# Automating/Scripting the Sage X3 Installation & Config

**15th Nov 2023**

**Raheel khan**

Sage

# Contents

- **Sage X3 Component Installation Files**
    - How the files are shipped
    - The format and how they are packaged
    - Automatic Install files

- **Automating the Installation of technical components**
    - Why use an automated Install
    - Scripting the Install
    - Demo

- **Using Launchersolution.exe for technical config**
    - Configure the X3 Solution
    - Configure technical components
    - Demo

- **Using Launchersolution.exe for console tasks**
    - Import Folder
    - Export Folder
    - Copying folders
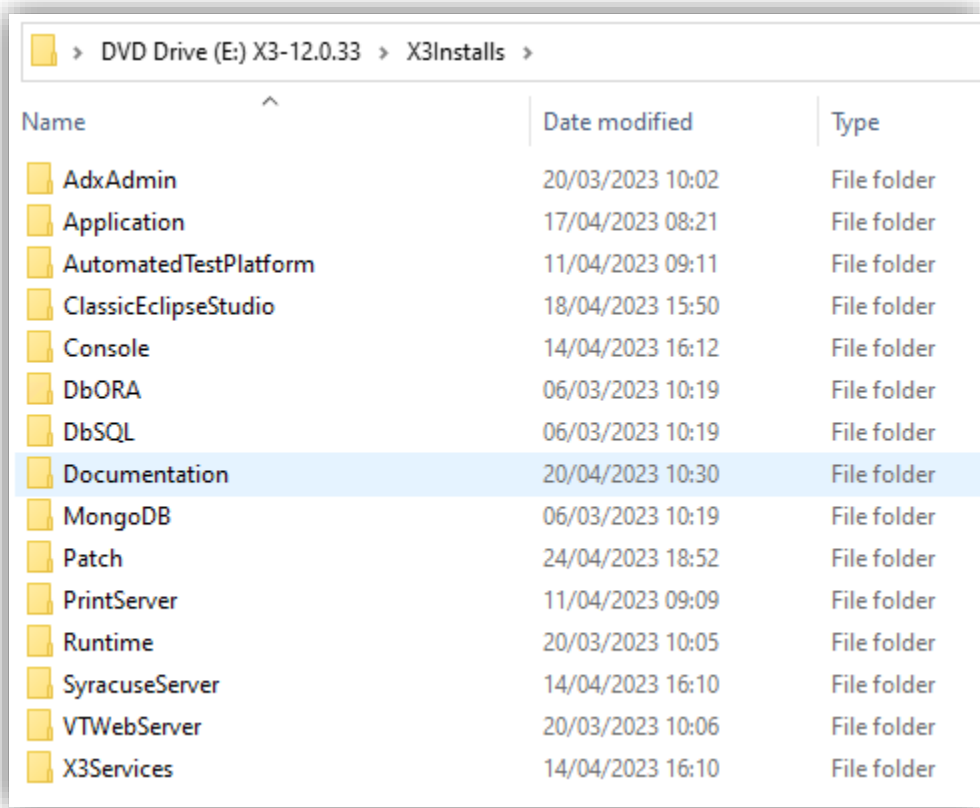
# Sage X3 Component Installation Files

# Sage X3 Component Installation Files

With each release of Sage X3, the ISO distribution is made available on the Sage X3 UK FTP site , which contains the following key folders.
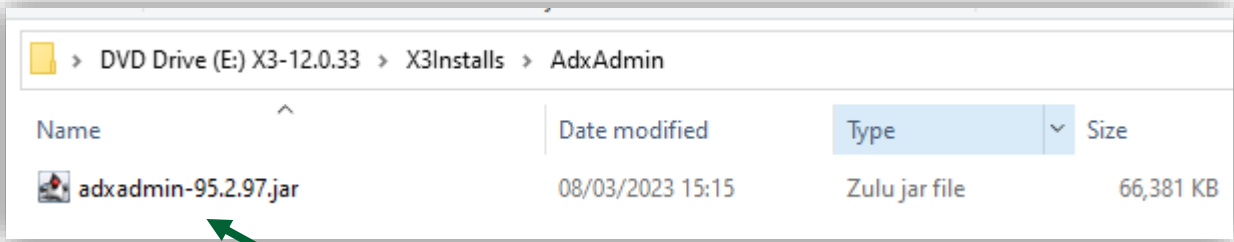
- **Documents:** This directory contains the documents about changes in this version.

  - Components readme, which contains information on the component changes in the release,

  - X3 Readme, which contains the applicative fixes information.

  - Version Information Note which contains the instructions that you need to follow after the installation or upgrade

- **FREECOMPONENTS:** This directory contains the prerequisite linked to certain components.

- **Resources:** This directory contains the resources patch (updates menus, sitemap, homepages)

- **X3Installs:** This directory contains the installation files for each component

- **Index.htm:** file which contains a list of all the components in the X3installs folder and versions

# Sage X3 Component Installation Files

The X3install directory which we will be covering contains a folder for each of the technical components which contains the installation file.

| | DVD Drive (E:) X3-12.0.33 > X3Installs > | | |
|---|---|---|---|
| **Name** | **Date modified** | **Type** | |
| AdxAdmin | 20/03/2023 10:02 | File folder | |
| Application | 17/04/2023 08:21 | File folder | |
| AutomatedTestPlatform | 11/04/2023 09:11 | File folder | |
| ClassicEclipseStudio | 18/04/2023 15:50 | File folder | |
| Console | 14/04/2023 16:12 | File folder | |
| DbORA | 06/03/2023 10:19 | File folder | |
| DbSQL | 06/03/2023 10:19 | File folder | |
| Documentation | 20/04/2023 10:30 | File folder | |
| MongoDB | 06/03/2023 10:19 | File folder | |
| Patch | 24/04/2023 18:52 | File folder | |
| PrintServer | 11/04/2023 09:09 | File folder | |
| Runtime | 20/03/2023 10:05 | File folder | |
| SyracuseServer | 14/04/2023 16:10 | File folder | |
| VTWebServer | 20/03/2023 10:06 | File folder | |
| X3Services | 14/04/2023 16:10 | File folder | |

| | DVD Drive (E:) X3-12.0.33 > X3Installs > AdxAdmin | | |
|---|---|---|---|
| **Name** | **Date modified** | **Type** | **Size** |
| adxadmin-95.2.97.jar | 08/03/2023 15:15 | Zulu jar file | 66,381 KB |

Each component installation contains a . Jar file. with the component name & and version number

The installer used for the for the application is IzPack.

IzPack - Installation of Sage Safe X3 AdxAdmin

sage

# Installation Packages IzPack

IzPack is an open-source Java-based software installation tool It allows developers to package their software with all necessary configuration files, libraries, and dependencies into a single installer package that users can run to install the software on their systems.

Some of the advantages over other installers

**1. Cross-Platform Compatibility:** IzPack-generated installers can be used on multiple operating in the case of Sage Windows and Linux

**2. Customizable User Interface:** Sage can design installer panels, screens, and layouts to match the branding and style of their application.

**3. Automatic Dependency Handling:** IzPack can manage the installation of required dependencies and libraries. For example, .NET framework version

**4. User Input and Configuration:** Developers can define prompts for user input during installation, allowing users to customize installation paths, settings, and other options.

**5. Uninstallation & Update Support:** IzPack installers include an uninstallation and update process to remove the installed application and related files cleanly.

**6. XML-Based Configuration:** The installer configuration can be defined using XML files, allowing you to define installation steps and behaviour.

IzPack.org

# IzPack XML Configuration

As mentioned, an XML configuration file can be used to automate the installation process when using the IzPack installer. The installer reads the configuration file to determine the installation steps, user inputs, and packages to install as part of the process.

Elements of the XML configuration file

**AutomatedInstallation**: This is the root element of the XML file and contains the entire installation script.

**HelloPanel**: This panel displays a greeting message to the user.

**ReadMePanel**: This displays a readme or informational message.

**LicencePanel**: This panel displays a license agreement with Sage.

**InstallTypePanel**: This panel specifies the installation type and path.

**UserInputPanel**: This panel collects user input. In this case, it sets values for the property's component.node.type and component.node.name.

**PacksPanel:** This panel allows the user to select which components or packs they want to install. It lists four packs with different names and selection statuses.

**SummaryPanel:** This panel provides a summary of the installation configuration.

**InstallPanel:** This panel initiates the installation process.

**ProcessPanel**: Displays progress during the installation process.

**FinishPanel:** This panel indicates the completion of the installation.

Sage

# IzPack Automatic Install File

Example IzPack XML automatic install configuration file for Sage X3 In this XML file, you have a series of panels defined, each for a specific step in the installation process. Each panel serves a different purpose in guiding the user through the installation process.

1. Hello, ReadMe, Licence Panels which display messages

2. Install Type Panel: Asks for installation type (modify or fresh install) and the install path

3. User Input Panel: Collects user input In this case component name and runtime install type

4. Packs Panel: Allows the user to select components to install

5. The Summary, Process & Finish panel display information to the user & Install panel performs the installation.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<AutomatedInstallation langpack="eng">
<com.izforge.izpack.panels.HelloPanel id="UNKNOWN (com.izforge.izpack.panels.HelloPanel)"/>
<com.izforge.izpack.panels.sage.ReadMePanel id="UNKNOWN (com.izforge.izpack.panels.sage.ReadMePanel)"/>
<com.izforge.izpack.panels.HTMLLicencePanel id="UNKNOWN (com.izforge.izpack.panels.HTMLLicencePanel)"/>
<com.izforge.izpack.panels.LicencePanel id="UNKNOWN (com.izforge.izpack.panels.LicencePanel)"/>
<com.izforge.izpack.panels.InstallTypePanel id="UNKNOWN (com.izforge.izpack.panels.InstallTypePanel)">
<MODIFY.IZPACK.INSTALL>false</MODIFY.IZPACK.INSTALL>
<installpath>D:\Sage\X3V12\runtime</installpath>
</com.izforge.izpack.panels.InstallTypePanel>
<com.izforge.izpack.panels.UserInputPanel id="UNKNOWN (com.izforge.izpack.panels.UserInputPanel)">
<userInput>
<entry key="component.node.type" value="MAIN"/>
<entry key="component.node.name" value="X3V12"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.TargetPanel id="UNKNOWN (com.izforge.izpack.panels.TargetPanel)">
<installpath>D:\Sage\X3V12\runtime</installpath>
</com.izforge.izpack.panels.TargetPanel>
<com.izforge.izpack.panels.PacksPanel id="UNKNOWN (com.izforge.izpack.panels.PacksPanel)">
<pack index="0" name="Eula files" selected="true"/>
<pack index="1" name="Safe X3 Runtime Base files for Windows X64" selected="true"/>
<pack index="2" name="Safe X3 Runtime Third party files for Windows X64" selected="true"/>
<pack index="3" name="Safe X3 Runtime Service files for Windows X64" selected="true"/>
</com.izforge.izpack.panels.PacksPanel>
<com.izforge.izpack.panels.SummaryPanel id="UNKNOWN (com.izforge.izpack.panels.SummaryPanel)"/>
<com.izforge.izpack.panels.InstallPanel id="UNKNOWN (com.izforge.izpack.panels.InstallPanel)"/>
<com.izforge.izpack.panels.ProcessPanel id="UNKNOWN (com.izforge.izpack.panels.ProcessPanel)"/>
<com.izforge.izpack.panels.FinishPanel id="UNKNOWN (com.izforge.izpack.panels.FinishPanel)"/>
</AutomatedInstallation>
```

# Generating The File

When we complete the manual installation of each component, we are presented with the option to save the XML configuration for the installation. This produces a file with all the parameters that we specified during the installation.

Existing XML configurations can also be modified.
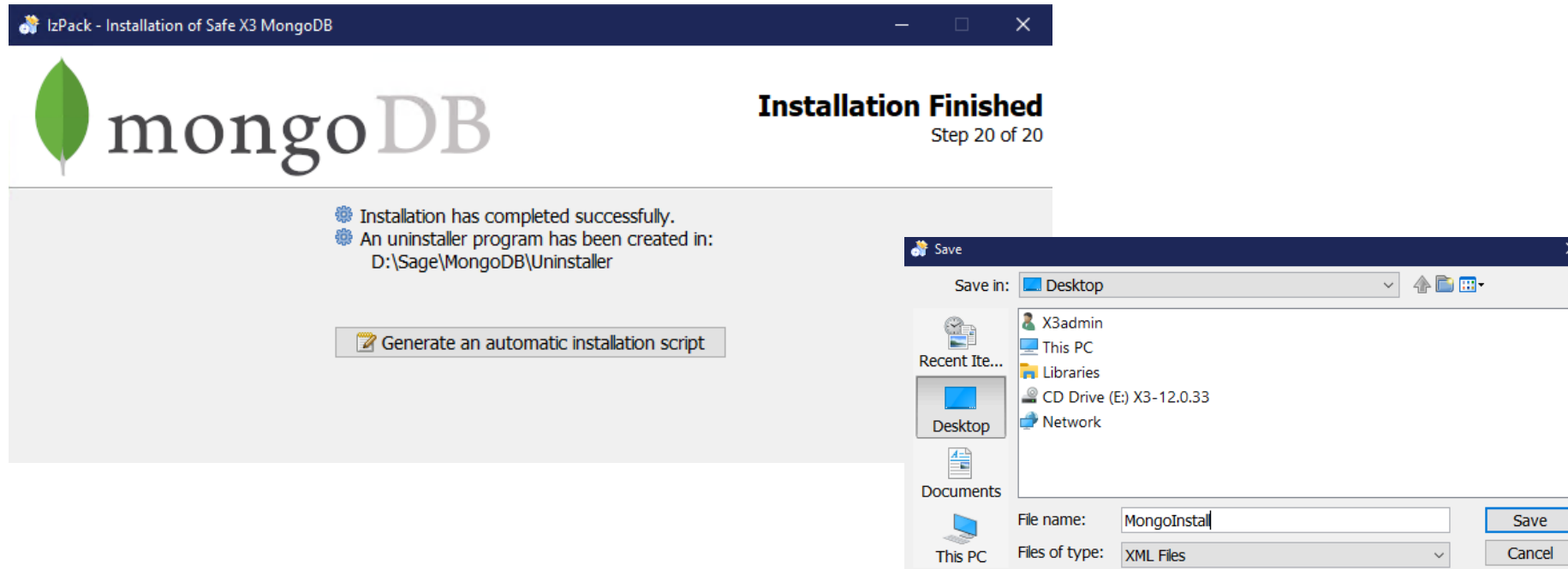
# AdxAdmin

Example of an AdxAdmin automated Installation file



Adxadmin.xml

```
1    <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2    <AutomatedInstallation langpack="eng">
3    <com.izforge.izpack.panels.HelloPanel id="UNKNOWN (com.izforge.izpack.panels.HelloPanel)"/>
4    <com.izforge.izpack.panels.HTMLLicencePanel id="UNKNOWN                     anels.HTMLLicencePanel)"/>
5    <com.izforge.izpack.panels.LicencePanel id="UNKNOWN (com.izforge.izpack.panels.LicencePanel)"/>
6    <com.izforge.izpack.panels.TargetPanel id="UNKNOWN (com.izforge.izpack.panels.TargetPanel)">
7    <installpath>D:\Sage\SafeX3\ADXADMIN</installpath>
8    </com.izforge.izpack.panels.TargetPanel>
         forge.izpack.panels.UserInputPanel id="UNKNOWN (com.izforge.izpack.panels.UserInputPanel)">
         nput>
11   <entry key="adxadmin.service.port" value="1895"/>
12   <entry key="adxadmin.administrator.enable" value="true"/>
13   </userInput>
14   </com.izforge.izpack.panels.UserInputPanel>
15   <com.izforge.izpack.panels.PacksPanel id="UNKNOWN (com.izforge.izpack.panels.PacksPanel)">
16   <pack index="0" name="Eula files" selected="true"/>
17   <pack index="1" name="Safe X3 AdxAdmin Base files for Windows X64" selected="true"/>
18   <pack index="2" name="Safe X3 AdxAdmin Service files for Windows X64" selected="true"/>
19   </com.izforge.izpack.panels.PacksPanel>
20   <com.izforge.izpack.panels.SummaryPanel id="UNKNOWN (com.izforge.izpack.panels.SummaryPanel)"/>
21   <com.izforge.izpack.panels.InstallPanel id="UNKNOWN (com.izforge.izpack.panels.InstallPanel)"/>
22   <com.izforge.izpack.panels.ProcessPanel id="UNKNOWN (com.izforge.izpack.panels.ProcessPanel)"/>
23   <com.izforge.izpack.panels.FinishPanel id="UNKNOWN (com.izforge.izpack.panels.FinishPanel)"/>
24   </AutomatedInstallation>
```

**Installation Path**

**AdxAdmin Port**

**Windows service started as Administrator.**

**What's installed**
-Eula Files
-Base files
-Service files

# Runtime

Example Runtime automated Installation file



**Installation Path**

```
 1  <?xml version="1.0" encoding="UTF-8" standalone="no"?>
 2  <AutomatedInstallation langpack="eng">
 3  <com.izforge.izpack.panels.HelloPanel id="UNKNOWN (com.izforge.izpack.panels.HelloPanel)"/>
 4  <com.izforge.izpack.panels.sage.ReadMePanel id="UNKNOWN (com.izforge.izpack.panels.sage.ReadMePanel)"/>
 5  <com.izforge.izpack.panels.HTMLLicencePanel id="UNKNOWN (com.izforge.izpack.panels.HTMLLicencePanel)"/>
 6  <com.izforge.izpack.panels.LicencePanel id="UNKNOWN (com.izforge.izpack.panels.LicencePanel)"/>
 7  <com.izforge.izpack.panels.InstallTypePanel id="UNKNOWN (com.izforge.izpack.panels.InstallTypePanel)">
 8  <MODIFY.IZPACK.INSTALL>false</MODIFY.IZPACK.INSTALL>
 9  <installpath>D:\Sage\X3V12\runtime</installpath>
10  </com.izforge.izpack.panels.InstallTypePanel>
11  <com.izforge.izpack.panels.UserInputPanel id="UNKNOWN (com.izforge.izpack.panels.UserIn
12  <userInput>
13  <entry key="component.node.type" value="MAIN"/>
14  <entry key="component.node.name" value="X3V12"/>
15  </userInput>
16  </com.izforge.izpack.panels.UserInputPanel>
17  <com.izforge.izpack.panels.TargetPanel id="UNKNOWN (com.izforge.izpack.panels.TargetPan
18  <installpath>D:\Sage\X3V12\runtime</installpath>
19  </com.izforge.izpack.panels.TargetPanel>
20  <com.izforge.izpack.panels.PacksPanel id="UNKNOWN (com.izforge.izpack.panels.PacksPanel)">
21  <pack index="0" name="Eula files" selected="true"/>
    <pack index="1" name="Safe X3 Runtime Base files for Windows X64" selected="true"/>
    <pack index="2" name="Safe X3 Runtime Third party files for Windows X64" selected="true"/>
    <pack index="3" name="Safe X3 Runtime Service files for Windows X64" selected="true"/>
    </com.izforge.izpack.panels.PacksPanel>
    <com.izforge.izpack.panels.SummaryPanel id="UNKNOWN (com.izforge.izpack.panels.SummaryPanel)"/>
    <com.izforge.izpack.panels.InstallPanel id="UNKNOWN (com.izforge.izpack.panels.InstallPanel)"/>
    <com.izforge.izpack.panels.ProcessPanel id="UNKNOWN (com.izforge.izpack.panels.ProcessPanel)"/>
    <com.izforge.izpack.panels.FinishPanel id="UNKNOWN (com.izforge.izpack.panels.FinishPanel)"/>
    </AutomatedInstallation>
```

**Component type (main runtime, additional, test etc)**

**Component Name**

**What's installed**
- Eula Files
- Base files
- Third-Party files
- Service files

# MongoDB

MongoDB automated Installation file

```xml
<userInput>
 <entry key="mongodb.ssl.capassphrase" value="S@GE2019_CA"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.6">
<userInput>
 <entry key="mongodb.ssl.certificate.email" value="X3amin@sage.com"/>
 <entry key="mongodb.ssl.certificate.organization" value="UK"/>
 <entry key="mongodb.ssl.certificate.city" value="UK"/>
 <entry key="mongodb.ssl.certificate.name" value="X3admin"/>
 <entry key="mongodb.ssl.certificate.validity" value="3650"/>
 <entry key="mongodb.ssl.certificate.countrycode" value="UK"/>
 <entry key="mongodb.ssl.certificate.state" value="UK"/>
 <entry key="mongodb.ssl.certificate.organisationalunit" value="UK"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.7">
<userInput>
 <entry key="mongodb.ssl.serverpassphrase" value="S@GE2019_SERVEUR"/>
 <entry key="mongodb.ssl.certificate.hostname" value="x3erpv12sqlvm"/>
</userInput>
```

```xml
<userInput>
 <entry key="mongodb.dir.dbpath" value="D:\Sage\MongoDB\data"/>
 <entry key="component.node.name" value="MONGO01"/>
 <entry key="mongodb.service.creation" value="true"/>
 <entry key="mongodb.dir.configpath" value="D:\Sage\MongoDB\config"/>
 <entry key="mongodb.dir.logpath" value="D:\Sage\MongoDB\logs"/>
 <entry key="mongodb.service.port" value="27017"/>
 <entry key="mongodb.net.ipv6" value="false"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.1">
<userInput>
 <entry key="mongodb.dir.certs" value="D:\Sage\MongoDB\certs"/>
 <entry key="mongodb.ssl.createcert" value="true"/>
 <entry key="mongodb.ssl.enable" value="true"/>
</userInput>
```

MongoDB Installation configuration file, we can see certificate-related information also

- Validity
- Passphrases in clear text **
- Hostname

MongoDB configuration
-Directory paths, data, logs, config
-Certificate paths
-Enable SSL

# Syracuse

Syracuse automated Installation file

Syracuse configuration

Certificate Passphrases Service user passwords in clear text**

```
</com.izforge.izpack.panels.InstallTypePanel>
<com.izforge.izpack.panels.TargetPanel id="UNKNOWN (com.izforge.izpack.panels.TargetPanel)">
<installpath>D:\Sage\Syracuse</installpath>
</com.izforge.izpack.panels.TargetPanel>
<com.izforge.izpack.panels.PacksPanel id="UNKNOWN (com.izforge.izpack.panels.PacksPanel)">
<pack index="0" name="Syracuse Base files" selected="true"/>
<pack index="1" name="Syracuse Windows files" selected="true"/>
<pack index="2" name="Syracuse Configuration files" selected="true"/>
<pack index="3" name="Syracuse Windows Control files" selected="true"/>
<pack index="4" name="Syracuse Documentation files" selected="false"/>
</com.izforge.izpack.panels.PacksPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.0">
<userInput>
<entry key="syracuse.service.webnumber" value="2"/>
<entry key="syracuse.winservice.username" value="x3run"/>
<entry key="syracuse.service.procnumber" value="2"/>
<entry key="component.node.name" value="NODE0"/>
<entry key="syracuse.service.port" value="8124"/>
<entry key="syracuse.service.licencepath" value="C:\X3InstallScripts\X3V12_License-SAGE_GLOBAL_MyX3_NFR_ADCbadges_MAXATP=5-Endval=20240131.json"/>
<entry key="syracuse.dir.logpath" value="D:\Sage\Syracuse\syracuse\logs"/>
<entry key="syracuse.winservice.password" value="s@ge2023"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.1">
<userInput>
<entry key="syracuse.dir.certs" value="D:\Sage\Syracuse\syracuse\certs"/>
<entry key="syracuse.certificate.install" value="true"/>
</userInput>
</com.izforge.izpack.panels.UserInputPanel>
<com.izforge.izpack.panels.UserInputPanel id="UserInputPanel.2">
<userInput>
<entry key="syracuse.certificate.capassphrase" value="S@GE2019_CA"/>
</userInput>
```

Syracuse licence file

Mongo Certs location

```
<userInput>
<entry key="mongodb.service.hostname" value="x3erpv12sqlvm"/>
<entry key="mongodb.ssl.client.pemkeyfile" value="D:\Sage\MongoDB\certs\client.key"/>
<entry key="mongodb.ssl.client.certfile" value="D:\Sage\MongoDB\certs\client.crt"/>
<entry key="mongodb.ssl.enable" value="true"/>
<entry key="mongodb.service.port" value="27017"/>
<entry key="mongodb.ssl.pemcafile" value="D:\Sage\MongoDB\certs\ca.cacrt"/>
</userInput>
```

# Automating the Installation of Sage X3 Technical Components

# Why Automate the Install

A scripted/ automated install of Sage X3 can be useful in various scenarios. It helps to streamline the installation process and minimise manual intervention during the installation of the components. Some of the potential use cases

- Automation ensures that the installation process is consistent every time it's executed. This minimizes the chances of human error that might occur during manual installations.

- Automated installations can be performed with minimal human intervention, freeing up time and resources for other tasks.

- Automated installation files can be saved and reused, making it easy to replicate the installation on different machines or in different environments.

- There is a requirement to clone the existing install you can use the automated install files

- Automated installations are typically faster than manual ones.

- Install could be scheduled for a specific time (using Windows Task Scheduler or other scheduling software to launch the install)

# Scripting the Component Install

Once the XML installation configuration files are populated, we can use a command to call the installation file, parse the automated install file and launch the component installation.

Example, in CMD, we can execute to install MongoDB using the parameters from the  MongoDB.xml file.

```
java -jar E:\X3Installs\MongoDB\mongo-db-4.4.12.9.jar C:\X3InstallScripts\MongoDB.xml
```

This part of the command is invoking the Java Virtual Machine (JVM), which is required to execute Java programs. You need to have Java installed and a JAVA home path is set.

This is the full path to the JAR file that you want to execute. In this case, it's located on the E: drive under the directory X3Installs\MongoDB.

This is an argument being passed to the Java program (JAR file). It's a path to the automatic installation file located on the C: drive under the directory X3InstallScripts. This XML file contains the configuration settings for the installation.

# Example Script

Once the XML installation configuration files are ready, we can use a script which calls the installation file and parses the configuration file to launch the component installation.

```
1    @echo off
2    setlocal
3    set DIR_ERROR=0
4
5    :: Mount point for ISO and directory for Jar distributions
6    set ISO_INSTALL_PATH=E:\X3Installs
7
8    :: Directory that contins the automatic install files
9    set INSTALL_SCRIPT_FOLDER=C:\X3InstallScripts
10
11   :: Jar file names for the version you are installing
12   set MONGODB_JAR=mongo-db-4.4.12.9.jar
13   set ADXADMIN_JAR=adxadmin-95.2.97.jar
14   set DBSQL_JAR=db-sql-3.0.0-win.jar
15   set APPLICATION_JAR=x3-application-12.0.33.jar
16   set RUNTIME_JAR=runtime-95.2.97.jar
17   set SYRACUSE_JAR=syracuse-server-12.18.0.50.jar
18   set CONSOLE_JAR=console-2.57.0.11-win.jar
19   set PRINTSERVER_JAR=print-server-2.28.0.10-win.jar
20
21
22   :: Test if ISO is present
23   if not exist %ISO_INSTALL_PATH%\. (
24     set DIR_ERROR=1
25     echo ISO distribution directory %ISO_INSTALL_PATH% does not exist !!!
26   )
27
28   if %DIR_ERROR%==0 goto PROCEED
29
30   Pause
31   goto EXITSCRIPT
32
33   :PROCEED
34   :: Now working...
```

Install directory

Install Script Location

The Install files

Test for Install directory.

The Script executes each installation program and passes the automatic install file which contains the install parameters.

```
36   :: 1 - MONGODB
37   java -jar %ISO_INSTALL_PATH%\MongoDB\%MONGODB_JAR% %INSTALL_SCRIPT_FOLDER%\MongoDB.xml
38
39   :: echo Waiting about 10 secs before next step...
40   ping -n 10 localhost >NUL: 2>&1
41
42   :: 2 - ADXADMIN
43   java -jar %ISO_INSTALL_PATH%\AdxAdmin\%ADXADMIN_JAR% %INSTALL_SCRIPT_FOLDER%\Adxadmin.xml
44
45   echo Waiting about 10 secs before next step...
46   ping -n 10 localhost >NUL: 2>&1
47
48   :: 3 - DBSQL
49   java -jar %ISO_INSTALL_PATH%\DbSQL\%DBSQL_JAR% %INSTALL_SCRIPT_FOLDER%\DbSql.xml
50
51   echo Waiting about 10 secs before next step...
52   ping -n 10 localhost >NUL: 2>&1
53
54   :: 4 - X3 APPLICATION
55   java -jar %ISO_INSTALL_PATH%\Application\%APPLICATION_JAR% %INSTALL_SCRIPT_FOLDER%\Application.xml
56
57   echo Waiting about 10 secs before next step...
58   ping -n 10 localhost >NUL: 2>&1
59
60   :: 5 - RUNTIME
61   java -jar %ISO_INSTALL_PATH%\Runtime\%RUNTIME_JAR% %INSTALL_SCRIPT_FOLDER%\Runtime.xml
62
63   echo Waiting about 10 secs before next step...
64   ping -n 10 localhost >NUL: 2>&1
65
66   :: 6 - SYRACUSE
67   java -jar %ISO_INSTALL_PATH%\SyracuseServer\%SYRACUSE_JAR% %INSTALL_SCRIPT_FOLDER%\Syracuse.xml
68
69   echo Waiting about 10 secs before next step...
70   ping -n 10 localhost >NUL: 2>&1
71
72   :: 7 - CONSOLE
73   java -jar %ISO_INSTALL_PATH%\Console\%CONSOLE_JAR% %INSTALL_SCRIPT_FOLDER%\Console.xml
74
75   echo Waiting about 10 secs before next step...
76   ping -n 10 localhost >NUL: 2>&1
77
78   :: 8 - PRINTSERVER
79   java -jar %ISO_INSTALL_PATH%\PrintServer\%PRINTSERVER_JAR% %INSTALL_SCRIPT_FOLDER%\Print.xml
80
81   echo Check above if all components installed successfully.
82   pause
83
84   :EXITSCRIPT
```

Sage

# Prerequisites Overview for Install

- **Windows users need to have the correct rights**

    - Log on as a service – Allow running of a program as a service on the system. To allow software or services that need to run continuously in the background, even when no user is logged in.

    - Act as part of the OS – Account can perform tasks that are normally restricted to the operating system. Used by system-level processes and services that need to interact with system resources and perform critical operations.

    - Full access to the Installation directory

- **Java 8 needs to be installed with Java home set so that the .jar install files can execute.**

    https://www.azul.com/downloads/?version=java-8-lts&os=windows&package=jdk#zulu

- **Microsoft PowerShell version 7.2 minimum needs to be installed command-line shell and scripting language**

    https://learn.microsoft.com/en-us/powershell/scripting/install/installing-powershell-on-windows?view=powershell-7.3&viewFallbackFrom=powershell-7&WT.mc_id=THOMASMAURER-blog-thmaure

- **SQL Server module for PowerShell needs to be installed 21.1.18256**

    https://www.sagecity.com/gb/sage-x3/f/announcements/199626/alert-sql-server-powershell-module

- **Microsoft .NET Framework 3.5 and 4 Required for the Sage X3 Console & Print Server**

- Refer to the prerequisites for more details General technical prerequisites for installing Sage X3 V12 components (sageerpx3.com)

# Overview of the Settings

**Server Setup**

- Create Users with appropriate permissions

- Create an Install folder with the appropriate permissions for install users

- Copy the Automated install files to the server and the Installation ISO

**Automated installation files**

- Set server name in automatic installation files default is X3ERPV12SQLVM

- Set the parameters in the automatic install files (Installation directories, service users, passphrases etc)

- Set path to Syracuse license file

**Automated install Script -** AutomateInstall.cmd

- Amend the file AutomateInstall.cmd with the ISO Install path default  E:\X3Installs

- Set the Jar file names for the version you are installing in automated_install.cmd

- Set the location of the Automatic install files

# Demo

- Setup of Server
  - Host name
  - Java Home
  - PowerShell & SQL Module
- Setup of automatic install scripts
- Setup of automatedinstall.bat file

# Limitations

- If the Installation dialogues (panels) change, the Automated install scripts will no longer work

- Familiarity with XML and the required parameters is required if building your own XML files

- Passwords cannot be encrypted in the automatic install files, so the files should be secured.

# Script the Sage X3 Console Configuration

**launchersolution.exe**

# Console Configuration

When an installation is completed, the next step is to create your solution and configure the technical components. Runtime server component & and application server component and initiate the Sage X3 database.
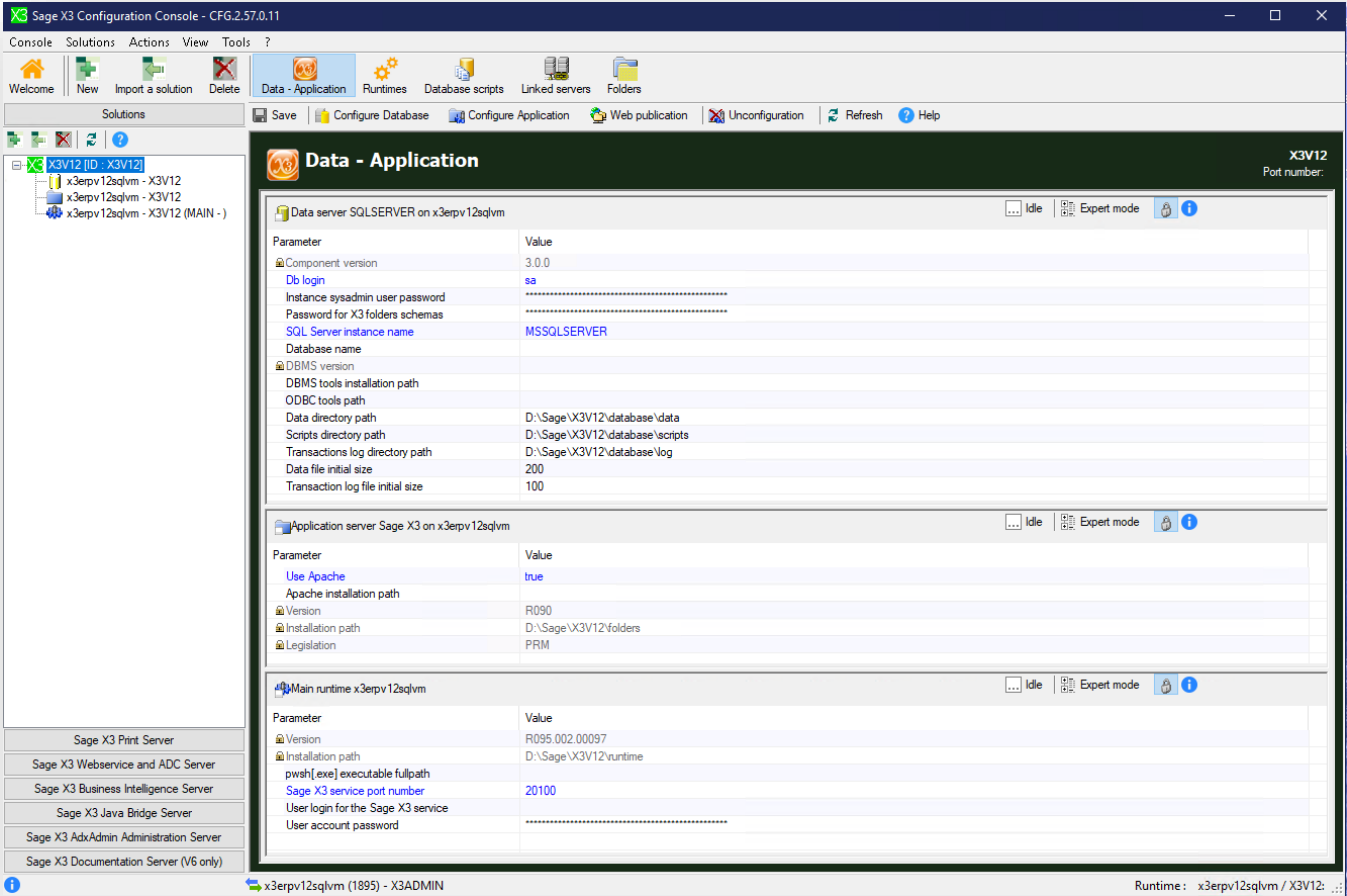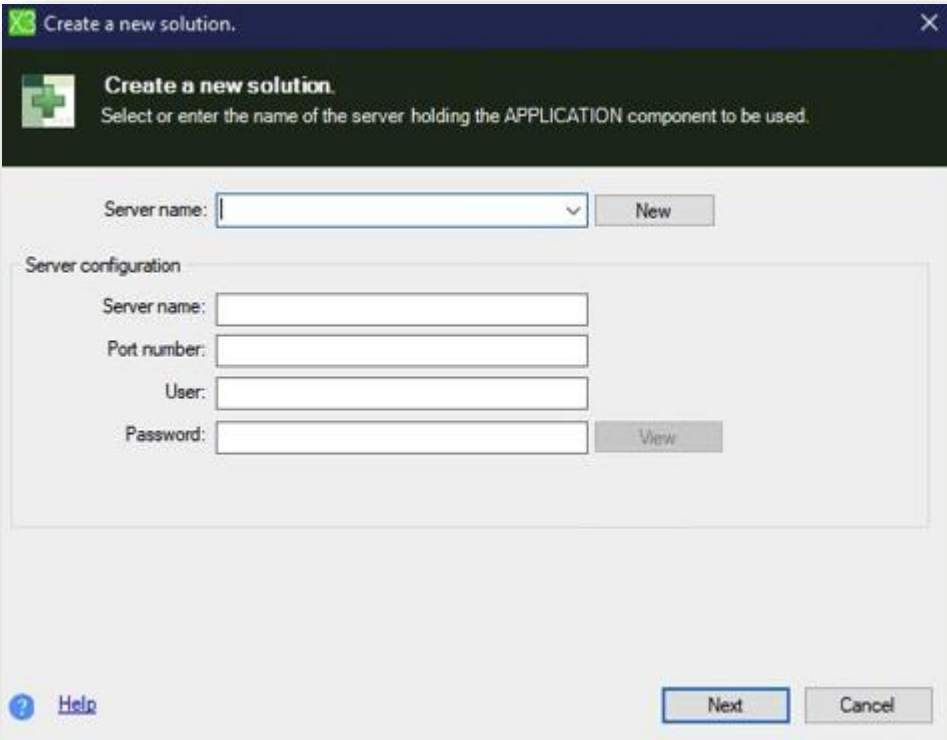
# LauncherSolution.exe
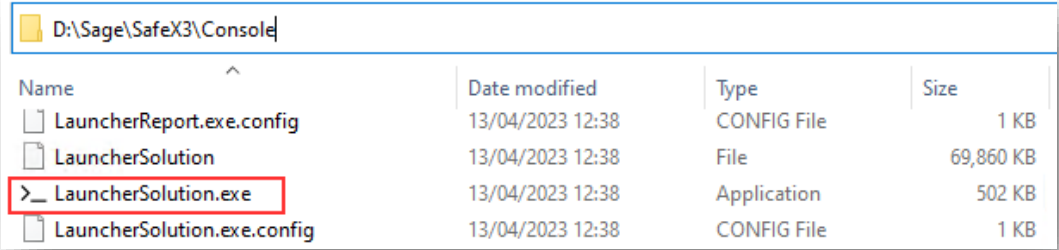
Launchersolution.exe is a command line tool that allows you to launch actions in the console without starting the graphical user interface. The program uses XML template files that contain component definitions and the configuration

- **Create Sage X3 Solution**

- **Database configuration**

- **Runtime configuration**

- **Application configuration**

It can also be used to script folder tasks
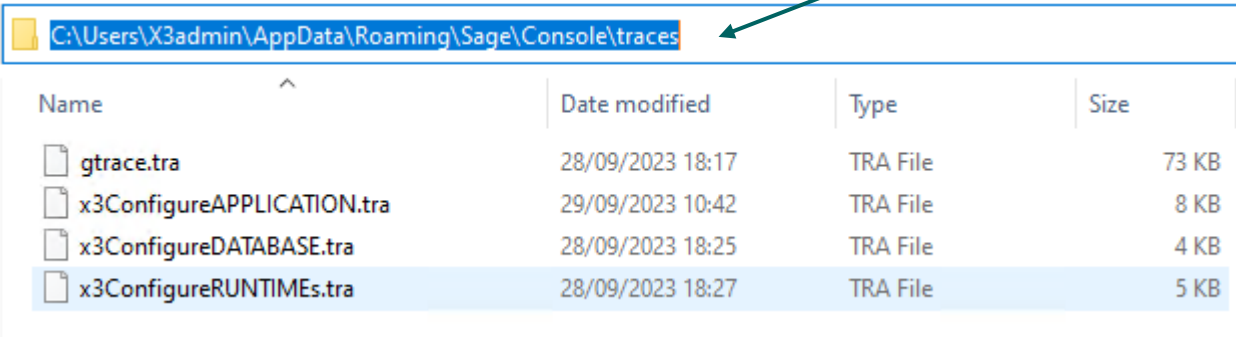
- **Folder Import**

- **Folder Export**

- **Copying existing Folder**

Location of launchersolution.exe
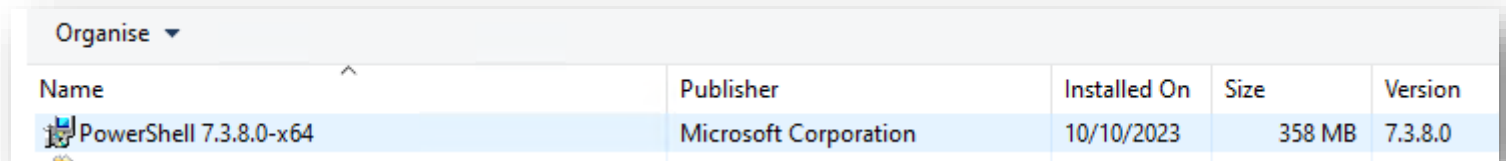


Traces are created for each action in this location
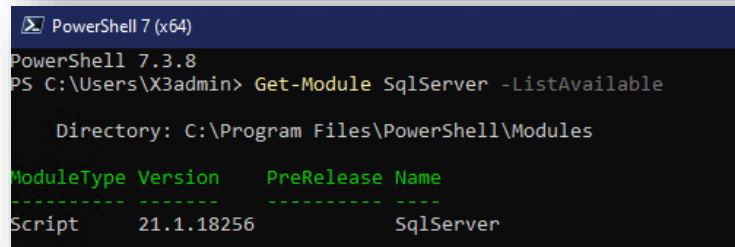
# Prerequisites for Console Config

- SQL server installed with an SQL user that can be used for the configuration of the database (SQL sa recommended)

- Windows users need to have the correct rights as the configuration creates services (runtime)

    - Log on as a service – Allow running of a program as a service on the system. To allow software or services that need to run continuously in the background, even when no user is logged in.

    - Act as part of the OS – Account can perform tasks that are normally restricted to the operating system. Used by system-level processes and services that need to interact with system resources and perform critical operations.

- PowerShell 7.2 (minimum version)



- PowerShell module for SQL 21.1.18256

- If in a multi-server environment, the appropriate ports need to be open.

# Creating the Solution

Using launchersolution.exe, we can use the command prompt to create the Sage X3 solution entry
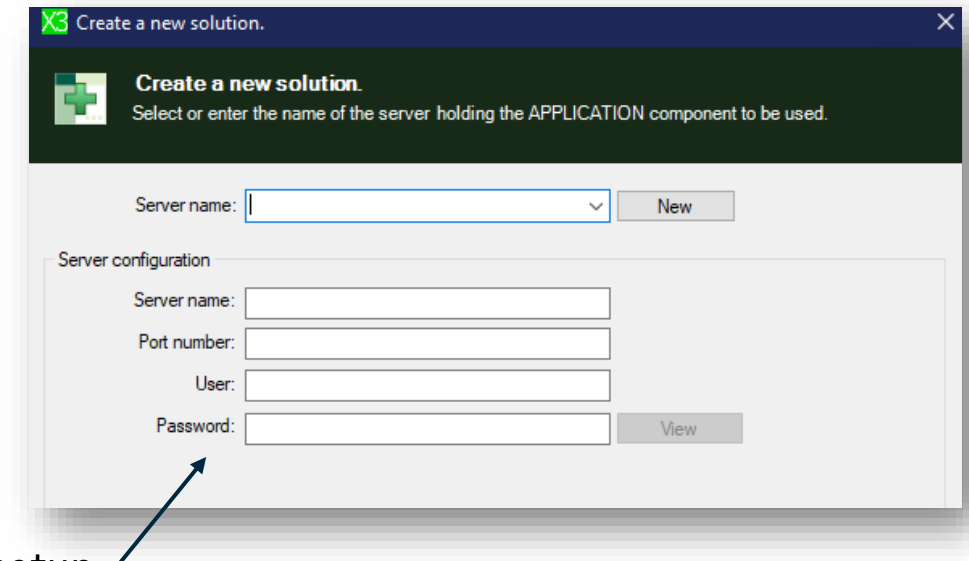
```
D:\Sage\SafeX3\Console\LauncherSolution.exe
/target:"C:\X3InstallScripts\ConsoleConfig\x3ConfigureSOLUTION.xml"
/solutioninfo:'name=X3V12;label=X3_Solution;servername=x3erpv12sqlvm;configpath=C:\Sage\X3V12\folders;'
/account:'host=x3erpv12sqlvm;user=.\x3admin;pass=s@ge2023;port=1895;' /create
```

Path to launcher solution.exe

Target points to the solutiontemplate.xml that will have the solution config

Solution info contains the server hosting the AdxAdmin component (component name, server name, folders path)

The account Section contains the hostname, windows user credentials and port to connect to AdxAdmin

GUI Equivalent solution setup

# Creating the Solution

When the creation is successful this is what you would see in your command prompt window





Exit code 0 tells that the command was successful

**Other error codes** C:\Users\X3admin\AppData\Roaming\Sage\Console\traces

# Database configuration

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"
-target:'C:\X3InstallScripts\ConsoleConfig\x3ConfigureDATABASE.xml' -database:3
-account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'
-logfile:'x3ConfigureDATABASE.tra'
```

# Runtime configuration

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"
-target:'C:\X3InstallScripts\ConsoleConfig\x3ConfigureRUNTIMEs.xml' /main
-account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'
-logfile:'x3ConfigureRUNTIMEs.tra'
```
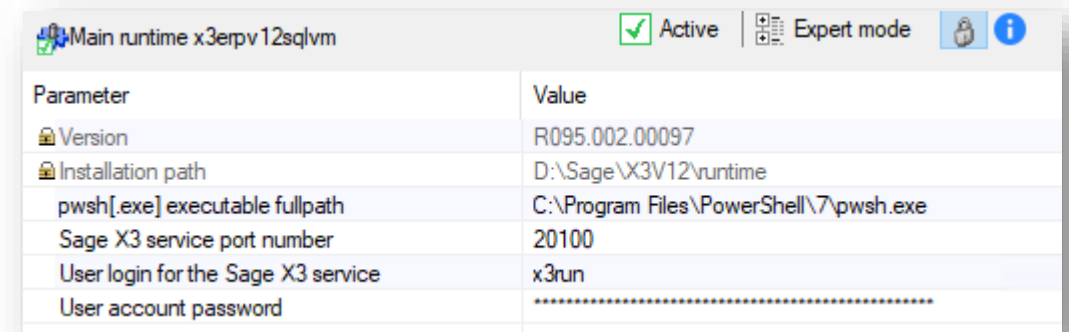
Specify the runtime tag.

# Application configuration

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"
-target:'C:\X3InstallScripts\ConsoleConfig\x3ConfigureAPPLICATION.xml'
-application ^ -account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'
-logfile:'x3ConfigureAPPLICATION.tra'
```

# Complete Configuration

We can also configure the technical components using a single script.

- Note the order of the components

- Status will be active

- Logs – X3admin\AppData\Roaming\Sage\Console\traces

# Generating the Scripts

You can put together your scripts manually following the console help. But there is also a method of generating the scripts from a preconfigured solution, this involves launching the console in debug mode which makes the generate scripts options available in the action menu.

Debug mode – D:\Sage\SafeX3\Console\Console.exe –debug
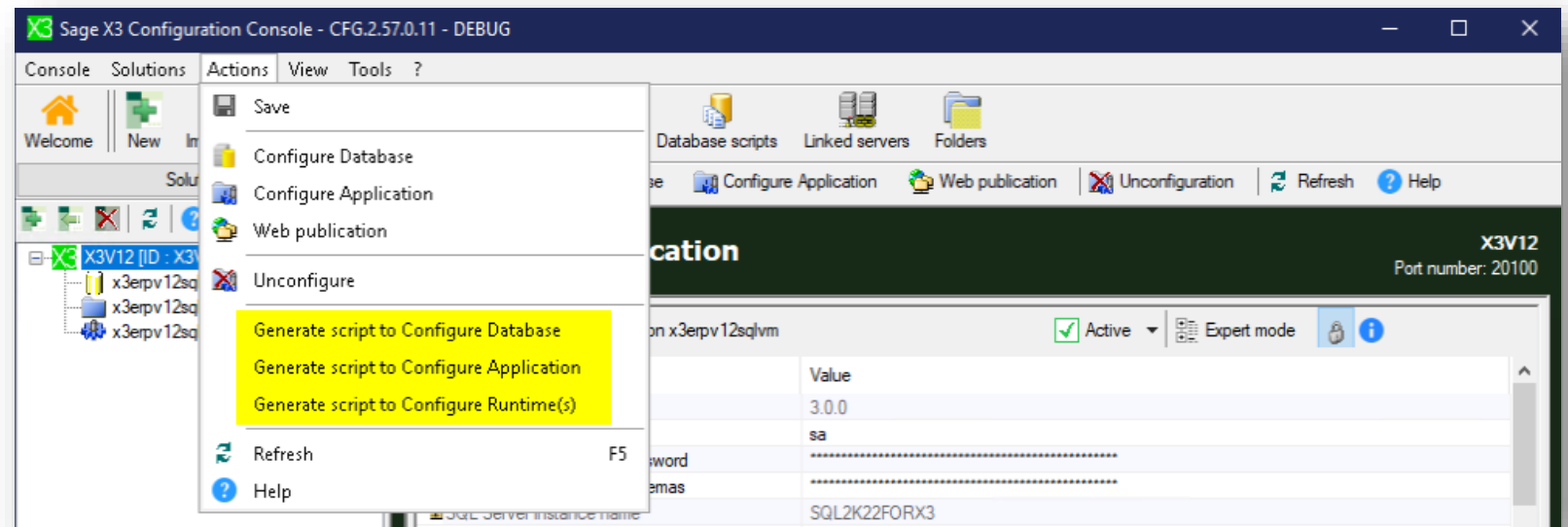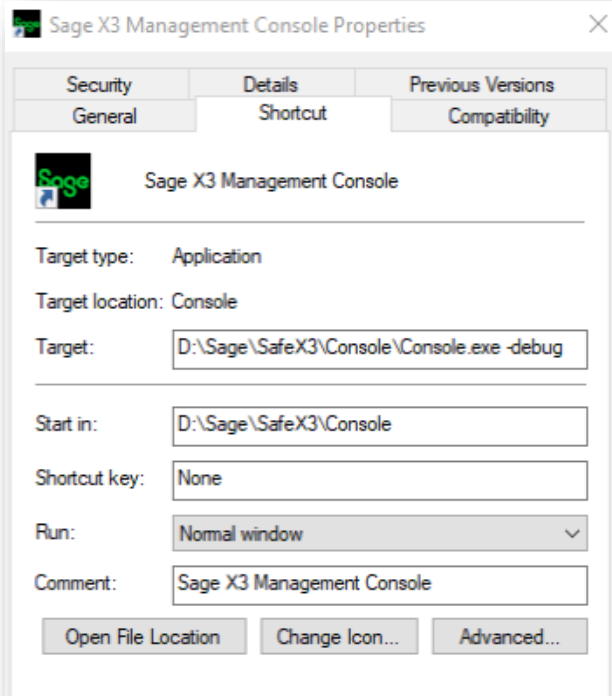
# Demo

1. Show the scripts

2. Create solution

3. Create database

4. Configure application

# Scripting Folder Tasks

## launchersolution.exe

# Scripting of Folder Tasks
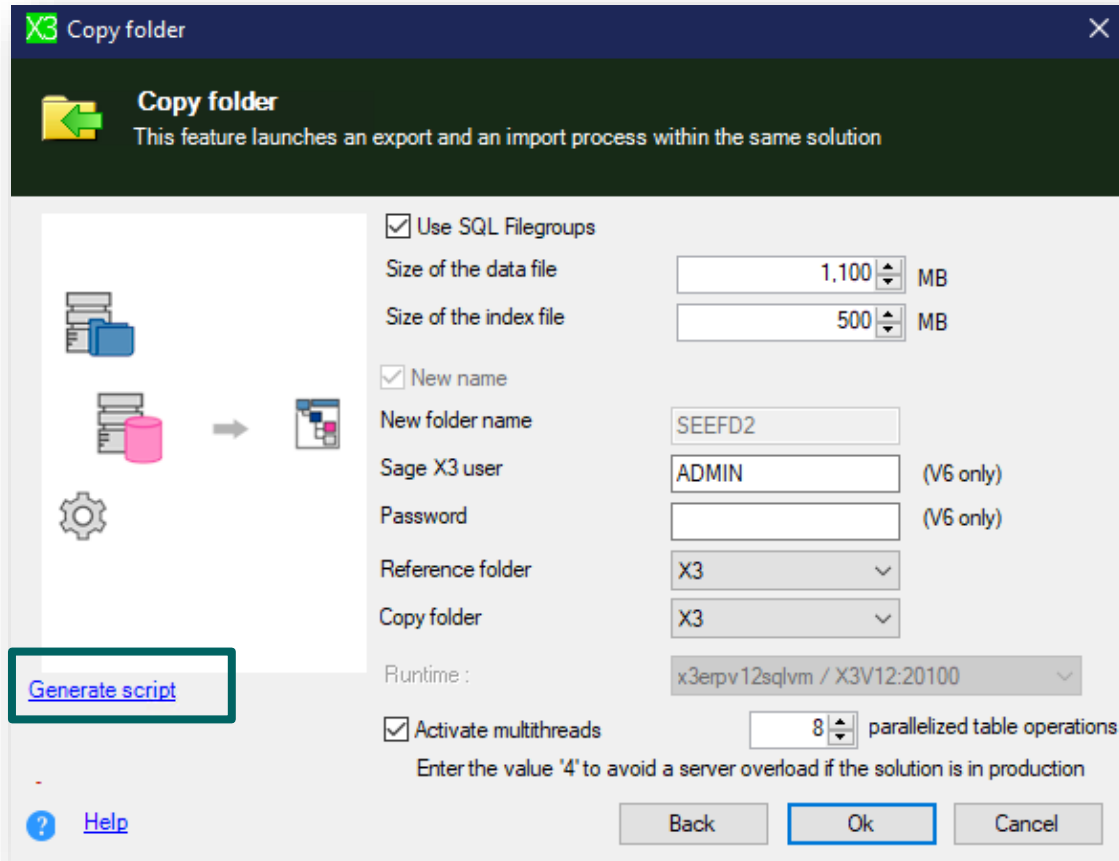
With Launchersolution.exe, we can also complete the common folder tasks using the command line interface.
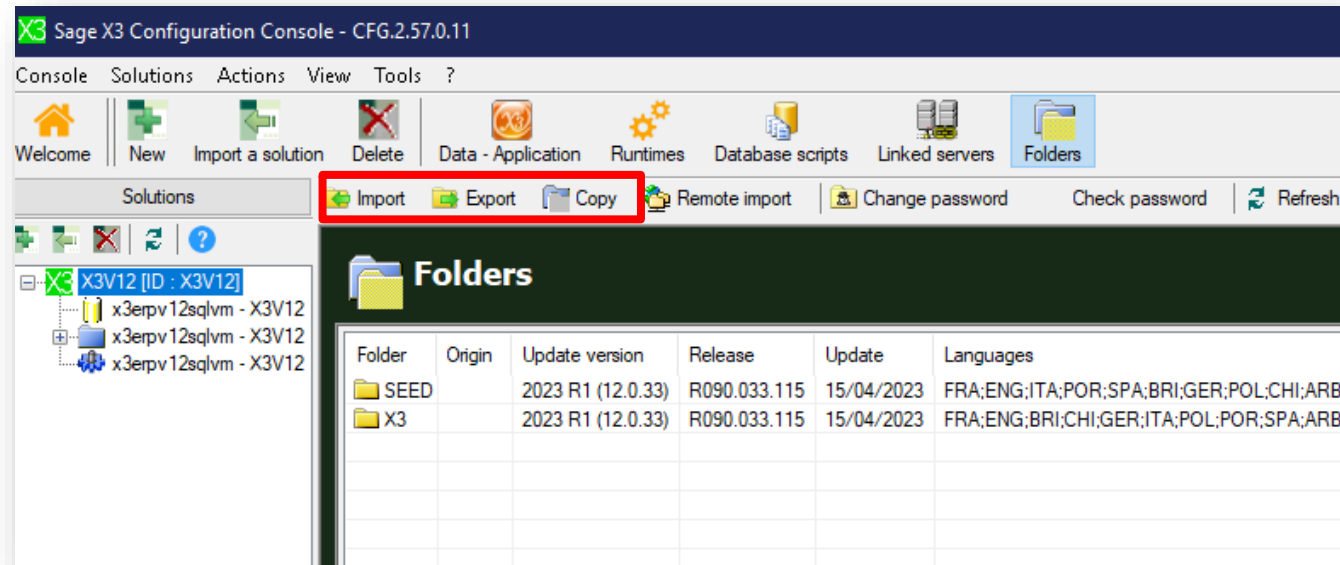


- Import folder data – The folder is the container or location where data files are stored. Importing folder data involves selecting and processing these files to update or add information to your Sage X3 database.

- Exporting folder data – Process of exporting data from the Sage X3 database and saving it in an external folder or directory

- Copy the existing folder – This function launches an export and then import process within the same solution.

# Generating the Scripts

The Scripts for folder tasks can be generated from the console (2.57+). The script will contain all the settings you have specified on-screen, as well as the credentials used to connect to the console.



- Populate the settings
- Generate script
- Paste the script into Notepad++ to amend it accordingly

# Export Folder

Example script exporting the SEED Folder to SEEDSVGBPC

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"  ^
-solutioninfo:'name=X3V12;configpath=D:\Sage\X3V12\folders;servername=x3erpv12sqlvm;'  ^
-export:'folder=SEED;svg=SEEDSVGBCP;deletesvgfolder=1;nodata=0;usedbimport=1;multithreads=1;maxthreads=8;'  ^
-account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'  ^
-logfile:'exportfolder.tra'
```

- Path to the program
- Solution details
- Export folder settings
  - Folder Name to export
  - SVG Directory Name
  - Delete Existing SVG 1/0
  - Export data 1/0
  - Use BCP 1/0
  - Multithread settings
- AdxAdmin connection settings
- Trace File name



Traces are created for each action in this location

C:\Users\X3admin\AppData\Roaming\Sage\Console\traces



Export folder SEED
Export folder SEED
Please confirm the backup folder name

SVG directory          SVGSVGBPC                    ...

☐ Do not export the data                              ❓ Help

☑ Export data to SQLServer native dump files
  This function uses Bulk Copy (BCP) utility on MS SQLServer

☑ Delete the SVG directory before export, if it exists

Runtime :              x3erpv12sqlvm / X3ERPV12RUN:2010C ⌄

☑ Activate multithreads          8 ⬍  parallelized table operations
  Enter the value '4' to avoid a server overload if the solution is in production

Generate script
Script copied in clipboard              Ok      Cancel

# Importing Folder

Example script for exporting the SEED Folder

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"  ^
-solutioninfo:'name=X3V12;configpath=D:\Sage\X3V12\folders;
servername=x3erpv12sqlvm;'^
-folder:'folder=SEED;svg=SVG;usegrp=1;data=1100;index=500;usedbimport=1;
originalsize=0;dosref=X3;doscop=X3;multithreads=1;maxthreads=8;'^
-account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'^
-logfile:'importfolder.tra'
```

- Path to the program
- Solution details
- Import folder settings
  - Folder Name
  - SVG Directory Name
  - Data tablespace settings
  - Multithread options
- AdxAdmin connection settings
- Trace file name

# Copy Existing Folder

Example script copying the SEED folder to SEED2 using SVG SVG241023

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"  ^
-solutioninfo:'name=X3V12;configpath=D:\Sage\X3V12\folders;servername=x3erpv12sqlvm;'  ^
-copyfolder:'sourcefolder=SEED;destfolder=SEED2;svg=SVG241023;dosref=X3;doscop=X3;usedbexportimport=1;usegrp=1
;data=1100;index=500;multithreads=1;maxthreads=8;'  ^
-account:'host=x3erpv12sqlvm;user=x3admin;pass=s@ge2023;port=1895;'^
-logfile:'copyfolder.tra'
```

- Path to the program
- Solution details
- Copy folder settings
  - Source Folder Name
  - Destination Folder name
  - SVG Directory Name
  - Data tablespace settings
  - Use BCP 1/0
  - Multithread options
- AdxAdmin connection settings
- Trace file name
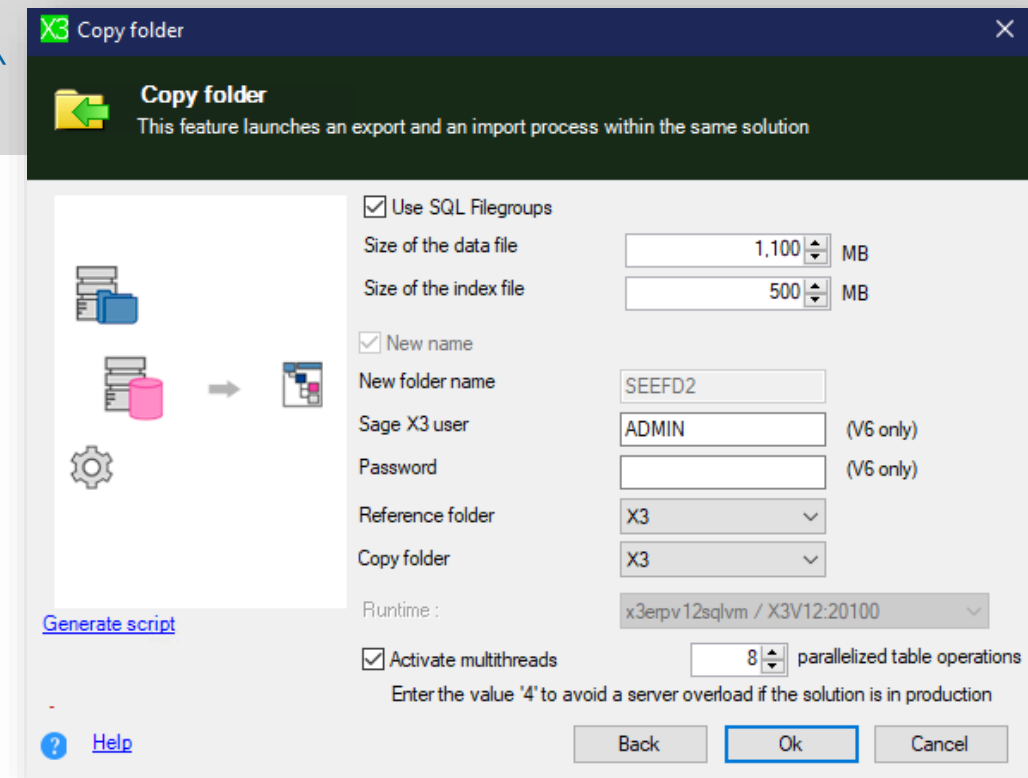
# Encrypt the Password

When the wizard generates the scripts, it also provides the command so you can also encrypt the password for example

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe" -cryptpassphrase:s@ge2023 -logfile:'encrypt.tra'
```

This will return an encrypted value for your password which you can use in your scripts

```
C:\Users\X3admin>"D:\Sage\SafeX3\Console\LauncherSolution.exe" -cryptpassphrase:s@ge2023  -logfile:'encrypt.tra'
Running on Microsoft Windows Server 2022 Datacenter / 10.0.20348
Total Memory : 8089 MB / Free Memory : 2727 MB
Processor : 2 Core(s)  64-bit
MS .Net Framework : 4.8
Machine Name : X3ERPV12SQLVM
CurrentCulture : en-GB - CurrentUICulture : en-GB
23/10/2023 16:52:53 INFO - Sage X3 Configuration Console CFG.2.57.0.11 23/10/2023 16:52:53
Encrypted password:
CRYPT:teueQrtvaXvavdoxvascc   ◄─────────
Program finished.
 Exit code = 0 - SUCCESS
```

The encrypted password can then be used in your script

```
"D:\Sage\SafeX3\Console\LauncherSolution.exe"  ^
-solutioninfo:'name=X3ERPV12;configpath=D:\Sage\X3ERPV12\folders;servername=x3erpv12sqlvm;'  ^
-export:'folder=SEED;svg=SVG;deletesvgfolder=1;nodata=0;usedbimport=1;'  ^
-account:'host=x3erpv12sqlvm;user=x3admin;pass=CRYPT:teueQrtvaXvavdoxvascc;port=1895;'  ^
-logfile:'exportfolder.tra'
```

# Limitations

Limitations of using launchersolution.exe as of 2023 R1

- As of 2023 R2, it is not possible to configure the print server from the command line interface. We are expecting a tool to be available in the future (laucnchereport.exe)

- Folder import can only be done once the supervisor has been initialised from Syracuse

  - Setup solution in X3

  - X3 endpoint

  - Menu & personalization

  - Access classic function

- Multi threading is only available from Sage X3 management Console version 2.57

# Useful Links

| Resource | Link |
| --- | --- |
| Sage X3 Release notes | All V12 release notes |
| Prerequisites information for V12 | Prerequisites overview |
| Technical online help | Technical help |
| Sage X3 Console Help | Console Help |
| Test System Build Diaries | Test System Build Diaries |
| | |
| | |

# Thank you