SOFTWARE
DESIGN SOLUTIONS

CTC BIM Data Suite Servers
Installation and Configuration Guide

Contents

Audience 3
BIM Data Suite Servers Overview 3
General Requirements 4
Installation Process 7
Custom Installation (Using Command-Line Parameters) 13
Silent Installation 13
Selecting Services to NOT Install 13
Configuring Microsoft SQL Server and Security 14
Creating the Database 14
Permissioning the Database 14
Using a SQL Account (Using SQL Authentication) 15
Using Windows Authentication (Active Directory User Account) 16
Project Activity Server 18
Overview 18
Project Activity Server Configuration 19
BIM Data Admin Server 21
Overview 21
BIM Data Admin Server Configuration 22
Project Snapshot Server 24
Overview 24
Project Snapshot Server Configuration 25
Database Structure 27
Licensing 31
Appendix A – Restoring the Default Empty Database Backup File into SQL Server 32
Appendix B – How to Run a Script in Microsoft SQL Server 36
Appendix C – Changing the Domain Account a Windows Service "Runs As" 39
Appendix D – Configuring Database Connection Strings 45
Determining the SQL Server Instance Name 45
Determining the Database Name 45
Types of Database User Authentication 46
Using SQL Authentication 46
Using Active Directory Authentication 46
Putting it All Together (Examples) 46
Appendix E – How to Open Firewall Ports for CTC Services 48

Audience

The audience for this document consists of IT professionals with experience installing software on servers and with experience working with and administering Microsoft SQL Server databases.

IMPORTANT: Knowledge of administering Windows servers *and Microsoft SQL Server databases* is required to correctly install, configure and use this product.

BIM Data Suite Servers Overview

BIM Data Suite is a product from CTC Software. It consists of tools for recording data about Revit usage and Revit projects.

This suite includes multiple Windows "server" services which communicate between Revit workstations and a Microsoft SQL Server database. This setup installs those services.

There are 4 types of server computers that could be involved for BIM Data Suite:

1) Project Activity Server computer*
2) Project Snapshot Server computer(s)
3) BIM Data Admin Server computer*
4) MS SQL Server Computer(s) running full SQL Server (**NOT** the free Express version) 2008 R2 or later

*These services can readily be run on the same computer. Project Snapshot Server *may* run on the same computer as well, but performance will be dependent on the amount of data and frequency of snapshots being taken.

For implementations to be successful:

- CTC is unable to provide report writing services or technical support for Microsoft SQL Server. As such, **you should have your own Microsoft SQL Server Database Administrator ("DBA") either on staff or readily at your disposal.** *This role is typically very different from a standard IT / infrastructure role.* A person in this role should have special training and expertise with installing, configuring, tuning, reporting from and generally maintaining Microsoft SQL Server. Without a SQL Server DBA available, it is not likely you will have a good experience with BIM Data Suite.

- **SQL Server Express is not supported.** This is primarily due to limitations imposed by Microsoft on this free version of the SQL Server database engine. While SQL Server Express will physically function with BIM Data Suite, and may be acceptable for a small trial environment, for long-term production use it is not supported.

- Microsoft SQL Server 2008 R2 or later is required.

- If using Project Activity Logger (PAL), the CTC Project Activity Server service is required, but is very lightweight and can be installed on most any application server. CTC does not recommend installing this service directly on your SQL Server computer in order to maximize the performance of the SQL database engine.

- If using Project Snapshot Exporter, the CTC Project Snapshot Server service is required and should be installed on a separate, dedicated server computer. This is because it is NOT a lightweight service, and should generally be installed on its own dedicated server computer. Actual server resources needed will vary depending on the size and complexity of the snapshots being taken, as well as the number of snapshots being processed

simultaneously. Disc space requirements are minimal, but a general rule of thumb is that **the dedicated Project Snapshot Server computer should have at least 32 GB of memory and 4 cores.**

- CTC does not make any hardware recommendations for the SQL Server computer(s). This is because the SQL Server computer needs for BIM Data Suite are much harder to quantify. The needs can vary dramatically based on the environment. Are there 30 Revit workstations or 300 Revit workstations? Is data being gathered about 10 different projects or 1000? Will data be kept long-term, or deleted for each project as the project is completed? Are project snapshots being recorded, or only project activity data?

  Project snapshots can be very large. For project snapshot data, it is more likely that a clustered SQL server environment (using multiple SQL server computers working together) may be in order. For example one single, full (unfiltered) snapshot for a medium-sized MEP project may generate several million records.

- For both the Project Snapshot Server computer(s) and the SQL server computer(s), only your IT and SQL teams can make judgements for hardware components after using the software for a reasonable trial period and evaluating the needs based on their environment. For the SQL server in particular, CTC recommends setting up a separate, isolated Test SQL server environment for the trial period, so as not to risk impacting any production environments. This is especially true if capturing snapshot data.

- Project Activity Server requires a license file from CTC for the server computer on which it is installed. Project Snapshot Server does not require a server license because the Project Snapshot Exporter Revit add-in itself is licensed, at the client.

- Reasonably fast, stable network connections are needed between the Revit workstations and the CTC service servers. Reasonably fast, stable network connections are also needed between the CTC service servers and the SQL servers being used.

General Requirements

While the services will install and function correctly on a Windows workstation operating system, it is preferable to install them on a server operating system. They can run on either a physical or virtual machine.

The person installing all software must have administrative privileges on the computer(s) to which it will be installed.

The installation program requires the Microsoft .NET Framework 3.5 to already be installed on the server. The services themselves require the Microsoft .NET Framework 4.5 (or later) be installed as well.

On Windows Server the installed features dialog should look something like this:

If installing on Windows workstation, the "Turn Windows features on or off" dialog should look something like this:

Software which can read Adobe Acrobat files is required in order to read the help documentation.

In order to work with Microsoft SQL Server you will need to have SQL Server Management Studio software installed. This software is required in order to get the database structure installed and configured on SQL server. This software is freely available from Microsoft, and is available from them as a separate download. It may be installed on the SQL server itself and/or on another computer (e.g. your workstation) which has a network connection to the SQL server.

CTC services require TCP ports to be open on the server firewall and accessible through the network from all Revit workstations.

For information about how to open firewall ports, please see Appendix E at the end of this document.

Project Activity Server: **TCP port 5058**
.NET Framework 4.5 or later must already be installed on the server

Project Snapshot Server: **TCP port 5056**
.NET Framework 4.0 or later must already be installed on the server

BIM Data Admin Server: **TCP port 5059**
.NET Framework 4.0 or later must already be installed on the server

These servers can create log files. It is generally not a good practice to have log files be written to an operating system drive (e.g. the C: drive) because they can accumulate and having an operating system drive fill up with data can cause problems with Windows. As such, **at least one secondary local drive (for example, D: or E:) should be available for writing log files**, though log files can be written to another location such as a network share on a different server.

Installation Process

The **CTCBIMDataSuiteServersSetup.msi** file is the installation program. This program must be run by someone with administrative privileges on the server computer. After this installer finishes, the services selected for installation will be started automatically.

IMPORTANT: If prompted for permission to install components, possibly more than one time, grant it each time (e.g. by clicking the "Yes" button).

On the next screen, you must accept the license agreement before you can click the “Next” button to move on to the next screen:

On the next screen you can select which services to install:

The next screen starts the actual installation process:

The next screen shows the progress of the install:

The final setup screen will appear as well as other dialogs, discussed below.

If the checkbox on the last setup screen remains checked when clicking the Finish button, the installation and configuration guide (this document) will be displayed.

Once the above screen appears, one or more configuration screens will appear, depending on which services were selected for installation.

The first is a screen which provides information about getting a license file for the Project Activity Server service:

Once you receive a Project Activity Server license file from CTC, simply placing the license file in the folder shown in the screen above is all that is needed.

IMPORTANT: Project Activity Server licenses expire when your subscription expires. You will be contacted by CTC before your license expires to provide you with the opportunity to renew your subscription and get an updated license file. If the license expires, data gathered will be cached on the local Revit workstations but will NOT be sent to the database. Once the license is renewed, data that had been cached will then be sent to the database.

The other screens that may appear once the installer is finished are used for configuring each service (note the title bar text in each window), as seen below. More details about configuring these services can be found in the sections below, but these are the applications which may appear.

The services are installed to the program files folder, typically:

C:\Program Files\CTC Software\BIM Data Suite Servers

The windows services that get installed will be seen in the Windows Services list, with names which start with:

"CTC"

NOTE: CTC provides other windows services, such as the CTC Network License Server, which have separate installation programs. They may also appear in this list but would not have been installed by this setup program.

By default, the services run as the Local System account, as seen in the next image:

Only the services selected for installation during the setup will appear on this list.

On the Windows Start Menu, under *CTC Software* are menu items for configuring the services and for viewing server information, which is needed for license file generation.

These choices allow you to manually run the programs that appear automatically at the end of the installation (as seen above) in order to change the configuration for each service.

Custom Installation (Using Command-Line Parameters)

Silent Installation

VERY IMPORTANT: Just as when running the setup interactively, when running it silently it must be run with elevated privileges ("as Administrator").

The msi installer supports performing silent installations. A silent installation does not show any dialogs on the screen during the install.

This is accomplished by using the command-line parameter: /q

So an example command to install it silently would be:

CTCBIMDataSuiteServersSetup.msi /q

IMPORTANT: By choosing to do a silent installation, you will NOT be prompted to configure each service installed or shown the computer information for license file generation. If you will be installing the services silently you will need to deploy configuration files, and possibly license files, separately for the services to work correctly.

IMPORTANT: By choosing to do a silent installation, you are automatically agreeing to the software license agreement.

Selecting Services to NOT Install

By default, all services will be installed.

However, command-line parameters can be used to keep one or more services from being installed. The supported command-line parameters are:

InstallActivityServer=No	Can be used to turn OFF installing the CTC Project Activity Server
InstallSnapshotServer=No	Can be used to turn OFF installing the CTC Project Snapshot Server
InstallBIMDatAdminServer=No	Can be used to turn OFF installing the CTC BIM Data Admin Server

Here is an example command line for silently installing BIM Data Suite Servers and not installing the Project Snapshot Server, which is normally installed on its own dedicated server:

CTCBIMDataSuiteServersSetup.msi /q InstallSnapshotServer=No

Configuring Microsoft SQL Server and Security

Creating the Database

Included with the installer is an MS SQL server empty database backup file which contains all of the schema and stored procedures needed by BIM Data Suite.

The backup file is named:

CTCBIMData.bak

The file was created by Microsoft SQL Server 2008 R2, and can only be restored into Microsoft SQL Server version 2008 R2 or later.

The database backup file is installed to the "Default Database" subfolder of the installation folder, typically:

C:\Program Files\CTC Software\BIM Data Suite Servers\Default Database

This empty database backup file needs to be restored into SQL server, thus creating the empty database on the SQL server for the services to use for storing their data.

IMPORTANT: If you need assistance with how to restore the empty database backup file into SQL server, please see Appendix A.

IMPORTANT: This data structure was specifically designed to make it as easy as possible to generate reports about Revit project data using general reporting tools such as PowerBI. It is therefore not in a normalized state, nor does it very often have multi-field primary keys. While both of those aspects could make the database more compact, making changes like that would make report writing much more difficult and possibly report generation much slower.

Permissioning the Database

Once the database has been restored from the backup file, the next step is to define the permissions such that the services can execute the stored procedures and access tables in the database.

A SQL User named ExecuteAllDBStoredProcedures has been pre-defined in the CTCBIMData database, and has permissions to execute the stored procedures in the database by being in the "db_ExecuteAllDBStoredProcedures" Security Role inside the CTCBIMData database.

A SQL User in the CTCBIMData database that's in the "db_ExecuteAllDBStoredProcedures" role (also defined in the database) must be linked to a SQL Login, so it can be called from any of the services installed.

Permissions are typically granted to either a user account defined within the SQL server database itself (a "SQL account") or to a network user account ("Active Directory domain account").

IMPORTANT: **Using a SQL account is much simpler and is preferred for its simplicity**, but not all installations of Microsoft SQL Server allow defining SQL accounts. Some installations require that all security be based on network user accounts only, in which case using an Active Directory network account is the only option.

Using a SQL Account (Using SQL Authentication)

Using a SQL account is the simplest approach in part because it does not require changing who the service "runs as." It also makes testing database connections easier.

If SQL accounts are supported on the SQL server (in addition to Active Directory security) and it is desired to use a SQL account, creating and permissioning a new SQL Login for the restored CTCBIMData database-specific SQL User can be done with the following script, with the variables at the top edited appropriately (defaults are shown).

IMPORTANT: If you need assistance with how to run a script in SQL server, please see Appendix B.

This script is available as a text file in the *Default Database* folder:

C:\Program Files\CTC Software\BIM Data Suite Servers\Default Database

It is called:

PermissionForSQLAccount.sql

```
USE CTCBIMData;
GO

DECLARE @NewLoginUserName varchar(30) = 'CTCBIMDataSuiteUser';
DECLARE @NewLoginPassword varchar(30) = 'CTCB1MD@ta';

-- Create a SQL Login on this database server for which the existing SQL Database User to be associated
DECLARE @cmd varchar(200) = '';

IF NOT EXISTS (SELECT loginname FROM master.dbo.syslogins WHERE name = @NewLoginUserName)
BEGIN
        SET @cmd = 'CREATE LOGIN ' + @NewLoginUserName + ' WITH PASSWORD=''' + @NewLoginPassword + ''', DEFAULT_DATABASE=CTCBIMData;'
        EXEC (@cmd)
END
ELSE print 'User ' + @NewLoginUserName + ' already exists.'

-- Associate the existing SQL Database User with the new SQL Login
SET @cmd =
        'sp_change_users_login ''UPDATE_ONE'', ''ExecuteAllDBStoredProcedures'', ' + @NewLoginUserName
EXEC (@cmd)

GO
```

When using this approach, the connection string for each of the service configurations (see below) should look like this:

```
server=SQLServerInstanceName;database=CTCBIMData;User Id=CTCBIMDataSuiteUser;password=CTCB1MD@ta
```

Where SQLServerInstanceName is replaced with the actual SQL server computer name (and instance name if using named SQL instances). Any other changes made to the query listed above (such as the password for the SQL Login) would need to be changed in the connection string accordingly as well.

Connection strings are discussed in more detail below, in the sections describing how to configure each service.

IMPORTANT: If you to move the database to a different SQL server, you will likely need to run this script on the new server.

Using Windows Authentication (Active Directory User Account)

A Windows domain account can be used to control the access to executing the stored procedures and to the tables in the database. Typically this is done with a dedicated "service account" – a Windows Active Directory network domain user defined solely for the purpose of being used by an application.

It may, for example, have a network domain login ID of: SA_CTCService, with a suitable password.

IMPORTANT: For this to work, once the Active Directory user account exists **the Windows services must each be stopped and changed to "Run As" the new domain account.** Then the Windows services must be started again.

If you need assistance with how to change who a Windows service "Runs As" please see Appendix C.

Next, the following script should be run in SQL server with the variables at the top set appropriately:

IMPORTANT: If you need assistance with how to run a script in SQL server, please see Appendix B.

This script is available as a text file in the *Default Database* folder:

C:\Program Files\CTC Software\BIM Data Suite Servers\Default Database

It is called:

PermissionForActiveDirectoryServiceAccount.sql

```sql
USE CTCBIMData;
GO

-- Set these variables to match your environment
DECLARE @DomainName varchar(30) = 'MyDomain';
DECLARE @ActiveDirectoryUserID varchar(30) = 'SA_CTCService';


DECLARE @cmd varchar(200) = '';

-- Create a SQL Login on this database server for the Active Directory service account if it doesn't already exist.
IF NOT EXISTS (SELECT loginname FROM master.dbo.syslogins WHERE name = @DomainName + '\' + @ActiveDirectoryUserID)
BEGIN
         SET @cmd = 'CREATE LOGIN [' + @DomainName + '\' + @ActiveDirectoryUserID + '] FROM WINDOWS WITH DEFAULT_DATABASE=CTCBIMData;'
         EXEC (@cmd)
END
ELSE print 'Login ' + @DomainName + '\' + @ActiveDirectoryUserID + ' already exists.'


-- Create a new user in the database which matches the domain account user ID if it doesn't already exist.
IF NOT EXISTS (SELECT name FROM [sys].[server_principals] WHERE name = @DomainName + '\' + @ActiveDirectoryUserID)
BEGIN
         print 'not exist'
         SET @cmd =
                  'CREATE USER [' + @ActiveDirectoryUserID + '] FOR LOGIN [' + @DomainName + '\' + @ActiveDirectoryUserID + '];'
         EXEC (@cmd)
END
ELSE print 'User ' + @DomainName + '\' + @ActiveDirectoryUserID + ' already exists.'


-- Grant the new database user permissions to execute all stored procedures
SET @cmd =
         'sp_addrolemember ''db_ExecuteAllDBStoredProcedures'', ' + @ActiveDirectoryUserID
EXEC (@cmd)

-- Grant the new database user read permissions
SET @cmd =
         'sp_addrolemember ''db_datareader'', ' + @ActiveDirectoryUserID
EXEC (@cmd)

-- Grant the new database user write permissions
SET @cmd =
         'sp_addrolemember ''db_datawriter'', ' + @ActiveDirectoryUserID
EXEC (@cmd)

GO
```

When using this approach, the connection string for each of the service configurations (see below) should look like this:

```
server=SQLServerInstanceName;Database=CTCBIMData;Trusted_Connection=True;
```

Where SQLServerInstanceName is replaced with the actual SQL server computer name (and instance name if using named SQL instances).

Connection strings are discussed in more detail below, in the sections describing how to configure each service.

IMPORTANT: If you to move the database to a different SQL server, you will likely need to run this script on the new server.

Project Activity Server

Overview

Project Activity Server is a lightweight Microsoft® Windows® service that is the connection link between the Project Activity Logger add-in for Revit and a Microsoft SQL Server database.

A typical installation looks something like this:

Project Activity Server software can be installed directly on the MS SQL Server computer, or on a separate server computer (strongly recommended), as shown above.

Project Activity Server is a lightweight service, handling very small amounts of data from each Revit user.

Using Project Activity Server, the Project Activity Logger add-in for Revit only needs to know the name of the computer on which the Project Activity Server software is installed as part of their configuration, without having to know anything about the Microsoft SQL Server being used, how to connect to that SQL server, or requiring any permissions in SQL server by the Revit user as well.

If the SQL database needs to be moved to another server, or the security on it needs to change, simply changing the configuration of the Project Activity Server software to match the SQL server changes is all that's needed. **Each of the Revit workstations do not need to be changed.**

The only time each of the Revit workstations needs to be updated is if the Project Activity Server itself is installed to a new server computer.

Project Activity Server Configuration

Project Activity Server's configuration is controlled by an XML file located in the installation folder. Typically, this file will be:

C:\Program Files\CTC Software\BIM Data Suite Servers\Project Activity Server\PALDataServiceConfig.xml

This file can be edited using a text editor (such as Notepad) or by using the provided configuration tool, which runs automatically at the end of the software installation:

This tool can also be launched from the Start menu, under CTC Software, called "Configure Project Activity Service"

NOTE: Changing the service configuration file DOES NOT require restarting the Windows service. Changes take effect immediately.

The connection string will need to be changed to match the location of the MS SQL Server database (CTCBIMData) with which Project Activity Server will be working. At the very least the "SQLServerInstanceName" value will need to be changed.

The above example shows a connection string configured for using a SQL account for database security (SQL authentication). If using integrated Active Directory security in the database instead, a different connection string will be needed.

IMPORTANT: If you need guidance for configuring the connection string(s), please see Appendix D.

The "Test" button can be used to ensure the Project Activity Server will be able to communicate correctly with the database when using the connection string.

IMPORTANT: Once the correct permissions are set on the database **the "Test" results will only be valid if you're either using SQL authentication or if you're using Active Directory authentication and are logged in with the same credentials the service is using ("running as").**

If you're using Active Directory authentication and are logged in and running this program as a different account (e.g. yours) than the service account "runs as", that account (yours) may not have permissions in the database and thus a false security error will occur when testing the connection.

Alternately, if the network account you are currently logged in as has permissions in the database but the service account does not have permissions, a false success result may be returned.

This is one of the reasons why using SQL authentication is simpler.

IMPORTANT: The default location for log files is within the folder to which the services have been installed. However, this is located on the C: drive. As log files can accumulate over time, **it is generally NOT a good idea to have log files be written anywhere on the C: drive** because having the operating system drive become full can cause problems with Windows. As such, it is recommended to write log files to a different local drive, such as a D: or E: local drive.

IMPORTANT: If log files are to be written to a different location than the default location and Active Directory security is to be used with SQL server, read and write permissions for service account being used to run the services may need to be added to the log folders. This is discussed in the later portion of Appendix C.

BIM Data Admin Server

Overview

BIM Data Admin Server is a lightweight Microsoft® Windows® service that is the connection link between the BIM Data Admin Windows application and a Microsoft SQL Server database.

A typical installation looks something like this:

BIM Data Admin Server software can be installed directly on the MS SQL Server computer, or on a separate server computer (strongly recommended), as shown above.

BIM Data Admin Server is a lightweight service, handling very small amounts of data from users designated for managing project definitions in the database and providing some data cleanup functionality as well.

Using BIM Data Admin Server, the BIM Data Admin application only needs to know the name of the computer on which the BIM Data Admin Server software is installed as part of its configuration, without having to know anything about the Microsoft SQL Server being used, how to connect to that SQL server, or requiring any permissions in SQL server by the Windows user as well.

If the SQL database needs to be moved to another server, or the security on it needs to change, simply changing the configuration of the BIM Data Admin Server software to match the SQL server changes is all that's needed. **Each of the Revit workstations do not need to be changed.**

The only time each of the workstations needs to be updated is if the BIM Data Admin Server itself is installed to a new server computer.

BIM Data Admin Server Configuration

BIM Data Admin Server's configuration is controlled by an XML file located in the installation folder. Typically, this file will be:

C:\Program Files\CTC Software\BIM Data Suite Servers\BIM Data Admin Server\BIMDataAdminServiceConfig.xml

This file can be edited using a text editor (such as Notepad) or by using the provided configuration tool, which runs automatically at the end of the software installation:

This tool can also be launched from the Start menu, under CTC Software, called "Configure BIM Data Admin Service"

NOTE: Changing the service configuration file DOES NOT require restarting the Windows service. Changes take effect immediately.

The connection string will need to be changed to match the location of the MS SQL Server database (CTCBIMData) with which BIM Data Admin Server will be working. At the very least the "SQLServerInstanceName" value will need to be changed.

The above example shows a connection string configured for using a SQL account for database security (SQL authentication). If using integrated Active Directory security in the database instead, a different connection string will be needed.

IMPORTANT: If you need guidance for configuring the connection string(s), please see Appendix D.

The "Test" button can be used to ensure the BIM Data Admin Server will be able to communicate correctly with the database when using the connection string.

IMPORTANT: Once the correct permissions are set on the database **the "Test" results will only be valid if you're either using SQL authentication or if you're using Active Directory authentication and are logged in with the same credentials the service is using ("running as").**

If you're using Active Directory authentication and are logged in and running this program as a different account (e.g. yours) than the service account "runs as", that account (yours) may not have permissions in the database and thus a false security error will occur when testing the connection.

Alternately, if the network account you are currently logged in as has permissions in the database but the service account does not have permissions, a false success result may be returned.

This is one of the reasons why using SQL authentication is simpler.

IMPORTANT: The default location for log files is within the folder to which the services have been installed. However, this is located on the C: drive. As log files can accumulate over time, **it is generally NOT a good idea to have log files be written anywhere on the C: drive** because having the operating system drive become full can cause problems with Windows. As such, it is recommended to write log files to a different local drive, such as a D: or E: local drive.

IMPORTANT: If log files are to be written to a different location than the default location and Active Directory security is to be used with SQL server, read and write permissions for service account being used to run the services may need to be added to the log folders. This is discussed in the later portion of Appendix C.

Project Snapshot Server

Overview

Project Snapshot Server is a Microsoft® Windows® service that is the connection link between the Project Snapshot Exporter add-in for Revit from the CTC BIM Data Suite and a Microsoft SQL Server database.

A typical installation looks something like this:

Project Snapshot Server software can be installed directly on the MS SQL Server computer, but installing it on a separate, **dedicated** server computer is STRONGLY recommended, as shown above.

IMPORTANT: Project Snapshot Server is **NOT** a lightweight service if many or very large snapshots are to be saved. Large snapshots may require a significant amount of memory be available on the server computer on which Project Snapshot Server is installed. Having a 64-bit server with 32 GB of memory and at least 4 cores should generally be a good rule of thumb, however monitoring the server during a large snapshot save to the database to gauge server load for your environment is the best approach to determining actual server hardware needs.

IMPORTANT: Project snapshots can be large. It is extremely unlikely that using a free version of MS SQL Server (e.g. SQL Server Express) will allow storing more than a handful of snapshots before the database size limits imposed by the free version will prevent saving more data, thus generating errors. As such, except for small trials of the software, **taking project snapshots using MS SQL Server Express is neither recommended nor supported**.

Using Project Snapshot Server, the Project Snapshot Exporter add-in for Revit only needs to know the name of the computer on which the Project Snapshot Server software is installed as part of their configuration, without having to know anything about the Microsoft SQL Server being used, how to connect to that SQL server, or requiring any permissions in SQL server by the Revit user as well.

If the SQL database needs to be moved to another server, or the security on it needs to change, simply changing the configuration of the Project Snapshot Server software to match the SQL server changes is all that's needed. **Each of the Revit workstations do not need to be changed.**

The only time each of the Revit workstations needs to be updated is if the Project Snapshot Server itself is installed to a new server computer.

The service is installed to the 32-bit program files folder, typically:

C:\Program Files\CTC Software\BIM Data Suite Servers\Project Snapshot Server

The windows service can be found in the Windows Services list, with the name "CTC Project Snapshot Server."

By default, the service runs as the Local System account.

Project Snapshot Server Configuration

Project Snapshot Server's configuration is controlled by an XML file located in the installation folder. Typically, this file will be:

C:\Program Files\CTC Software\BIM Data Suite Servers\Project Snapshot Server\ ProjectSnapshotServiceConfig.xml

This file can be edited using a text editor (such as Notepad) or by using the provided configuration tool, which runs automatically at the end of the software installation:

This tool can also be launched from the Start menu, under CTC Software, called "Configure Project Snapshot Service"

NOTE: Changing the service configuration file DOES NOT require restarting the Windows service. Changes take effect immediately.

The connection string will need to be changed to match the location of the MS SQL Server database (CTCBIMData) with which Project Snapshot Server will be working. At the very least the "SQLServerInstanceName" value will need to be changed.

The above example shows a connection string configured for using a SQL account for database security (SQL authentication). If using integrated Active Directory security in the database instead, a different connection string will be needed.

IMPORTANT: If you need guidance for configuring the connection string(s), please see Appendix D.

The "Test" button can be used to ensure the Project Snapshot Server will be able to communicate correctly with the database when using the connection string.

IMPORTANT: Once the correct permissions are set on the database **the "Test" results will only be valid if you're either using SQL authentication or if you're using Active Directory authentication and are logged in with the same credentials the service is using ("running as").**

If you're using Active Directory authentication and are logged in and running this program as a different account (e.g. yours) than the service account "runs as", that account (yours) may not have permissions in the database and thus a false security error will occur when testing the connection.

Alternately, if the network account you are currently logged in as has permissions in the database but the service account does not have permissions, a false success result may be returned.

This is one of the reasons why using SQL authentication is simpler.

IMPORTANT: The default location for log files is within the folder to which the services have been installed. However, this is located on the C: drive. As log files can accumulate over time, **it is generally NOT a good idea to have log files be written anywhere on the C: drive** because having the operating system drive become full can cause problems with Windows. As such, it is recommended to write log files to a different local drive, such as a D: or E: local drive.

IMPORTANT: If log files are to be written to a different location than the default location and Active Directory security is to be used with SQL server, read and write permissions for service account being used to run the services may need to be added to the log folders. This is discussed in the later portion of Appendix C.

Database Structure

On the services server resides an image file (**DatabaseDiagram_CTCBIMData.png**) which shows the database diagram for the entire database. The image can be found in the Default Database folder:

C:\Program Files\CTC Software\BIM Data Suite Servers\Default Database

This diagram is also the only diagram in the CTCBIMData database, and is named "dbo.Diagram_All".

IMPORTANT: This data structure was specifically designed to make it as easy as possible to generate reports about Revit project data using general reporting tools such as PowerBI. It is therefore not in a normalized state, nor does it very often have multi-field primary keys. While both of those aspects could make the database more compact, making changes like that would make report writing much more difficult and possibly report generation much slower.

In these diagrams, parent tables are shown to the left of child tables. In the first diagram below, the tables shown are used for project activity logs. In the second diagram, the tables shown are used for project snapshots. Keep in mind all of these tables reside in the same database, but only the pertinent ones are shown on each page below for clarity.

Project activity data is only collected when using central files.

NOTE: Except for primary key fields, only a few other fields in the tables used for project snapshots are specially indexed. Specifically the ElementDefinition table has an index for the combination of ProjectFileSnapshotID and CategoryName. The ElementInstance table has an index defined for its same fields (ProjectFileSnapshotID and CategoryName).

Other than those two, along with primary keys, no other indexes have been predefined for snapshot tables in order to keep from placing additional load on the SQL server for managing those indexes which may not be used by most customers of this product. Given that snapshots can be so large, the load on the database server for unused indexes could be large as well.

As such, you may get better reporting performance by adding your own indexes which are helpful to your environment.

CTCBIMData Schema (project activity tables only):

CTCBIMData Schema (project snapshot tables only):

NOTE: For snapshot tables, the term "Element" can often be mentally substituted for the more common Revit term "Family." For example:

ElementDefinition → FamilyDefinition (something you'd see at the top level under *Families* in the Project Browser)
ElementType → FamilyType (something you'd see within each Family in the Project Browser)
ElementInstance → FamilyInstance (each item that has been placed in the model)

The reason the term Element is used instead of Family is because not all items stored in the tables are considered Revit families. For example, information about levels, views and materials are stored in these tables. None of these things is considered a "family" in Revit.

In the case of views a single, common definition (within a snapshot) for all views exists in the ElementDefinition table whose name is: "<System> Views"

For that view, multiple rows may exist in the ElementType table, whose names are the view type name (e.g. "Floor Plan", "Detail" etc.)

Information about each view in the project is then stored in its own separate row in the ElementInstance table. The ElementParameterValue table will then contain rows about all of the parameters tied to each view instance.

ElementDefinition table, note several Door definitions exist but only one for all views:

Results | Messages

	ID	ProjectFileSnapshotID	Name	Category	CategoryIsAnnotation	CategoryIsInstanceCreatable	CategoryIsFamilyFileLoadable	IsSystemFamily	TypeCount	InstanceCount	Param
1	-2000279	39AC5926-5F9D-40B4-AF6B-C858AFF36862	<System> Views	Views	0	1	0	1	7	27	1368
2	218942	39AC5926-5F9D-40B4-AF6B-C858AFF36862	Single-Flush	Doors	0	1	1	0	1	7	492
3	768490	39AC5926-5F9D-40B4-AF6B-C858AFF36862	Pocket_Slider_Door_5851	Doors	0	1	1	0	1	3	272
4	907033	39AC5926-5F9D-40B4-AF6B-C858AFF36862	M_Double-Flush	Doors	0	1	1	0	1	1	116
5	910123	39AC5926-5F9D-40B4-AF6B-C858AFF36862	Curtain Wall Dbl Glass	Doors	0	1	1	0	1	3	230
6	930303	39AC5926-5F9D-40B4-AF6B-C858AFF36862	Entrance door	Doors	0	1	1	0	1	2	157

Here are the rows in the ElementType table for the ElementDefinition for all views (with ID -2000279):

Results | Messages

	ID	ProjectFileSnapshotID	ElementDefinitionID	Name	IsSystemFamilyType	InstanceCount	ParameterValueCount
1	49552	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Floor Plan	0	3	184
2	49554	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Building Section	0	3	173
3	49555	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Detail	0	4	241
4	49557	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Building Elevation	0	4	229
5	49559	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	3D View	0	7	330
6	49563	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Rendering	0	5	147
7	958300	39AC5926-5F9D-40B4-AF6B-C858AFF36862	-2000279	Wall Section	0	1	64

As there were 3 floor plan views in the project, the ElementInstance table contains the records for each floor plan view instance in the model (with Type ID 49552).

Results | Messages

	ID	ProjectFileSnapshotID	ElementTypeID	Category	CreatedPhase	DemolishedPhase	IsDesignOptionPrimary	LevelElementID	LevelName
1	312	39AC5926-5F9D-40B4-AF6B-C858AFF36862	49552	Views	NULL	NULL	0	NULL	NULL
2	8251	39AC5926-5F9D-40B4-AF6B-C858AFF36862	49552	Views	NULL	NULL	0	NULL	NULL
3	245424	39AC5926-5F9D-40B4-AF6B-C858AFF36862	49552	Views	NULL	NULL	0	NULL	NULL

Licensing

Some services, such as Project Activity Server, require a license file in order to process requests correctly. For CTC to generate a license file, information about the server computer on which the service has been installed must be provided.

The server information for the computer is displayed when the installation completes in this dialog:

This program can be manually launched from the Start menu as well ("Show Computer Information") from the CTC Software section.

Once you receive the license file, simply place it in the installation folder seen in the image above. Restarting the service is NOT required.

Appendix A – Restoring the Default Empty Database Backup File into SQL Server

In order for the BIM Data Suite Servers to store data in a Microsoft SQL Server instance, a database with the correct name and structure must be defined.

The BIM Data Suite Servers installation includes an empty database backup file which needs to be restored into the SQL server.

To restore the database, you must connect the SQL Management Studio software to the SQL server using credentials with enough permissions to create a database.

The procedure for doing this is as follows:

1) Copy the CTCBIMData.bak file from the installation folder on the server to which the services have been installed to a LOCAL hard drive (e.g. C: or D:) on the SQL server.

   On the server where the services have been installed, this file should be found at this location:

   C:\Program Files\CTC Software\BIM Data Suite Servers\Default Database\CTCBIMData.bak

   For the purposes of this example, copy this file to a temporary folder on the C: drive of the SQL server, for example:

   C:\Temp

2) From any computer, launch SQL Management Studio and connect to the SQL Server computer. For example, if logged on to the SQL server itself:

3) Right-click on the “Databases” node in Object Explorer and select the “Restore Database…” option:

4) Select the “Device” option, then click on the ellipses (…) button to browse for the file to restore:

5) In the next dialog, click the Add button:

6) Use the resulting "Locate Backup File" dialog to select:

C:\Temp\CTCBIMData.bak

7) Back on the "Select backup devices" dialog, click the OK button:

8) Back on the "Restore Database" dialog, simply click the "OK" button. The following dialog should appear:

You should now see the “CTCBIMData” database appear in the list of available databases:

IMPORTANT: Although the database now exists in SQL server, permissions still need to be applied to it before the BIM Data Suite servers can use the database.

Please refer to the section above, “Permissioning the Database” for instructions on how to apply the correct permissions.

Appendix B – How to Run a Script in Microsoft SQL Server

Sometimes you need to give commands to SQL server, such as to query data or to change system settings. For the CTCBIMData database you'll need to apply permissions to the database so the services can use it, which is done by running a script.

To run a script in SQL Server:

1) Open the SQL Server Management Studio software and connect to the SQL server which houses the database to query or modify.

2) In the Object Explorer, expand the Databases node and select the database of interest. This will help automatically tell the script to which database it should be applied.

3) In the toolbar, click the "New Query" button.

4) A new query window will appear, as will a new toolbar:

The new query window is where you enter the text of the query to run.

IMPORTANT: Ensure the name of the correct database on which to act appears in the selection list in the new toolbar. If the correct name isn't displayed, change the selected database using the dropdown arrow.

This step is not strictly necessary if the script itself tells SQL Server Management studio the name of the database on which to operate with the USE command, which can be seen at the top of the script in the next example.

5) Once the script to run has been entered into the query window, click the "Execute" button in the ribbon to run the script.

6) After the script completes executing a results messages window will appear below the script. If there were any errors in the script or problems executing the script these issues will appear in red text in the messages window.

7) When closing the query window, SQL Management Studio may ask you if you want to save the query. This is typically not needed.

Choosing one of the permissioning scripts and running it using this approach is needed in order to configure the CTCBIMData database correctly for use by the BIM Data Suite Servers services.

This is explained in the section above, "Permissioning the Database."

Appendix C – Changing the Domain Account a Windows Service "Runs As"

When accessing a SQL Server database, as the BIM Data Suite Servers services do, information has to be provided to SQL Server about **who** is accessing it in order for SQL Server to verify that the user has permissions to do what they request.

This can be done either by using a "SQL Account" (defined as a user in the database itself) or by using an "Active Directory" (network) user account. An Active Directory account is usually the type of account used by all people who login to the network.

While SQL Accounts are simpler to use, some I.T. people prefer to always use Active Directory accounts, as those can be more secure.

To use Active Directory security with the CTCBIMData SQL server database requires changing the domain user account the various services (Project Activity Service, Project Snapshot Service and BIM Data Admin Service) "run as." This means that when those services are running, they run as if they were a specific person who logged into the network and is requesting access to something.

By default these services are installed to run as the "Local System" account, which is the standard account for Windows services. However, the Local System account is not recognized by other computers. So when the services run as Local System and try to access a database hosted on another computer which requires Active Directory security, the other computer (SQL Server computer) doesn't know who they are and will therefore deny access.

In order for both servers to recognize the same user account, a network domain (Active Directory) user account must be used.

For this example, we will use a domain user that has been set up by the I.T. group which is named: SA_CTCService

IMPORTANT: You will need to know the password for the user account that is to be used.

To change a service to "run as" the newly-created SA_CTCService domain user account:

1. Be logged into the server computer as someone with Administrative privileges
2. Open the Computer Management program
3. Expand the "Services and Applications" item in the navigation tree on the left
4. Click on the "Services" item
5. Right-click on the service of interest (e.g. "CTC BIM Data Admin Server") and select Properties:

6. Switch to the “Log On” tab:

7. Select “This account” and specify the domain user name and password, then click OK:

(For this example, substitute “MyDomain” for whatever your domain name is)

8. The first time you assign a domain user account to a service on this computer, you’ll likely see this message:

   The account MyDomain\SA_CTCService has been granted the Log On As A Service right.

   This message will not appear if this domain user account had previously been granted Log On As A Service rights.

9. Once the setting has been made, you’ll likely see this message:

   The new logon name will not take effect until you stop and restart the service.

   To restart the service, right-click on it in the Services list and select the Restart choice:

10. Repeat this process for all the affected BIM Data Suite Servers services that have been installed on this computer.

11. Once the service(s) have been configured to run as the domain user account and been restarted, we must also grant that account permissions to read and write files to the folder in which the services were installed, **and any other folders the services need to read from or write to, for example, folders where log files are to be written.**

    For a default BIM Data Suite servers installation, the services all use this folder, and folders beneath it:

    C:\Program Files\CTC Software\BIM Data Suite Servers

    At a minimum, the SA_CTCService domain account must be granted read and write permissions to this folder.

    IMPORTANT: The default installation of BIM Data Suite Servers grants "Authenticated Users" read and write permissions to all the folders it installs. Because of this, an Active Directory domain account the service is running-as should automatically have all the permissions it needs, but if you use a service configuration program(s) to change the folder(s) in which the log files are to be written, the SA_CTCService domain account must also be manually granted read and write permissions to those folders.

    For example, if the configuration programs are configured to store their respective log files in:

    D:\BIM Data Suite Servers Logs\BIM Data Admin Server
    D:\BIM Data Suite Servers Logs\Project Activity Server
    D:\BIM Data Suite Servers Logs\Project Snapshot Server

    The SA_CTCService domain account must be granted read and write permissions to those folders.

    For this example, all we need to do is change permissions on the parent folder:

    D:\BIM Data Suite Servers Logs

because all of the subfolders will automatically get those permissions applied as well.

12. Use File Explorer to navigate to the parent folder of the folder whose permissions need to be changed. In this example: D:\

13. Right-click on the BIM Data Suite Servers Logs folder and select Properties from the pop-up menu

14. On the resulting dialog, switch to the Security tab and select the Edit button:

15. In the resulting dialog, click the Add button:

16. On the resulting dialog, type in the name of the domain user account being used to run the services. In our example, this is the SA_CTCService domain account:

17. Back on the previous dialog the CTC Service account has been added to the list. You must select this account and then place a checkmark the "Modify" box in the "Allow" column in order to grant the SA_CTCService account read and write permissions to this folder.

18. Click OK on this dialog, and OK on the previous dialog ("BIM Data Suite Servers Logs Properties") to apply the new permissions settings to the folder.

Appendix D – Configuring Database Connection Strings

In order to have software be able to work with a SQL Server Database it must be able to communicate with the database, and in defining that communication convey to the database who is trying to work with it. This allows the database engine to verify that the user trying to communicate with it has permissions to do what they ask.

This is done by defining a "connection string" which contains commands for communicating with the database. Sections of the connection string are separated by the semicolon (;) character.

The first part of the connection string typically defines the SQL Server instance name.

Determining the SQL Server Instance Name

SQL Server can be installed multiple times on the same server computer. Because this is an option, *every time SQL Server is installed it must have a unique name*. This is called a SQL Server Instance Name.

During the installation of SQL Server, if you do not configure a specific instance name a default name will be used, if it has not already been by a previous SQL Server installation on the same computer.

The default instance name for the free version of SQL Server (Microsoft SQL Server Express) is: SQLEXPRESS

The default instance name for the full versions of SQL Server is: MSSQLSERVER

When connecting to the database, the instance name structure is:

server=<Server Computer Name>\<Instance Name>;

Examples:

server=MyComputer\SQLEXPRESS;
server=MyComputer\MSSQLSERVER;

IMPORTANT: ONLY if the instance name is MSSQLSERVER can just the server computer name be used. For example:

server=MyComputer;

You must know the server computer name and instance name in order to define the connection string.

Determining the Database Name

The second part of the connection string is typically the name of the database being accessed. For all BIM Data Suite Servers, this name is:

database=CTCBIMData;

Types of Database User Authentication

The last part of the connection string defines which user is trying to access the database.

There are two types of ways to tell the SQL server what user is trying to access the database:

1) SQL Authentication. This is the simplest. It involves creating a user account (user name and password) in the database itself.

2) Active Directory Authentication. This can be more complex to implement in the rest of the system, as discussed in more detail in Appendix C. It involves using a domain user account, just like the one you use to login to the network every day.

Using SQL Authentication

When using SQL Authentication, the connection string components for specifying the user credentials are:

User Id=<User name>;password=<password>;

Because the password is visible as plain text in the connection string, some people feel it is less secure. Given that this information is only stored on the service computer (and not each user's workstation) it is secure as the access given to that service computer.

Using Active Directory Authentication

When using SQL Authentication, the connection string components for specifying the user credentials are always:

Trusted_Connection=True;

Putting it All Together (Examples)

Example 1 – SQL Authentication with SQL Server Default Instance

In this example, SQL authentication is being used on a server called MyServer with the default SQL Server instance (MSSQLSERVER) using the default username and password provided by the *PermissionForSQLAccount.sql* script:

server=MyServer;database=CTCBIMData;User Id=CTCBIMDataSuiteUser;password=CTCB1MD@ta

Example 2 – SQL Authentication with SQL Server Named Instance

In this example, SQL authentication is being used on a server called MyServer with a SQL Server instance name of MyInstance using the default username and password provided by the *PermissionForSQLAccount.sql* script:

server=MyServer\MyInstance;database=CTCBIMData;User Id=CTCBIMDataSuiteUser;password=CTCB1MD@ta

Example 3 – SQL Authentication with SQL Server Express

In this example, SQL authentication is being used on a server called MyServer using MS SQL Server Express edition with the SQL Server default instance name of SQLEXPRESS using the default username and password provided by the *PermissionForSQLAccount.sql* script:

server=MyServer\SQLEXPRESS;database=CTCBIMData;User Id=CTCBIMDataSuiteUser;password=CTCB1MD@ta

Example 4 – Active Directory Authentication with SQL Server Default Instance

In this example, Active Directory authentication is being used on a server called MyServer with the default SQL Server instance (MSSQLSERVER) after having run the *PermissionForActiveDirectoryServiceAccount.sql* script:

server=MyServer;database=CTCBIMData;Trusted_Connection=True;

Example 5 – Active Directory Authentication with SQL Server Named Instance

In this example, Active Directory authentication is being used on a server called MyServer with a SQL Server instance name of MyInstance after having run the *PermissionForActiveDirectoryServiceAccount.sql* script:

server=MyServer\MyInstance;database=CTCBIMData;Trusted_Connection=True;

Example 6– Active Directory Authentication with SQL Server Express

In this example, Active Directory authentication is being used on a server called MyServer using MS SQL Server Express edition with the default SQL Server instance name of SQLEXPRESS after having run the *PermissionForActiveDirectoryServiceAccount.sql* script:

server=MyServer\SQLEXPRESS;database=CTCBIMData;Trusted_Connection=True;

Appendix E – How to Open Firewall Ports for CTC Services

In order for a computer to send a message to another computer over a TCP/IP network, both computers must allow messages to be sent over what are called firewall ports.

For BIM Data Suite Servers services, the following TCP ports must be open on both the server and any remote computer (e.g. a Revit workstation) that needs to communicate with the server:

Project Activity Server: **TCP port 5058**
Project Snapshot Server: **TCP port 5056**
BIM Data Admin Server: **TCP port 5059**

On most server computers the needed ports WILL NOT be open by default, but on most workstation computers the ports WILL be open by default.

In this example we will create two rules, one "inbound" rule and one "outbound" rule that each allow a range of TCP ports to be open. Inbound rules allow messages to be sent to this computer while outbound rules allow messages to be sent from this computer to another computer. Since all CTC products work within a range of ports, we'll open all the ports any CTC product might need. However, it's worth noting that the best practice is to open exactly and only the ports known to be needed.

This example will be from Windows Server 2012, however the screens and settings are essentially the same for many versions of Windows.

1) Open Control Panel
2) Select *Windows Firewall* (or *System and Security* then *Windows Firewall* depending on your screen layout)
3) If the "Windows Firewall state" is Off, you don't need to do any more and can skip the remaining steps.
4) In the left column, select *Advanced settings*:

5) In the left column, click the *Inbound Rules* choice, then in the right column click the *New Rule…* choice:

6) On the first screen of the New Inbound Rule Wizard dialog, select *Port* and click the *Next* button:

7) On the next screen of the wizard, select *TCP* and *Specific local ports*, then enter the port range: 5050-5070

Then click the Next button

8) On the next screen, leave *Allow the connection* selected and click the *Next* button:

9) The next choices may vary based on your environment. Typically selecting *Domain* and *Private* should work for the majority of environments.

10) Provide a name and description for the new rule, then click the Finish button.

The new rule should appear in the list:

11) To create the outbound rule, select the *Outbound Rules* option in the left column, then click the *New Rule…* option in the right-most column:

12) On the first page of the New Outbound Rule Wizard, select the *Port* option and click the *Next* button:

13) On the next screen, select the TCP option and enter the same ranges of ports: 5050-5070, then click *Next*.

14) Select the *Allow the connection* option and click the *Next* button:

15) The next choices may vary based on your environment. Typically selecting *Domain* and *Private* should work for the majority of environments.

16) Provide a name and description for the new rule, then click the Finish button.

17) The new rule should appear in the list:

With the inbound and outbound rules now created, the CTC services should be allowed to communicate to and from this computer.