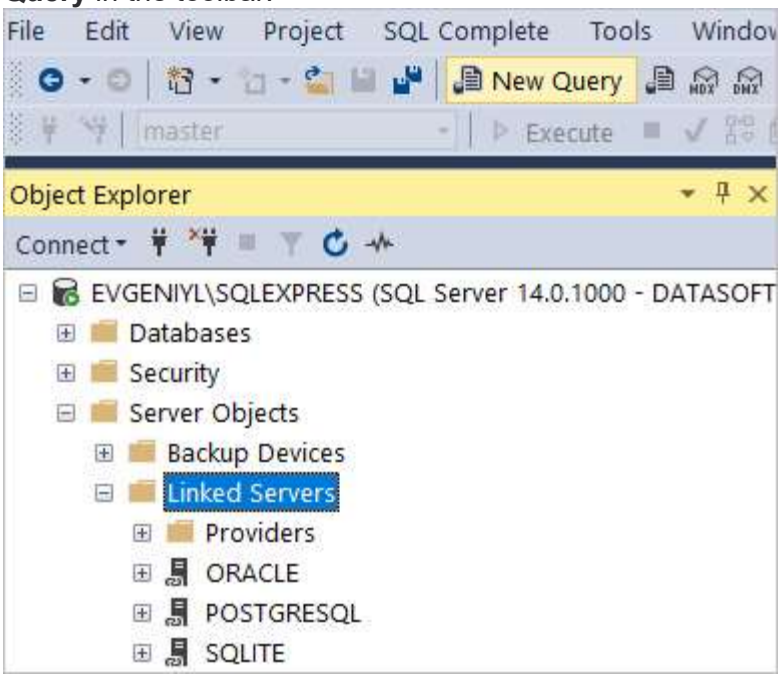
Create a new **Linked Server**



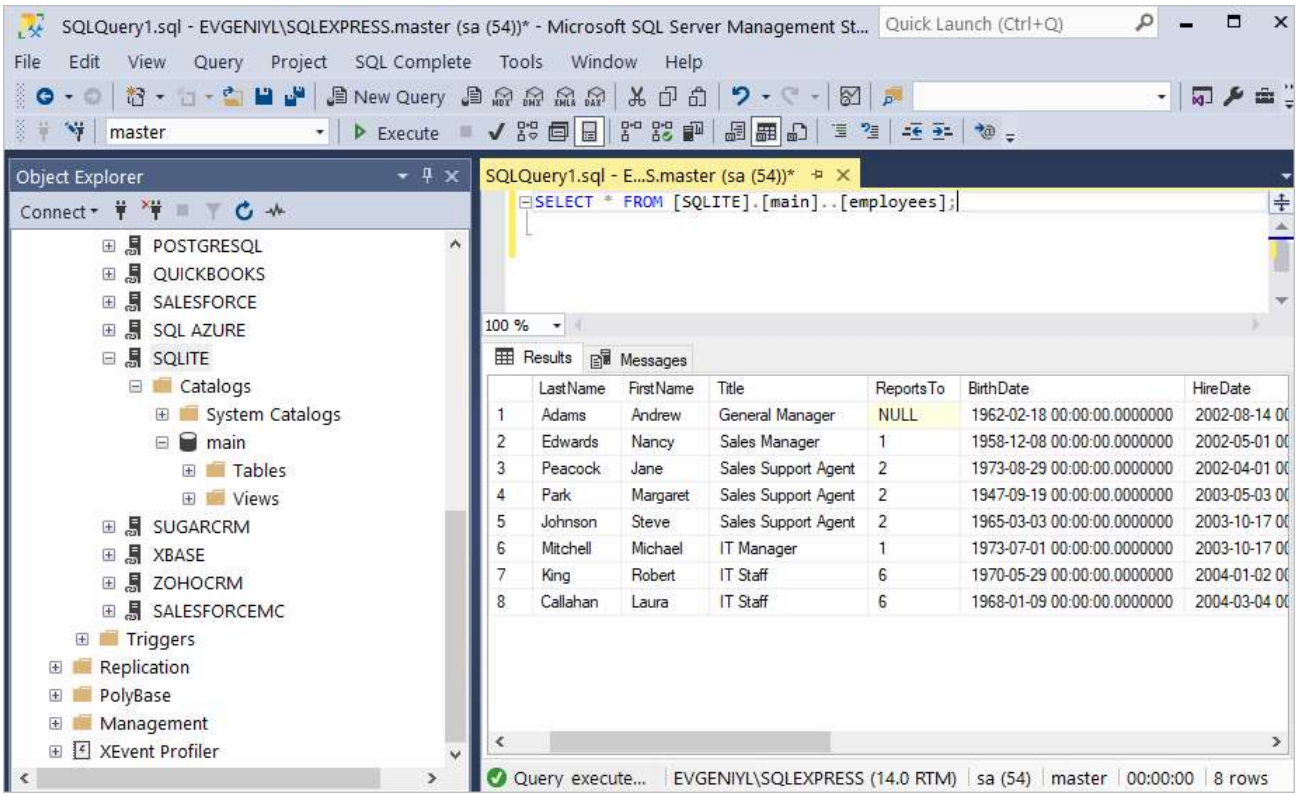
Make sure to select **Microsoft OLE DB Provider for ODBC Drivers** and specify the following parameters:



The SQLite tables are already available to be fetched. To query the linked server, click **New Query** in the toolbar:



Enter your SQL query in the editor window and click **Execute** to run the query:



As a result, you can see the contents of the selected table retrieved directly from the SQLite account you are connected to.

Attention

If the Linked Server was created with the Allow inprocess option enabled, then you should delete this Linked Server and create it again with the Allow inprocess option disabled.