Sage 100 Premium & Microsoft SQL FAQ

Note: This info is supplemental and not intended to replace Sage's official supported platforms or the advice of your Sage partner. These FAQ items were initially generated in October 2018, and some portions may not have been updated for subsequent changes to either Microsoft or Sage products.

Use this information to supplement and not replace your technical resources.

Please review the Sage Installation and System Administrator's Guide for your version of Sage 100 for detailed installation and configuration information.

Important Note On SQL Password Limitation (9/2024): Please note that one user reports that the Sage 100 Premium password cannot exceed 15 characters or you may receive an error message **Error 10: Illegal pathname specified.** Read more at Sage KB 225924850098671

Essential Tips for Migrating Sage 100 Premium

- Have IT set aside a temporary storage space with enough storage to maintain several full-copy backups. This temporary storage can be removed after you've successfully migrated/converted and gone live.
- Once you install the Sage 100 Premium version that you are upgrading/migrating TO:
    - **Make a full copy** of **\MAS90** (and subfolders) **Important**
    - Take **SQL Backups of all the MAS_XXX folders plus MAS_SYSTEM**. **Important**
        - If there are any issues during migration, you may be unable to get into your new Sage system due to a corrupt **MAS_SYSTEM**, which - if you don't have a backup to restore - may require you to uninstall/reinstall Sage 100 fully or restore the MAS_SYSTEM.BAK to SQL. Having a backup of MAS_SYSTEM, which you restore to the SQL server, can save you many hours of reimplementation.
        - **Recovery and error situations during migration are not well documented,** so making backups is your best protection against having to reinstall (start over)
        - Premium requires at least local admin user rights to run the service. If moving/migrating SQL from one server to another, you want domain admin rights.
        - Make these backups

        - Before attempting a migration/conversion
        - After a successful migration/conversion
    - During an re-migration of a Sage 100 upgrade (v2021 to 2024) the PL log files did NOT appear to re-migrate despite answering “YES” to migrate system files. The fix requires SQL knowledge and copying the prior records from the old SQL to new SQL (definitely make backups)

What’s **NOT** Available in Sage 100 Premium

- Old-style custom financial statement (*UNHIDEGL)
- Work Order Module or MRP (Must use Production Management / IRP) - The Work Order module is no longer available on any upgrade past Sage 100 2021 as it has been fully replaced by Production Management
- Business Insights ***REPORTER***
- Access to the **SOTAMAS90 ODBC** for reporting (use a direct SQL connection instead, which is significantly faster)
    - Previously, SOTAMAS90 would prompt for what company code to report on - with the SQL ODBC, you will either need to manually change the “default” database or create a separate DSN for each company (if multi-company), or use custom coding to swap the DSN info on-the-fly.
- Automatic Update (Library Master)
- ODBC Security (Library Master Role Maintenance)
- C/S ODBC Driver (use a direct SQL connection which is much faster)
- Change Data Location (Library Master - Company Maintenance)
- SY0_CompanyParameters is not available in Premium (used to pull in parameters like company code, and company name)
    - 90 Minds Tip: you can restore SY0_CompanyParameters by running the following SQL in each MAS_xxx database:

```
CREATE VIEW SY0_CompanyParameters AS
SELECT CompanyCode, CompanyName
FROM MAS_SYSTEM.dbo.SY_Company
WHERE CompanyCode = RIGHT(db_name(),3)
```

- View more in the Sage 100 2019 Help Files

Other Important Considerations

- As of **March 2023**, Sage has announced that beginning with Sage 100 2023, **specific license keys will no longer be needed for Premium**. This platform choice will be

made at installation time. It is still being determined what impact this may have—if any—on subscription costs for anyone with migration credits.

- ~~As of **January 1, 2022**, your **Sage subscription may change if you have migration credits on your account and your platform changes to Sage 100 Premium**. Please verify your subscription renewal before changing your platform to Premium (SQL).~~
- Users are hosted on Azure but with full (not **SQL as Service**) versions of MS SQL.
    - It appears the issue is with these versions of SQL you can't do specific cross-database queries by using the full qualified name of the database as `you can with on-prem sql when in another database. So I can't say select * from mas_abc.dbo.AR_Customer when i'm in mas_system, it doesn't seem to understand the mas_abc.dbo portion.
    - https://www.sqlservercentral.com/articles/cross-database-queries-in-azure-sql-database
- If you create individual user security rights for ODBC to access separate company SQL databases (MAS_XXX) and during testing or go-live you've dropped the MAS_XXX and re-added then you also need to re-add security permissions to the newly added table(s) for any individuals that you had given rights. MAS_REPORTS and MAS_USERS do not appear to have this issue.
- Test registration keys may be available using part **G9QPREMTRIAL (This is obsolete starting with Sage 100 v2023, which no longer requires Sage 100 Premium specific registration keys)**
- **Importance of security permissions:** During the initial installation of Sage 100 Premium you must have either (a) the **SA password for the SQL server** or (b) If you plan to select the **Windows authentication** option when running the installation wizard, the **account must be a member of the sysadmin role on the server**. This is necessary to ensure that the MAS_SYSTEM SQL database is created correctly. If you do not have the required permissions to create these databases your installation will likely fail and in some cases, the failure may not be apparent until starting Sage 100.
- Sage 100 Crystal temporary Work tables are generally not exposed for use. This impacts very few people (aside from advanced Crystal Reports users) as these temporary tables are generally used internally.

  The temp tables are created in TempDB for performance but also done so for concurrency reasons by using a "#" (no quotes) in front of the name.

  Using a "#" means it's a local temporary table. Each session, which means each instance of form or report printing, gets its own unique temp table that is not visible to other sessions. That's very good for multiple users running the same form/report concurrently but not good if you're trying to say run a .rpt file directly from Crystal Reports Designer to see the data.

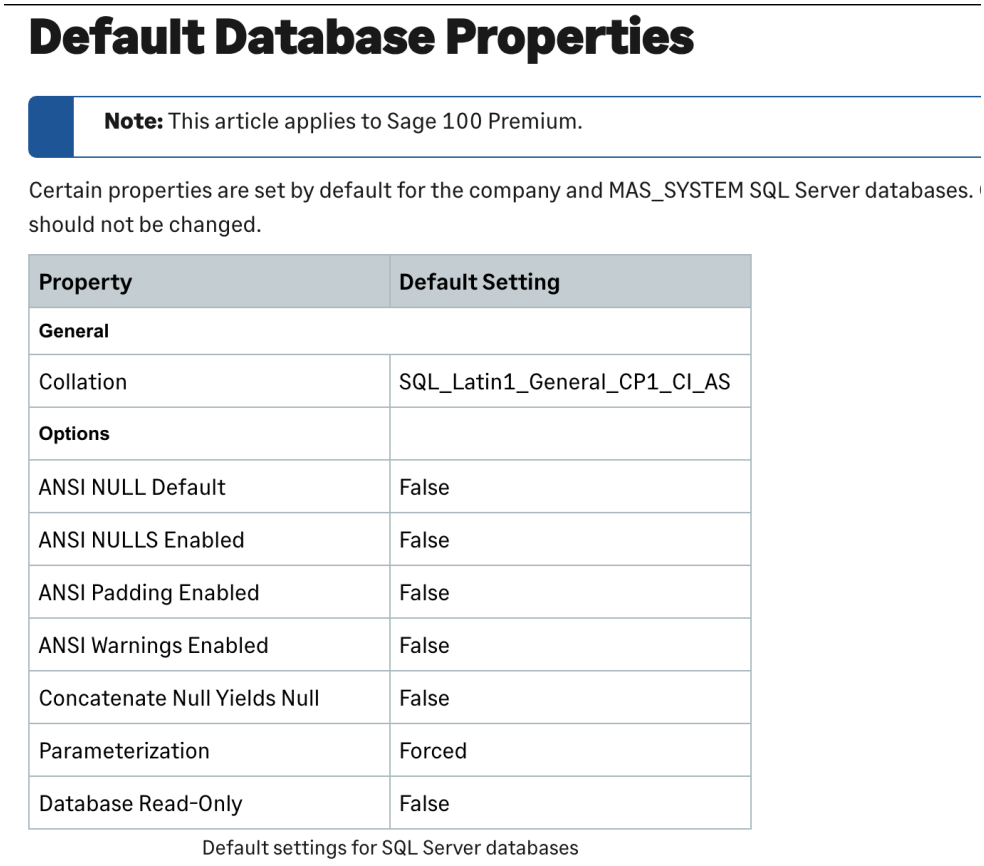
  The trick in Standard/Advanced to swap a base work file for a populated temp work file during Preview is not possible.

  Having said all that, you can still see all the local temporary table names with this command

and if you can figure out which one is yours for the form/report you just previewed, you can possibly do something more with it.

Sage OEM Runtime Version of Microsoft SQL

Sage promotes their OEM version of Microsoft SQL Server Runtime Edition as optimally



Default Database Properties

Note: This article applies to Sage 100 Premium.

Certain properties are set by default for the company and MAS_SYSTEM SQL Server databases. should not be changed.

Property	Default Setting
General	
Collation	SQL_Latin1_General_CP1_CI_AS
Options	
ANSI NULL Default	False
ANSI NULLS Enabled	False
ANSI Padding Enabled	False
ANSI Warnings Enabled	False
Concatenate Null Yields Null	False
Parameterization	Forced
Database Read-Only	False

Default settings for SQL Server databases

configured.

As of April 14, 2023, Sage sells the Microsoft SQL Server OEM CAL for $120 per user per year.

The SQL Server Runtime license is a license that lets an independent software vendor (ISV) embed the complete SQL Server code into a solution for use only by the ISV application. The customers of the ISV cannot use this SQL Server product to run other applications or to develop new applications, databases, or tables. There may also be limitations that don't allow the Sage OEM SQL license(s) to be used in the cloud.

The “optimizations” are listed on pages 99 and 100 in the Admin guide, and it saves checking the right components and clicking Next a few times during the installation.

"Optimized" is a bit misleading. It means "configured for basic compatibility with Sage 100 Premium". The installer defaults to Mixed Mode Authentication (required for Premium and prompts to create a 'sa' password). It uses the default collation method (basically, it chooses Case Insensitive for you) and the default language of English.

Sage recommends tempdb database should be at least 1/3 the size of largest database. Here is more information on best practices for tempdb sizing.

It does not "optimize" for performance since it does not, for example, ask you where to create the Sage 100 databases (like on a drive with more disk space), what setting to use for Database Recovery Model, etc. You have to move/set those things AFTER installing.

(Link to Sage 100 v2018 Admin Guide)

The OEM license runs SQL Server 2014 (2016, 2017, and now 2019 are available) and cannot legally use that SQL Server instance for anything other than Sage. If SQL Server is hosted in a data center, there are license restrictions on the OEM SQL license. Check with your Microsoft licensing specialist for further information.

ISVR licenses/Licensing Mobility Eligibility

The ISVR restrictions are outlined in the ISVR Agreement. It is not currently an option with ISV Royalty Licenses because LM (License Mobility) is a benefit of Software Assurance and ISVR does not offer Software Assurance, only Embedded Maintenance. The End Customer can have the Unified Solution running on Outsourced Servers, but **not on shared (Cloud hosted) servers**.

Several reports from consultants that the OEM download link only goes to the end user and can be difficult for a consultant to obtain from Sage.

On July 31, 2020, Sage sent a request to users asking them to review that any SQL license purchased from Sage was in compliance with Microsoft licensing. There was also a flowchart included aiding in reviewing compliance.

How Many Microsoft SQL Licenses Are Required To Run Sage 100 Premium?

The short (and safest) answer is one for each Named user (not just the concurrent count) of Sage, plus the service account. The official Microsoft response is that any user that could access the environment (even temporarily) should be fully licensed.

Per Sage - Sage 100 Premium requires two licenses; one for the MAS_User account and another for the MAS_Reports account in SQL. You will need more licenses depending on individual needs: Administration, other databases or 3rd party methods of accessing Sage 100 data, etc.

However, the consensus among consultants seems to be:

I'm not sure the exact licensing for the Sage run-time, but as far as Microsoft's concerned with SQL Server Standard or Enterprise, you either license by the processor core (all physical cores on the box or at least 4 if SQL is virtual) or per-User, and per-User means any user that could ever potentially touch the SQL server for data. If you have more than 30 users, per-core is usually the cheapest. If you have less than 30, per User is the way to go.

So for per-User licensing, any Sage user would require a SQL CAL.

Additional Server recommendations:

Sage has the following MS SQL server configuration recommendations. They more fully outline the SQL server setup recommendations in their Administrator Guide here.

Server RAM:

A rough estimate for the amount of RAM could be calculated as 1 GB for each concurrent user or size of database (in GB) whichever is higher but not less than 32 GB. There are more sophisticated ways to monitor this but minimum should be 32 GB or even 64 GB RAM.

SQL Server Memory Allocation:

SQL Server will allocate all the RAM to itself until the server is rebooted or the service restarted. To prevent the operating system from being robbed of RAM use the "Memory Allocation' setting.

In Object Explorer, right-click the Server and select Properties.

Click the Memory node.

Under Server Memory Options, enter the amount that you want for Minimum server memory and Maximum server memory. Microsoft recommends the Maximum to be set from ½ to ¾ of the Server's total RAM.

Tempdb:

The tempdb is a temporary global workspace for storing tables and processing queries. If your application is causing the tempdb to grow larger this will cause bottleneck until the tempdb is increased. Instead it is wise to bump up the size of this table.

The tempdb database should be at least 1/3 the size of largest database.

Hard Drives:

Hard drives with optimum speeds should be considered ever Solid State Drives. Also the .mdf log files are competing for read and write access to the hard drive. There will be a separate .mdf file for the data, transaction, and backup logs. To eliminate the competition these .mdf files can be placed on separate physical drive letters. The location of these logs can be determined from Server Properties -> Database Settings.

FAQ: How do I set up SQL to Sage 100's requirements? Review the Sage 100 Administrator Guide

Consultant Provided Additional Recommendations:

· Windows 2012 R2 Server or 2016. (Verify this with the Sage SPM)
· SQL Server 2014 or 2016 standard (Verify this with the Sage SPM)

- with mixed mode authentication enabled.
- Be sure to set the server collation method as SQL_Latin1_General_CP1_CI_AS (if not, there will be data integrity issues).
- This collation method is normally set by default when the server is setup with a default language of US English.
- If using a different language setting, be sure to set this collation method as part of the SQL Server installation.
- The Default SQL data path should have sufficient space for the databases / log files.

Recommended Cores
Generally 4 cores and 16 GB of RAM, but ideally, you should have enough RAM to fit the entire MAS_xxx databases. It is crucial to ensure that the storage is SSD. Please note that this is not official Sage guidance and is provided as-is and believed to be accurate.

Required access:

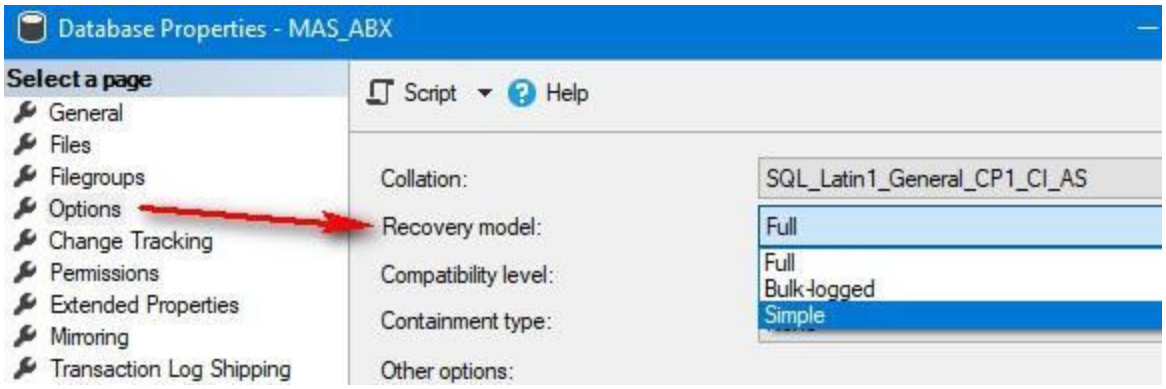
- Full administrator control over the server.
- SQL “sa” password.
- Remote access using a method that allows us to reboot the machine and reconnect afterward.
- We run the services under a Windows login. If there is a password policy which involves expiring passwords, we’ll want a utility Windows login with admin access over the server and a password that does not expire.

SQL Logs and Backup Tips:

- SQL Logs are part of a backup & recovery strategy
- If you run **a "full" backup within SQL**, the log file space will be freed and can be recycled (right-click and shrink files, log file, and set a reasonable size if they grew too large). The idea is that you can combine a full backup with the LDF to catch up to the latest transaction if the MDF is lost. The architecture was mainly designed for physical drive failures (two drives... one with backup & LDF, one with the MDF), before VMs and RAID arrays. You can also do a full backup, then smaller (more frequent) log backups, again to protect the data in case of disaster. It's mainly for advanced environments, with mission-critical minute-to-minute data... which is not Sage 100 (usually).
- You can also **change the backup and recovery model to "Simple" (from "Full"), to disable the log files... and shrink them to a reasonable working size (500MB is usually what I set)**. This is what we normally do now, unless the client has sophisticated IT who want to do it their way (i.e. log backups during the day to minimize data loss in the case of a disaster).

- **Important Note**: Never shrink the main database files. Only shrink database log files (which will auto-grow if more space is needed).

If you are not running SQL backups from within SQL, change the "Recovery model" to **Simple**... for all Sage databases.



Simple recycles log file space, with backups done as file copies. (Most important is the MDF, but BAK files and server snapshots are fine too).

Full retains the log files (which grow without limit) until a SQL backup command is run against the database. (The backup releases the log file space for re-use). This is part of a backup and recovery strategy that can pair a full backup with the log file to restore a database up to a point in time (beyond what the backup contains). This architecture was from the pre-RAID days, where you would put backup and log file onto one physical drive, and the main data file on another physical drive. That way you could always restore up-to-the-minute if either drive failed. (MDF = full data, and BAK + LDF can be used to re-create the MDF). Obviously this configuration needs a high level of skill in whoever is supporting the SQL environment, which is rarely the case in the Sage 100 world.