

finPOWER Connect SQL Server 2019 Installation and Configuration

Version 1.00

26 November 2020

Table of Contents

Disclaimer	3
Version History	4
Introduction	5
Backwards Compatibility.....	6
SQL Server Database Modifications	6
Configuring the Server	7
SQL Server Hardware Guidelines	7
Installing SQL Server	8
Notes for the Express Edition	8
Windows Server Configuration	9
Add the finPOWER Group	9
To add the finPOWER Security Group.....	9
Assign users to the finPOWER group	9
To assign Users to the finPOWER Group	9
Windows 10 Configuration	9
To add the finPOWER Security Group.....	9
SQL Server Configuration	10
SQL Server Security Properties	10
To configure SQL Server	10
System Administrator Login	11
To assign a new password to sa	11
SQL Server Network Protocols	12
Enable TCP/IP protocol	12
Firewall Configuration.....	13
Opening a port in Windows firewall	13
Model database	14
To check Model database options	14
finPOWER Login	16
To add the Intersoft Login	16
Backups	17
Moving an Access database to SQL Server	18
Database Naming.....	18
To move an existing Access database to SQL Server	18
Duplicating a SQL Server Database	19
Duplicating a Database using SQL Server Backup and Restore.....	19
To Copy a Database using SQL Server Backup and Restore	19

Disclaimer

This document is for informational purposes only.

All information contained in this document, including code samples, are provided "as is" without warranty of any kind, and Intersoft accepts no liability for any decisions made on the basis of this information

This document contains information that may be subject to change at any stage.

Copyright Intersoft Systems Ltd, 2020.

Version History

Date	Version	Name	Changes
25/11/2020	1.00	JR	Created.

Introduction

The finPOWER Connect Enterprise Edition adds support for Microsoft SQL Server databases.

The Enterprise Edition incorporates support for Microsoft's industrial strength database engine "SQL Server", which is designed to handle hundreds of simultaneous users and very large databases.

NOTE: The default database engine used within finPOWER Connect is Microsoft Access. However, we recommend this database engine only be used for up to 10 users and databases less than 200MB. Once this threshold has been reached, you should consider licensing for the Enterprise Edition of finPOWER Connect.

Whilst finPOWER Connect supports SQL Server 2008 and higher, this document covers SQL Server 2019.

SQL Server supports several versions, including:

- Standard Edition
- Enterprise Edition
- Express Edition
 - Free version
 - Limitations:
 - ✦ Maximum Database size: 10 GB
 - ✦ Compute Capacity: Limited to lesser of 1 CPU socket or 4 cores
 - ✦ Maximum Memory: 1410 MB
 - ✦ Whilst there is no limit on connections, the available memory and compute capacity will limit the effective number of connections
 - ✦ Lacks some facilities, e.g. Maintenance Plans for backups

We expect most users will either use the Standard or Express editions, so the focus of this document will be on these two editions.

For more information see:

- <https://docs.microsoft.com/en-us/sql/sql-server/editions-and-components-of-sql-server-version-15?view=sql-server-ver15>

NOTE: Microsoft SQL Server and Client licenses to login to the server must be purchased separately (except for the Express version).

WARNING: Make sure your backups are working correctly before using your finPOWER Connect database.

Backwards Compatibility

Once a SQL Server database has been upgraded to SQL Server 2019 there is no way to run it on an older SQL Server platform.

This means you should carefully test your systems in a SQL Server test environment before moving them to production. It also means you should ensure you have suitable backups before transferring to SQL Server 2019.

WARNING: Once a database has been loaded and used on SQL Server 2019 it cannot be restored back to a previous SQL Server version.

SQL Server Database Modifications

Unlike Access databases, where database access is tightly controlled by Intersoft, SQL Server **System Administrators** have full control over all aspects of SQL Server and everything within it, including the database structure and the data contained in each table.

In other words, there is nothing specifically stopping the System Administrator from altering the database structure or the data contained in the database. However, altering the database structure or information stored in the database may stop finPOWER Connect from working correctly, and even corrupt information.

WARNING: It may be tempting to log in as the System Administrator to change information or add an extra column to a table etc – BUT this will invalidate your software licence.

Under Software Warranty section

This Warranty is also subject to the additional condition that it immediately becomes void if any attempt by the Licensee to decompile, disassemble, reverse engineer or in anyway modify the System, database structure or data is made without written consent of the Licensor.

Under Termination section

The Licensor may immediately terminate this Agreement if any attempt by the Licensee to decompile, disassemble, reverse engineer or in anyway modify the System is made without written consent of the Licensor.

Configuring the Server

There are a large number of options that are configurable in SQL Server.

It is recommended that changing SQL Server settings is only carried out by a technically able person.

finPOWER Connect expects SQL Server and indeed the Server itself to be configured in a certain way so that a number of tasks can be automated from within finPOWER Connect.

SQL Server Hardware Guidelines

Whilst there are minimum hardware requirements outlined by Microsoft the recommended or ideal configuration will be quite different.

For example, SQL Server loads data into RAM. So, how much RAM should your SQL Server have? RAM is used to buffer data of your database, so ideally should be large enough to hold active/ commonly used data. Queries and data manipulation of each SQL request are best performed in RAM, so if your SQL Server is not configured with enough RAM, then the older data will be pushed to disk, which is slower.

We suggest using the following as minimum guidelines:

- A dedicated Server
- Windows Server 2019
 - Windows Server 2016 or greater is required
 - SQL Server 2019 Standard and Express editions will run under Windows 10
- High-speed disk drives for database storage
 - E.g. SSD drives
- Express Edition
 - CPU with at least 4 cores
 - Limited to 1.4GB of RAM, but allow more for Windows OS, so we suggest 8 to 16 GB
- Standard/ Enterprise Editions
 - CPU with at least 8 cores
 - ✦ More if your database is larger than 100GB
 - At least 32GB RAM (7GB will be used by Windows, leaving 25GB for SQL Server)
 - ✦ More if your database is larger than 100GB
 - Also check the limits - see <https://docs.microsoft.com/en-us/sql/sql-server/editions-and-components-of-sql-server-version-15?view=sql-server-ver15> for more information – there is no point installing more RAM than can be used

This is all an educated guess, and it pays to get an expert involved and regularly review performance going forward.

Note, you should have a separate environment for your test and production systems, i.e. do not use your production environment for testing.

Remember also that technology is changing rapidly, so consider upgrading to faster processors, disk drives, networks etc as they become available and cheaper.

NOTE: Intersoft Systems is always updating its software – and this includes performance improvements. If you notice a performance bottleneck, please feel free to talk to your Intersoft Dealer so that we can review it further.

Installing SQL Server

This section is not a full guide to installing SQL Server, rather it includes any important tips to help you.

SQL Server can be downloaded from Microsoft at <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>.

NOTE: For SQL Server Editions except Express you will need to purchase Server and Client licences.

We strongly suggest getting an expert, e.g. your Intersoft Dealer, to install SQL Server as they will have experience in doing so.

Notes for the Express Edition

- Installing using the Basic option is simpler but does not allow you to specify items such as the System Administrator's password.
- We suggest you install using the Custom option and specify these options:
 - "Express Core" package
 - Include SQL Server product updates
 - There will be a warning regarding Windows Firewall. This is covered later in the document, but you can drilldown to the link to review the warning information
 - Give the instance a name like "SQLEXPRESS"
 - ✦ Within finPOWER Connect the Server Name will be [Domain]\SQLEXPRESS, e.g. INTERSOFT\SQLEXPRESS
 - Use the standard Collation order
 - ✦ We suggest **Latin1_General_CI_AS**
 - Select **Mixed Mode (SQL Server authentication and Windows authentication)**
 - ✦ Enter a strong password for your "sa" (System Administrator), and make sure you record this in a safe place
- Once complete you should install SQL Server Management Studio (SSMS)
 - This facility is very important and, for example, provides facilities to backup your databases
 - If the SQL Server Installation Center is still onscreen:
 - ✦ Click Install SQL Server Manager Tools
 - If not, go to <https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15> to download

Windows Server Configuration

For security purposes you need to create a finPOWER Security Group and assign to it those users that will require access to finPOWER Connect.

Add the finPOWER Group

The finPOWER Group allows you to quickly assign new Users to this group, giving them access to all finPOWER Connect databases – rather than assigning individual Users to the database.

To add the finPOWER Security Group

1. Run **Server Manager**
2. Open **Tools** and click **Active Directory Users and Computers**
3. Expand your Domain and click on **Users**
4. From the **Action** menu, point to **New** and click **Group**
5. In the New Object wizard enter details as follows:
 - a. **Group name** as **finPOWER**
 - b. **Group scope** as **Global**
 - c. **Group type** as **Security**
6. Click **OK**.

Assign users to the finPOWER group

Next you will need to add the users that have permission to open finPOWER databases.

To assign Users to the finPOWER Group

1. Run **Server Manager**
2. Open **Tools** and click **Active Directory Users and Computers**
3. Expand your Domain and click on **Users**
4. Right click the **finPOWER** group and click **Properties**
5. Click on the **Members** tab
6. Click **Add** and add the Users required

Windows 10 Configuration

When running SQL Server under Windows 10, the procedure to add a finPOWER Group is slightly different – as you will need to add a Local Group.

To add the finPOWER Security Group

1. From the **Start** menu, click **Run** and run **lusrmgr.msc**
 - a. The Local Users and Groups console should open
2. Right-click **Groups** and click **New Group...**
3. In the New Group wizard enter details as follows:
 - a. **Group name** as **finPOWER**
 - b. Add Users under Members
4. Click **OK**.

Once added you can easily add or remove Users by drilling down to the **finPOWER** group.

SQL Server Configuration

SQL Server uses "**logins**" to control access to the information contained in databases.

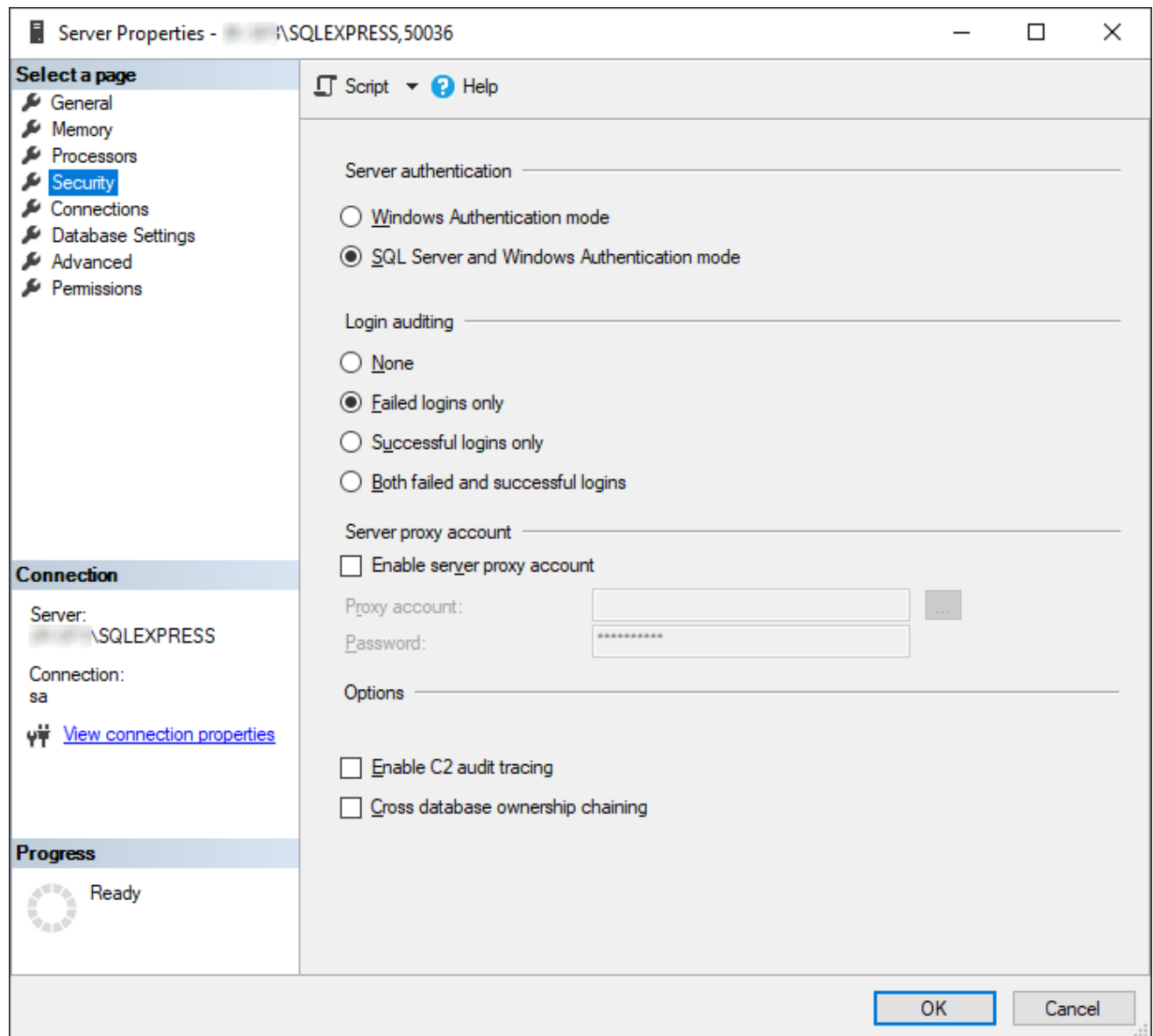
finPOWER relies upon the previously added **finPOWER** login group to control general database access. finPOWER also uses Application Roles to control database permissions.

SQL Server Security Properties

You must setup the SQL Server Security Authentication mode as "**SQL Server and Windows**", sometimes called "Mixed Mode". Whilst some documentation does not recommend this setting it allows finPOWER Connect to automatically create and configure databases.

To configure SQL Server

1. Open SQL Server Management Studio
2. Connect to your Database Engine
3. Expand your Database Engine
4. Right-click on your Database Engine, and click **Properties**
5. Select the **Security** page



6. Under **Server Authentication**, ensure that **SQL Server and Windows Authentication mode** is checked
7. Click **OK**

System Administrator Login

As SQL Server is now configured in "**SQL Server and Windows**" authentication mode the "**sa**" or system administrator login should be properly password protected.

To assign a new password to sa

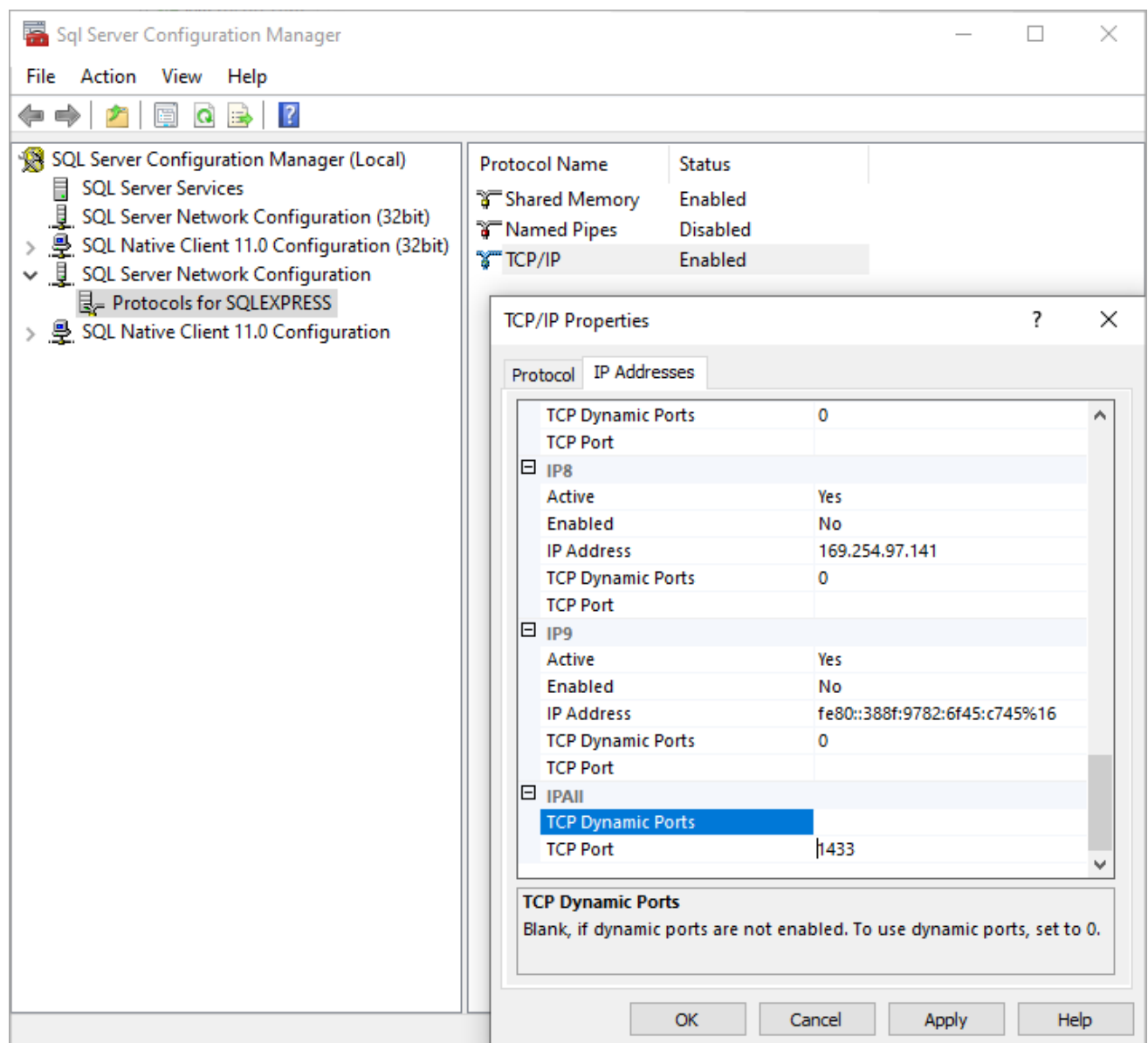
1. Open SQL Server Management Studio
2. Connect to your Database Engine
3. Expand your Database Engine
4. Expand **Security, Logins**
5. Right-click on your "**sa**" Login and click **Properties**
6. Under **SQL Server authentication** update the password

SQL Server Network Protocols

One area that you should make sure is correctly configured is which SQL Server protocols are enabled.

Enable TCP/IP protocol

1. Run **SQL Server Configuration Manager**
2. Expand **SQL Server Network Configuration**
3. Enable **TCP/IP** if disabled
 - a. Right-click **TCP/IP** and click **Enable**
 - b. You may be required to restart your computer at this time
4. Right click **TCP/IP** and click **Properties**
5. Select the **IP Addresses** tab
6. Scroll to **IPAll** section and update the **TCP Port** to **1433**



7. Click **OK**
8. Restart SQL Server so changes take effect

Firewall Configuration

Next you need to ensure the firewall on the Server allows connections via the port specified previously.

Firewalls help prevent unauthorized access to computer resources. If a firewall is turned on but not correctly configured, attempts to connect to SQL Server might be blocked.

Opening a port in Windows firewall

1. From the **Start** menu, click **Run** and run **wf.msc**
2. In **Windows Firewall with Advanced Security**, in the left pane, select **Inbound Rules**.
3. In the **Actions** menu, click **New Rule...**
4. In the Rule Type page, select **Port**, and then click **Next**
5. In the Protocol and Ports page, select **TCP**. In **Specific local ports**, enter the port number of the instance of the Database Engine, 1433 for the default instance (as defined previously)
6. Click **Next**
7. In the Action page, check **Allow the connection**, and then click **Next**
8. In the Profile dialog page, select the profiles that describe the computer connection environment when you want to connect to the Database Engine, and then click **Next**
9. In the Name page, type a name and description for this rule
10. Click **Finish**

Model database

Within SQL Server there are several "system" databases, including the **master**, **model**, **msdb** and **tempdb** databases.

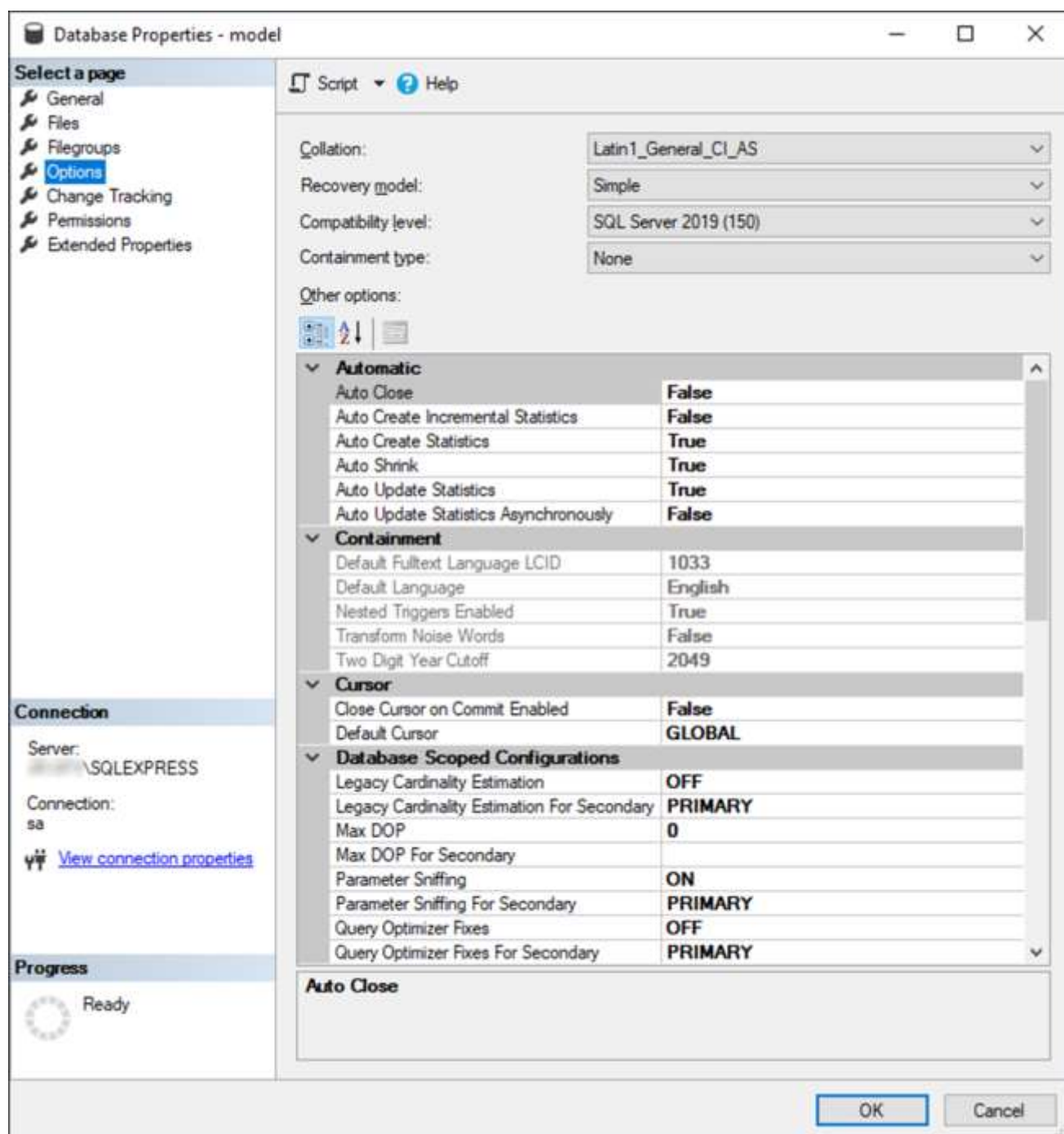
The **model** database is used as the template for all databases created. When a CREATE DATABASE statement is issued, the first part of the database is created by copying in the contents of the model database.

This means all the base settings of a new database are copied from the model database. Therefore, it is strongly recommended that the model database is correctly setup prior to doing anything else.

To check Model database options

1. Open SQL Server Management Studio
2. Connect to your Database Engine
3. Expand **Server, Databases, System Databases**
4. Right-click the **model** database and click **Properties**
5. Select the **Options** page

6. We suggest you configure options as follows:



7. Ensure the following options are set:

- Collation** is either **Latin1_General_CI_AS** or **SQL_Latin1_General_CI_AS**
- Recovery Model** is **Simple**
 - For more information on Recovery options see the Backup topic.
- Auto Create Statistics** is True
- Auto Shrink** is True
- Auto Update Statistics** is True
- ANSI NULL Default** is False

finPOWER Login

The **finPOWER** login is required for general day to day use of finPOWER Connect databases. This maps directly to the Windows **finPOWER** Security Group defined previously.

This means all Users belonging to the **finPOWER** Security Group can access the finPOWER Connect database.

To add the Intersoft Login

1. Open SQL Server Management Studio
2. Connect to your Database Engine
3. Expand **Security**, and then click **Logins**
4. Right-click on **Logins** and click **New Login...**
5. On the **General** page enter details as follows:
 - a. Enter **Login name** as **[Domain]\finPOWER**, e.g. **Intersoft\finPOWER**
 - i. If using **Search**, click **Object Types** and ensure **Groups** are checked
 - b. Check **Windows authentication**
6. Click **OK**

Backups

Database backups are one of the most important safeguards against hardware failure and database corruption problems.

WARNING: You must ensure not only the backup procedure is correctly put into place, but also that the restore process works as expected.

The backup and restore component of SQL Server provides a very easy to use solution for backing up SQL Server databases. One of the most important facets is that a database can be backed up whilst the database is in use – unlike Microsoft Access databases.

SQL Server includes three "**Recovery**" models to choose from, but for most users the "**Simple**" model is sufficient. This model only allows full backups to be made but simplifies transaction log requirements.

We suggest you also have several generations of backups. For example, you might take a backup twice a day (midday and after hours) and keep backups for 10 days. This means if one backup fails you have an older copy.

As noted above, it is prudent to regularly test your backup is working correctly by performing a restore on another Server. In fact, you should not underestimate the importance of a disaster recovery plan.

This is why we strongly recommend talking to your Intersoft Dealer and Networking expert about setting up Backups and a Disaster Recovery Plan.

NOTE: SQL Server Standard and Enterprise Editions include Maintenance Plans that allow you to automate backups. Note, SQL Server Express does not include Maintenance Plans, but this can be scripted.

WARNING: A final word, however you generate backups, make sure that they are also stored offsite in a safe and secure place. This might be in the cloud or at a private residence, either way this protects the backup should the location of the server be destroyed, e.g. by fire or water.

Moving an Access database to SQL Server

All databases start from an Access database template, whether it is a new database or an existing database. There is a very important step which takes the Access database and moves it to a SQL Server database.

For a new database, this process occurs without any visible acknowledgment. However, when an existing database is copied to SQL Server a special finPOWER facility must be used.

Database Naming

All databases are prefixed with **finPOWERConnect_**, e.g. **finPOWERConnect_finDEMO**.

Normally the **finPOWERConnect_** prefix does not need to be entered.

To move an existing Access database to SQL Server

1. Open **finPOWER Connect**
2. From the **Tools** menu, point to **Database** and then click **Copy Database...**

Follow the wizard, selecting the Access database and SQL Server as required. For more information refer to finPOWER Connect help.

Note, this facility can also be used to move a SQL Server database back to an Access database.

Duplicating a SQL Server Database

Because SQL Server stores the database on its own Server, it is not possible to simply copy the raw data files that make up the database (unlike an Access database).

Duplicating a Database using SQL Server Backup and Restore

You can use Microsoft SQL Server Management Studio to create a copy of your database.

To use this facility, you must be logged into SQL Server with **System Administrator** rights.

To Copy a Database using SQL Server Backup and Restore

1. Open Microsoft SQL Server Management Studio
2. Connect to your Database Engine
3. Expand **Databases** and select your Database
4. Right-click your Database and point to Tasks and click **Back Up...**
5. Backup your database as per normal
6. Right-click **Databases**, and click **Restore Database...**
7. On the **General** page change the following:

Source

Enter the source as the Backup just completed

Destination

The name of the new database that you want to create. This must be another valid SQL Server database name, remembering to preserve the **finPOWERConnect_** prefix

8. Click **OK**

The database will then be restored as the new database name and should then be available to use.