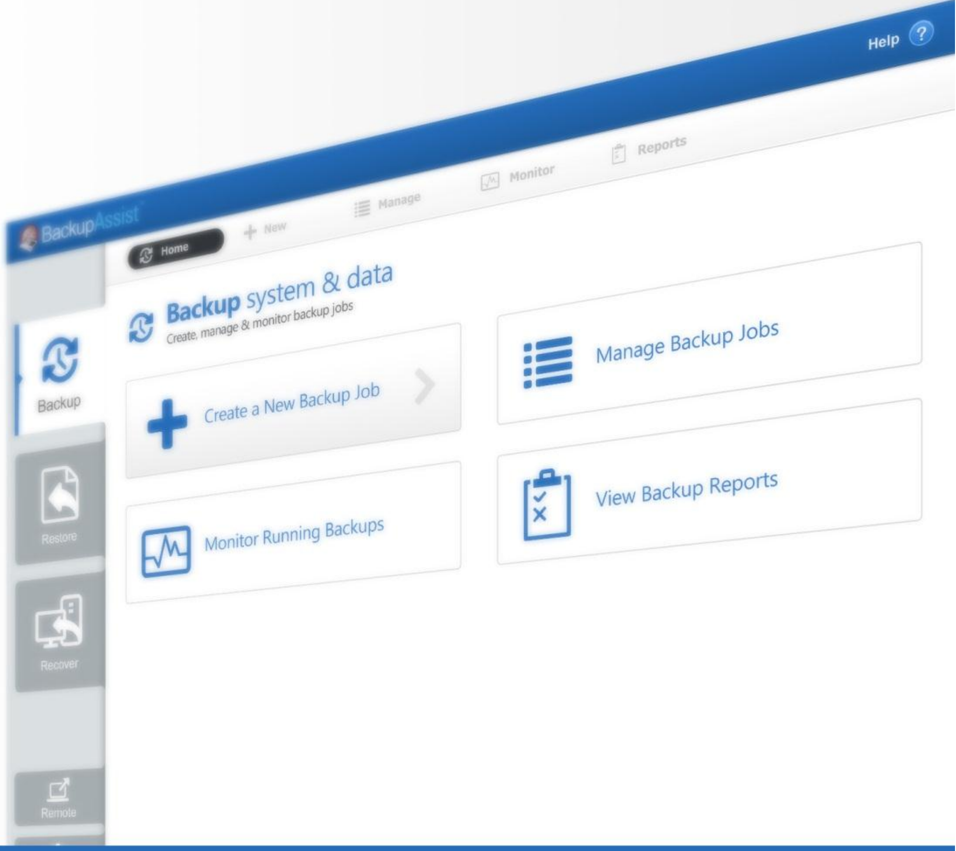


BackupAssist™

BackupAssist v7

SQL Server Protection

Whitepaper

backupassist.com

Cortex I.T.

Contents

1. Introduction

SQL Server Protection
For near continuous database protection
Method: SQL Backup
Backup to Disk
Requires add on license

BackupAssist SQL Server Protection allows you to back up specific databases or complete servers. The backups can be daily or near-continuous (minimum every 15 minutes) for maximum data protection.

BackupAssist System Protection, File Protection and File Archiving solutions can all back up SQL servers using the Volume Shadow copy Service (VSS), but these solutions have limitations in their ability to restore specific databases and to specific points in time.

SQL Server databases are dynamic, shared and critical. For these reasons, BackupAssist SQL Server Protection has been developed to perform live, online backups of SQL Server databases. The backups are stored in the standard SQL Server backup format (.bak file), which allows for the restoration of an entire SQL Server (including the master database) or individual databases. You can also perform advanced tasks such as migrating databases from one server to another.

Documentation

This whitepaper provides a comprehensive guide to BackupAssist SQL Server Protection and can be used in conjunction with other BackupAssist guides.

- For information on the BackupAssist Backup tab, see the BackupAssist Backup Tab Whitepaper.
- For information on the BackupAssist Restore tab, see the BackupAssist Restore Tab Whitepaper.
- For information on the BackupAssist Recover tab, see the BackupAssist Recover Tab Whitepaper.
- For information on performing remote backups see the BackupAssist Central Administration Guide

Licensing

SQL Server Protection requires the *SQL Add-on* license, once the initial trial period has expired. Please contact your local BackupAssist reseller or distributor for pricing information, or visit www.BackupAssist.com.

For instructions on how to activate / deactivate license keys, visit our Licensing BackupAssist page.

The benefits of using the SQL Server Add-on

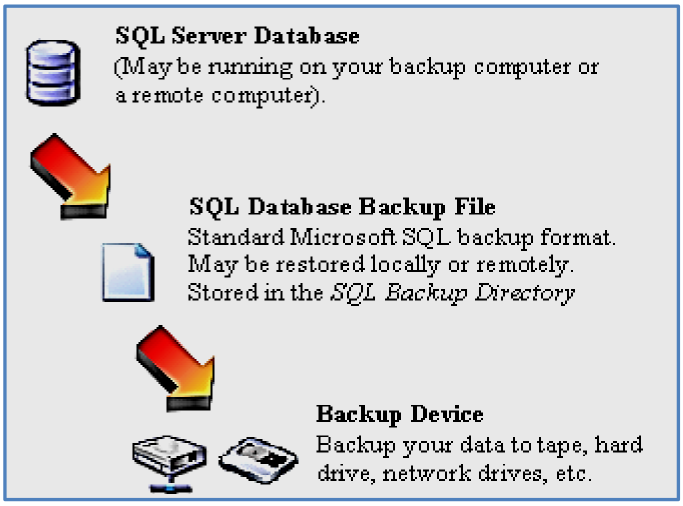
- Back up any number of local or remote SQL servers from one machine (using one add-on license).
- Capture a "live snapshot" of your databases, so your SQL Server can be running during the backup.
- Choose between daily backups or near-continuous data protection.
- Perform transaction log backups of your SQL databases for specific point-in-time restores.
- Restore an entire SQL Server or individual databases using the built-in SQL Restore utility.

Requirements

- Windows XP, Vista, 7, 8, Server 2003, Server 2008/R2 or Server 2012/R2 operating system.
- SQL Server Protection can be used for Microsoft SQL Server 2005, 2008/R2, 2012 and 2014. (Including variants: SQL Express and MSDE).

2. SQL Protection overview

BackupAssist will log in to your SQL Server and perform a back up of the selected databases in .bak file format. These SQL backup files are then transferred to your backup device (hard drive, network server, local directory etc).



SQL Server Database
(May be running on your backup computer or a remote computer).

SQL Database Backup File
Standard Microsoft SQL backup format.
May be restored locally or remotely.
Stored in the *SQL Backup Directory*

Backup Device
Backup your data to tape, hard drive, network drives, etc.

User databases

User databases are the databases containing the information that you have used SQL to store and manage. This is the business data, shown in the below example, that is backed up to protect your company's information. This information should be backed up on a regular basis so that it can be recovered in the event of data loss or corruption.

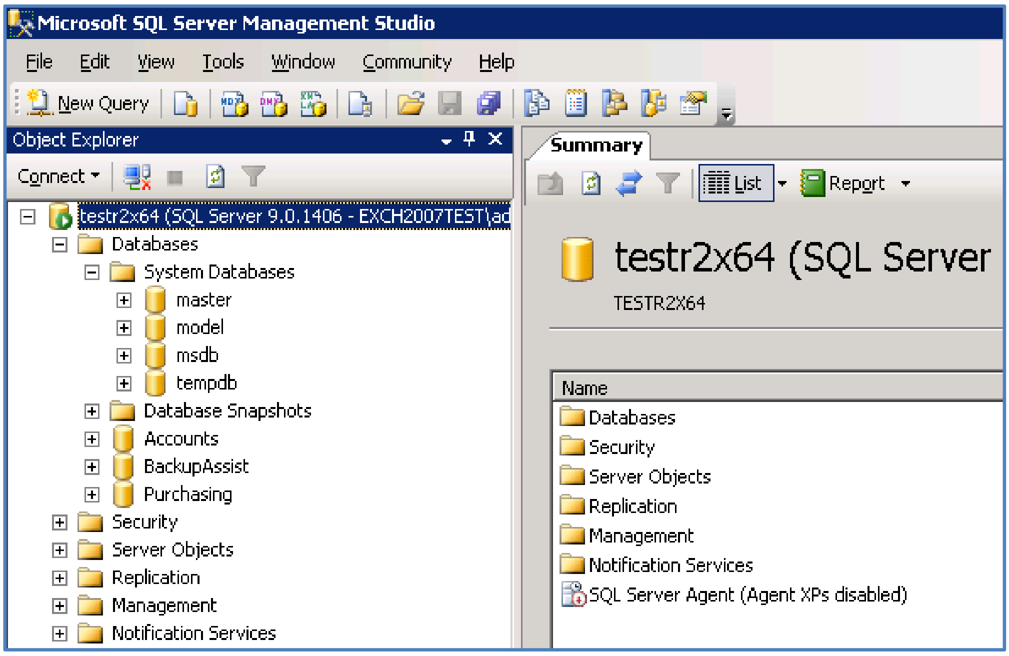


Figure 1: SQL Server Manager - databases

In the above example, you can see SQL user databases such as Accounts and Purchasing.

System databases

SQL system databases are the databases used by the SQL Server itself. These need to be backed up in addition to the user databases, so that the entire SQL Server can be rebuilt in the event of system or database failure.

The system databases vary depending on the version of SQL you are running, but generally include the following:

Master	Records all the system-level information for an SQL Server instance. The master database is composed of system tables and catalogs, and is critical to your SQL Server. If you are restoring your SQL Server after a disaster or new installation, it is advised that you restore the master database before restoring any other database. The BackupAssist SQL Restore tool can be used to restore all other databases remotely but not the master database. If you need to restore the master database to a remote SQL Server, you must install BackupAssist on the server running SQL. You can only perform full backups of the master database. Transactional log, differential or file group backups of the master database are not allowed.
Msdb	Used by the SQL Server agent for scheduling alerts and jobs. The msdb database is used to store data, including scheduled job information, backup and restore history information and DTS packages (data transmission services). Usually the msdb database is rather small, therefore full database backups provide a fast alternative to transaction log backups of the database.
Model	The model database is a template used by Microsoft SQL Server when creating other databases, such as tempdb or user databases. When a new database is created, the entire contents of the model database is copied to the new database. If the master or msdb databases are rebuilt, the model database is also rebuilt and any changes are lost.
Resource	A read-only database that contains system objects included with SQL Server 2005. The resource database makes upgrading to a new version of SQL Server an easier and faster procedure. In earlier versions of SQL Server, upgrading required dropping and creating system objects. Because the resource database file contains all system objects, an upgrade is now accomplished by copying the single resource database file to the local server. Similarly, rolling back system object changes in a service pack only requires overwriting the current version of the resource database with the older version.
Tempdb	A workspace for holding temporary objects or intermediate results sets. It is important to have regular backups of these system databases, however, it is not necessary to back up the tempdb system database because it is rebuilt each time SQL Server is started. When SQL Server is shut down, any data in tempdb is deleted permanently. For this reason, do not store any application specific data in the tempdb database. Leave it exclusively for the SQL Server to use.

Transaction logs

BackupAssist provides two forms of backup schedules: A full backup, at a set time each day, and a transactional backup. The transactional backup creates a full backup then updates at pre-defined user configured intervals as changes are made to the databases.

SQL Server utilizes the transaction log to track all of the modifications performed within a database. This log ensures that the database is able to recover when it is abruptly interrupted (such as a loss of power) and that users are able to undo (or 'rollback') the results of a database transaction if required.

Each time the transaction log is backed up, the SQL Server removes all of the committed transactions in the log and writes them to the backup media. Due to this incremental process, transaction logs are not cumulative and you must maintain a complete set of transaction logs reaching back to your most recent full or differential backup.

The transaction log is a critical component of the database and, if there is a system failure, the transaction log might be required to restore your databases to a consistent state.

The transaction log supports the following operations:

- Recovery of individual transactions
- Recovery of all incomplete transactions when an SQL Server is started
- Rolling a restored database, file, file group or page forward to the point of failure

For more information please refer to http://msdn.microsoft.com/en-us/library/ms190925.aspx

Transaction log backups of SQL Server databases are only possible if the databases are configured using the full or bulk Logged recovery models option. For additional in-depth information on SQL database recovery models, please refer to http://msdn.microsoft.com/en-us/library/ms175477.aspx.

SQL database restores

Most SQL restores will be for SQL databases used by your business, but an SQL restore can also be used to recover an SQL Server. It is important to understand the difference before starting a restore.

SQL database restoration
BackupAssist provides a tool for restoring one or more SQL databases to either a local or remote SQL Server. The tool also lets you restore an SQL database to a specific point in time, provided you have a transaction log backup to restore from. The databases restored for an SQL database restoration are the business related databases used to manage information.

Master database recovery
BackupAssist can restore a master database to a local SQL Server. The master database is composed of important system tables and catalogues required to rebuild your SQL Server. If you are restoring your server after a disaster or performing a new installation, it is advised that you restore the master database before restoring any other databases.

The databases restored for an SQL database restoration are system databases. You must select these system databases from the SQL database list when your backup is made. For information on what databases are system databases, refer to the section, System Databases, on page 4.

3. BackupAssist settings

When creating a backup job, there are some global settings that should be configured in BackupAssist. If they are not configured, you will be prompted to complete them during the creation of your first backup. It is recommended that this is done in advance.

BackupAssist's settings can be entered and modified using the selections available in the **Settings tab.** Clicking on the *Settings* tab will display the selections as icons. Four of these are used when creating new a backup job and each is described below:

Backup user identity

Backup jobs require an administrator account with read access to the data source, and full read-write access to the backup's destination. It is recommended that a dedicated backup account is created for this purpose. The account's details are entered here and your backup jobs will be launched using these credentials. The account's permissions will be validated both when the backup user identity is entered and when the job is executed. If no account is specified or the account has insufficient permissions, the backup job will fail and note the error in the backup report.

A video explaining the creation of a backup user identity can be found on our, Videos Webpage.

Email server settings

This menu item is used to enter the details of the SMTP server used by BackupAssist to send email notifications. The SMTP server must be configured if you want to have an email *Notifications* step enabled when you create a backup job.

Email address list

This menu item is used to define and store the email addresses of potential notification recipients. The list will be used to populate the recipient selection screen when configuring an email notification for a backup job. Any email addresses entered during the creation of a new notification are automatically added to the *Email address list*.

Network paths

This option allows you to enter access credentials for networks, domains and drives that the default account (specified in the *Backup user identity)* does not have access to. Enter or browse to the location and add it to the *Path list*. The *Edit* option will allow you to enter an authentication account, specifically for that path. When you create a backup job to a remote location, that location will be automatically added here.

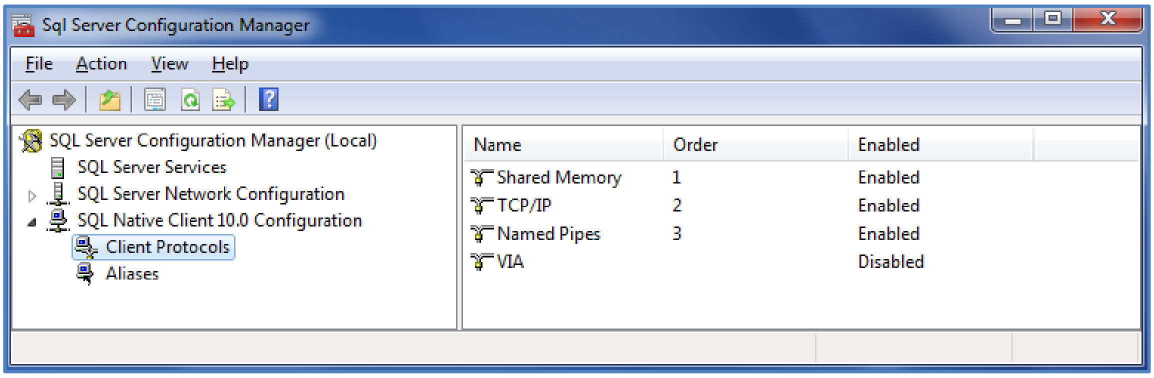
Having multiple connections to a resource using the same logon credentials can generate a Windows error, such as the BA260 NAS error. It is therefore recommended that you avoid having mapped shares on the computer running BackupAssist that are the same as the paths configured in BackupAssist.

4. Remote SQL Server configuration

To back up a remote SQL database, you need to make sure that the SQL Server accepts remote TCP/IP connections. To do this:

1. Go to Start > All Programs > SQL Server.
2. Select the SQL Server Configuration Manager.
3. Enable TPC/IP and Named Pipes for both Client protocols and Protocols for MSSQLSERVER.

   The below screenshot shows the enabled settings:


Sql Server Configuration Manager
File
Action
View
Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration
SQL Native Client 10.0 Configuration
Client Protocols
Aliases

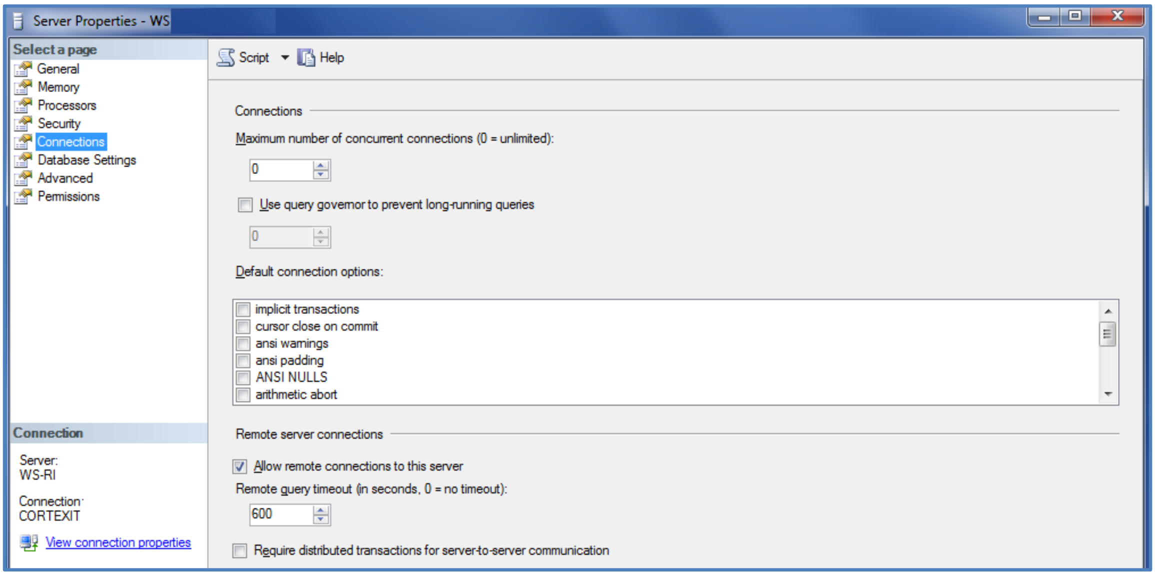
Name	Order	Enabled
Shared Memory	1	Enabled
TCP/IP	2	Enabled
Named Pipes	3	Enabled
VIA		Disabled

Figure 2: Server Configuration manager – SQL settings

   You will now need to enable remote connections on your SQL Server.

4. Open your SQL Server's Enterprise Manager or Management Studio.
5. Under *Server Properties* > *Connections*, enable *Allow remote connections to this server.*

   The below screenshot shows the enabled settings:



Server: WS-RI

Connection: CORTEXIT

Connections

Maximum number of concurrent connections (0 = unlimited): 0

□ Use query governor to prevent long-running queries

0

Default connection options:

□ implicit transactions
□ cursor close on commit
□ ansi warnings
□ ansi padding
□ ANSI NULLS
□ arithmetic abort

Remote server connections

☑ Allow remote connections to this server

Remote query timeout (in seconds, 0 = no timeout): 600

□ Require distributed transactions for server-to-server communication

Figure 3: SQL Server Management Studio

After completing these steps, your Windows SQL Server *Protocols* and *Connections* settings, will be ready to receive remote connections from BackupAssist.

5. Creating an SQL Server Protection backup

BackupAssist SQL Server Protection can back up SQL system databases for an SQL Server Master recovery, and use a transaction schedule to restore databases to a specific point in time, on both local and remote SQL machines.

Launch BackupAssist and follow the steps outlined below:

1. Select the **Backup** tab, and click **Create a new backup Job**

2. Select **SQL Server Protection**

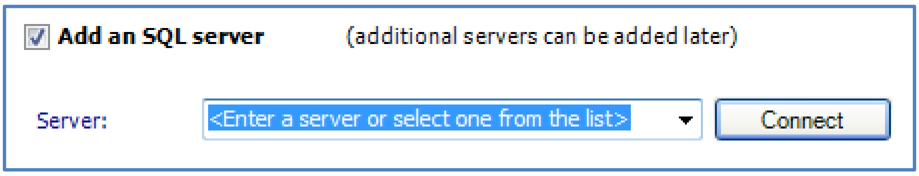
   If this is the first time you have created a backup job, you will be asked to provide a *Backup user identity* if one has not been defined. See the section above, BackupAssist settings, for guidance.

3. **Selections:** The selections screen is used to locate your SQL servers, authenticate them and select the databases to be backed up.

   a. **Server selection**

   BackupAssist will scan your local machine for SQL servers. Any servers found will be selectable from the drop-down list. Select **connect.**

   To select an SQL Server on your network, enter the name of the SQL Server or its IP address into the *Server* field. Select **connect**.



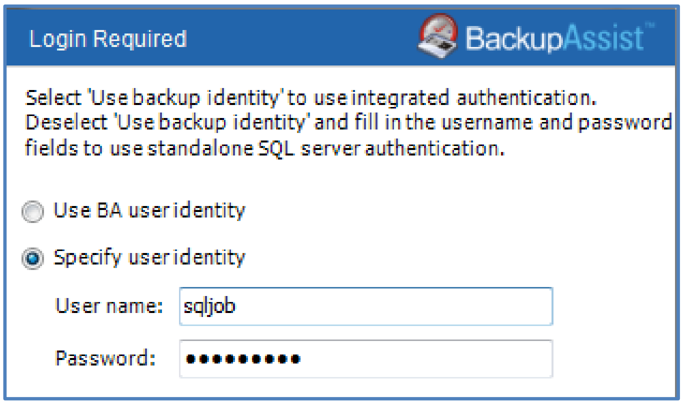
☑ **Add an SQL server** (additional servers can be added later)

Server: <Enter a server or select one from the list>

Figure 4: SQL Server connection

   b. **Server authentication**

   When you select *Connect*, you will be prompted to provide login credentials using the BA user identity, another user account (such as a Windows account), or an SQL Server user account.



Login Required

Select 'Use backup identity' to use integrated authentication.
Deselect 'Use backup identity' and fill in the username and password fields to use standalone SQL server authentication.

○ Use BA user identity

● Specify user identity

User name: sqljob

Password: ●●●●●●●●●

Figure 5: SQL Server authentication credentials

The authentication account, BA user identity account (if different to the account used for this backup) and the SQL Server's service account, ALL need access to the backup destination.

c. **Database selection**

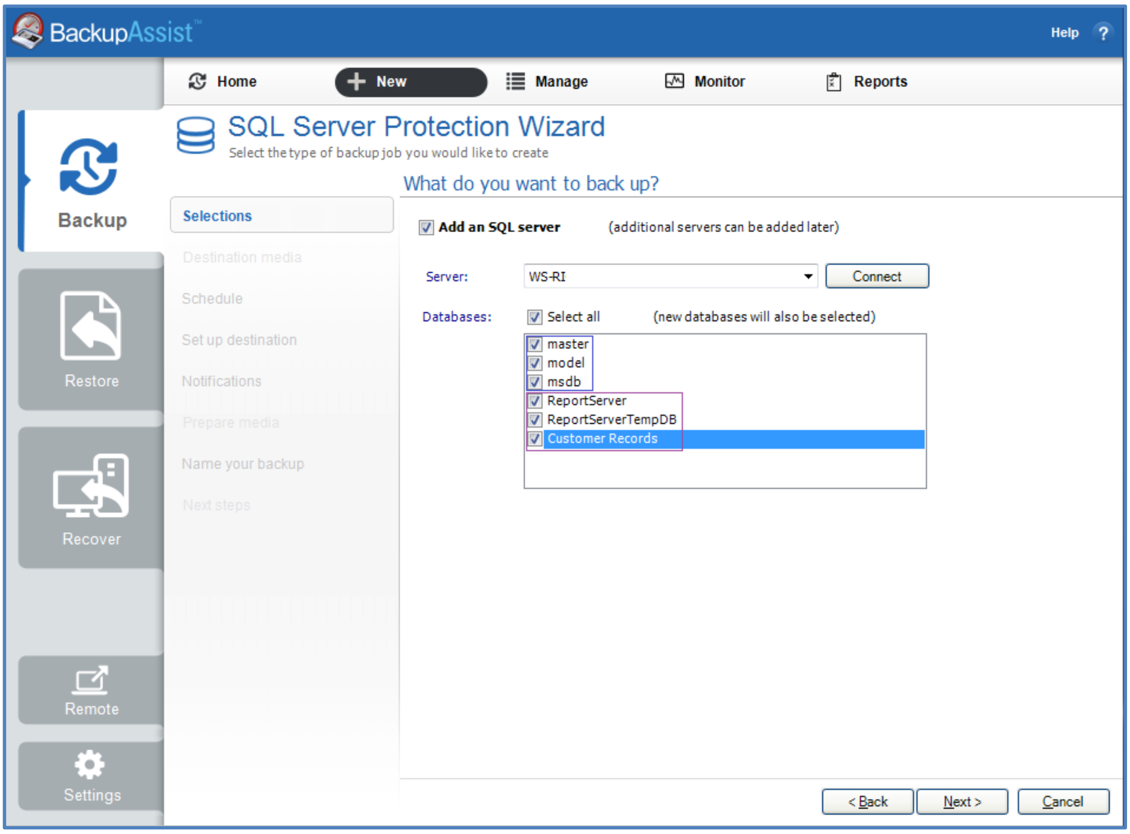
A list of databases will be displayed for each SQL Server. The databases displayed will be both *system databases* and *user databases*. It is important to understand the difference, and to select the correct databases for your backup.

User databases: These contain your business and user data. Select the databases you wish to backup. In the example below, the user databases are circled in purple.

System databases: These are used by the SQL Server to operate. If they are included in a backup, then that backup can be used to perform a master database restore. All SQL protection plans should include a backup cycle that includes system databases so that an SQL Server can be recovered. In the example below, 3 system database are circled in blue.

- For a list of system databases, refer to the section, System databases.
- For more information on database types, refer to the section, SQL Protection overview.

Select the databases to back up, and click Next.



SQL Server Protection Wizard

Select the type of backup job you would like to create

What do you want to back up?

☑ Add an SQL server (additional servers can be added later)

Server: WS-RI

Databases: ☑ Select all (new databases will also be selected)

☑ master
☑ model
☑ msdb
☑ ReportServer
☑ ReportServerTempDB
☑ Customer Records

Figure 6: SQL Server backup - Server and database selection screen

4. **Destination media**: You will not be prompted to provide a destination. This step will be skipped.

   SQL Server Protection backups use fixed media only, therefore, the media type does not need to be selected. The fixed media's location and configuration will be entered at the, **Set up Destination** step.

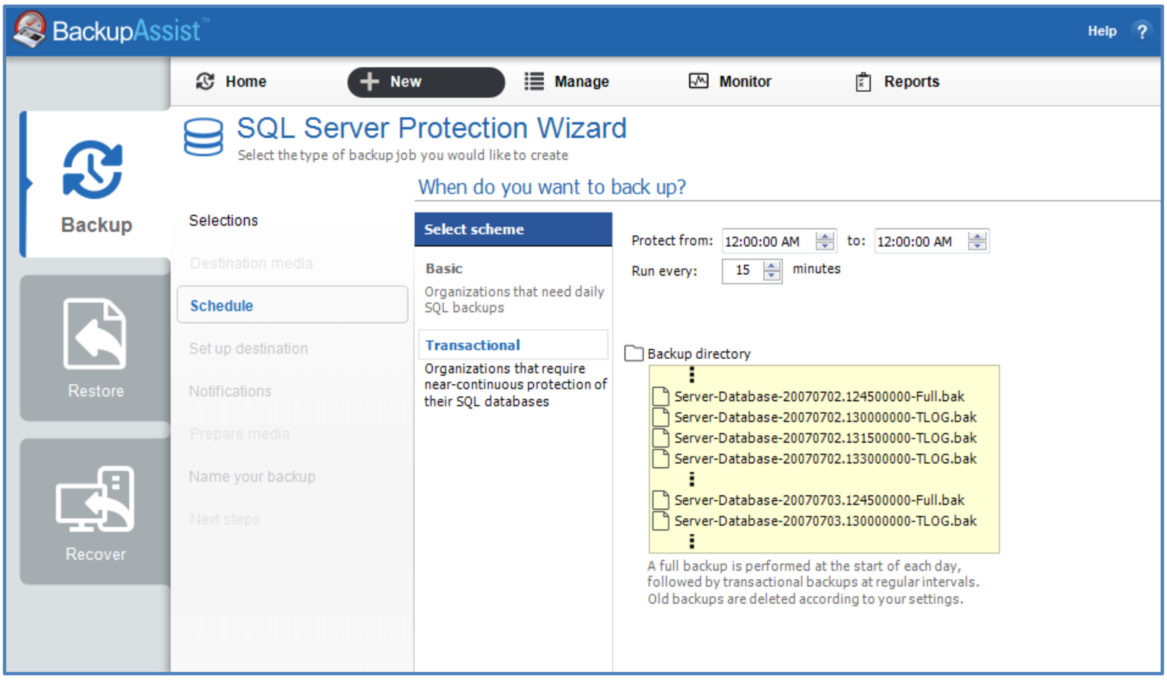
5. **Schedule**: There are two schedules available for SQL Server Protection, *Basic* and *Transactional*.

   - **Basic:** Runs a full daily backup of your SQL databases. A backup is run once per day at the time set in this screen.
   - **Transactional:** Creates a full daily backup at the start of the day, and then runs transaction log backups at pre-defined intervals (minimum of every 15 min) throughout the day.

     This provides the functionality of an incremental backup because SQL can use the transaction log to recover data. Each day's database backup is saved in the same location along with each transaction log.

   The schedule can be changed after the backup has been created, using the *Manage* > *Schedule* menu on the *Backup* tab.

   Select an appropriate schedule. Click **Next**.



SQL Server Protection Wizard

Select the type of backup job you would like to create

When do you want to back up?

Basic
Organizations that need daily SQL backups

Transactional
Organizations that require near-continuous protection of their SQL databases

Protect from: 12:00:00 AM to: 12:00:00 AM

Run every: 15 minutes

Backup directory

Server-Database-20070702.124500000-Full.bak
Server-Database-20070702.130000000-TLOG.bak
Server-Database-20070702.131500000-TLOG.bak
Server-Database-20070702.133000000-TLOG.bak

Server-Database-20070703.124500000-Full.bak
Server-Database-20070703.130000000-TLOG.bak

A full backup is performed at the start of each day, followed by transactional backups at regular intervals. Old backups are deleted according to your settings.

Figure 7: SQL Server Protection – schedule selection

6. **Set up destination**. Enter the path of the directory where you want to store your SQL backup files.

   Once a backup has been created, you can configure how long backups are kept for, how many are retained and how much space to allocate. This is done using the *Manage* > *Destination* menu on the *Backup* tab.

   The SQL backups are taken from .bak files (one for each database). The file name is given a suffix, indicating the day it was created on.

➢ **Mail Server:** If you have not configured an SMTP mail server for BackupAssist, you will be prompted to provide those details after the backup destination step has been completed. See the BackupAssist settings section for guidance.

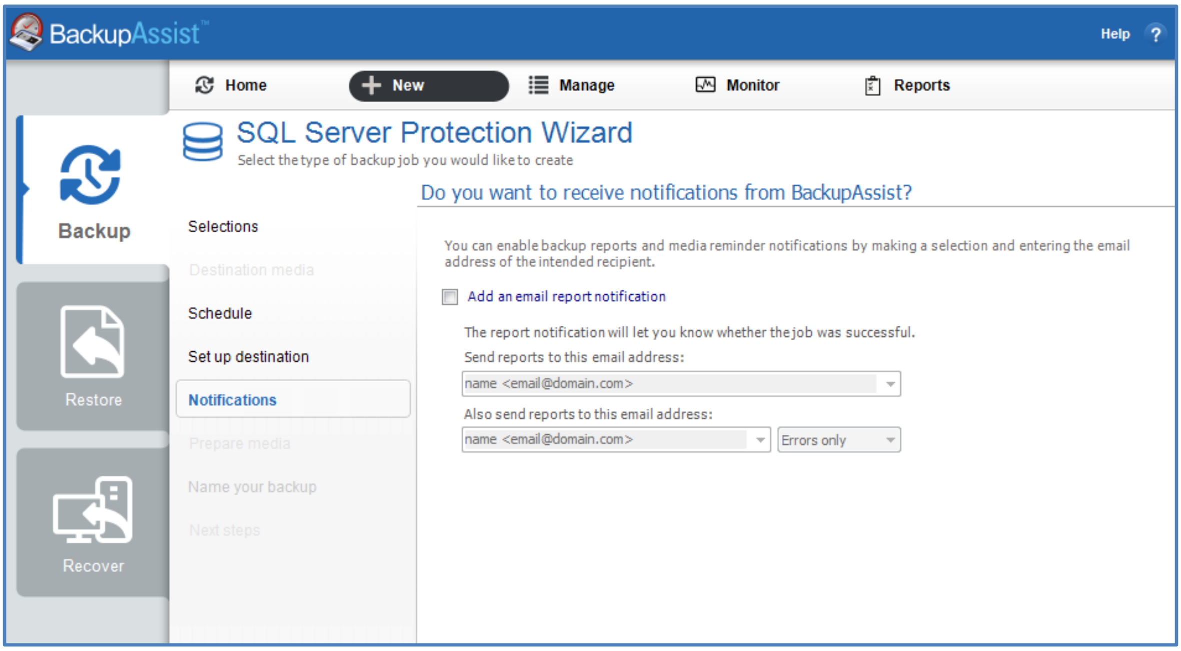
7. **Notifications:** Once a backup job has completed, BackupAssist can send an email to inform selected recipients of the result. This email notification can be enabled during the creation of a backup job, if the mail server has been configured.

To enable email notifications:

a. Select, **Add an email report notification.**

b. Enter recipients into the **Send reports to this email address** field.

c. Enter recipients into the **Also send reports to this email address** field. You can then select the condition under which the email should be sent, using the drop-down box.



SQL Server Protection Wizard

Select the type of backup job you would like to create

Do you want to receive notifications from BackupAssist?

You can enable backup reports and media reminder notifications by making a selection and entering the email address of the intended recipient.

□ Add an email report notification

The report notification will let you know whether the job was successful.

Send reports to this email address: name <email@domain.com>

Also send reports to this email address: name <email@domain.com> Errors only

Figure 8: SQL Server Protection – backup job Notifications

After a backup job has been created, you can modify the notifications by adding and removing recipients, setting additional notification conditions and including print and file notification types.

To learn more about notification options, see the BackupAssist Backup tab whitepaper.

8. **Prepare media:** SQL Server Protection backups use fixed media only. Media preparation applies to removable media only, therefore this step will be skipped.

9. **Name your backup:** Provide a name for your backup job, and click **Finish**.

▶ **Your SQL Server Protection backup job has now been created.**

Important: Once a backup job has been created, it should be reviewed and run using the *Manage* menu. This menu provides additional options to configure your backup. See the section, SQL Server Protection backup management, for more information.

Important: Once a backup job has been run and a backup created, a MANUAL test restore should be performed to ensure the backup is working as intended. To perform a test restore, refer to the section, Restoring an SQL Server Protection backup.

6. Restoring an SQL Server Protection backup

BackupAssist provides an inbuilt SQL restore tool that can be used to restore SQL databases to both local and remote SQL servers. If the backup used a transactional schedule, the databases can be restored to a specific point in time.

To restore data from an **SQL Server Protection** backup, start BackupAssist and follow these steps:

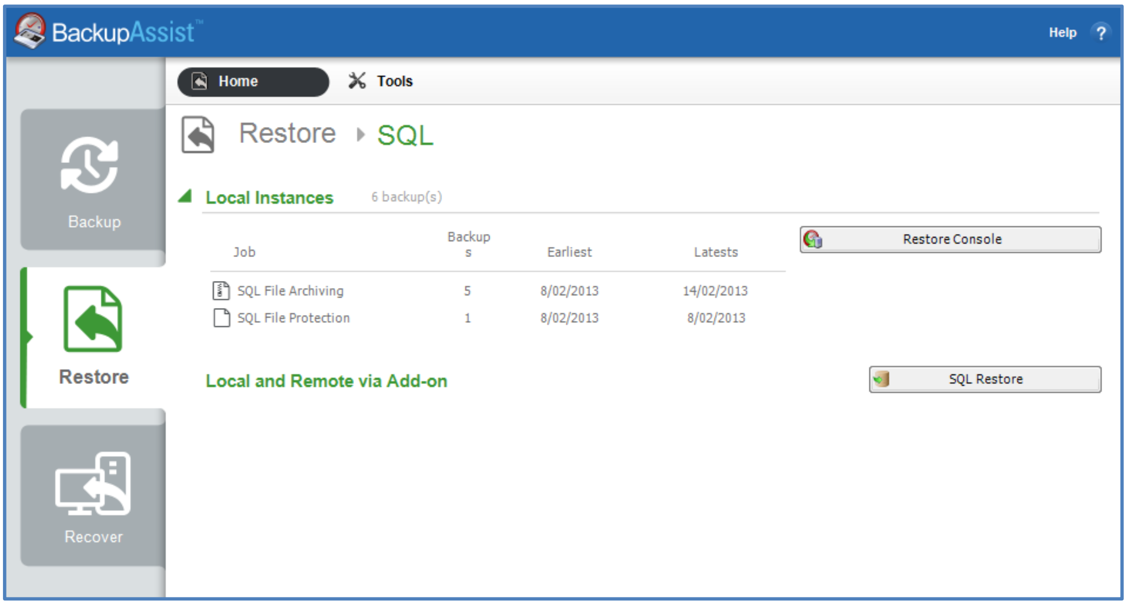
1. Select the **Restore tab**

   The *Restore tab* has a *Home page* and a *Tools menu*. The *Home page* is the default screen and the recommended starting point for performing a restore.

2. From the **Home page**, select **SQL**

3. The **Home page** will now display all backups that contain SQL data.

   - If you have SQL VSS backups created using File Protection or File Archiving, they will appear with the *Restore Console*.
   - Your *SQL Server Protection* backup will be associated with the *SQL Restore* tool. These backups are represented by the title *Local and Remote via Add-on*. The individual backups created using SQL Server Protection will be located and displayed within the *SQL Restore* tool.



Job	Backups	Earliest	Latests
SQL File Archiving	5	8/02/2013	14/02/2013
SQL File Protection	1	8/02/2013	8/02/2013

Figure 9: Restore Home page – results screen

4. Select **SQL Restore.** This will open the SQL Restore tool.

   Most SQL restores will be for SQL databases used by your business, but an SQL restore can also be used to recover an SQL Server. It is important to understand the difference.

   a. **An SQL database restoration:** The databases restored are the business related databases that are used to information.

   b. **A master database recovery:** BackupAssist can restore a master database to a local SQL Server. The master database is composed of system tables and catalogues required to rebuild an SQL Server. You must select these system databases from the database list when your backup is made. It is advised that you restore the master database before any other databases.

5. **Select Server**

This screen is used to locate and authenticate the SQL Server that you will restore to.

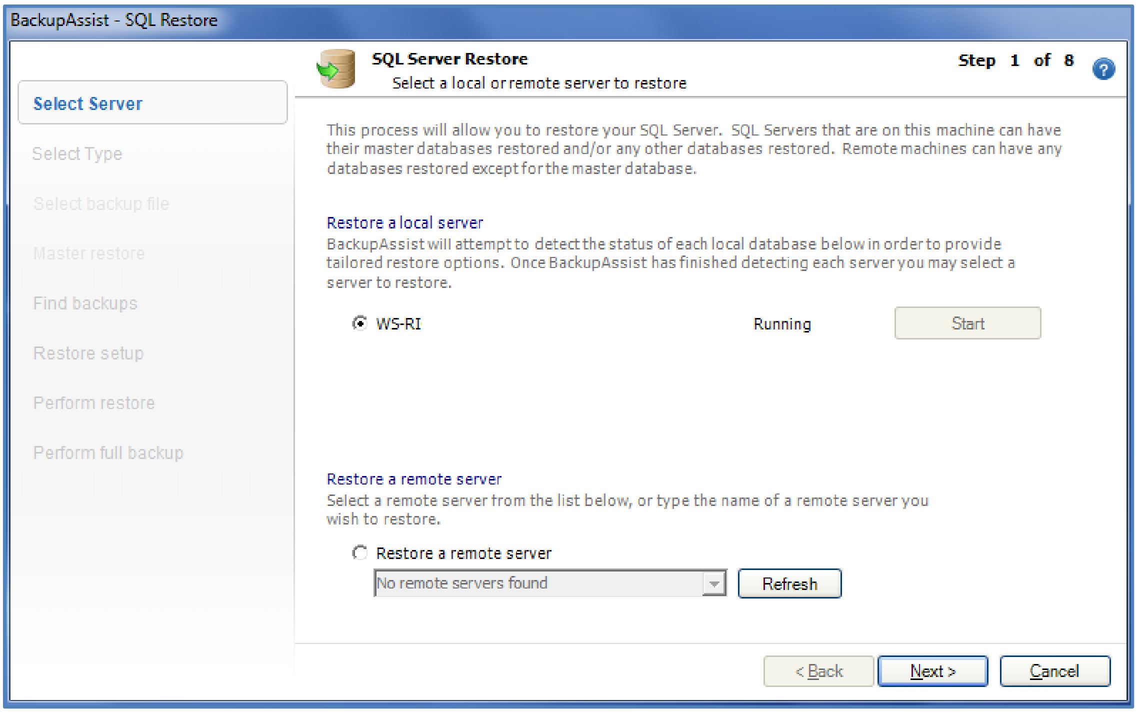
The databases you wish to restore must already exist on the SQL Server. If they do not, you can choose to either restore the master database first, which will result in recreating all databases on the server, or to manually recreate each database you wish to restore. If you are restoring your SQL Server after a disaster or new installation, it is advised that you restore the master database before restoring any other databases.

a. **Server selection:** Select the SQL Server to restore to.

Restore a local Server: The console will display a list of the SQL servers on your local machine. If an SQL Server is present but not running, you can click **Start**. Any servers that cannot be started may indicate a fault with the server itself. You can only restore to SQL servers that are running.

Restore a remote server: Any remote SQL servers detected will appear in the drop down list. If the server is not shown, select the *Restore a remote server* button, and enter the name of the SQL Server or its IP address.

Select the SQL Server to restore to and click Next.



SQL Server Restore

Select a local or remote server to restore

Step 1 of 8

This process will allow you to restore your SQL Server. SQL Servers that are on this machine can have their master databases restored and/or any other databases restored. Remote machines can have any databases restored except for the master database.

Restore a local server

BackupAssist will attempt to detect the status of each local database below in order to provide tailored restore options. Once BackupAssist has finished detecting each server you may select a server to restore.

◉ WS-RI Running

Restore a remote server

Select a remote server from the list below, or type the name of a remote server you wish to restore.

○ Restore a remote server

No remote servers found

Figure 10: SQL Restore tool – server selection

b. **Server authentication**

When you select Next, and you will be prompted to provide logon credentials.

- *Use integrated authentication*: Backup user identity.
- *Specify user identity*: Another user account (such as a Windows account) or an SQL Server user account.

  The authentication account, the BA account (if different to the account used for this backup) and the SQL Server's service account, all need access to the backup destination. Some SQL servers are configured to accept only SQL authentication.

6. Select Type

This screen is used to identify the type of restore you want to perform.

- **Master Database recovery:** Only select this if your SQL database is damage, missing or needs to be replaced / recreated. This option will restore the system databases that the SQL Server uses to operate. This is not for user or business databases; it is only for SQL recovery scenarios.

  To restore the master database you need to have BackupAssist installed on the server itself.

- **Selected Databases:** Select this option to restore your user / business databases to any point in time for which you have backups.

Only select both if you wish to recover your SQL database, as well as restore your user databases.

7. Select backup file

If you selected *Master database recovery* in step 6, you will be asked to select a master database to restore. A list of backups containing master databases will be listed for selection.

8. Master restore

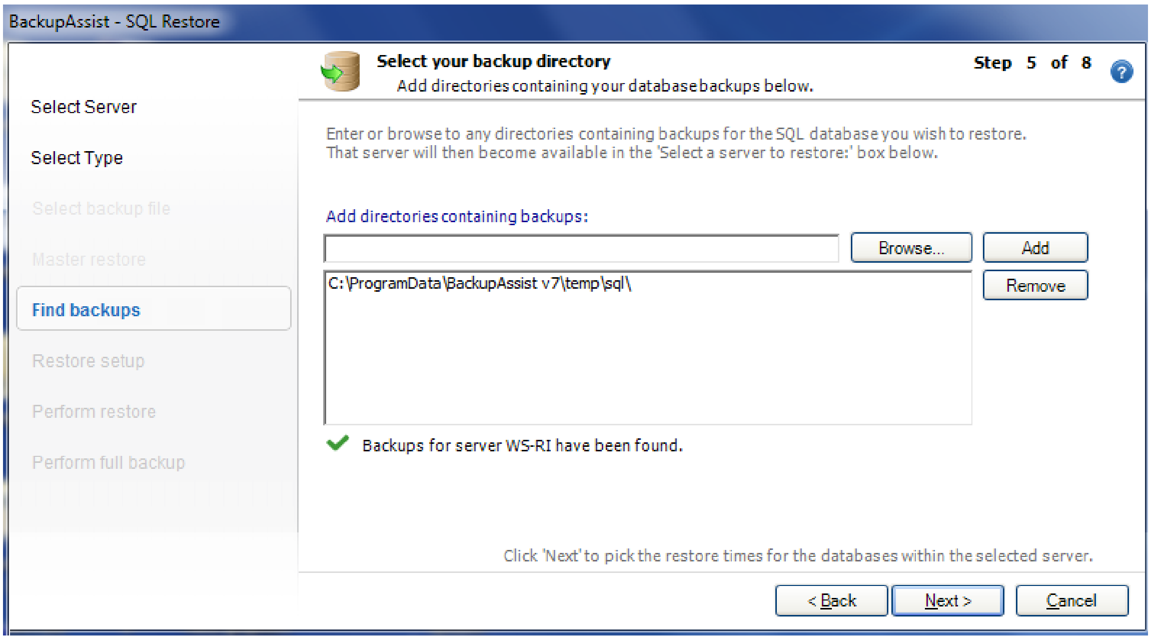
If you selected *Master database recovery* in step 6, you will now be asked to start the master database restore. After the master database recovery has finished, you can continue with the SQL database restore steps.

9. Find backups

This screen is used to enter the location of your SQL database backup files. This is typically the SQL directory used in your SQL backup job.

Browse to the backups' location or type in the path and press *Add*. If the location contains SQL databases, a confirmation tick and message will appear below the list.

Add the location of your SQL Server Protection backups, and click next.



Select your backup directory

Add directories containing your database backups below.

Enter or browse to any directories containing backups for the SQL database you wish to restore. That server will then become available in the 'Select a server to restore:' box below.

Add directories containing backups:

C:\ProgramData\BackupAssist v7\temp\sql\

✓ Backups for server WS-RI have been found.

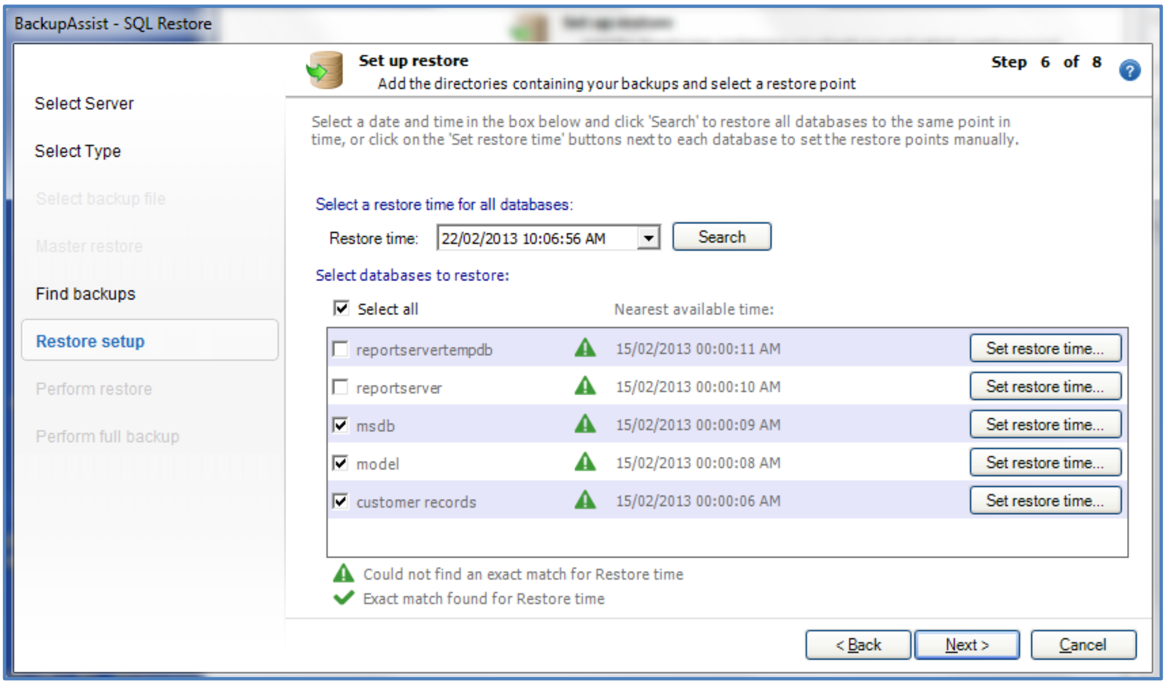
Click 'Next' to pick the restore times for the databases within the selected server.

Figure 11: SQL Restore – SQL backup location selection

10. Restore setup

You will be presented with a list of databases that can be restored using the backups identified in step 9. The screen is used to select what databases to restore and what time to restore them from.

Databases can be restored to a specific point in time if the *Transactional* scheme was selected (during the *Schedule* step) when the backup job was created.



BackupAssist - SQL Restore

Set up restore — Step 6 of 8

Add the directories containing your backups and select a restore point

Select a date and time in the box below and click 'Search' to restore all databases to the same point in time, or click on the 'Set restore time' buttons next to each database to set the restore points manually.

Select a restore time for all databases:

Restore time: 22/02/2013 10:06:56 AM [Search]

Select databases to restore:

☑ Select all — Nearest available time:

Database	Nearest available time:	
□ reportservertempdb	⚠ 15/02/2013 00:00:11 AM	Set restore time...
□ reportserver	⚠ 15/02/2013 00:00:10 AM	Set restore time...
☑ msdb	⚠ 15/02/2013 00:00:09 AM	Set restore time...
☑ model	⚠ 15/02/2013 00:00:08 AM	Set restore time...
☑ customer records	⚠ 15/02/2013 00:00:06 AM	Set restore time...

⚠ Could not find an exact match for Restore time
✓ Exact match found for Restore time

Figure 12: SQL Restore – backup selection

a. Select the time that you want to restore your databases from using:

- The **Restore time** field: Edit the data and time shown or select a date from the drop-down calendar, and select **Search**. This point in time will apply to all databases listed.

  The backups that are closest to the selected time will appear next to each database, and show the time that the backup was made.

- The **Set restore time** button: This button will select a restore point for only that database.
  - **Select a date** from the Calendar. Dates that have backups will be indicated in bold.
  - **Select a time range** will show a list of time ranges available for that date**.** You can restore the database to any point in time between any of the time ranges listed. Select a time range from the list.
  - Select a time from within that range using the field at the bottom of the window.

b. Select the databases that you want to restore using the tick-box next to each database, or the *Select all* option.

- If the restore time specified is not available from the supplied backup files, a **warning** icon will appear to the right of the database/ the closest available time to that specified (that the database can be restored to) will be indicated.
- If the restore time selected is available, a **success** icon will appear to the right of the database.

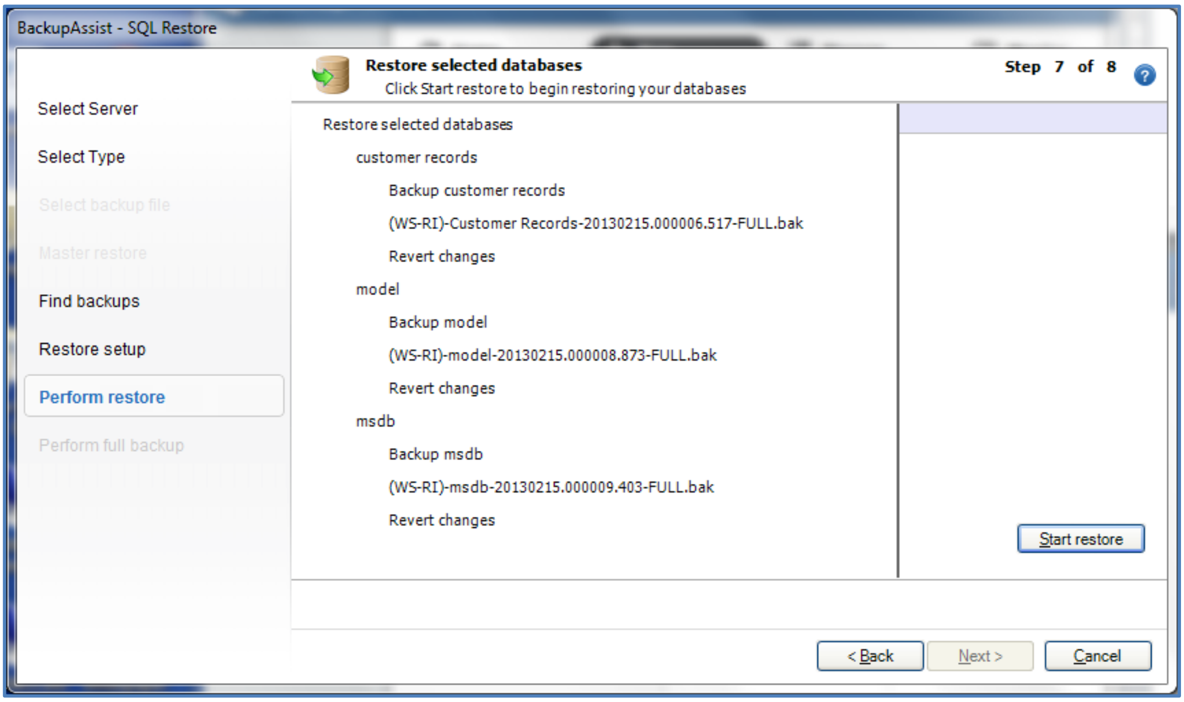
c. Click *Next.*

11. **Perform restore**

Review the restore job settings.

Select **Start restore** to the right of the window to restore all databases selected.

Click **Next** once the restore process has completed.



Restore selected databases

Click Start restore to begin restoring your databases

Restore selected databases
- customer records
  - Backup customer records
  - (WS-RI)-Customer Records-20130215.000006.517-FULL.bak
  - Revert changes
- model
  - Backup model
  - (WS-RI)-model-20130215.000008.873-FULL.bak
  - Revert changes
- msdb
  - Backup msdb
  - (WS-RI)-msdb-20130215.000009.403-FULL.bak
  - Revert changes

Figure 13: SQL Restore – Start restore screen

12. **Perform full backup**

After restoring an SQL database, future transaction log backups cannot be used to restore a database to a specific point in time until a full backup runs. It is therefore recommended that you now run a full backup of the databases that were restored.

Click **Start full backup**.

Full backups of these databases will be stored in the same folder as the backup files you just restored from.

If your backup schedule already includes a **full** backup as the next SQL backup and you are certain you will not need to restore these databases to a specific point in time between the restoration and the next full backup, you can check **Skip full backup** and click **Finish**.

▶ **Your SQL Server Protection restore has now been completed.**

7. SQL Server Protection backup management

Once you have created a backup job, you can modify the settings and access advanced configuration options using the *Manage* menu.

To access the backup management screen:

1. Select the BackupAssist, **Backup tab.**
2. Select **Manage** from the top menu. A list of all backup jobs will be displayed.
3. Select the backup job you want to modify, and select **Edit.**
4. Select the required configuration item on the left. Key configurations are described below.

To learn more about the backup management options, see the Backup tab whitepaper.

Manually running a backup job

All new and modified backup jobs should be manually run to ensure they work as intended.

1. Select the backup job, and select *Run.*
2. You will be prompted to *Rerun a past backup* or to *Run a future backup now.*
3. When the backup job starts, the screen will change to the *Monitor* view.
4. Once the backup has been completed, select the *Report* button and review the results.

SQL Servers

Use this screen to change the databases that have been selected for backup, and to remove and add SQL Servers.

Scheduling

Selecting *Scheduling* will display your **Scheduling options.** You can use this screen to change the default time and days of your scheme's daily backups. If you are running a transactional scheme, you can also modify the *Time between backups*. The default time is every 15 minutes.

Select a new Schedule: This will display the *Basic* and *Transactional* backup schemes that you chose from during the creation of your backup job. You can change the current scheme using this option.

- **Basic:** Runs a full daily backup of your SQL databases. A backup is run once per day at the time set in this screen.
- **Transactional:** Creates a full daily backup at the start of the day, and then runs transaction log backups at pre-defined intervals (minimum of every 15 min) throughout the day.

  This provides the functionality of an incremental backup because SQL can use the transaction log to recover data. Each day's database backup is saved in the same location along with each transaction log.

For additional information on the *Scheduling* screen, please refer to the Backup tab whitepaper.

Concurrent backup

This feature allows two backup jobs to run at the same time.

Concurrent backup combinations:

- Two File Archiving backup jobs can run at the same time if both have 'Enable concurrent backup' selected.
- An SQL Protection or Mailbox Protection backup job, with 'Enable concurrent backup' selected, can run concurrently with a System Protection, File Protection or File Archiving backup job. The File Archiving backup job does not need to have 'Enable concurrent backup' selected. (System Protection and File Protection do not have an 'Enable concurrent backup' option).
- An SQL Protection and a Mailbox Protection backup job can run at the same time, in any combination, if both have 'Enable concurrent backup' selected.
- In all cases, only two backup jobs can run concurrently.

Concurrent backup considerations:

- If two concurrent backups are scheduled to start at the same time, one backup will start first and begin preparing the job. Once the preparation phase has completed, the second backup will start
- If a third scheduled backup job has 'Enable concurrent backup' selected, it will be queued and run once one of the two existing concurrent backup jobs has finished.

Concurrent backup limitations:

- Only two backup jobs can run concurrently.
- Concurrent backups cannot write to the same destination device (e.g. local drive, NAS, RDX etc.).
- If another backup job is already running when the concurrent backups are scheduled to start, then one of the concurrent backups will start if it meets the criteria defined in the concurrent combinations section.
- A backup job cannot run concurrently if it is backing up a Hyper-V environment or an Exchange server using VSS (VSS enabled).

8. Support and Resources

Contacting Technical Support

Should you have any questions regarding either BackupAssist or SQL Server Protection, please email support@backupassist.com and we will respond to you as soon as possible.

Similarly, if you have any suggestions for additional functionality in BackupAssist, or new products or add-ons, please also forward your feedback to support@backupassist.com

References

- For additional information regarding recovering your SQL databases to a certain point in time, and creation of multiple recovery points, refer the following Microsoft Knowledge Base article: http://msdn.microsoft.com/en-us/library/ms175078.aspx
- Tips for scheduling and testing SQL Server backups:

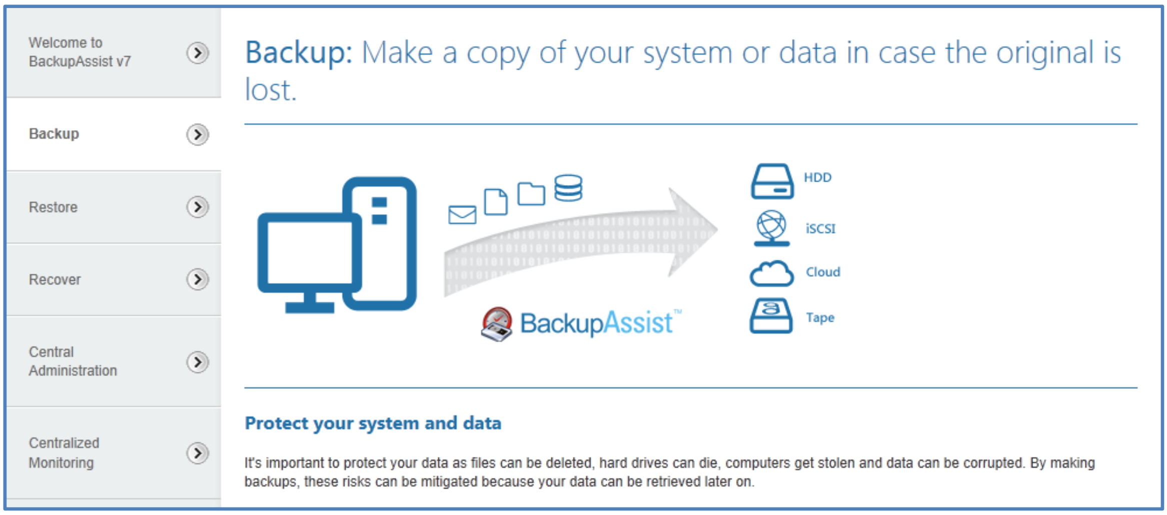
http://searchsqlserver.techtarget.com/tip/0,289483,sid87_gci1307822,00.html?asrc=SS_CLA_301326&psrc=CLT_87

- SQL system databases explanations:
  http://vyaskn.tripod.com/sql_server_administration_best_practices.htm
- MSDN knowledge base on recovery paths:
  http://msdn.microsoft.com/en-us/library/ms175078.aspx

Learn more – The Welcome Screen

Each tab in BackupAssist includes a "Learn More" link on the tab's **Home** page.

For example, selecting the ***Learn more about Backup*** link will open the **Welcome Screen** with the Backup introduction selected. This screen provides an overview of the tab's functions and features, and links to documentation and resources.



Backup: Make a copy of your system or data in case the original is lost.

Protect your system and data

It's important to protect your data as files can be deleted, hard drives can die, computers get stolen and data can be corrupted. By making backups, these risks can be mitigated because your data can be retrieved later on.

Figure 14: Backup tab – Welcome Screen