Sage X3 – Housekeeping suggestions

X3 PU9

Disclaimer

This document is provided "as is" and is for your guidance and educational purposes only. It does not replace the Online documentation, nor is any warranty expressed nor implied for the steps described below.

Document Information

Author: Mike Shaw, Sage UK X3 Support Team

Contents

Introduction

This document is designed to give an X3 Applications administrators some ideas as to what housekeeping tasks could be useful to perform

This is not a complete list of all tasks you should perform on your own system, but gives some pointers to common tasks that would generally be recommended by Sage to be performed

As this document is generic, you will need to adapt it for your own situation. If you need help to determine which housekeeping tasks are relevant for your specific site, you could consider engaging our Professional Services Group (PSG) to assist

Confirming X3 version information

A common question when requesting help from Sage Support will be “What version of X3 are you using?”

There are various different components that make up your X3 system, but the most important to know the versions for are:

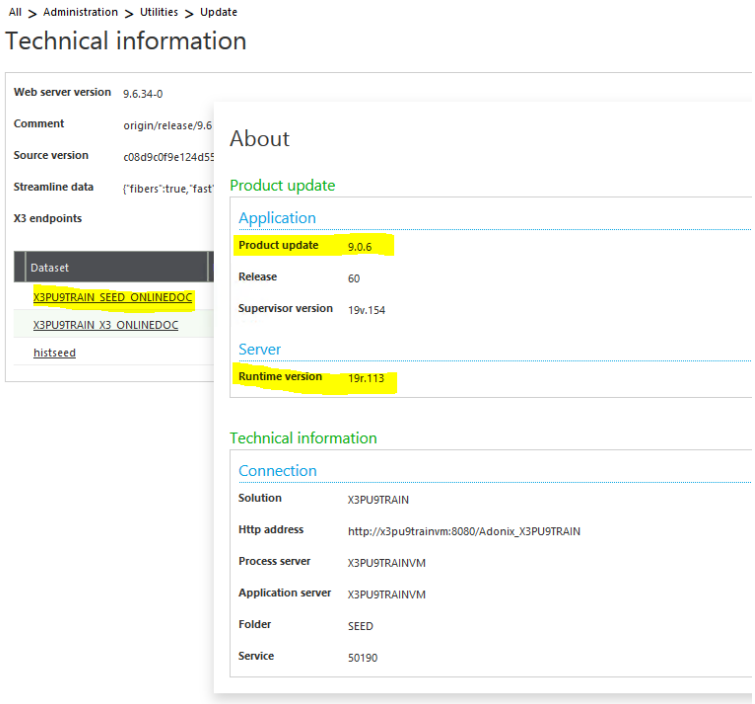
a. X3 Patch level
b. Runtime
c. Syracuse Web Server

Luckily you can confirm the versions for all three of these from one screen within X3 itself

Navigate to Administration→ Utilities→ Update→ About

On this first screen you can see the “Web Server” (i.e. Syracuse) version, in this case 9.6.34-0

Each X3 folder could potentially have a slightly different version, although for most customers this is not the case. You can check each folder individually by clicking the link for the relevant folder. For example, click the “X3PU9TRAIN_SEED_ONLINEDOC” folder name to get the X3 folder version and runtime version information



Here we can see the “Product Update” version is 9.0.6 which is the X3 folder version, also called the “patch level” (the third digit indicates Patch 6 has been applied in this folder)

The “Runtime version” is 19r.113 in this example

Setting up separate users for Batch Server and Web Services

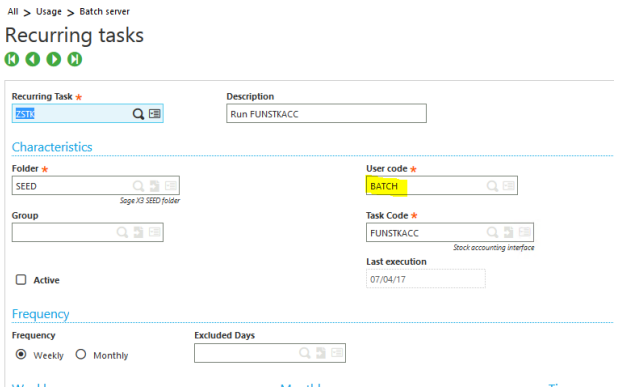
When setting up Web Services and Batch Jobs, it is tempting to just set these up using the ADMIN user, as it's quick and easy to do so. Whilst this will work, you are giving your batch users and web service users the same security equivalence as your Administrator (i.e. access to everything) and also may run into situations where changes to the ADMIN user will effect these other services. It is best practice to setup new users to be used for batch server and also web services. Indeed, you may decide to have multiple users in both categories, if it better suits your business requirements

The steps are:

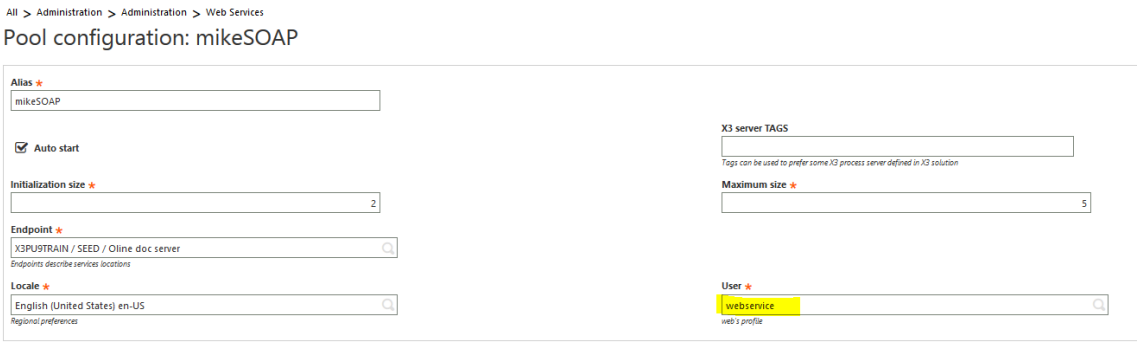
1. Setup new users as required in Administration→ Administration→ Users→ Users with the relevant permissions

2. Create corresponding users for the relevant folders, Parameters→ Users→ Users with the relevant permissions

3. For Batch Server, navigate to Usage→ Batch Server→ Recurring Task Management and set the User Code to your batch user

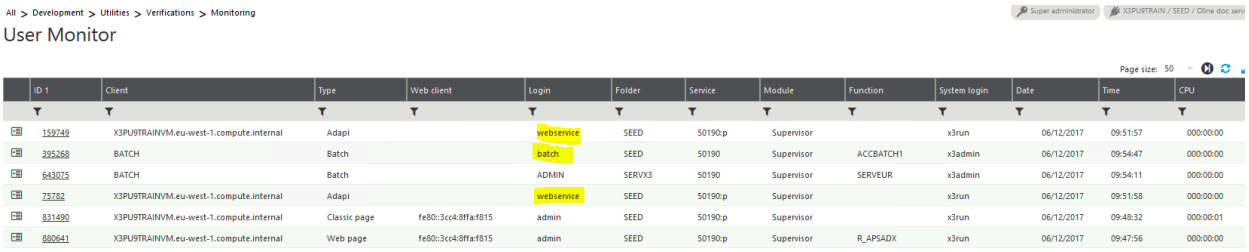


4. For Web Services, navigate to Administration→ Administration→ Web Services→ Classic SOAP pools configuration and set the "user" to your web services user. NOTE: the Web Service user (and language) is just a default setting. When calling web services, the calling program may well use a different user and/or language setting



Now when you want to confirm which users are using the system, you can easily see which users are the Web Services or Batch server users, as well as being able to better control and manage the permissions and accesses for the users using these functions

For example: Navigate to Development→ Utilities→ Verifications→ Monitoring→ User Monitoring



ID 1	Client	Type	Web client	Login	Folder	Service	Module	Function	System login	Date	Time	CPU
159749	X3PU9TRAINVM.eu-west-1.compute.internal	Adapi		webservice	SEED	50190:p	Supervisor		x3run	06/12/2017	09:51:57	000:00:00
395268	BATCH	Batch		batch	SEED	50190	Supervisor	ACCBATCH1	x3admin	06/12/2017	09:54:47	000:00:00
643075	BATCH	Batch		ADMIN	SERVX3	50190	Supervisor	SERVEUR	x3admin	06/12/2017	09:54:11	000:00:00
75782	X3PU9TRAINVM.eu-west-1.compute.internal	Adapi		webservice	SEED	50190:p	Supervisor		x3run	06/12/2017	09:51:58	000:00:00
831490	X3PU9TRAINVM.eu-west-1.compute.internal	Classic page	fe80::3cc4:8ffa:f815	admin	SEED	50190:p	Supervisor		x3run	06/12/2017	09:48:32	000:00:01
880641	X3PU9TRAINVM.eu-west-1.compute.internal	Web page	fe80::3cc4:8ffa:f815	admin	SEED	50190:p	Supervisor	R_APSADX	x3run	06/12/2017	09:47:56	000:00:00

Archive / Purge

Some data is useful for only a finite period of time, such as log files or temporary tables, other data is needed long term but is perhaps only referred to in detail occasionally, such as transactional data from previous financial years. X3 provides the ability to clear out certain data by purging (deleting permanently) and some data can be archived to a separate area, leaving it online for query purposes.

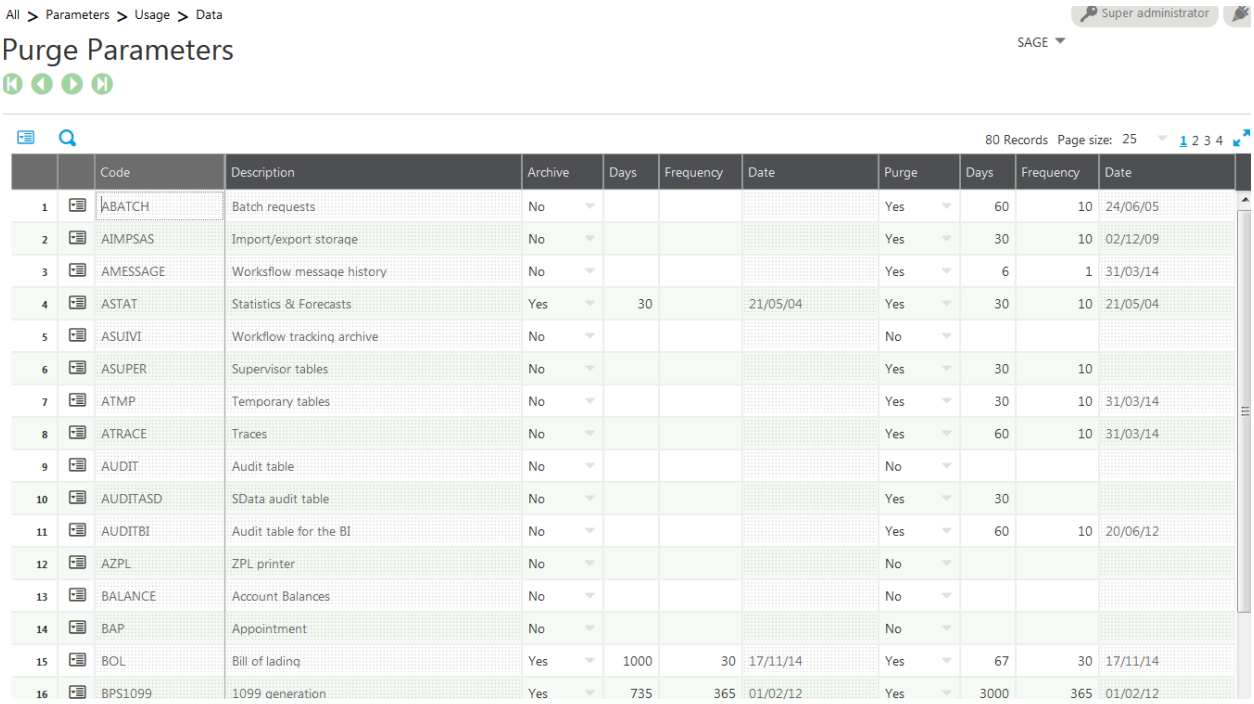
Whilst both purging and archiving are optional, it is prudent to consider using either or both these facilities in order to give best possible performance and to keep disk usage to a minimum

Navigate to Parameters→ Usage→ Data→ Purge Parameters

This shows the available Archive and Purge routines. You should review the list and configure according to your business needs

Only data considered as closed (i.e. that which will not change any more) can be purged or archived

Some routines relate only to purging, some relate only to archiving and some allow both activities



		Code	Description	Archive	Days	Frequency	Date	Purge	Days	Frequency	Date
1		ABATCH	Batch requests	No				Yes	60	10	24/06/05
2		AIMPSAS	Import/export storage	No				Yes	30	10	02/12/09
3		AMESSAGE	Worksflow message history	No				Yes	6	1	31/03/14
4		ASTAT	Statistics & Forecasts	Yes	30		21/05/04	Yes	30	10	21/05/04
5		ASUIVI	Workflow tracking archive	No				No			
6		ASUPER	Supervisor tables	No				Yes	30	10	
7		ATMP	Temporary tables	No				Yes	30	10	31/03/14
8		ATRACE	Traces	No				Yes	60	10	31/03/14
9		AUDIT	Audit table	No				No			
10		AUDITASD	SData audit table	No				Yes	30		
11		AUDITBI	Audit table for the BI	No				Yes	60	10	20/06/12
12		AZPL	ZPL printer	No				No			
13		BALANCE	Account Balances	No				No			
14		BAP	Appointment	No				No			
15		BOL	Bill of lading	Yes	1000	30	17/11/14	Yes	67	30	17/11/14
16		BPS1099	1099 generation	Yes	735	365	01/02/12	Yes	3000	365	01/02/12

You control how long to keep data with the “Days” setting. Any data that is purgeable (closed) and is older than the specified number of days, then qualifying for being purged on the next run

“Frequency” controls the gap between Purges. For example if set to 10 days as shown above, then after a purge run it will not attempt another purge run for another 10 days. It is suggested you set this to 1 day for all tasks you are using, then create a scheduled batch task to perform the purge at an appropriate frequency for your requirements, for example weekly or monthly.

Purging

Purging should be configured for each folder, including the X3 folder itself

You should pay particular attention to the purge jobs starting with “A” as most of these will likely need to be enabled and will apply to most systems and most folders

The ABATCH task is a special case as this applied only to the X3 folder, as it stores the information relating to the Batch Server itself

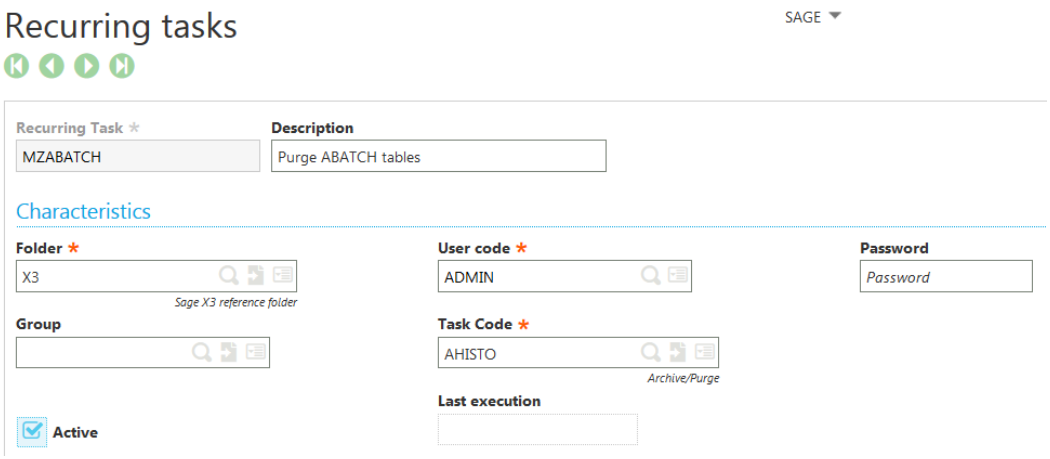
As another example, ATRACE is used to manage the log files which is created by most batch jobs. These files are retained in the folder TRA directory until purged

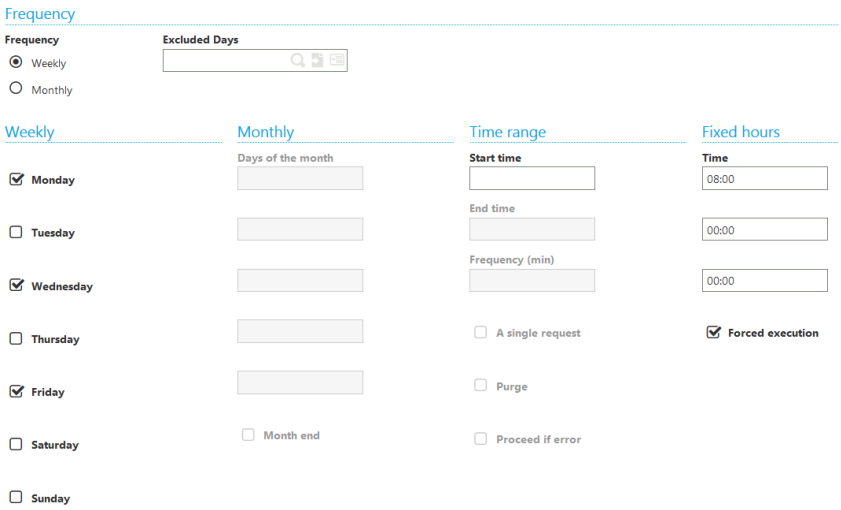
For the purposes of demonstration, this document will show the setup for purging for these two jobs, but you should review and decide which jobs are applicable for your own circumstances

Example of setup for ABATCH task

Log into the X3 Folder itself and then go to option Usage--> Batch Server--> Recurring Task Management (GESABA)

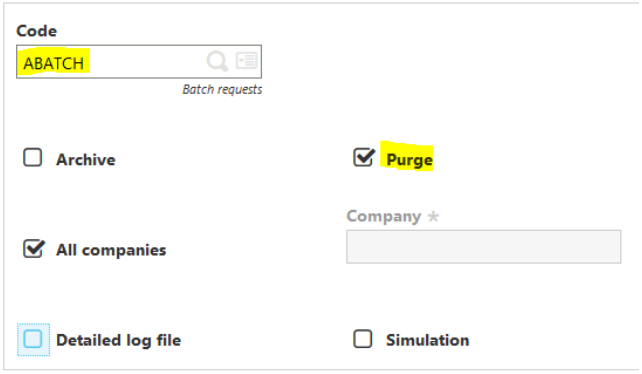
Setup the recurring task as shown below:





- Use the task code AHISTO which is the Archive/Purge code
- Set the task frequency as desired for your requirements
- Selecting "Forced Execution" ensures the job executes when the batch server is re-started and the execution time has lapsed
- Online help is available at http://online-help.sageerpx3.com/erp/9/staticpost/recurring-task-management/
- You will not be able to activate the task until you have defined any required parameters (Parameter Definitions)

Archive/Purge



- Enter the code for the Purge or Archive routine
- Ensure you select "Purge" and / or "Archive" as appropriate
- "Detailed log file" and "Simulation" would not normally be selected

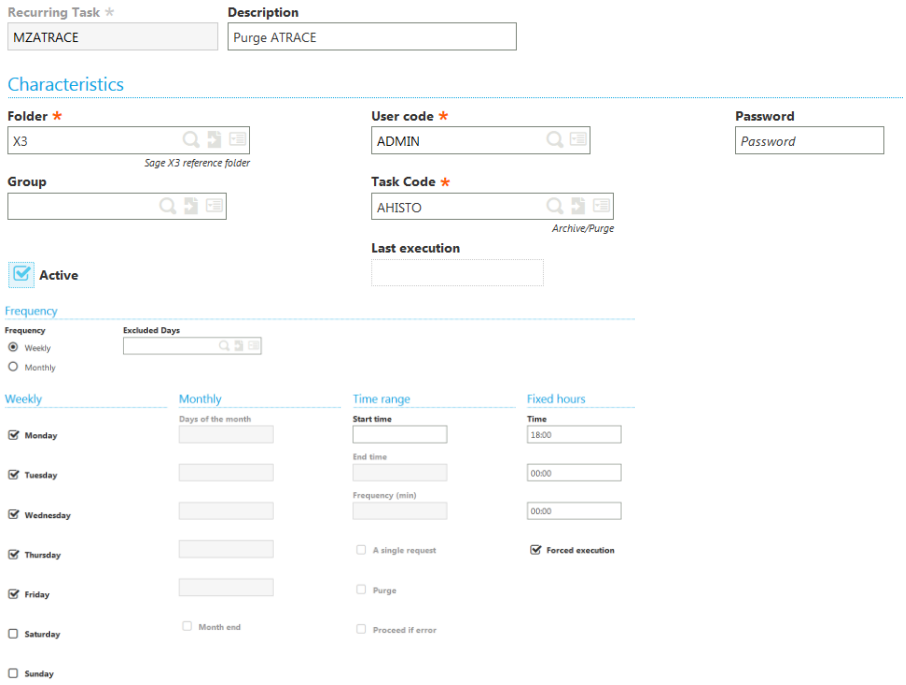
Once the task has been saved, you can see the executions of the task by navigating to Usage→ Batch Server→ Request Management

The recurring task may not appear on the list immediately, as it won't be in the list until the morning of its first run

Example of setup for ATRACE task

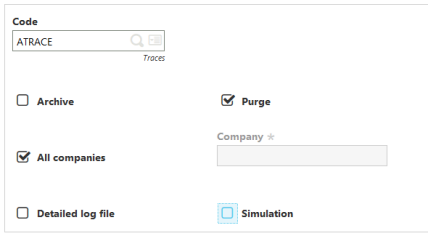
This task relates to each folder individually. For our example, we will setup for the X3 Folder itself
Navigate to option Usage--> Batch Server--> Recurring Task Management (GESABA)

Setup the recurring task as shown below:



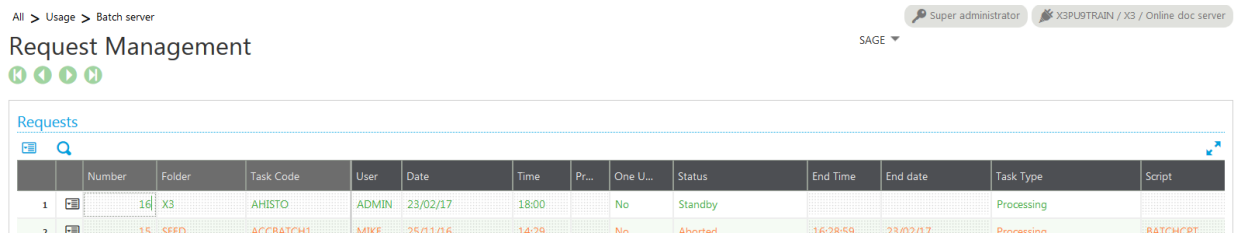
Parameters

Archive/Purge



Repeat these same setup steps for your other folders in this solution

In this case, as the execution is for later on today, we can see the tasks immediately in the Request Management screen



Archiving

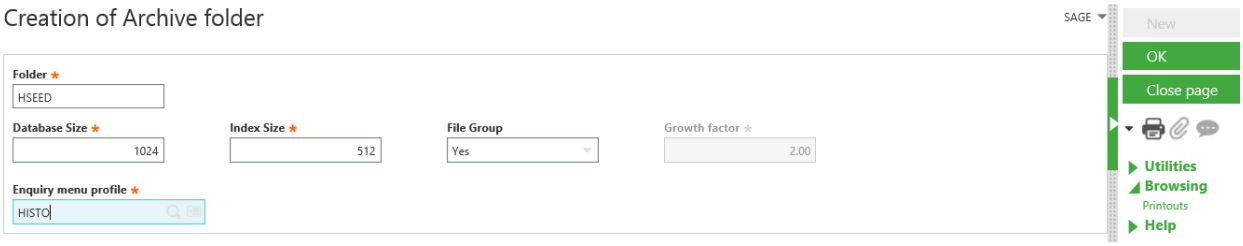
Archiving allows you to retain historical data for read only review purposes in a separate area from the live transactional data. This is useful where you want to retain the historic information, but not want to impact overall system performance by keeping too much data in the main folder itself

You should review your data volumes in conjunction with your business requirements and decide which tables (if any) could or should be archived to achieve these business objectives

Example of set up for Archiving

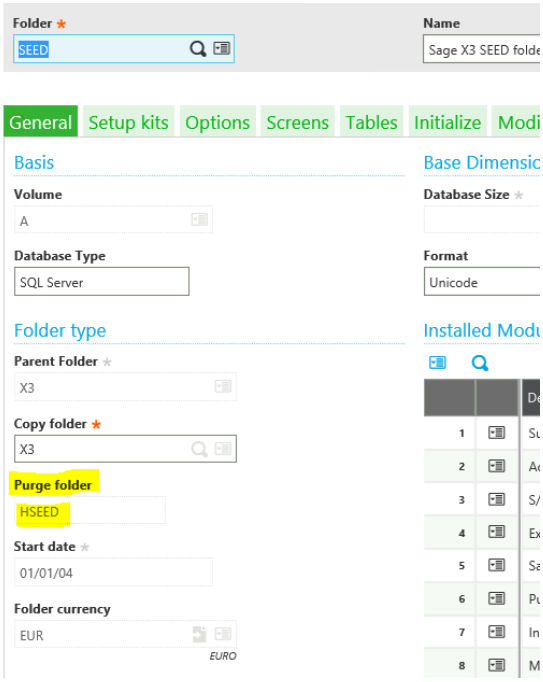
a. Setup Archive folder

Development→ Utilities→ Folders→ Creation of Archive folder

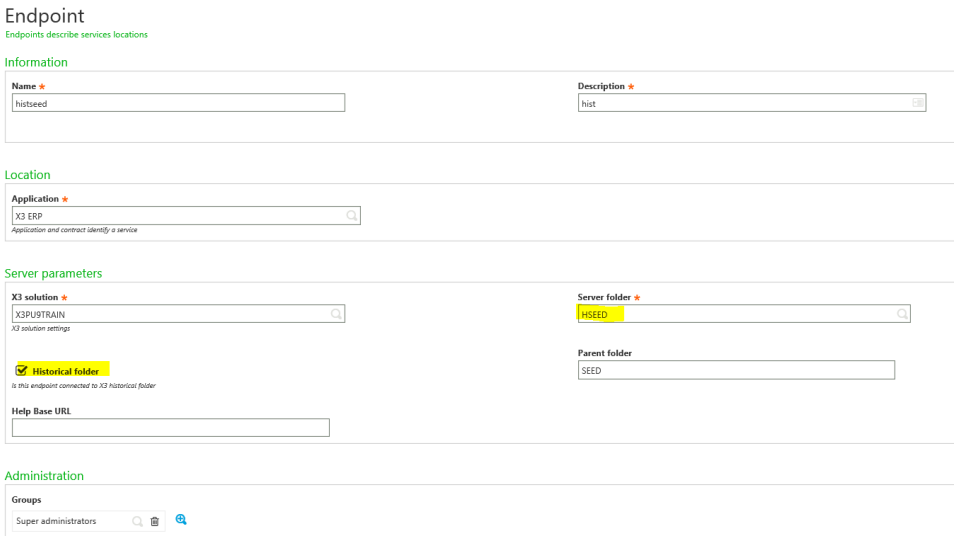


This could take several minutes to complete

Check the Folder setup to confirm after completion. Parameters→ General Parameters→ Folders



Create end point for history folder
Administration→ Administration→ Endpoints



b. Check the Archive parameters

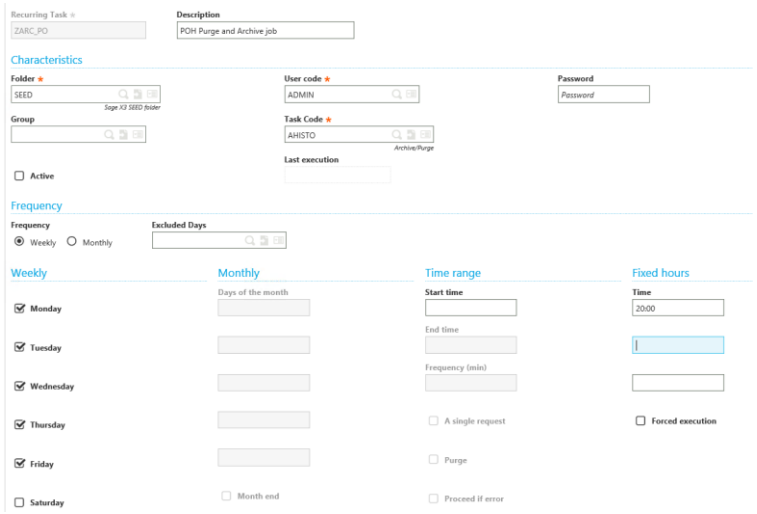
Parameters→ Usage→ Data→ Purge Parameters

See the discussions under “Purge” section above

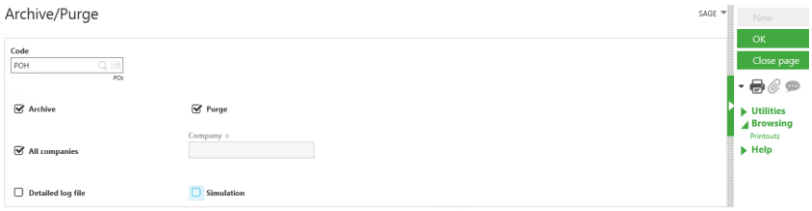
c. Setup Batch Job to run the Archive process

Help on the archive process is available online at http://online-help.sageerpx3.com/erp/9/staticpost/archivepurge/

This task relates to each folder individually. For our example, we will setup for the SEED Folder
Navigate to option Usage--> Batch Server--> Recurring Task Management (GESABA)



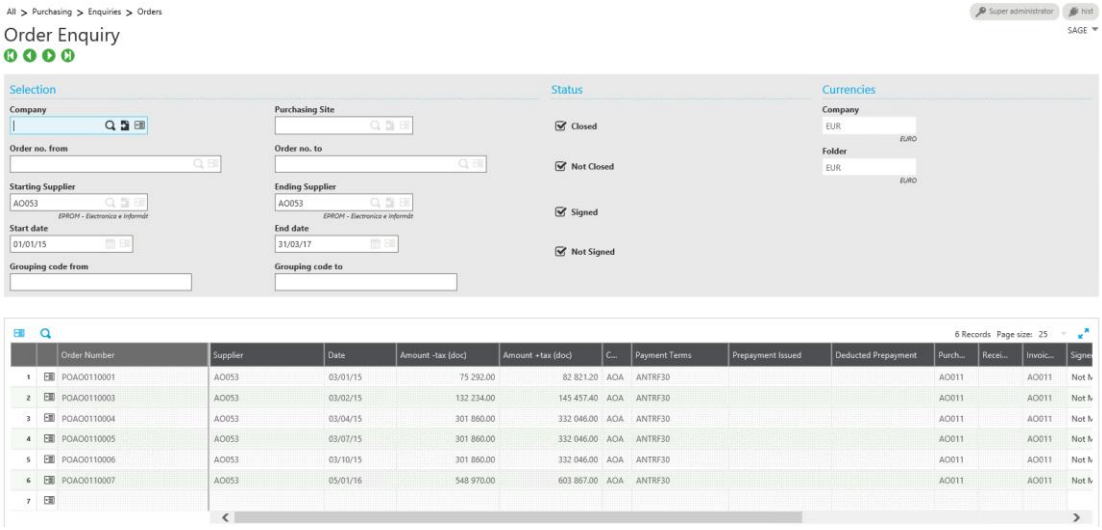
Add parameters



You can run the Archive/Purge interactively, by navigating to Usage--> Usage--> Archive/Purge

d. Check results by doing Enquiry on historic data

Connect to your history folder, then do any queries you are interested in to see the historical data



Batch Server Management

Daily re-start of Accounting Tasks

Sage recommend to schedule a restart of the accounting task every day

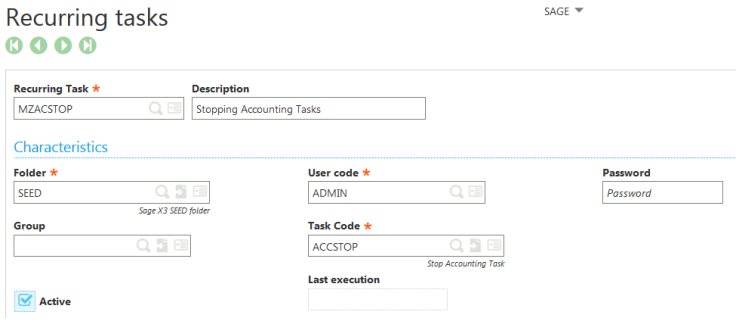
- To stop, use the ACCSTOP batch task
- To start, use the ACCBATCH batch task

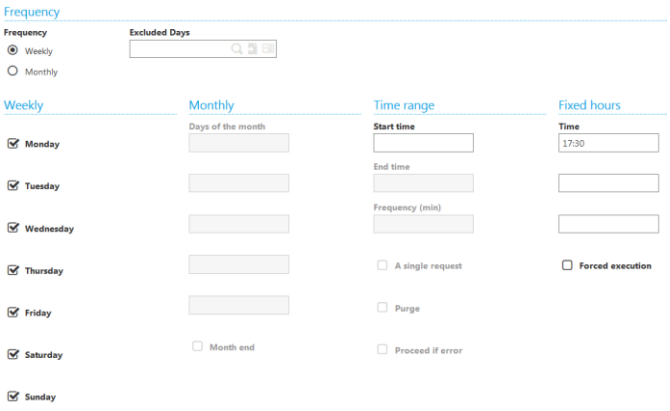
It is IMPORTANT you stop the Accounting Task process before you stop the batch server, and you stop the batch server process itself before a server restart or before shutting down SQL Server. Not doing so can cause issues when trying to restart these processes

Example of setup for ACCSTOP and ACCBATCH tasks

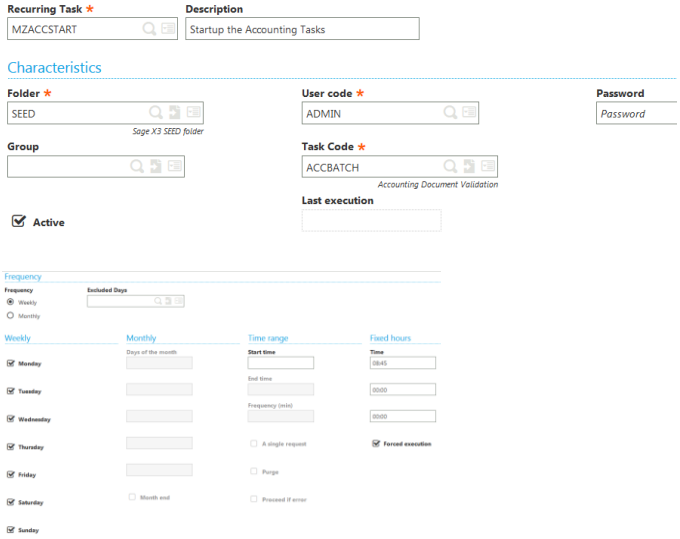
For the SEED Folder
Navigate to option Usage--> Batch Server--> Recurring Task Management (GESABA)

Define recurring job to stop the accounting task





Define recurring job to start the accounting task



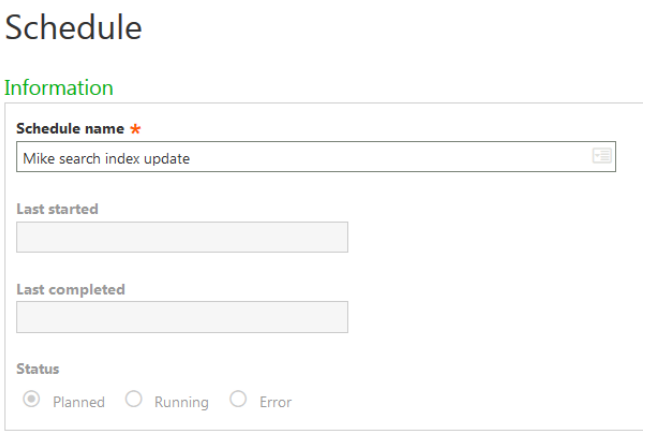
Elastic Search indexes

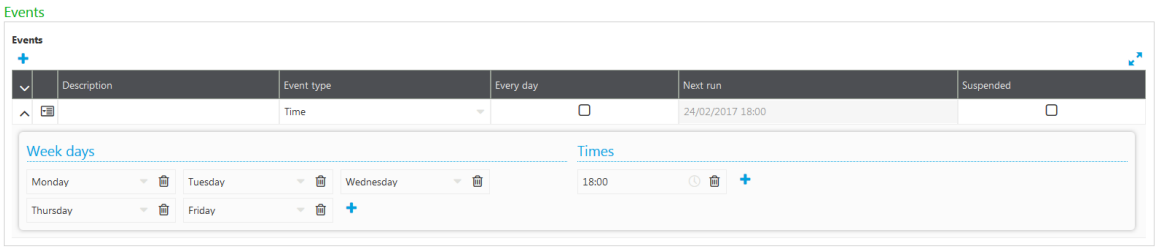
The Elastic Search indexes needs to be regularly updated in order to ensure the search results reflect recently added data

This task can be automated by scheduling to run at specific times. This process uses a different scheduler and is setup as described below:

Create a new schedule

Navigate to Administration→ Usage→ Automate→ Schedule
Click the “New Schedule” button and create a new schedule as shown below, using whichever days/times are suitable for your requirements



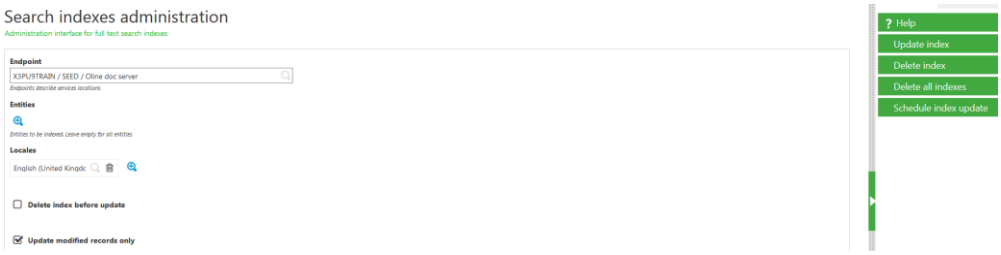


Then click “Save”

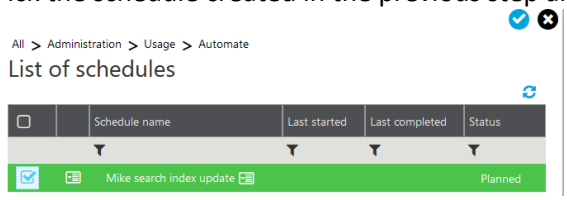
Setup the Elastic Search Index update

Navigate to Administration→ Usage→ Automate→ Search Index Management

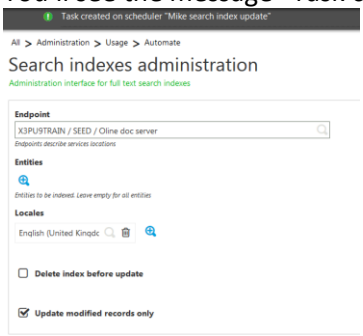
The default settings should be sufficient for the scheduled update, so you can just click the “Schedule index update” option



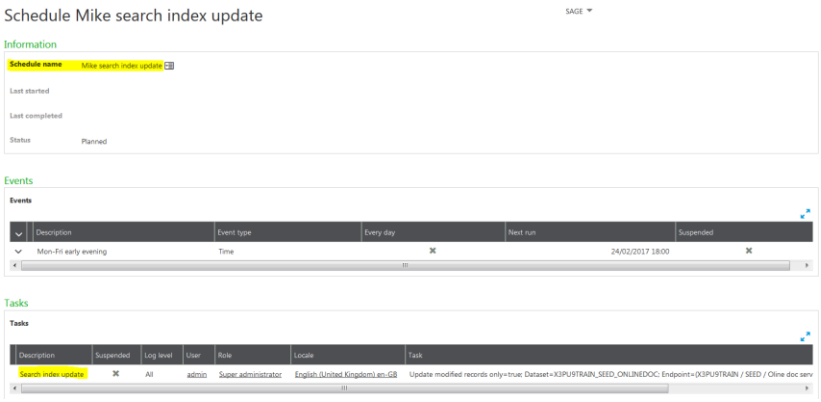
Pick the schedule created in the previous step and click the blue tick to save



You’ll see the message “Task created on scheduler …”



You can also check the tasks for your scheduler to confirm “Search index Update” is shown



Monitor/manage to ensure batch jobs not taking up too much resources

Depending on your users active hours and busiest times, you may want to ensure that some heavy processing batch jobs, such as described in this document, are not executed during the users' main working hours

You should additionally consider when the system backups are taking place as there may be some tasks which should not be run during these times also

You should therefore draw up a list of times during which it is acceptable to run the batch tasks and schedule them accordingly. You can consider if you need to enforce these hours using "Hourly Constraints" and/or "Batch server calendar" for the Batch Server tasks

Navigate to Parameters→ Usage→ Batch Server where you will find these options to allow you to configure allowable dates and allowable days/hours of batch task execution

Once hourly constraints have been configured, you can modify the Task configuration to ensure it conforms. Navigate to usage→ Batch Server→ Task Management and configure the tasks as needed

Backup / Restore

Your backup strategy will need to reflect the Business' Disaster recovery objectives and policies, so the first step is to confirm and understand these objectives and policies, specifically:

Recovery Time Objective (RTO)
How much downtime is acceptable, in other words the time it takes to get the service back to a state where users can login and work normally again after a failure

Recovery Point Objective (RPO)
How much data it is acceptable to lose once recovery has been achieved. In other words, how much work the users will need to redo after a successful recovery

These two items alone should provide a good guide to the type and frequency of backups that need to be taken, in order to satisfy these requirements

If you have a multi-server Sage X3 instance (different X3 components spread out across different servers), you should consider all these servers as one whole in a backup strategy. i.e. you will need to backup all the servers and perhaps also need to synchronize these backups for some servers

The items you need to consider for backup include:

- SQL Server
- MongoDB
- Filesystem files
  - Relatively static data (such as binaries)
  - Regularly changing data (such as log files and Elastic Search index files)
  - Don't forget some X3 specific data is stored under the Windows User home directory, for example some Management Console information
- Windows Server registry entries

You need to ensure you take SQL database backups and SQL Server log file backups such that any business recovery objectives are achieved

Essential configuration data and other user data, such as documents, are stored in the Mongo Database, so you therefore also need to ensure you backup MongoDB database

The file system and Windows registry should also be backed up regularly to ensure you capture regularly changing files such as log files, and maintain backups of relatively static files, after patching for example

Testing the recovery/restore procedures

You should regularly test your recovery processes, which includes restoring the data from backups. This would need to be done on separate hardware from your LIVE system. Testing your restore processes allows you to:

- Ensure your backups are working and usable
- Confirm your recovery processes and procedures work well and are up to date
- Give the X3 Administrators a chance to practice the recovery steps, so they are well versed in the processes

Change Control procedure

There are many changes that can be made to even a single node X3 instance, with any such changes having potential to disrupt the correct functioning of the instance:

- Operating System patches (Windows Updates)
- Changes to operating system parameters (Windows registry edit, changes to firewall settings)
- X3 Technology Stack patches (Syracuse, Runtime, etc.)
- X3 patches or hotfixes
- Updates to X3 configuration files (Syracuse, Management Console, MongoDB, SQL Server, etc.)

In multi-node instances, the list gets a bit longer:

- Load balancer setup
- Network topology

It is therefore important to have a change control procedure that allows you to plan and understand what changes are applied to any component of your X3 instance or the supporting infrastructure, so that:

- Changes can be applied in a controlled manner
- Any issues introduced by any change can be identified and reverted if necessary
- The business can understand any risks from proposed changes
- Business users can be scheduled to be involved in testing and changes

For some, this process may be as simple as a spreadsheet listing any changes that have been made, but in other cases there may be formalized systems to request and authorize changes before they are applied

Change control will often only apply to LIVE instances, although there is an argument for it to also be applied to TEST instances also

Patching and Testing

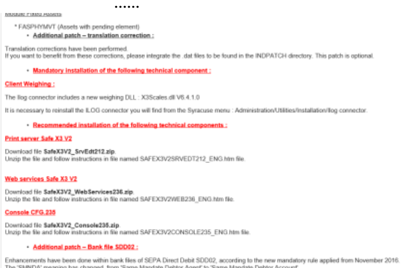
A "patch" is where new code needs to be installed, but is still within the same major version. E.g. Applying Patch 5 to a PU9 instance that is already on Patch 4 is "patching" but going from X3 v7 to X3 PU9 is "migrating" or "upgrading"   In this section we will only consider "patching"

With Sage X3, there are generally multiple patching activities that need to be undertaken to apply a patch. This is generally controlled by the nature of the main patch and is documented in the patch itself. For example, when you review the patch documentation for PU9 Patch 5 you will find there are mandatory pre-patch steps, which include applying the Syracuse 9.5 patch, as well as both Mandatory and Recommended post-patch activities, such as applying the latest Print Server patch

Update 9.0.5

Update (Patches no.)	Informations	Sage X3 functions
	File : X3_ProductUpdate_9.0.5.zip **Before integration** • Mandatory installation of the following technical component : Syracuse server Download file **SafeX3_Syracuse_9.5.zip** Unzip the file and follow instructions in file named SAFEX3SYRACUSE_9.5_ENG.htm **After updating the Syracuse server** : empty your Internet browser cache before logging in. • **Modified tables :** Attention to the integration time of this update on large volumes for tables : AWRKHISMES, AWRKHISSUI, AWRKTRND, COMPANY, CONTACTCRM, DCLVATPORP, GDIE, LTAPAR, MODSCALE, PAYFRM, SCALES, SEPAR, STKMVTADJ, TMPBLA, TRANNOTEH, TYPACE, WEIGHING Pay attention to have enough space in Oracle tablespaces. **After integration** • **New activity codes:** Module Common Data * KMA (Moroccan localization) • **New files:** Module Manufacturing * MODSCALE (Weighing scale templates)	

......



* FASPHYMVT (Assets with pending element)
• **Additional patch – translation correction :**

Translation corrections have been performed.
If you want to benefit from these corrections, please integrate the .dat files to be found in the INDPATCH directory. This patch is optional.

• **Mandatory installation of the following technical component :**

Client Weighing :

The Ilog connector includes a new weighing DLL : X3Scales.dll V6.4.1.0

It is necessary to reinstall the ILOG connector you will find from the Syracuse menu : Administration/Utilities/Installation/Ilog connector.

• **Recommended installation of the following technical components :**

Print server Safe X3 V2

Download file **SafeX3V2_SrvEdt212.zip**.
Unzip the file and follow instructions in file named SAFEX3V2SRVEDT212_ENG.htm file.

Web services Safe X3 V2

Download file **SafeX3V2_WebServices236.zip**.
Unzip the file and follow instructions in file named SAFEX3V2WEB236_ENG.htm file.

Console CFG.235

Download file **SafeX3V2_Console235.zip**.
Unzip the file and follow instructions in file named SAFEX3V2CONSOLE235_ENG.htm file.

• **Additional patch – Bank file SDD02 :**

Enhancements have been done within bank files of SEPA Direct Debit SDD02, according to the new mandatory rule applied from November 2016.

NOTE: when applying a "Hotfix" you should still go through these same steps, as for any other patch. Even though the impact of a HotFix is likely to be less, you still need to understand the impact and perform testing to confirm its effect

The general flow of a patching activity could be summarized as described below:

- Patch Analysis and planning
  - You perform a patch analysis to determine the areas of functionality that are effected, not forgetting any customisations you may have implemented
  - Identify and understand any pre-requisite and post-patch steps you need to take
  - You will also need to be able to backout the patch if it fails for some reason. You should therefore have a backout plan which can be implemented in your LIVE instance if necessary
  - Document all the expected steps, which should include links to all the additional notes or documentation that needs to be referred to. This document can then become your "Patch diary" to give a clear and repeatable process

NOTE : Sage Support strongly advise that you should always apply all the latest Technology patches, even though some may be flagged as "Recommended" rather than "Mandatory" in the patch documentation

- Apply patch in TEST environment

It is important to have a test environment which is completely separate from the LIVE environment, i.e. on separate hardware and with no shared components. This is to ensure that there is no cross-contamination between the environments, and no performance impact on LIVE when running the TEST instance

  - Take a pre-patch backup
  - Perform pre-requisite tasks
  - Apply the patch

    For the X3 patch itself, you can apply these using the "classic" patch function (Development→ Utilities→ Patches→ Patch Integration) however Sage recommend you utilize the newer "Updates Management" function (Administration→ Utilities→ Update→ Updates)
  - Perform any post-patch tasks

    This may include applying additional patches, functional configuration steps or other activities

- Validate and perform testing in TEST environment
  - Review the patch logs to ensure they went in without errors and that you have understand any warning messages
  - It is recommended you always test all business critical functions to ensure there are no unexpected side effects in these most crucial areas
  - Test all areas affected by the patching activity, as identified by your earlier patch analysis
  - Ensure you test any external links and any partner applications
    - For example, Web Services, Business Intelligence and any other third party or customized interfaces

NOTE: if you hit any problems or issues you should resolve and document the solutions, then perform a re-test before proceeding. This will ensure you have a "clean" patching run and will have a well-documented and repeatable process in your "Patch diary" document

- Test your backout/restore strategy
  - Once you are happy the patch is successfully applied, you may wish to test your backout plan, although in many cases this may be to restore your environment to the pre-patch backup

- Promote the patch through your test system hierarchy
  - If you have separate test instance (for example, you have a separate patch test and UAT) then you should apply the documented patch process to the next TEST instance

- Once all testing is successfully completed, schedule the patching activity for your LIVE instance. You will go through similar steps as per the TEST instance:
  - Ensure all users are logged out
  - Take a pre-patch backup
  - Perform pre-requisite tasks
  - Apply the patches
  - Perform post-patch tasks
  - Validate and perform non-destructive testing in LIVE environment
  - Allow key users onto the system for final checks
  - Release to all users

- Update your Change Control documentation

Auditing

Your auditing strategy will need to reflect the Business' Auditing objectives and policies, so the first step is to confirm and understand these objectives and policies

These policies would often include the following objectives:

- Sustaining accountability
- Ensuring compliance with standards and policies
- Monitoring for inappropriate or unusual activity
- Monitor health and performance of a system

From an X3 perspective, this boils down to deciding what data you need to audit, for example failed logins, updates to key fields on certain tables, etc.

WARNING: the more auditing you enable, the more overhead you create on system performance and could also potentially generate large amounts of audit data, which then needs to be stored and managed. You should therefore setup auditing for the minimum amount needed that achieves the business objectives

Steps to implement Auditing

- Check the AUDIT activity code has been enabled. This controls the overall availability of auditing (Should be enabled by default)
- Auditing needs to be setup for each table using "Audit" tab in Tables function (Development--> Data and parameters--> Tables--> Tables) Setup whichever fields should be audited and which events trigger the audit (create, update and/or delete) Database triggers are created automatically to enable this functionality. Remember to "Validate" the table once you have updated it

Additional notes

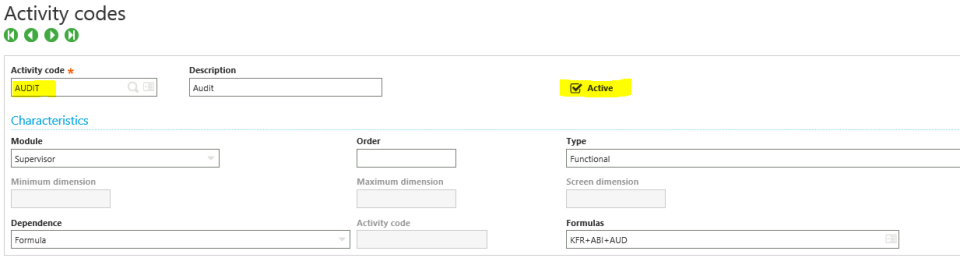
- Audit data writes to AUDITH and AUDITL tables
- Workflow batch task triggers for each line of AUDITH, if workflow option is selected
- Use the functions in Usage--> Audit to review the data
- Audit data can be purged via Usage--> Archive/Purge

Example of Auditing setup

In this example, we will setup Auditing on BPCUSTOMER table for changes to the Credit Limit

a. Check the activity code

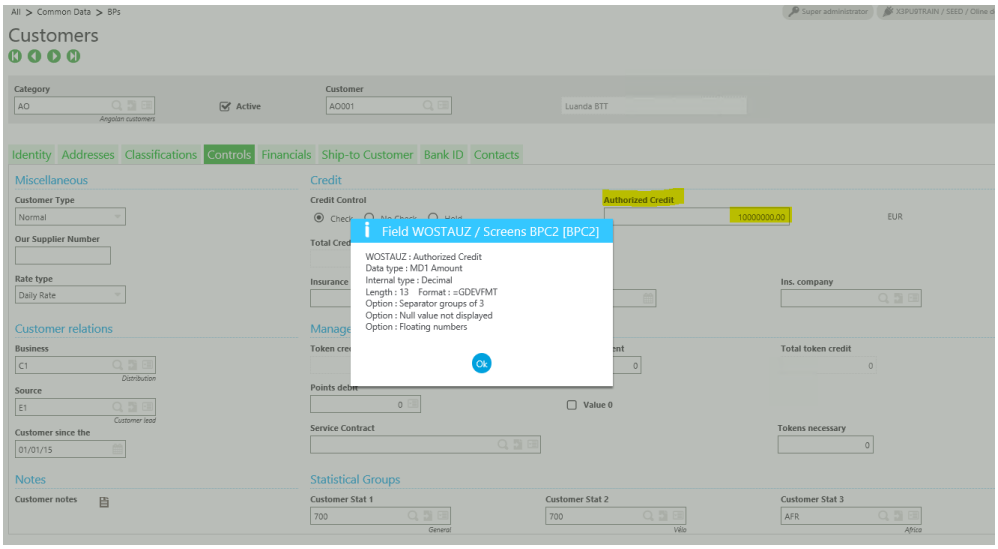
Development→ Data and Parameters→ Development Setup→ Activity Codes



b. Identify the table and field(s) for auditing

Navigate to Common Data→ BPS→ BPs

Click the field on screen and use ESC+F6 to see the field name and table



c. Setup Auditing on the BPCUSTOMER table

Navigate to Development--> Data and parameters--> Tables--> Tables

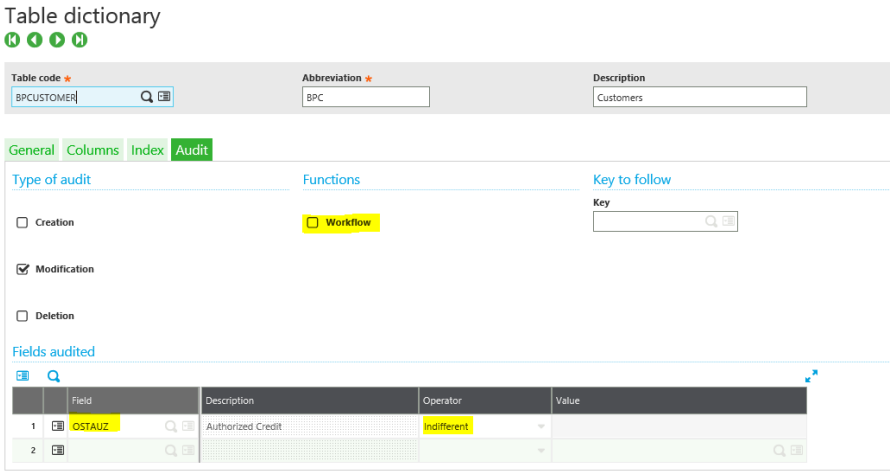
Query back the BPCUSTOMER table then go to the “Audit” tab

I will deselect “Workflow” in my case

Add the “Authorized credit” field (OSTAUZ)

Set “operator” to “Indifferent” as we want to see all changes

Save the changes, then click “Validation”



d. Make some changes to Credit Limits

Navigate to Common Data→ BPS→ BPs

Pick a couple of Customers and modify the "Authorized Credit" field

For customer AO001 change from 10'000'000 to 999'999
Customer BE001 change from no value to 132'000

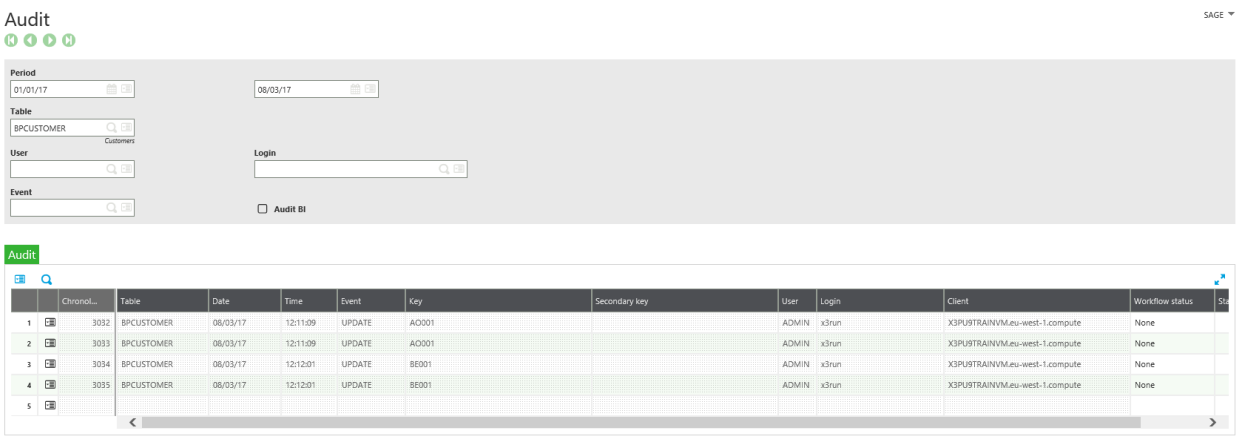
e. Check the audit tables

Navigate to Usage--> Audit→ Tables

For online help see http://online-help.sageerpx3.com/erp/9/staticpost/tables/

Query back Date range that covers todays date and enter BPCUSTOMER for the table

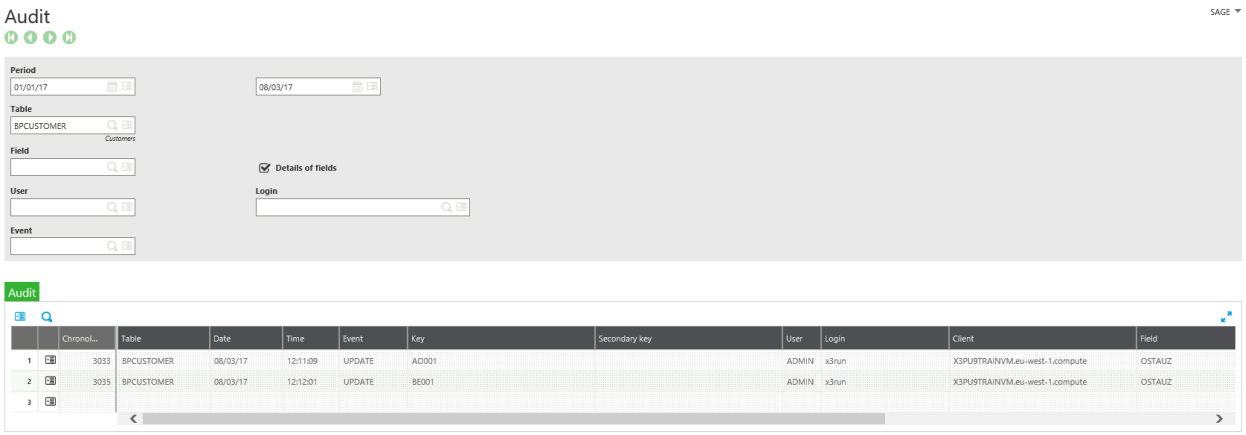
You will see four rows for the two updates, one before and one after the update. You can drill into the "Details of fields" from here if you wish



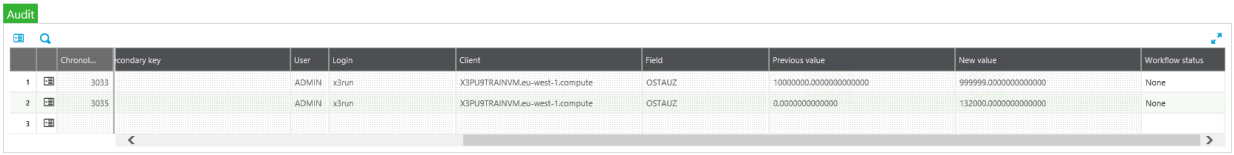
Navigate to Navigate to Usage--> Audit→ Fields

Query back Date range that covers todays date and enter BPCUSTOMER for the table
Also check the “Details of fields”

This shows two records for the two updates, but also has the field information immediately available, showing the before and after values



Scroll across to see the field details



Syracuse / Elastic Search / MongoDB

You should monitor and also archive the Syracuse, Elastic Search and MongoDB log files on a regular basis.

For all three components, you may wish to regularly scan the log files for any errors or unusual messages for further investigation, as a proactive measure to identify potential user issues

Over time, you will find a lot of log files will accumulate and some of the log files will grow quite large. It is prudent to periodically archive these log files to a different location in order to control disk space usage and make it easier to use the log files when they are needed

Syracuse

There is no option to change the level of logging so you cannot change the information level in the log files

There is also no automated way to archive the log files themselves, so you should regularly archive these logs, for example by using ZIP or similar tool to archive the old logs every month or so. This allows you to keep the number and size of the log files to a manageable level. The Syracuse service needs to be shutdown in order to archive the latest log files

The log files are locate in the <SYRACUSE INSTALL DIRECTORY>\Syracuse\logs for example "C:\Sage\Syracuse\syracuse\logs"

Elastic Search

The Elastic Search configuration file "logging.yml" located in the <ELASTIC SEARCH INSTALL DIRECTORY>/config allows you to change the level of logging. For example, "C:\Sage\ElasticSearch\config" By default it has INFO level logging for many components, which is quite verbose, so on your LIVE installation you may wish to reduce this log level to WARN instead

logging.yml

```
1  rootLogger: INFO, console, file
2  logger:
3    # log action execution errors for easier debugging
4    action: WARN
5    # reduce the logging for aws, too much is logged under the default INFO
6    com.amazonaws: WARN
```

There is no automated way to archive the log files themselves, so you should regularly archive these logs, for example by using ZIP or similar tool to archive the old logs every month or so. This allows you to keep the number and size of the log files to a manageable level. The Elastic Search service needs to be shutdown in order to archive the latest log files

The log files are locate in the < ELASTIC SEARCH INSTALL DIRECTORY>\logs for example "C:\Sage\ElasticSearch\logs"

MongoDB

Log file management

The MongoDB configuration file “mongodb.conf” located in the < MONGODB DIRECTORY>\config does allow you to change the level of logging. For example, “C:\Sage\MongoDB\config” By default it has relatively minimal logging configured, although this is still quite verbose. It is not recommended to reduce the logging level, even on a LIVE installation.

There is no automated way to archive the log file “mongodb.log” and it can grow quite quickly. You should regularly archive this log file, although you will need to stop MongoDB service in order to do this. For example use ZIP or similar tool to archive the old log every month or so. This allows you to keep the size of the log file to a manageable level, so when it is needed for diagnostic purposes it is easy to manage and search through for relevant messages

The log file is locate in the <MONGODB INSTALL DIRECTORY>\logs for example “C:\Sage\MongoDB\logs”

MongoDB Performance monitoring tools

MongoDB has its own command line performance monitoring tools you can use for monitoring performance and investigating poor MongoDB performance.

These are all located in the MongoDB bin directory.

- mongoperf
- mongostat
- mongotop

You can find the documentation for these tools on the MongoDB web site
https://docs.mongodb.com/manual/

Proactive Performance Monitoring / Tuning

System Performance monitoring is often not considered necessary by Business, that is until there is a performance problem

The trouble with this approach is that you may gather a lot of performance data with a performance problem in-situ, but it may not be clear what is the root cause or even worse you may make incorrect assumptions as you do not know what is considered “normal” for the performance statistics you are reviewing

You may wish to consider an alternative approach, which would be to regularly gather performance data whilst the system is running normally

This allows you to:

- Gather a “Normal” performance baseline which can then be used as a comparison when performance is poor
- Be able to see historic trends and react appropriately if there is a trend which indicates resources may run out, such as CPU usage trending upwards over time, or disk space being reduced at an alarming rate

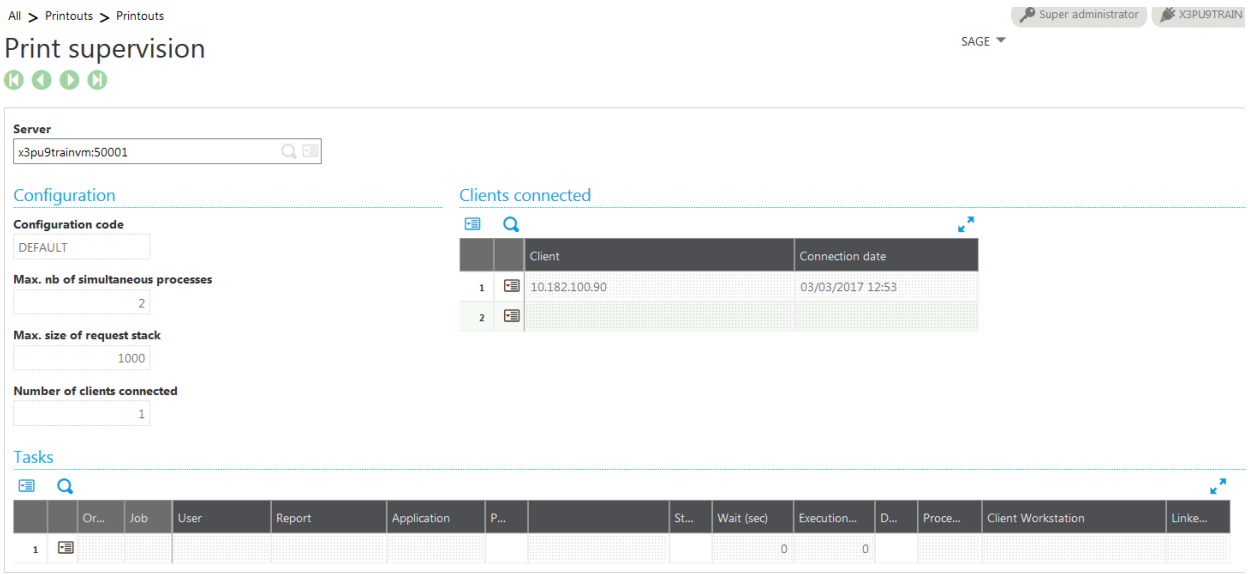
There are various tools available for both Windows and Linux platforms. For example, Windows Performance Monitor can be used to schedule the regular gathering of a wide range of system statistics, including SQL Server information

Miscellaneous topics

Sage X3

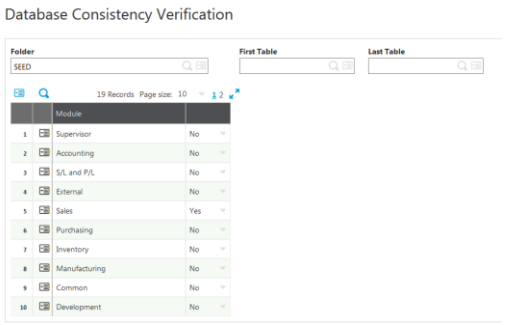
There are various X3 functions that can be used to check or manage your X3 instance which could be useful to an X3 System Administrator, although many would only be used when required. The most notable are discussed below:

Printouts—Printouts → Print supervision



This function shows the print jobs currently running and allows users with the appropriate authorization level to delete tasks or to change their priority

Development → Utilities → Data → Data Consistency



Whilst it can be difficult to analyse the output this function generates, the objective of this function is to compare the links between tables described in the X3 data dictionary with the actual tables stored in the Database itself. This process is resource intensive, as it reviews all the data in any tables you choose to run against, so should only run at quiet times.

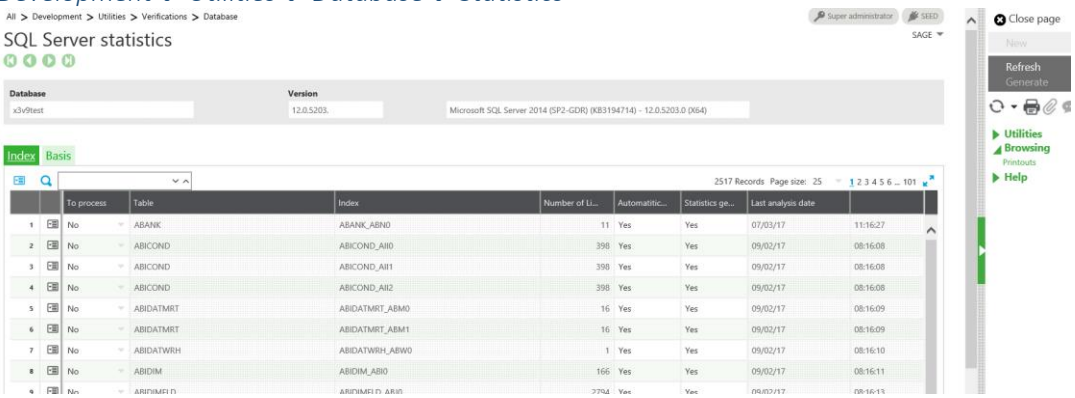
WARNING: you should not attempt to correct any standard tables if they are shown as having potential issues in the output, but instead to log a call with Sage Support to ask for assistance

Development → Utilities → Verifications → Database → Search index

Online help is available via the URL http://online-help.sageerpx3.com/erp/9/staticpost/database/ for the Development→ Utilities→ Verifications→ Database utilities

This routine should complete quite quickly. It provides a report comparing the X3 Data Dictionary description of the indexes against the indexes that actually exist in the database.

Development → Utilities → Database → Statistics



It is important that the table statistics are up to date to reflect the current data volumes and distribution. By default, this is managed automatically by SQL Server

You can check the database tables' statistics are being automatically generated and see the last date/time the statistics were gathered. If needed, you can also use this screen to select certain tables and then force a new statistics generation for those tables.

Administration → Certificates → Certificates (and Certificate Authorities)

You should be aware of what expiry dates your certificates have, in order to be able to renew them before they expire, as needed

Administration → Usage → Automate → Server logs

These log files relate to Elastic Search index process and also any jobs that have been configured through the Scheduler

These log files are not automatically purged, so you should also regularly monitor the log file usage and delete as and when these logs are no longer needed

Administration → Usage → Logs → Host Traces

See the online documentation at http://online-help.sageerpx3.com/erp/9/staticpost/host-trace/
You can review the automated logs generated, but can also configure your own manual logging of various components on an ad-hoc basis.

The automatically generated log files are kept for 10 days, but any manually generated ones are not automatically purged. You should regularly monitor the log file usage and delete as and when these logs are no longer needed

Conclusion

Hopefully this document provides additional clarifications to accompany the Online help, and will provide you some ideas of the administrative tasks you should be considering as an X3 System Administrator

Appendix A – SQL Script to gather data about Archive/Purge Parameters

```
--#####################################################################
--######             START OF SCRIPT                             ######
--#####################################################################
--##   Filename:  mzArchPurge.sql
--##   Updated:   23 February 2017
--##   Author:    Mike Shaw (Sage UK, X3 Support)
--##   Purpose:   Script to gather data of archive/purge config
--#####################################################################
---
---  Can also see most of this data if you navigate to:
---    Development--> Data and Parameters--> Development Setup--> Archive/Purge
---
SELECT
      AH.[COD_0] 'Code',
      TXT.[TEXTE_0] 'Description',
      TXT2.[TEXTE_0] 'Short Desc',
      CASE AH.[ENAFLG_0]
             WHEN '1' THEN 'No'
             WHEN '2' THEN 'Yes'
      END "Enabled",
      AH.[CTLTRT_0] 'Processing',
      AD.[TBL_0] 'Table',
      AD.[LNKTBL_0] 'Linked Table',
      AD.[DATFLD_0] 'Date field',
      CASE AH.[FLG1_0]
             WHEN '1' THEN 'No'
             WHEN '2' THEN 'Yes'
      END "Archive",
      AH.[TIM1_0] 'Retained days',
      AH.[FRQ1_0] 'Run Freq (days)',
      FORMAT (AH.[DAT1_0], 'd', 'en-gb' ) as 'Last Archived',
      CASE AH.[FLG2_0]
             WHEN '1' THEN 'No'
             WHEN '2' THEN 'Yes'
      END "Purge" ,
      AH.[TIM2_0] 'Retained Days',
    AH.[FRQ2_0] 'Run Freq (Days)',
    FORMAT (AH.[DAT2_0], 'd', 'en-gb' ) as 'Last Purged'
FROM
      [$(mzschema)].[AHISTO] as AH
             INNER JOIN [$(mzschema)].[AHISTOD] as AD
                    ON AH.COD_0 = AD.COD_0
             LEFT OUTER JOIN [$(mzschema)].[ATEXTE] as TXT
                    ON AH.DES_0 = TXT.NUMERO_0 and TXT.LAN_0 = 'BRI'
             LEFT OUTER JOIN [$(mzschema)].[ATEXTE] as TXT2
                    ON AH.DESSHO_0 = TXT2.NUMERO_0 and TXT2.LAN_0 = 'BRI'
ORDER BY AH.ENAFLG_0, AH.COD_0, AD.LIG_0;
--
-- Example of where we can use the date field column
--   SELECT * from X3.ABATRQT
--   WHERE DFIN_0 >= DATEADD(day,-60,CURRENT_TIMESTAMP);
--#####################################################################
--######               END OF SCRIPT                             ######
--#####################################################################
```

Appendix B – SQL script to get count of rows in X3 tables that are being purged

```
--#################################################################
--######             START OF SCRIPT                         ######
--#################################################################
--##   Filename:  mzArchPurge_count.sql
--##   Updated:   21 February 2017
--##   Author:    Mike Shaw (Sage UK, X3 Support)
--##   Purpose:   Script to gather data for archive/purge tables
--#################################################################
---
CREATE TABLE #mzResult
(
  rSchema NVARCHAR(261),
  rTableName NVARCHAR(261),
  rCount INT
);
DECLARE @mzSchemaName AS NVARCHAR(261);
--- Determine which schema names are relevant for X3 instance
DECLARE schemaCursor CURSOR FAST_FORWARD FOR
      SELECT NAME FROM sys.schemas WHERE (name not like 'db%' and name not in
('guest','INFORMATION_SCHEMA','sys') )
OPEN schemaCursor
FETCH NEXT FROM schemaCursor into @mzSchemaName
WHILE @@FETCH_STATUS = 0
--- Loop through the tables for each X3 schema in turn
BEGIN
      DECLARE  @mzSQL AS NVARCHAR(4000), @mzTableName AS NVARCHAR(261);
      ---  Pick out the table names we are intersted in
      DECLARE C CURSOR FAST_FORWARD FOR
            select TBL_0 from [X3].[AHISTOD] where TBL_0 like 'A%';
      OPEN C;
      FETCH NEXT FROM C INTO @mzTableName;
      WHILE @@FETCH_STATUS = 0
      ---  For each table, we want to get a row count and insert into the temp table
      BEGIN
            SET @mzSQL = 'INSERT INTO #mzResult(rSchema, rTableName, rCount) SELECT
''' +@mzSchemaName + ''','''' + @mzTableName + ''', COUNT(*) FROM ' + @mzSchemaName +'.'+
@mzTableName;
            EXEC(@mzSQL);
            FETCH NEXT FROM C INTO @mzTableName
      END
      CLOSE C;
      DEALLOCATE C;
FETCH NEXT FROM schemaCursor INTO @mzSchemaName;
END
CLOSE schemaCursor;
DEALLOCATE schemaCursor;
--- Report back on the number of rows we have gathered
SELECT * FROM #mzResult;
DROP TABLE #mzResult;
--#################################################################
--######               END OF SCRIPT                         ######
--#################################################################
```