Steve Stedman

Managing Technology Partner at SQL Data Partners

/in/SteveStedman

@SqlEmt

SQL Data Partners

Managing Technology Partner and Podcast Co-Host

Doing SQL Server performance tuning and general DBA tasks.

SQL Server Experience

Using SQL Server for 27 years

Creator of the Database Health Monitor

Founder of the Database Corruption Challenge

Blog regularly at http://SteveStedman.com

Bellingham SQL Server Users Group

Chapter founder and chapter leader.

Agenda

Your Backup and Recovery Strategy

- Recovery Point Objectives and Recovery Time Objectives
- Types of Backups
- RPO and RTO Scenarios
- Demo: Backup
- Demo: Restore
- Building a Restore Script
- Changes in SQL Server 2017

Schrodinger's Backup

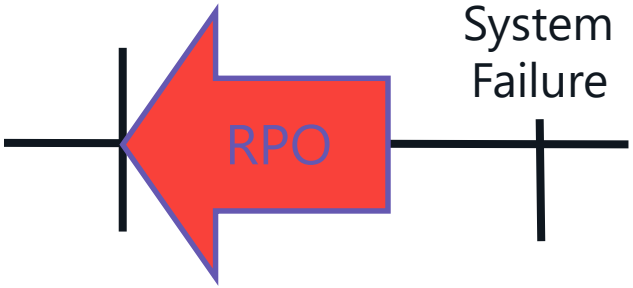
"The condition of any backup is unknown until a restore is attempted"

Test restoring of your backups!

RPO and RTO

Recovery Point Objective

When the system fails, the Recovery Point Objective is the worst case for a point that you can restore up to.



The point you can recover from

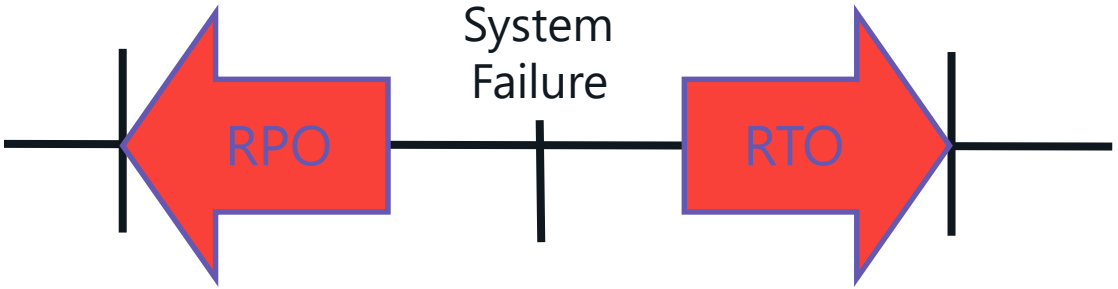
RPO and RTO

Recovery Point Objective

When the system fails, the Recovery Point Objective is the worst case for a point that you can restore up to.

Recovery Time Objective

Recovery Time Object is the worst case for how long it will take to get the system restored and running again.



The point you can recover from

The time it takes to recover

PASS

Factors with Recovery Point Objectives

Frequency of Backups

Weekly, Daily, Hourly, Every Minute, or something else

Type of Backups

Full, Differential, Transaction Log

Are Your Backups Reliable?

Have they been tested?

Factors with Recovery Time Objectives

Notification Time

How long is it before you know that a restore is required?

Response Time

How long does it take before you can start working on the restore?

Preparation Time

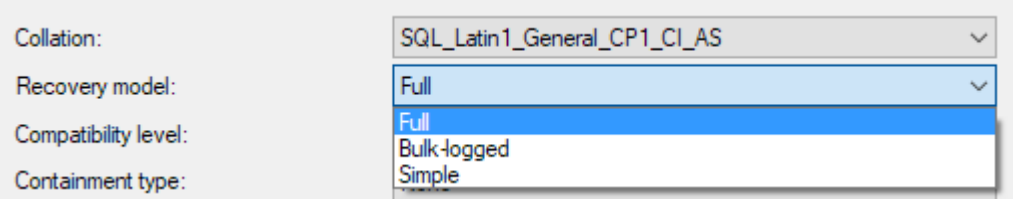
What do you need to do in order to start the restore?

Restore Time

Once the restore has been started, how long will it take?

PASS

Recovery Model and Backups



Simple Recovery Model

Transactions are written to the transaction log as they occur.

Once the transaction completes (commit or rollback), and the log is written to disk, the log is marked as re-usable.

The transaction log grows as large as is needed for currently running transactions.

Full Recovery Model

Transactions are written to the transaction log as the occur.

When the log is backed up the transaction log is marked as re-usable.

The transaction log continues to grow until a log backup occurs.

Bulk-Logged Recovery Model

Similar to Full Recovery Model.

In bulk logged recovery mode certain bulk operations are minimally logged.

The transaction log continues to grow until a log backup occurs.

PASS

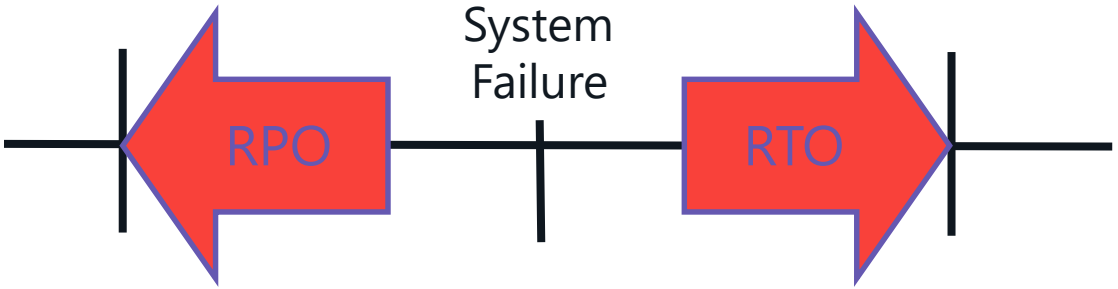
What if… would you still have a job?

Recovery Point Objective

What if your recovery point was 7 days?

Recovery Time Objective

What if it took you 2 days to get the system running again?



The point you can recover from

The time it takes to recover

PASS

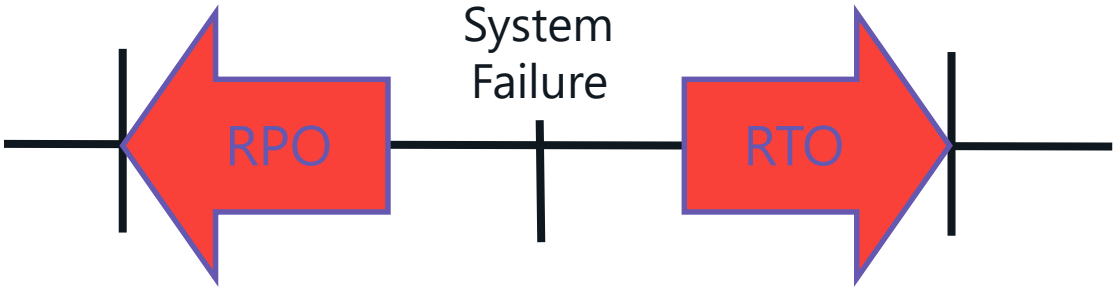
What if it was better?

Recovery Point Objective

Recovery point of 5 minutes

Recovery Time Objective

Recovery time of 30 minutes



The point you can recover from

The time it takes to recover

PASS

Types of Backups

Types of Backups

Full

A complete copy of the entire database.

Available with Simple, Bulk-logged and Full Recovery Models

Differential

Back up of all changes since the last full backup.

Available with Simple, Bulk-logged and Full Recovery Models

Transaction Log

Backup of all changes since the last transaction log backup

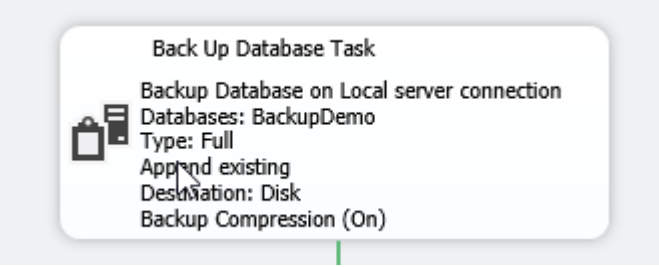
Available with Bulk-logged and Full Recovery Models

Copy Only

Similar to Full, but does not reset backup status.

Useful for taking an out of process backup.

Full Backups



A complete copy of your database

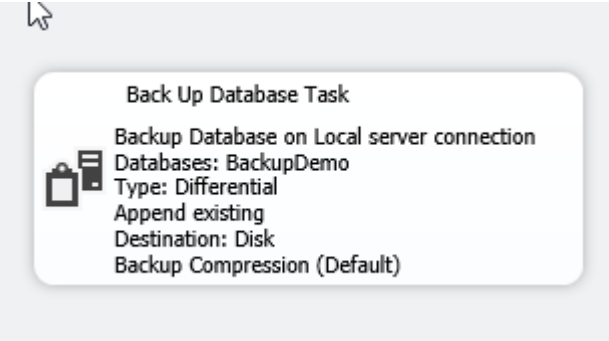
All you need to restore is the full backup file or files.

Full

- Restore Full Backup
- Single Step

PASS

Differential Backups



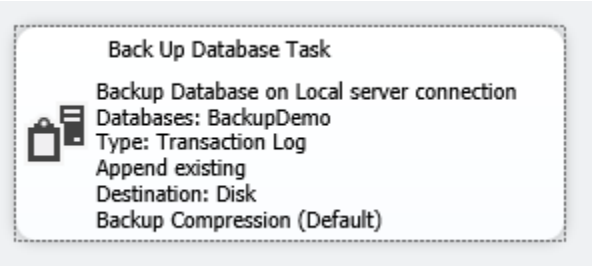
A copy of everything that changed since the last full backup

To restore you need the last full backup prior to the differential plus the last differential backup.

Full
- Restore Full Backup
- With NO_RECOVERY

Diff
- After restoring the full, restore a single differential backup
- Restore With RECOVERY to bring the database online

Transaction Log Backups



A copy of every transaction since the transaction log backup

To restore, you need the last full backup, or last full + differential, then all of the transaction logs since the last full or last full + differential.

Full
- Restore Full Backup
- With NO_RECOVERY

Diff
- After restoring the full, restore a single differential backup
- With NO_RECOVERY

Log
- Restore one or more transaction log backups in order
- Only restore With RECOVERY after the final log is applied.

RTO and RPO

Example Scenarios

RTO 24 hours – RPO 24 hours

Daily Full Backup – Manually restore when needed

RPO: up to 24 hours

RTO: depends on your manual restore process, and the database restore time. Could be up to 24 hours.

Will work in all recovery models, Simple, Full and Bulk-Logged.

RTO 8 hours – RPO 1 hour

Daily Full Backup, Hourly Differential Backup

At most lose 1 hour of data.

Manually restore when needed

The system could be back up and running in 8 hours.

Will work in all recovery models, Simple, Full and Bulk-Logged.

RTO 2 hours – RPO 5 minutes

Daily Full Backup, Differential Backup every 2 hours, Transaction logs every 5 minutes.

At most lose 5 minutes of data

Manually restore when needed

The system will be back up and running in 2 hours. Need to practice regularly and confirm that the database can be restored in under 2 hours

Will work in Full and Bulk-Logged recovery models.

RTO 1 hours – RPO 1 minutes

Daily Full Backup, Differential Backup every 2 hours, Transaction logs every minute.

At most lose 1 minutes of data.

Manually restore when needed

The system could be back up and running in 1 hours.

Need to practice regularly and confirm that the database can be restored in under 1 hours, including scripting of the restore.

Need to have the restore scripts created before the outage occurs.

Will work in Full and Bulk-Logged recovery models.

Configuring Backups with Maintenance Plans

Demo

Demo

Simulating weekly backups in just a few minutes

- Full backups every 5 minutes – Simulating a weekly full backup.
- Differential backups every 1 minute – Simulating a daily or twice daily differential backup.
- Transaction log backups every 10 seconds – Simulating ever 5 minutes.

Viewing the backup history

```sql
select top 1000
    bs.backup_start_date, bs.backup_finish_date,
    bs.backup_size, bs.compressed_backup_size,
    bs.compressed_backup_size / bs.backup_size * 100  as CompressionRation,
    case bs.type
        when 'L' then 'Log'
        when 'I' then 'Differential'
        when 'D' then 'Full Database' else bs.type end +
    case is_copy_only when 1 then '(COPY_ONLY)' else '' end as Type,
    bmf.physical_device_name as name, database_name,
    isnull(bs.[description], '') as [description]
  FROM msdb.dbo.backupset bs
 INNER JOIN msdb.dbo.backupmediafamily bmf ON bs.media_set_id = bmf.media_set_id
 WHERE database_name = 'BackupDemo'
  ORDER by backup_set_id desc;
```

Restoring your backups

Demo

Restore Full Backups

```
RESTORE DATABASE [BackupDemo]
 FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_184500_9077657.bak'
WITH MOVE N'BackupDemo' TO N'D:\SQLDATA\BackupDemo.mdf',
     MOVE N'BackupDemo_log' TO N'D:\SQLDATA\BackupDemo_log.ldf',
     REPLACE, RECOVERY, STATS = 1;
```

Full

- Restore Full Backup
- Restore with Recovery

Differential Backups

```
-- First restore the Full Backup
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185002_6323661.bak'
WITH MOVE N'BackupDemo' TO N'D:\SQLDATA\BackupDemo.mdf',
     MOVE N'BackupDemo_log' TO N'D:\SQLDATA\BackupDemo_log.ldf',  REPLACE, NORECOVERY, STATS = 1;
-- Then Restore the Differential Backup
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185203_7271247.bak'
    WITH NORECOVERY, STATS = 1;
```

Full

- Restore Full Backup
- With NO_RECOVERY

Diff

- After restoring the full, restore a single differential backup
- Restore With RECOVERY to bring the database online

Transaction Log Backups

```
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185002_6323661.bak'
WITH MOVE N'BackupDemo' TO N'D:\SQLDATA\BackupDemo.mdf',
     MOVE N'BackupDemo_log' TO N'D:\SQLDATA\BackupDemo_log.ldf',  REPLACE, NORECOVERY, STATS = 1;



-- And finally one or more log backups – Repeat as needed
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185220_7191334.trn' WITH NORECOVERY, STATS = 10;
RESTORE DATABASE [BackupDemo] WITH RECOVERY;
```

Full

- Restore Full Backup
- With NO_RECOVERY

Log

- Restore one or more transaction log backups in order
- Only restore With RECOVERY after the final log is applied.

PASS

Transaction Log Backups – STOP AT

STOPAT allows for you to specify a time partway through the log file to restore up to and stop.

```
-- First restore the Full Backup
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185002_6323661.bak'
WITH MOVE N'BackupDemo' TO N'D:\SQLDATA\BackupDemo.mdf',
     MOVE N'BackupDemo_log' TO N'D:\SQLDATA\BackupDemo_log.ldf',  REPLACE, NORECOVERY, STATS = 1;
-- Then Restore the Differential Backup
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185203_7271247.bak' WITH NORECOVERY, STATS = 1;
-- And finally one or more log backups
RESTORE DATABASE [BackupDemo] FROM DISK =
'C:\BackupDemo\BackupDemo\BackupDemo_backup_2017_09_11_185220_7191334.trn' WITH NORECOVERY,
  STOPAT = '20170911 18:52:15', STATS = 10;
RESTORE DATABASE [BackupDemo] WITH RECOVERY;
```

PASS

DEMO

Building a restore script when a database is backed up

Demo

Restore Script

Wouldn’t it be easier

- Create your restore script where there is no crisis
- Have your restore script ready when you need it
- Not have to sort out the order of log files, and which differential to use

Demo

```sql
EXECUTE [master]..[dbhLogChain] 'BackupDemo';
-- or
EXECUTE [master]..[dbhLogChain] 'BackupDemo', 'display';
-- or
EXECUTE [master]..[dbhLogChain] 'BackupDemo', 'FileInBackupDir';
```

Changes in SQL Server 2017

Demo

sys.dm_db_file_space_usage

```
SELECT * FROM sys.dm_db_file_space_usage;
```

New Columns in 2017 modified_extent_page_count

Results | Messages 2016

	database_id	file_id	filegroup_id	total_page_count	allocated_extent_page_count	unallocated_extent_page_count	version_store_reserved_page_count	user_object_reserved_page_count	internal_object_reserved_page_count	mixed_extent_page_count
1	26	1	1	1024	400	624	NULL	NULL	NULL	176

Results | Messages 2017

	database_id	file_id	filegroup_id	total_page_count	allocated_extent_page_count	unallocated_extent_page_count	version_store_reserved_page_count	user_object_reserved_page_count	internal_object_reserved_page_count	mixed_extent_page_count	modified_extent_page_count
1	1	1	1	512	456	56	NULL	NULL	NULL	240	0

PASS

sys.dm_db_file_space_usage

Can use this to calculate if a full backup or differential backup would be a better option.

2017 changes.sql -...ackupDemo (sa (56))*

```sql
SELECT df.name, df.physical_name,
    total_page_count,
    allocated_extent_page_count,
    modified_extent_page_count,
    100.0 * modified_extent_page_count / allocated_extent_page_count as PercentChanged
FROM sys.dm_db_file_space_usage fsu
INNER JOIN sys.database_files df on df.file_id = fsu.file_id;
```

119 %

Results | Messages

	name	physical_name	total_page_count	allocated_extent_page_count	modified_extent_page_count	PercentChanged
1	BackupDemo	G:\SQL2017\Data\BackupDemo.mdf	7424	4240	4008	94.528301886792452

Query executed successfully. | 192.168.21.111 (14.0 RC2) | sa (56) | BackupDemo | 00:00:00 | 1 rows

PASS

Other Backup Tools

Maintenance Solution by Ola Hallengren

Replace the need to use Maintenance plans with direct SQL jobs.

https://ola.hallengren.com/

- It is free
- Used by many
- Simplifies the backup process
- I use it with almost all clients

I am in no way affiliated with this product. I only know Ola by reputation, but I recommend you take a look at it.

Summary

Your Backup and Recovery Strategy

- Recovery Point Objectives and Recovery Time Objectives
- Types of Backups
- RPO and RTO Scenarios
- Demo: Backup
- Demo: Restore
- Building a Restore Script
- Changes in SQL Server 2017