

SQL Server Database Backup and Restore Planning

By Bryan Oliver



- General Issues..... 3
 - What is a backup? 3
 - Backup and Recovery Planning 3
 - Server Data Classification 4
 - What should you be backing up? 4
 - When should backups occur? 4
 - Where should backups be stored?..... 4
 - Testing backups? 5
 - Who should be responsible for backups? 5
- Server Specifics 5
 - Server Classification 5
 - Server High Availability Requirements 5
 - Server Backup Schedule 5
- SQL Server Specifics 6
 - System Databases 6
 - Master 6
 - MSDB 6
 - Distribution 6
 - Model..... 6
 - User Databases..... 7
 - Backup Reports 7
- SQL Server Backup Planning Work Sheet 8
 - Server Profile 8
 - Server Classification 8
 - Server High Availability Requirements 8
 - Server Backup Schedules..... 8
 - SQL Server Specific 8
 - SQL Server High Availability Requirements 8
 - SQL Server Backup Requirements 9
 - SQL Server Database Backup Requirements 9
 - System Database..... 9
 - User Databases 9
 - Database High Availability Requirements..... 9
 - Database Backup Requirements 9

SQL Server Database Backup and Restore Planning

SQL Server Restore Planning Work Sheet.....10

 Server Restore.....10

 SQL Server Restore.....10

 SQL Server Database Restore.....10

General Issues

What is a backup?

In simple terms a backup is a copy of the data contained within a system. For a DBA, a backup can consist of either a single file or multiple files for one database to the entire server. These files can be classified into three types: a full backup, a partial or differential backup, and finally, a transaction log backup. By utilising all three types of backups, the amount of data that needs to be backed up at specific times and the overhead placed upon the hardware to complete the backups at critical times is reduced.

A Typical Backup Scenario

Full Backup at 00:00	Differential Backup at 12:00
Log Backup at 02:00	Log Backup at 14:00
Log Backup at 04:00	Log Backup at 16:00
Differential Backup at 06:00	Differential Backup at 18:00
Log Backup at 08:00	Log Backup at 20:00
Log Backup at 10:00	Log Backup at 22:00

The problem with this “Typical Backup Scenario” is that restoring a database is a time consuming process. For Example: To restore the database to the actual state it was at 16:00, you have to restore the Full Backup, the Differential at 12:00, and the Log at 14:00 and 16:00.

Why backup?

Although people differ, computers do fail and they aren’t perfect yet. The main reason to backup is to protect against the disruption of business from a variety of scenarios including disasters:

- hurricanes, a building fire, a flood or terrorist event
- hardware failures including disk drives, motherboards and CPUs
- software malfunctions leading to file/data corruption
- users that make mistakes, or undertake fraudulent or malicious activity

Companies with lost data incur extensive costs due to a combination of factors including loss of business, low productivity, legal action, and the cost of re-creating data. In 2002, a study from OnTrack, a company that provides data recovery services, showed that the cost of re-creating just 20 MB of data can be extensive:

Data type	Time to re-create 20 MB	Cost
Sales and marketing	19 days	\$17,000
Accounting	21 days	\$19,000
Engineering	42 days	\$98,000

Critical data loss can result in business collapse. A Study from the National Archives and Records Administration have shown that 80% of companies without well-conceived data protection and recovery strategies go out of business within two years of a major disaster.

Backup and Recovery Planning

Whether a business is small, medium or large business, it must have a well-written plan for backing up the servers. Planning a backup strategy up-front and documenting not only the backup process but also the restore process, will save you a ton of time in the end. Because of its value to the company and the sensitive nature of it, the classification of data must be carefully considered in the planning stage. Based upon these classifications, the

backup and restore plan will need to be tested and adjusted. While planning stage, data should be ranked according to sensitivity and value to the business.

Server Data Classification

Sensitivity of Data Held	Least	1	2	3	4	5	6	7	8	9	10 Highest
Value to Business of Data Held	Least	1	2	3	4	5	6	7	8	9	10 Highest

With data that is highly valuable to a company, plans must include an increased backup frequency due to the nature of the costs incurred while recapturing data in case of a to disaster. Recoverability plans must also consider the availability requirements of this data.

With highly sensitive data, plans must include encryption of backups, especially when this data is stored offsite.

What should you be backing up?

With a SQL Server, DBAs should also be concerned with the OS, the applications that the server runs and finally the databases. In other words, the entire server needs a backup and recovery plan. User databases are critical to backup plan, but system databases that contain significant information like Users, SQL Jobs and other system functionality, must also be taken into account.

When should backups occur?

When planning your backup strategy, each server and each database needs to be assessed on the six criteria below. Based upon this assessment, and agreement between the business owners of the database, a backup strategy should be implemented to reduce the liabilities caused by the loss of a server or database. For example a data warehouse that is highly valued by the business, but has no rekey cost and requires high availability, might only require a daily backup. While an accounting system with a high rekey cost and high frequency of changes may require frequent backups, every 15 minutes or so.

Six Criteria for Assessing Backup Needs

Value to Business of the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Ability to Rekey the data into DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Cost to Rekey the data into DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Frequency of changes to the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Amount of data changed per hr	Least	1	2	3	4	5	6	7	8	9	10 Highest
Cost to Restore the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest

Where should backups be stored?

All backups should be eventually stored in a offsite location. You don't want to be like the system administrator at a Paris Bank who ran back into a building on fire in 1996, because they didn't have offsite copies of their tape backups. In today's world, files can be transferred to storage sites via the internet. Companies can collect backup media and tapes or disk drives and store them at secure locations or transfer it to an another data centre within the company. Strategically backups should either go directly to another device like a SAN, iSCSI, NAS or file server

and from there backups should be archived to tape or removable disk drives so they can be sent to an offsite storage centre.

Testing backups?

Test restores should occur on a regular basis, including the entire system as well as individual databases. The only way to guarantee the integrity of a backup is to actually restore it followed by a DBCC check. There should always be a last complete backup set onsite so that immediate restores can be undertaken.

Who should be responsible for backups?

Ultimately the IT manager is responsible for backups, but you as the DBA should be taking responsibility for the backups of the SQL Servers and the databases contained within these servers. You should make sure the all documentation is complete and thoroughly tested. Good DBAs know how long it takes to backup the system and each database; exceptional ones know how long it takes to restore the system and each of the databases.

Server Specifics

The operating system and applications on the server need to be backed up at scheduled intervals and after any system changes. If you are applying service packs and updates regularly, than regular backups should be occurring along with these changes. When making changes to the underlying system user accounts, policies and installing third party products you should be undertaking at least an incremental backup so that these changes can be restored easily in the event of the server needing to be restored.

Server Classification

Value of the Server to Operations	Least	1	2	3	4	5	6	7	8	9	10	Highest
Cost to Business for Lost Time	Least	1	2	3	4	5	6	7	8	9	10	Highest
Server Availability	Least	1	2	3	4	5	6	7	8	9	10	Highest
Changes to OS	Least	1	2	3	4	5	6	7	8	9	10	Highest
Changes to Applications	Least	1	2	3	4	5	6	7	8	9	10	Highest

Server High Availability Requirements

Servers scored in the 10s should be clustered for hardware and OS redundancy along with a possible warm standby server either using a Physical or Virtual Server at another location.

Server Backup Schedule

Hourly incremental backup when file additions or updates occur frequently

Daily incremental backups to capture file additions and updates that have occurred during the day. If using hourly incremental backups consider differential backups to roll up all the changes that have occurred during the day..

Weekly this will be either a full or differential backup depending on the number of changes occurring to the non sql server database files.

Monthly and Yearly should always be a full backup of the entire system.

Major Changes should always be a full backup of the entire system.

Application Changes should be either a full backup or differential backup.

Operating System Changes should be either a full backup or differential backup.

SQL Server Specifics

System Databases

All system databases, excluding tempdb and resource, should be backed up on a strict schedule. In addition, they should be backed up after certain events to maintain the integrity of the server and enable recovery from a disaster.

Master

The master database should be backed up every day, without fail, and after any event that alters the SQL Server. These events include creating, altering or deleting a database, changing a data or log file of a user database, creating, altering or deleting logins, altering configuration options, etc.

MSDB

If MSDB is used for monitoring and scheduling of SQL Jobs and maintenance plans, a daily full backup with 6hrly differentials is recommended to minimize data loss from this database.

Distribution

For servers with replication, the distribution database needs to be backed up on a schedule based upon the method of replication. Transactional replication requires frequent log backups with a daily full backup, while snapshots require backups of either a log or differential after each snapshot is taken and either a daily or weekly full backup.

Model

Model is often the forgotten database in backup scheduling due to minimal changes occurring to it after installation. Still, a weekly full backup with ad-hoc full backups occurring after the database has any alterations or additions to it is recommended.

User Databases

Selecting the correct recovery model of a database makes it possible to select the correct backup strategy for that database. For instance, transaction log backups aren't possible if the recovery model is in simple mode. It is always safer to err on the side of caution and select full recovery model if you are unsure. For a data warehouse where large amounts of bulk operations will occur, it is possible to look at a bulk-logged recovery model to increase performance and reduce log file growth, but it reduces your abilities in undertaking point in time recovery operations.

Critical databases, including those where the data is costly to re-enter, and those that are important to the ongoing operations of the company, should be backed up on a very frequent schedule. In some instances, backups of the transactions should occur at the sub-second level as frequently as every 10 seconds. Servers that contain mission critical databases need to have redundancy built into them in the form of clustering and log shipping or mirroring of the actual database. In addition to undertaking frequent transactional backups, it is important to do differential backups on a frequent basis to reduce the number of files required to restore a database. Using software with next generation compression and de-duplication technology can help address this issue.

As the database becomes less critical, the number of transactional and the frequency of full backups can be reduced to a minimum. The least critical databases should only require a weekly full backup and daily differential backups for these databases. In these cases the recovery model should be set to simple so that the log file does not grow beyond the available disk space.

Backup Reports

Reporting of all backup and restore operations, not only the notification of failed backups and restores, but successful operations as well, should be part of your weekly schedule. From these reports it should be clear if backup periods and data are increasing. This allows you to make future adjustments to backup and restore plans before they become critical. It is also helpful for management to understand how long the restore operation will take both the entire server and in particular for individual databases. A standard report from SQL Server on Backup and Restore Events is a great start for reporting backups.

Backup and Restore Events
[Admin]
on BRYAN-PC\SQL2008 at 11/04/2009 12:29:17 AM



This report provides historical data about Backup and Restore actions performed on the Database.

Average Time Taken For Backup Operations
The average duration is calculated using the history of various types of backup operations performed on this database.

Backup Type	Average Duration (min.)
Database	0.02
Log	0.00

Successful Backup Operations
Shows details of the successful backup operations performed on this database in the past.

Start Time	Duration (minutes)	Backup Type	Backup Size	Backup Name	Device Type	User Name	Recovery Model	Differential Base LSN	Last LSN
11/04/2009 12:28:58 AM	0.00	Log	74.00 KB	Admin-Transaction Log Backup	Disk (temporary)	Bryan-PC\Bryan	FULL	Not applicable	37000000035100001
11/04/2009 12:28:41 AM	0.02	Database	1.95 MB	Admin-Full Database Backup	Disk (temporary)	Bryan-PC\Bryan	FULL	Not applicable	37000000034300001

Backup Operation Errors
Shows details of the errors, which caused the backup operations failure in the recent past.
No backup operations errors occurred for [Admin] database in the recent past; or default trace is not enabled.

Successful Restore Operations
Shows details of the successful restore operations performed on this database in the past.
No successful restore operations were performed on [Admin] database.

SQL Server Backup Requirements

SQL Server Database Backup Requirements

System Database

Database	Full Backup	Differential	Transaction Log
master	Daily	N/A	N/A
msdb	Daily	6 Hourly	N/A
model	Weekly	N/A	N/A
distribution*	Daily	6 Hourly	Hourly

*if replication is enabled

User Databases

Database Specific Database Name : <database name>

Sensitivity to Business of the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Value to Business of the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Ability to Rekey the data into DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Cost to Rekey the data into DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Frequency of changes to the DB	Least	1	2	3	4	5	6	7	8	9	10 Highest
Amount of data changed per hr	Least	1	2	3	4	5	6	7	8	9	10 Highest
Cost to Restore the Data	Least	1	2	3	4	5	6	7	8	9	10 Highest

Database High Availability Requirements

Mirrored Log Shipped Replicated

Database Backup Requirements

Database Name	Full Backup	Differential	Transaction Log
Critical	Daily	6 Hourly	5 to 15 minutes
High	Daily	6 Hourly	Hourly
Medium	Weekly	Daily	Hourly
Low*	Weekly	Daily	N/A

*Database in simple recovery mode

SQL Server Restore Planning Work Sheet

Server Restore

Full Restore to Physical or Virtual Server		Quarterly <input type="checkbox"/>	Yearly <input type="checkbox"/>
Test Restore of Individual Files	Weekly <input type="checkbox"/>	Monthly <input type="checkbox"/>	Yearly <input type="checkbox"/>
Full Restore of Individual Drive		Monthly <input type="checkbox"/>	Quarterly <input type="checkbox"/>

SQL Server Restore

Full Restore of SQL Server Instance		Quarterly <input type="checkbox"/>	Yearly <input type="checkbox"/>
Full Restore of System Databases		Monthly <input type="checkbox"/>	Yearly <input type="checkbox"/>

SQL Server Database Restore

Database	Single Database	All Databases	Full Failover
Critical Databases	Weekly <input type="checkbox"/>	Monthly <input type="checkbox"/>	Quarterly <input type="checkbox"/>
High Databases	Monthly <input type="checkbox"/>	Quarterly <input type="checkbox"/>	Yearly <input type="checkbox"/>
Medium Databases	Monthly <input type="checkbox"/>	Quarterly <input type="checkbox"/>	Yearly <input type="checkbox"/>
Low Databases	Monthly <input type="checkbox"/>	Quarterly <input type="checkbox"/>	Yearly <input type="checkbox"/>

Every time you undertake a restore, you should collect information about the length of time to restore, difficulties encountered and if testing by users was performed to check for functionality of the database. This will enable you to improve your restore plans over time.