APS SQL MAINTENANCE TASKS AND BACKUP GUIDE

Practice Management (PM)
Practice IQ (PIQ)
Virtual Cabinet (VC)
Business Process Automation (BPA)
Author

Created and Published by Reckon Limited
Level 12, 65 Berry Street
North Sydney NSW 2060
Australia
ACN 003 348 730
All Rights Reserved
Copyright © 2015 Reckon Limited

Copyright & Trademarks

No part of these materials may be reproduced or transmitted in any form or by any means, including photocopying, without written permission of Reckon Limited.

Every effort has been made to ensure the accuracy and completeness of this manual. However, neither Reckon Limited, its subsidiaries, employees and agents, are liable for any errors or omissions. This document should not be relied on as a detailed specification of the system. In addition, Reckon Limited reserves the right to issue revisions, enhancements and improvements to Reckon APS software or any other Reckon software at any time without notice. Every effort will be made to ensure that the manual is updated concurrently and that the user is not inconvenienced as a result of any change.

Reckon, Reckon logos, R logos, Reckon APS, Advance and Accountable Technology are trademarks of Reckon Limited.
Contents

INTRODUCTION .................................................................................................................. 4

General SQL Server Recommendations ........................................................................ 4
  Microsoft SQL Server Service Accounts ................................................................. 4
  Database Shrink ........................................................................................................ 4

Backup and Recovery .................................................................................................... 4
  Types of SQL Backups ............................................................................................... 5
  Impact of the Recovery Model on Backup and Restore .............................................. 5
  Practice Management (PM) Database Backup .......................................................... 6
  PracticeIQ (PIQ) Backup and Recovery .................................................................... 7
    Data Warehouse ....................................................................................................... 7
    Analysis Services Databases ................................................................................... 7
    Performing an Analysis Services Database Backup ............................................... 7
    Reporting Services Backup and Recovery .............................................................. 9
    Integration Services Packages Backup .................................................................... 9
    SQL Server Agent Jobs Backup ............................................................................ 11
  Virtual Cabinet (VC) Backup ..................................................................................... 12
  Business Process Automation (BPA) Backup ............................................................ 12
  System Database Backup and Recovery ................................................................... 12
  Backup Validation ...................................................................................................... 12

Index Maintenance ....................................................................................................... 14

Database Statistics Maintenance ................................................................................. 18

Database Consistency Checks (DBCC) ........................................................................ 20

Clean-up Tasks ............................................................................................................. 23
  History Clean-up ....................................................................................................... 23
  General Clean-up Tasks ........................................................................................... 24

Appendix I – Performing an Ad-hoc backup .............................................................. 26
  Individual Database Backup ..................................................................................... 26
  Individual Database Backup Script .......................................................................... 27

Appendix II - Creating Scheduled Backups Using a Maintenance Plan ...................... 28

Appendix III – Online Resources ............................................................................... 32
INTRODUCTION

This document serves as a guide to some of the most common SQL Maintenance tasks required to ensure that SQL databases are adequately protected from data loss and that databases perform as expected. This document draws mainly from Microsoft recommendations but also from leaders in the SQL Database Administration field.

This document contains links to Microsoft’s MSDN and Technet libraries as well as other websites. These links may refer to a version of SQL Server that is different from your version. The information contained in these links is relevant across all versions of SQL Server. Reckon APS cannot guarantee the continued availability of these web pages.

The implementation and maintenance of SQL Maintenance Tasks is the responsibility of the client and not Reckon.

Microsoft SQL Server 2008 R2 was used in creating the instructions and screenshots in this document. A different version of SQL Server may mean the instructions and screens may vary slightly. If this is the case, consult Microsoft Books Online for the help files for the specific version of Microsoft SQL Server.

General SQL Server Recommendations

Microsoft SQL Server Service Accounts

Microsoft recommends creating individual domain user accounts for each of the Microsoft SQL Server Services and then running each service using its particular domain user account. These accounts should have minimal access rights assigned.

Using individual user accounts to run the Microsoft SQL Server Services will reduce the vulnerability of the entire network should the SQL Server ever face an external breach of security.

It is also a requirement of Virtual Cabinet to use a domain user account to run the SQL Server service.

Database Shrink

Never run a database shrink operation against any of the Reckon APS databases. The database shrink operation releases any free space from the database but it also causes all the indexes in the database to become 100% fragmented. This will have a detrimental effect of database performance.

This option is usually performed when transaction logs have grown very large due to the use an incorrect backup process when using the full recovery model (explained later in this guide). If this has happened, ensure that only the transaction log is shrunk and not the data file.

Backup and Recovery

Reckon APS recommend that SQL Databases are regularly backed up using the backup facilities available within the SQL Server Management Studio. These backup files should then be backed up to a secondary back-up location (Tape, Different server, SAN etc.) to be retained in accordance with your firm’s disaster recovery strategy.
When deciding on a backup strategy for your SQL databases it is important to know what the acceptable data loss is for your firm. Once this has been confirmed the relevant Recovery Model can be set for each of you databases and the type and frequency of backups can be determined.

It is also important to test your backups to ensure that if data is lost that it can be recovered.

Types of SQL Backups

SQL server provides three types of backups:

- **Full Backup** – The whole database is backed up and represents the database as at the time of the full backup’s completion.
- **Differential Backup** – Only data that has changed since the last full backup is captured. A differential backup is thus based on a preceding full backup
- **Transaction Log Backup** – Only backs up the transaction log and not the database. Must be used in conjunction with Full backups and optionally differential backups.

In addition to these backups SQL server also provides copy only versions of the full backup and transaction log backup. Copy only backups do not affect the sequencing of regular log backups. Copy only Transaction log backups are usually unnecessary. Copy only full backups are useful if the Full Recovery model is used and the need arises for a special backup to be taken that will not affect the normal sequence of your backup strategy.

Impact of the Recovery Model on Backup and Restore

Microsoft SQL Server offers three different recovery models: Simple, Full and Bulk Logged.

The recovery model is a database property; it controls how the database’s transaction log is managed. It also defines what type of backup and restore scenarios are supported for the database.

Generally a database will either have a Simple or Full Recovery model.

The best recovery model for your database will depend on your business requirements. To avoid Transaction log maintenance and simplify backing up and restoring a database use the Simple recovery model – however there is a potential for some data loss using this model. To minimise the potential for data loss use the full recovery model – However there is an administrative overhead associated with this recovery model, and the time required to recover the database to a point in time will be longer than with the simple model.

Simple Recovery Model

The simple recovery model requires full backups; optionally you could also perform differential backups between full backups. This model does not require log backups as SQL server will maintain the size and content of the log file. It allows for minimal transaction logging. Recovery is only possible to the last full backup and subsequent differential backups (if differential backups are used).

The simple recovery model can be used for:

- Test or development environment
- Production databases where a certain amount of data loss is acceptable

Full Recovery Model

The full recovery model requires full backups and transaction log backups; optionally you could also perform differential backups between full backups. All transactions are logged and are
stored in the transaction log. The transaction log will continue to grow until a transaction log backup is performed and the log is truncated by the transaction log backup.

The full recovery model with the correct implementation of full backups and transaction log backups allows for point in time restoring of the database but there is an administrative and time overhead associated with this model.

The full recovery model should be used for production databases where data loss should be almost eliminated.

Bulk Logged Model

The bulk logged recovery model is used in conjunction with the full recovery model. A database’s recovery model can be switched to bulk logged when bulk operations like Inserts or index creation is performed. These bulk operations will only be minimally logged thus increasing performance during these operations. Once the bulk operation is completed the database can be switched back to the full recovery model.

A database’s recovery model can be checked using Management Studio by right-clicking the database name, selecting Properties, and then clicking on Options in the left hand pane.

The recovery model can also be checked by executing the following script – change the name of the database to be checked in the WHERE clause if required.

```
SELECT [recovery_model_desc]
FROM [sys].[databases]
WHERE [name] = 'APS_DSQL' ;
```

For more details follow the link to Microsoft’s explanation of the different Recovery Models in the section Appendix III – Online Resources.

Practice Management (PM) Database Backup

The practice management database (usually called aps_dsql) contains all the client master records, their attributes and transactional data. This database should be backed up fully at a minimum of once a day. Please refer to the backup and recovery strategy of your firm to determine the correct backup frequency and type for the Practice Management database.

In addition to the daily backup(s) there is also a requirement to do ad-hoc backups of this database from time to time. E.g.: before month end rollover or before bulk updates. See Appendix I – Performing an Ad-hoc backup and Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.
PracticeIQ (PIQ) Backup and Recovery

The Advance PracticeIQ solution consists of the following:

- Data Warehouse database (aps_dss or similar name)
- Analysis Services database(s); Depending on the modules purchased one or more of the following Analysis services databases will exist: (Note, names may be different depending on deployment)
  - Advance PracticeIQ (PIQ Core)
  - Advance PracticeIQ Perspective
  - Advance PracticeIQ Workflow
  - Advance PracticeIQ Tax
  - Advance PracticeIQ CRM
- Integration Services Packages
- Reporting Services reports
- PracticeIQ website

The PracticeIQ solution does not have to be backed up on a daily basis but APS recommends the solution be backed up after the initial deployment, and then monthly thereafter, as well as whenever changes have been made to the solution.

Data Warehouse

The data warehouse consists of an application database, usually named aps_dss. It contains all the PIQ User data, PIQ Groups data, access rights to the reports, saved reports, site statistics and warehouse statistics. The data warehouse is built from data stored in the aps_dsql database.

The data warehouse does not have to be backed up on a daily basis as the transaction data is rebuilt every time the PIQ update job is run; however it also contains data that is of a more static nature (User information, access rights, saved views etc.) so it is therefore important to backup this database whenever changes have been made to the PIQ solution.

The Data Warehouse database backup can be scheduled using a maintenance plan “Back up Database Task”, or by performing an ad hoc database backup. See Appendix I – Performing an Ad-hoc backup and Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

Analysis Services Databases

The Advance PracticeIQ solution uses Analysis Services databases to store practice information specifically formatted for reporting and analysis. The data in these databases is updated every time the PIQ update job is run.

The Analysis Services databases do not have to be backed up on a daily basis as the data is rebuilt every time the PIQ update job is run; however it is important to backup these databases whenever changes have been made to the PIQ solution.

The Analysis Services databases cannot be backed up using maintenance plans.

Performing an Analysis Services Database Backup

As Analysis Services databases cannot be backed up using maintenance plans, to perform an Analysis Services database backup, the Backup Database dialog must be used.

1. Open SQL Server Management Studio (SSMS)
2. Change the Server Type to Analysis Services and click Connect
3. Expand the Databases node
4. Right click on the database to be backed up to bring up the context menu
5. Select Back Up…

Complete the form as required by the backup strategy. Key information for consideration:

- Database should display the name of the database to be backed up
- Backup file defines where to save the backup file to.
- Options offer replace (overwrite), compress and encrypt operations for the backup file. If choosing to encrypt the backup a password will need to be provided for decryption. Loss of this password will mean the backup cannot be decrypted and restored.

6. Select OK to accept your settings and backup the database.

To copy the backup database script to the clipboard:

1. Set the options as desired using the form above
2. Select the Script drop down arrow at the top of the dialog
3. Select Script Action to Clipboard (or press Ctrl + Shift + C)

The script can be pasted into a procedure document for record keeping, or pasted into an XMLA query window for execution. The script can also be used in an Agent Job to automate the backup. When executed, the script will perform the same actions as clicking the OK button after configuring the backup.
Reporting Services Backup and Recovery

Reporting Services consists of two databases: ReportServer and ReportServerTempDB.

ReportServer

The ReportServer database stores reports, linked reports, shared data sources, report models, folders, resources and all security settings and object properties for these objects.

Report subscriptions, schedule definitions and report history are also stored in the report server database. Reporting services is a stateless server; it uses the ReportServer and ReportServerTempDB to store application state and persistent data.

The ReportServer database should be backed up after the report solution has been configured and deployed and again after report or configuration changes have been deployed. Add the ReportServer database to a regular backup to ensure that report history is protected. See Appendix I – Performing an Ad-hoc backup and Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

ReportServer TempDB

The ReportServerTempDB database is a temporary database for storing session and execution data, cached reports, and work tables generated by Reporting Services. Each ReportServer database has a related ReportServerTempDB. The ReportServerTempDB database tables can be truncated occasionally; this operation will delete the report cache, resulting in a minor performance hit for report users. Backup the ReportServerTempDB database directly after installing Reporting Services. The empty backup will minimise recovery times. See Appendix I – Performing an Ad-hoc backup and Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

Integration Services Packages Backup

Integrations Services builds or updates the data warehouse by running Integration Services Packages. These packages are included in a SQL Server job that is scheduled to be run by the SQL agent at predetermined times. The PIQ solution will have one or more of the following packages depending on which PIQ modules have been purchased (note that package names may vary):

- DataWarehouseUpdate
- DataWarehouseUpdatePerspective
- DataWarehouseUpdateWorkflow
- DataWarehouseUpdateTax
- DataWarehouseUpdateCRM

The Integration Services packages do not have to be backed up on a daily basis. APS recommends that these packages be backed up after the initial installation of the PIQ solution and then once a month thereafter, and every time a change has been made to the PIQ solution.

Integration services packages cannot be backed up using Maintenance tasks. Creating a backup of these packages is a manual process: Follow the following steps to back up the packages:

1) Open SQL Management studio and change the Server Type to Integration Services and click connect.
2) Expand Stored Packages and then MSDB. It should look similar to the screenshot below (the packages may be in an APS folder under the MSDB folder):

3) Right click on the packaged to be backed up and select “Export Package”.
   - Change the Package Location to: File System
   - Specify the Package Path
   - Change Protection Level to: Do not save sensitive data
4) Click on “OK” and ensure that the package path specified forms part of the normal filesystem backup procedure.

**SQL Server Agent Jobs Backup**

The SQL Server Agent is used to create Jobs that will schedule the execution of tasks, which includes the PIQ Integration Services packages and any maintenance tasks.

The PIQ solution will have one or more of the following packages depending on which PIQ modules have been purchased (note that package names may vary):

- PIQ Update
- PIQ Workflow Update
- PIQ Tax Update

The SQL Server Agent jobs do not have to be backed up on a daily basis. APS recommends that these jobs be backed up after the initial installation of the PIQ solution, and then once a month thereafter and every time a change has been made to the PIQ solution.

SQL Server Agent jobs cannot be backed up using Maintenance tasks. Creating a backup of these packages is a manual process: Follow the steps below to back up the jobs:

1) Open SQL Management studio and change the Server Type to: Database Engine and click Connect

2) Expand “SQL Server Agent” and “Jobs”. It should look similar to the screenshot below:

3) Right click on the job to be backed up and select: Script Job as --> Create To -- > New Query Editor Window.
4) Save the query that has been created to a location that forms part of your normal filesystem backup.
Virtual Cabinet (VC) Backup

The Virtual Cabinet solution consists of two parts:

- The virtual Cabinet database (named VirtualCabinet) that contains all the metadata for each document
- The FileStore – The windows directory where the actual documents are stored

The Virtual Cabinet database should be backed up on a daily basis. The Virtual Cabinet database backup can be scheduled using the “Back up Database Task” task within a maintenance plan. See Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

The filestore is a normal windows directory that contains all the documents and should form part of your normal daily file backup procedure.

The location of the file store can be obtained by running the following SQL script on the SQL Server hosting the Virtual Cabinet database:

```sql
USE VirtualCabinet

SELECT [Path] from FileStore
```

Business Process Automation (BPA) Backup

The BPA (aps_dbpa) database should be backed up on a daily basis. The BPA backup can be scheduled using the “Back up Database Task” task within a maintenance plan. See Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

System Database Backup and Recovery

System Databases consist of the following:

- **Master** – The master database contains information about system and application databases, login accounts, server and user roles and system configuration settings.
- **Msdb** – The msdb database contains information relating to the SQL Agent. Jobs, Alerts, Schedules and database mail. It also contains the backup history and the SSIS package definitions.
- **Model** – The model database is the SQL server’s database template. It serves as the basis for each new database that is created.
- **Tempdb** – The Tempdb database is used to hold temporary user objects that are explicitly created, like temporary tables etc. It also stores objects that are created by the SQL Server Database Engine. It is not possible to back up the tempdb system database. This database is rebuilt each time an instance of Microsoft SQL Server is started.

The System databases also need to be backed up but less frequently than your production databases – usually when some sort of change has been made to the SQL server (e.g. new database created, access rights changed, new jobs created etc.). See Appendix I – Performing an Ad-hoc backup and Appendix II - Creating Scheduled Backups Using a Maintenance Plan for details on how this is done.

Backup Validation

Validating your SQL backups is just as important as backing up your databases. The only sure way to validate that your backups are working is to do an actual restore. For validation purposes
there is no need to restore over your current production database, restore to a test environment or if you do not use a test environment, restore the backup on your production server ensuring that you change the database name to that of a test database.
Index Maintenance

An Index is a data structure associated with a database table that stores the values for a specific column or columns in a table and is used to speed up the retrieval of data. Indexes are updated as data is added and removed from tables. As a result of this, indexes can become fragmented, resulting in a performance reduction when retrieving data.

It is important to maintain these indexes to ensure that the database is performing at its best. Microsoft SQL Server has two options for index maintenance:

- **Index Rebuilding** – When an index is rebuilt the whole structure is recreated and then populated. Except for SQL Server Enterprise Edition an index rebuild is always performed offline which means that the table will be locked and unavailable while the index is rebuilt.
- **Index Reorganising** – The index structure is not recreated but the ordering of pages within the leaf level of the index is fixed for optimised performance. Reorganising an index is always performed online, which means that the table is not locked and is available during this process.

It is recommended that for the APS Practice Management Database (aps_dsql), indexes are reorganised on a weekly basis and rebuilt on a monthly basis. Both these processes can be accomplished by setting up a SQL Maintenance plan. It is best practice to schedule these processes outside of normal working hours to minimise the impact on users. This can also be applied to other application databases.

Follow these steps to set up an Index Maintenance task.

1. Open Microsoft SQL Server Management Studio and expand the **Management** Section
2. Right Click on **Maintenance Plans** and select **Maintenance Plan Wizard**
3. Click Next on the **Maintenance Plan Wizard** window

   Complete the form as required.

   - Give the maintenance task a descriptive name and optionally a description
   - Select either **Separate schedules for each task**, or **Single schedule for the entire plan or no schedule**
4. To change the schedule click **Change**

Complete the scheduling page as per your backup strategy requirements

- Give the schedule a descriptive name
- Decide on the frequency of the schedule
- Depending on your choice above complete the Weekly or Monthly Frequency section
- Change the duration if needed
- Clicking OK will return to the previous page

5. Click **Next**
6. Select Reorganise Index or Rebuild Index, depending on which action is to be performed
7. Click Next

8. If more than one task is included, review their order and click Next
Complete the form as per your requirements:

- Select the database for which indexes are to be rebuilt (aps-dsql)
- Leave the objects as *Tables and Views*
- Do not change the Free Space options – always use: *Reorganise Pages with the default amount of free space*
- Click *Next*

9. Decide on report options

- Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document
10. Click Next

11. Review the actions to be performed and click Finish
12. Click Close

The newly created Maintenance Task will now be displayed under Maintenance Plans and a scheduled job will be available Under Jobs in the SQL Server Agent section.

Follow the same process to create a Reorganise Index maintenance task.

Database Statistics Maintenance

Database statistics are used by queries to determine the best execution plan when executed. The default settings in SQL server are to Auto Create and Auto Update statistics. These settings can be checked under the options page of the database properties.
Although SQL server automatically updates statistics there are circumstances when these statistics can become outdated. This usually manifests itself in degradation of performance, where performance does not improve after indexes have been defragmented.

It is recommended that the statistics be updated periodically. This can be done by executing the following query manually (preferably during non-working hours) in Microsoft SQL Management studio.

```
USE aps_dsql
EXEC sp_updatestats
GO
```

Database Statistics can also be updated as part of a Microsoft SQL Server Maintenance Task.

1. From within Microsoft SQL Server Management Studio expand Management; Right-click Maintenance Plans, Select Maintenance Plan Wizard and click Next on the splash screen
2. Give the maintenance plan a descriptive name and decide on the scheduling option to use; Select Change to define the schedule and click OK when done
3. Click Next

4. Click Next and select Update Statistics from the task list; click next
5. Confirm the task order and click Next

6. Complete the form to reflect the actions to be taken
   - Specify the databases against which the task should be run
   - Leave the default selection for Object
   - Select Update All existing statistics
   - Select Scan Type Full scan
   - Click Next

7. Decide on report options and click Next
   - Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document.

8. Review the actions to be performed and click Finish

9. Click Close

Database Consistency Checks (DBCC)

Even if a database is performing as expected it is useful to check the database for consistency from time to time.

This can be achieved by executing the following script (manually or as part of a SQL Agent Job). Change the database name to reflect the database that needs checking. This should only be done outside of working hours as this task is very resource intensive.

```
USE [aps_dsql]
DBCC CHECKDB
GO
```

The DBCC CHECKDB command will perform the following operations:

- Run DBCC CHECKALLOC on the database
- Run DBCC CHECKTABLE on every table and view in the database
- Run DBCC CHECKCATALOG on the database
- Validate the contents of each indexed view in the database
- Validate link-level consistency between table metadata and file system directories and files when storing varbinary(max) data in the file system using FILESTREAM
• Validate the Service Broker data in the database

On the message page that gets created, check the last line for any errors. Any errors should be investigated and corrected by the Database Administrator.

Database consistency can also be checked by creating a Check Database Integrity maintenance plan.

1. From Within Microsoft SQL Server Management Studio expand Management; Right-click Maintenance Plans, select Maintenance Plan Wizard and click Next on the splash screen.
2. Give the maintenance plan a descriptive name and decide on the scheduling option to use; Select Change to define the schedule and click OK when done

3. Click Next
4. Select *Check Database Integrity* from the task list; click *Next*

5. Confirm the task order and click *Next*

6. Complete the form to reflect the actions to be taken  
   - Specify the Databases against which the task should be run  
   - Select *Include indexes*  
   - Click *Next*

7. Decide on report options and click *Next*  
   - Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document.

8. Review the actions to be performed and click *Finish*

9. Click *Close*
Clean-up Tasks

History Clean-up

All historical data regarding Backups, Restores, the SQL Server Agent, and Maintenance plans are stored in the MSDB database. If this history is not cleaned out from time to time the MSDB database can grow very large.

This history can be cleared using a Maintenance plan performing the *Clean Up History* task.

2. Give the maintenance plan a descriptive name and decide on the scheduling option to use; Select *Change* to define the schedule and click *OK* when done.
3. Click *Next*.
4. Select *Clean Up History* from the task list and click *next*
5. Confirm the task order and click *Next*

![Image of Define History Cleanup Task dialog box]

6. Select the historical data to delete
   - Backup and restore history
   - SQL Server Agent job history
   - Maintenance plan history
7. Define the age at which the history is to be deleted and click *Next*
8. Decide on report options and click *Next*
   - Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document.
9. Review the actions to be performed and click *Finish*
10. Click *Close*

### General Clean-up Tasks

Backup files and Maintenance plan reports can consume a large amount of disk space. It is good practice to periodically delete these files. Make sure that backup files are no longer required or stored offsite before deleting these.

This task can be achieved manually or by setting up a Maintenance Plan executing the *Maintenance Cleanup Task*. The *Maintenance Cleanup Task* can only delete one type of file at a time, therefore three different maintenance tasks would have to be created to delete all the files (.bak, .trn and .txt). Creating a maintenance task with sub plans can accomplish all the deletions in one task but this falls outside the scope of this guide.

1. From Within Microsoft SQL Server Management Studio expand *Management*; right click *Maintenance Plans*, *Select Maintenance Plan Wizard* and click *Next* on the splash screen.
2. Give the maintenance plan a descriptive name and decide on the scheduling option to use; Select *Change* to define the schedule and click *OK* when done.
3. Click Next

4. Select Maintenance Cleanup Task from the task list and click Next
5. Confirm the task order and click Next
6. Complete the form to reflect the actions to be taken
   - Select the file types to be deleted
   - Ensuring Search folder and delete files based on extension is selected, browse to the folder location to be searched by clicking the ellipses button (…)
   - Specify the file extension (for backup files this will be .bak or .trn; for Maintenance Plan reports this will be .txt)
   - Ensuring Delete files based on the age of the file at task run time is ticked, specify the age of the files to be deleted
   - Click Next to continue
7. Decide on report options and click Next
   - Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document.
8. Review the actions to be performed and click Finish
9. Click Close
Appendix I – Performing an Ad-hoc backup

Individual Database Backup

The following steps can be used to perform an ad hoc database backup. Please note that SQL 2008R2 was used to produce this section. Depending on the version of SQL Server being used the process may vary.

1. Open SQL Server Management Studio (SSMS)
2. Right-click on the database to be backed up
3. Select Tasks --> Back Up…
4. Select General from the page options

Complete the form as required by the backup strategy. Key information for consideration:

- Database should display the name of the database to be backed up
- Backup type. (refer to the Types of SQL Backups and Appendix III – Online Resources sections of this guide)
- Backup set Name and Description are optional, use these fields to identify backup sets
- Backup set Expiration is optional, use these fields to set backup overwrite options
- Destination defines where to save the backup file.
  - Select “Disk” to write the backup to a file location or select “Tape” to write the backup directly to tape
  - If writing to disk, select “Add”, navigate to the backup folder by clicking the ellipses button (…), and set the File name for the backup, including the .bak extension.

5. Select “Options” from the left pane to display additional options

Complete the form as required by the backup strategy. Key information for consideration:

- Overwrite media settings determine if the backup is going to be added to, or will overwrite, an existing backup file
- Reliability settings specify additional quality checks to be performed on the backup file
- From SQL 2008 onward, backup files can be compressed by SQL Server

![Backup Database dialog box](image)

6. Select OK to accept your settings and backup the database.
7. Select OK when SSMS displays the backup completed information notification.

![Backup completed notification](image)

Individual Database Backup Script

To copy the backup database script to the clipboard:

1. set the options as desired using the forms above
2. select the Script drop down arrow at the top of the dialog
3. select Script Action to Clipboard or press Ctrl + Shift + C

The script can be pasted into a procedure document for record keeping, or pasted into a query window for execution. When executed, the script will perform the same actions as clicking the OK button after configuring the backup.

An example script (not suitable for production use):

```
BACKUP DATABASE [APS_DSQL] TO DISK = \
N"C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\Backup\aps_dsql.bak"
```
WITH NOFORMAT,

NOINIT,

NAME = N"APS_DSQ-L Full Database Backup"

GO

Appendix II - Creating Scheduled Backups Using a Maintenance Plan

Please note that you may need to create a number of backup maintenance tasks to accomplish your backup strategy.

The following steps can be used to create a scheduled backup using Microsoft Server Maintenance plans.

1. Open Microsoft SQL Server Management Studio and expand the Management Section

2. Right Click on Maintenance Plans and select Maintenance Plan Wizard

3. Click Next on the Maintenance Plan Wizard window

Complete the form as required.

- Give the maintenance task a descriptive name and optionally a description
- Select either Separate schedules for each task, or Single schedule for the entire plan or no schedule
4. To change the schedule click *Change*

Complete the scheduling page as per your backup strategy requirements

- Give the schedule a descriptive name
- Decide on the frequency of the schedule
- Depending on your choice above complete the Daily, Weekly or Monthly Frequency section
- Change the duration if needed
- Clicking OK will return to the previous page

5. Click *Next*
6. Select the Backup task to be performed – depending on your backup strategy select which Back up Database Task to be performed (Full, Differential or Transaction Log)
7. Click Next
8. If more than one task had been selected determine the order of execution
9. Click Next
10. Complete the Define Back up Database Task window according to your backup strategy
   - Select the database or databases that are included in this maintenance task. The options available are:
     - All Databases
     - System Databases
     - All User Databases – this will exclude system databases
     - These Databases – This allows you to select the particulate database(s) to be included
   - Decide on the backup component
   - Set when the backup will expire if needed
   - Set where to backup to (Disk or Tape)
   - Decide whether to backup databases across one or more files or to create a backup file for every database
   - Decide if databases should all be backed up to a single directory or whether each database should have its own sub directory
   - Specify the backup folder location
   - Decide on Verifying backup integrity
   - Set Backup compression if needed

11. Click Next

12. Decide on report options
   a. Note: In order to email the report, Database Mail must be set up, which falls outside the scope of this document.

13. Click Next
14. Review the actions to be performed and click Finish

15. Click Close

The newly created Maintenance Task will now be displayed under Maintenance Plans and a scheduled job will be available Under Jobs in the SQL Server Agent section.

Appendix III – Online Resources

The following web site references can be used to expand the topics covered in this guide.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>Location</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Transaction Log summary</td>
<td><a href="http://msdn.microsoft.com/en-us/library/d7be5ac5-4c8e-4d0a-b114-939eb97dac4d">http://msdn.microsoft.com/en-us/library/d7be5ac5-4c8e-4d0a-b114-939eb97dac4d</a></td>
</tr>
</tbody>
</table>