SQL Database Engine and SSAS Upgrade Plan

# Summary

This document provides a detailed plan for upgrading the SQL Server Database Engine and SQL Server Analysis Services (SSAS) from 2014 to version 2022. The following outlines the plan for upgrading the SQL Server Database Engine from version 2014 to 2022 by building new instances instead of upgrading the existing ones. This ensures the new environment is clean while preserving the original setup for fallback.

## 1.0 Data Migration Assistant (DMA) Analysis

This work is performed by a System Admin and resolved by Development   
The [Data Migration Assistant (DMA)](https://learn.microsoft.com/en-us/sql/database-engine/install-windows/prepare-for-upgrade-by-running-data-migration-assistant?view=sql-server-ver16&viewFallbackFrom=sql-server-ver1) is used to identify potential compatibility issues in your SQL Server 2014 databases before migrating them to SQL Server 2022. The process includes analyzing both system and non-system databases, where the SA   
will focus on executing the tool, and developers will address database-specific and non-server-level issues that arise.  
  
Steps for using the Data Migration Assistant:  
1. Download and install the tool from the official Microsoft website.  
2. Launch DMA and create a new project, selecting 'Assessment' as the project type. (this can be executed locally on a workstation and NOT on the server)  
3. Select 'SQL Server' as the source server type and version 2022 as the target version.  
4. Choose the type of assessment: Database engine (for databases) or Server level (for instance-level configurations).  
5. Connect to the SQL Server 2014 instance and select the databases you want to assess. System databases are automatically excluded.  
6. Review the assessment results, which will provide:  
 - Compatibility issues that may prevent migration  
 - Deprecated features that are no longer supported  
 - Recommendations for feature parity and performance  
  
7. The developers will address non-system database issues identified by DMA.  
8. Once the assessment is complete, and all issues have been resolved, proceed with the migration process.

## Request the Instance

This work is performed by a Database Administrator (DBA)

Once we are ready for the build of the new instance the DBA will request a new instance using the following:

CPU:

Depending on the load and system being used will start with four CPU's. If the system warrants more the system will be evaluated and increased if needed.

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

RAM:

Depending on the load and system being used, it will start with 16GB of RAM. If the system warrants will be evaluated and increased if needed.

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

Drive Configuration:

Standard configuration with a system starting out but space will be dependent on how much space is needed for the system.

C:\ (System) 60GB

D:\ (CD Drive)

E:\ (Programs) 50GB

F:\ (SQLData) 100GB

G:\ (SQLLogs) 50GB

H:\ (TempDB) 50GB

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

## 2.0 Building of the New Instance

This work is performed by a Database Administrator (DBA)

(from a previous machine if exists)

1. Generate the Configuration File from the Existing SQL Server Instance

To find the config it can found using the following method:

1. Open Powershell and create this function:

function Get-RemoteSQLVersion {

param (

[string]$remoteServer, # The remote server name or IP

[string]$instanceName = "localhost" # The SQL Server instance name (default: localhost)

)

# Define the common SQL Server installation paths on the remote server

$commonPaths = @(

"C:\Program Files\Microsoft SQL Server",

"C:\Program Files (x86)\Microsoft SQL Server",

"C:\ProgramData\Microsoft SQL Server"

)

# Define the file name you're searching for

$fileName = "configurationfile.ini"

# Script block to execute remotely

$scriptBlock = {

param ($instanceName, $commonPaths, $fileName)

# Loop through each path and search for the file

foreach ($path in $commonPaths) {

if (Test-Path -Path $path) {

$files = Get-ChildItem -Path $path -Recurse -Filter $fileName -ErrorAction SilentlyContinue

if ($files) {

foreach ($file in $files) {

"$($file.FullName) "

}

}

}

}

}

# Execute the script block on the remote server

Invoke-Command -ComputerName $remoteServer -ScriptBlock $scriptBlock -ArgumentList $instanceName, $commonPaths, $fileName

}

Once added execute the Get-RemoteSQLVersion with the server’s name it will give you a location that is directly to the ini file.

# Example usage

Get-RemoteSQLVersion -remoteServer "SERVERNAME" -instanceName "SQLInstanceName"

1. Creating a manual configuration.ini file

; SQL Server 2022 Configuration File

[OPTIONS]

ACTION="Install"

IAcceptSQLServerLicenseTerms="True"

FEATURES=SQLENGINE

INSTALLSHAREDWOWDIR="D:\Program Files (x86)\Microsoft SQL Server"

INSTANCEDIR="C:\Program Files\Microsoft SQL Server"

INSTANCENAME="MSSQLSERVER"

INSTALLSHAREDDIR="D:\Program Files\Microsoft SQL Server"

INSTANCEID="MSSQLSERVER"

SQLTELSVCSTARTUPTYPE="Automatic"

SQLTELSVCACCT="NT Service\SQLTELEMETRY"

ISTELSVCSTARTUPTYPE="Automatic"

ISTELSVCACCT="NT Service\SSISTELEMETRY160"

AGTSVCACCOUNT="svc\_name\_createdforsql"

AGTSVCSTARTUPTYPE="Automatic"

ISSVCSTARTUPTYPE="Automatic"

ISSVCACCOUNT="NT Service\MsDtsServer160"

SQLSVCSTARTUPTYPE="Automatic"

SQLSYSADMINACCOUNTS="ELDREDGE\_A\Database Administrators"

SQLSVCACCOUNT="svc\_name\_createdforsql"

SECURITYMODE="SQL"

SQLUSERDBDIR="correct\_drive\_filepath"

SQLUSERDBLOGDIR="correct\_drive\_filepath"

SQLTEMPDBDIR="correct\_drive\_filepath"

1. Go without and manual install using the UI if that the case then step 4 can be skipped.

3. Obtain SQL Server 2022 Installation Media

- Download the SQL Server 2022 installer from Microsoft's website.

- Extract or mount the downloaded installer ISO file.

4. Run SQL Server 2022 Setup in Silent Mode

- Open an elevated Command Prompt (Run as Administrator).

- Navigate to the directory where the setup.exe is located (inside the extracted/mounted ISO).

- Run the following command to install SQL Server 2022 using the configuration file:

setup.exe /ConfigurationFile="C:\path\to\your\ConfigurationFile.ini"

- Make sure that the file is in the path to ConfigurationFile.ini and is correct.

- The installation will proceed in silent mode based on the configuration file.

5. Monitor the Installation Progress

- To track the installation status, open the log files located in:

D:\Program Files\Microsoft SQL Server\160\Setup Bootstrap\Log\

- The most recent timestamp folder will contain detailed log files (Summary.txt and Detail.txt) that you can monitor for any errors or progress during the setup.

6. Verify the Installation

- Once the setup is complete, check the SQL Server Configuration Manager to confirm that the new SQL Server 2022 instance is running.

- Connect to the instance using SQL Server Management Studio (SSMS).

- Verify that all required settings from the configuration file are applied correctly.

## 2.1 Setting up of the New Instance

This work is performed by a Database Administrator (DBA)

Manual work for the DBA

Ensure the DBAs are local admins on the new host.

Create Database Administrators and the service account for GitHub (svc\_GTHB\_PRD01) to ensure both have sysadmin rights in SQL Server.

The “sa” account will need to be disabled.

SQL Instance Configuration: Please adjust default DATA and LOG locations to the F & G drives.

SQL TempDB Configuration:

Depending on the number of CPUs will determine the number of TempDB data files. The default configuration will start with two data files and increase the number to eight as the CPU requirement increases. 8GB will be the standard size of the data files starting. Auto growth should be in one Gb increments and unlimited max size.

Once that work is complete automated

SQL Configurations, the DBA Database, the PerfStats Database, the SQL Backups, and Maintenance will be completed by the GitHub Deploy.

## 3.0 Backup of the old instance

This work is performed by a Database Administrator (DBA)

Determine the best route

1. Backup Databases after services/IIS and no other user activity on the DB
   1. EXEC [backup].[BackupDatabase] @BackupLevel = 'FULL', @SQLDatabaseName = 'USER\_Databases', @force = 1, @DryRun = 0
   2. EXEC [backup].[BackupDatabase] @BackupLevel = 'LOG', @SQLDatabaseName = 'USER\_Databases', @force = 1, @DryRun = 0
2. Use Backup from the night before and all logs up to the most recent including a final log backup after services/IIS and no other user activity on the DB
   1. EXEC [backup].[BackupDatabase] @BackupLevel = 'LOG', @SQLDatabaseName = 'USER\_Databases', @force = 1, @DryRun = 0

## 3.1 Restore to new instance

This work is performed by a Database Administrator (DBA)

On the new instance

EXEC DBA.[backup].[RestoreDatabase] @ClientHost= ‘OldServer', @SQLDatabaseName = 'olddbname', @RestoreDatabaseName ='newname' , @DryRun =0

(If we are pulling from production environment to Dev we want to define the BackupPathFull, BackupPathDiff, and BackupPathLog directly to the Production network path and ensure we have access.)

## 4.0 Copy or validate all users and roles were replicated over

EXECUTE [DBA].[deploy].[CreateDatabase] @TargetDB ='DBName', @Dryrun =0

After executing the previous stored procedure (after the databases are restored it will ensure that all the correct permissions are deployed to the databases.

Any remaining accounts can be copied by using the following PowerShell script

(see below)

$server1 = ''

$server2 = ''

Copy-DBALogin -Source $server1 -Destination $server2 -Login $user

Sync-DbaLoginPermission -Source $server1 -Destination $server2 -Login $user

## 5.0 Validation

This work is performed by a System Admin, Development

Once restores are completed DBA steps through all steps are completed from DBA work.

1. System Admin and Developers are presented with instances to do the application-level scripts
2. System Admins turn on all services.
3. Any additional application needs that are handled by SA

This document will be updated periodically as newer resolutions, improvements, or configuration methods for SQL Server 2022 are discovered. Be sure to check for the latest updates to ensure you are following the most current best practices and procedures.

# SQL Server Analysis Services (SSAS) Upgrade (2014 to 2022)

## 1.0 Request the Instance

This work is performed by a Database Administrator (DBA)

Once we are ready for the build of the new instance the DBA will request a new instance using the following:

CPU:

Depending on the load and system being used will start with four CPU's. If the system warrants more the system will be evaluated and increased if needed.

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

RAM:

Depending on the load and system being used will start with 16GB of RAM. If the system warrants more the system will be evaluated and increased if needed.

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

Drive Configuration:

Standard configuration with a system starting out but space will be dependent on how much space is needed for the system.

C:\ (System) 60GB

D:\ (CD Drive)

E:\ (Programs) 50GB

F:\ (SQLData) 100GB

(For any upgrade to a newer SQL we will request the same specs as the previous instance as those specs have proven to work for the application previously and lessen the number of changes at one time)

## 2.0 Building of the New Instance

This work is performed by a Database Administrator (DBA)

1. Run Setup:
   * Double-click the SQL Server installation media file (typically an.exe file).
   * Follow the on-screen prompts to launch the SQL Server Installation Center.
2. Choose Installation Type:
   * Select "New SQL Server stand-alone installation or upgrade" and click "Next."
3. Installation Rules:
   * Review the installation rules and click "Next."
4. Product Selection:
   * Uncheck all components except "Analysis Services." This ensures you install only SSAS.
   * Click "Next."
5. Instance Configuration:
   * Choose the instance name for SSAS. You can accept the default or specify a custom name.
   * Click "Next."
6. Server Configuration:
   * Configure settings related to the SSAS instance, such as authentication mode, port number, and data directory.
   * Click "Next."
7. Database Engine Configuration:
   * Since you're installing only SSAS, you can leave the database engine configuration options blank or accept the defaults.
   * Click "Next."
8. Instance Configuration Rules:
   * Review the instance configuration rules and click "Next."
9. Ready to Install:
   * Review the installation summary and click "Install."
10. Installation Progress:

* Monitor the installation progress. This may take some time.

1. Completion:

* Once the installation is complete, click "Close."
  1. Copying database/setting and permissions over

(this has different flavors to accomplish)

* + 1. Create DB over and manually create roles per environment then allow DEV and SA to create string entries
    2. Create the XMLA script for the database per environment (do not use the same script for all environments; settings and permissions change per environment) that will give data sources, roles, etc auto-filled.

4.0 Validation

Handoff to SA/Dev to add connections

This document will be updated periodically as newer resolutions, improvements, or configuration methods for SQL Server 2022 are discovered. Be sure to check for the latest updates to ensure you are following the most current best practices and procedures.