

# PV226/MSSQL

## Microsoft SQL Server 2012


### Kapitola 3: Zálohování a obnova po havárii

**Bc. David Gešvindr**  
MCT | MSP | MCTS | MCITP | MCPD

# Osnova

- ➞ 1. Návrh strategie zálohování
- ➞ 2. Zálohování uživatelských databází
- ➞ 3. Obnova uživatelských databází
- ➞ 4. Obnova z databázového snapshotu

# 1. Návrh strategie zálohování

The background of the slide is a vibrant blue gradient. It features several overlapping, wavy, semi-transparent bands in various shades of blue, creating a sense of motion and depth. In the upper left corner, there is a soft, out-of-focus image of a sky with white clouds and a bright light source, possibly the sun, which adds a natural and airy feel to the design.

# Dostupné typy záloh

Typ zálohy	Popis
<b>Full</b>	Zálohuje všechny datové soubory a část transakčního logu
<b>Differential</b>	Zálohuje změněné datové stránky od poslední plné zálohy a nové záznamy v TL od poslední plné zálohy
<b>Transaction Log</b>	Zálohuje změny popsané v transakčním logu

# Recovery models

Recovery Model	Popis
<b>Simple</b>	<ul style="list-style-type: none"><li>• Záloha transakčního logu není podporována</li><li>• Nepotřebné části transakčního logu se automaticky odstraňují po dokončení transakce</li></ul>
<b>Full</b>	<ul style="list-style-type: none"><li>• Záloha transakčního logu je podporována</li><li>• Při havárii nedojde ke ztrátě žádných dat (Tail-Log Backup)</li><li>• Je možné provést obnovu databáze do konkrétního časové okamžiku</li></ul>
<b>Bulk Logged</b>	<ul style="list-style-type: none"><li>• Upravená verze Full režimu logující bulk operace jako celek nikoliv po řádcích</li><li>• Menší transakční log, potenciálně vyšší výkon</li></ul>

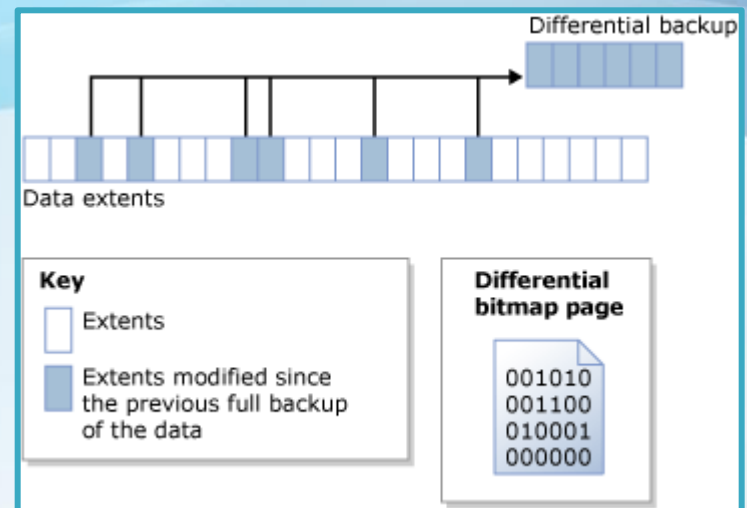


# Full Backup

- ➞ Provádí kompletní kopii databáze
- ➞ Vhodné pro malé databáze
- ➞ Je typicky prvním druhem zálohy, který je třeba obnovit po havárii

# Differential Backup

- ➔ Předpokládá provádění plných záloh
- ➔ Nahrazuje některé plné zálohy
- ➔ Ukládá změněné stránky od poslední plné zálohy
- ➔ Vhodná pokud se mění jen část databáze častěji než zbytek



# Transaction Log Backup

- Předpokládá provádění plných záloh
- Nahrazuje některé plné zálohy
- Zálohují se změny prováděné v databázi uložené ve formě transakčního logu
- Umožňuje databázi obnovit do libovolného časového okamžiku



# Kombinace záloh

- ➞ Zálohy je výhodné kombinovat
- ➞ Např.:
  - ➞ 2x do týdne plná záloha
  - ➞ 1x denně diferenciální
  - ➞ Každých 15 minut záloha transakčního logu
- ➞ Při plánování vycházejte z obsahu databáze
  - ➞ Kolik dat (za jaký časový úsek) je přípustné ztratit?

# Kam se ukládají zálohy

- ➞ Jsou zapisovány do zálohovacích zařízení
  - ➞ Soubor na pevném disku
  - ➞ Pásková jednotka
- ➞ SQL Server si v databázi MSDB uchovává historii záloh každé databáze

# Oprávnění zálohovat

- ➔ Právo zálohovat mají jen:
  - ➔ Sysadmin Members (Server Role)
  - ➔ Db\_owner Members (Database Role)
  - ➔ Db\_backupoperator (Database Role)

## 2. Zálohování uživatelských databází



# Kontrola integrity databáze

- ➔ Provedení kontroly integrity databáze provedete příkazem:

```
DBCC CHECKDB
```

- ➔ V databázi může dojít k poškození uložení některých objektů, případně jejich špatné alokaci



*ukázka*

Full Backup

*ukázka*

# Differential Backup

*ukázka*

Transaction Log Backup

# Tail Log Backup a Copy Only

## ➤ Tail Log Backup

- Zálohuje konec transakčního logu
- Potřeba pokud nechceme ztratit žádná data
- Není třeba když si vystačíme s již provedenými zálohami
- NO\_TRUNCATE

## ➤ Copy Only Backup

- Nenaruší pořadí obnovování databáze
- Záloha není uvedena v MSDB
- Neorízne se transakční log



# Kompresce záloh

- Novinka v SQL Serveru 2008
- Zmenší velikost zálohy
- Zvýší výkon I/O ale zvýší zátěž na CPU
- Podporováno v edicích Enterprise, BI i Standard



# Možnosti pro zaručení integrity záloh

- ⇒ Mirrored Media Sets
- ⇒ Backup Verification
- ⇒ Checksums
  
- ⇒ **Zálohovat je sice hezké, ale důležité je, aby šla záloha obnovit**

[http://msdn.microsoft.com/en-us/library/ms175053\(SQL.100\).aspx](http://msdn.microsoft.com/en-us/library/ms175053(SQL.100).aspx)

[http://msdn.microsoft.com/en-us/library/ms189587\(SQL.100\).aspx](http://msdn.microsoft.com/en-us/library/ms189587(SQL.100).aspx)

# 3. Obnova uživatelských databází



# Jak funguje obnovení

1. Fáze kopírování dat
2. Fáze zapsání již dokončených transakcí na disk (Redo Phase)
3. Pokud:
  - a) **WITH RECOVERY**  
Fáze zrušení nepotvrzených transakcí a přepnutí databáze do použitelného stavu (Undo Phase)
  - b) **WITH NO RECOVERY**  
Ponechání rozpracovaných transakcí aby mohla navázat další obnova zálohy

# Postup při obnově zálohy

1. Podle situace proved'te Tail-log backup
2. Ujasněte si, do jakého okamžiku chcete databázi obnovit
3. Ujasněte si postup obnovy
4. Zjistěte, které zálohy bude pro obnovu třeba
5. Začněte obnovu
  1. Poslední Full Backup
  2. Poslední Differential Backup
  3. Všechny zálohy transakčního logu od poslední FB/DB po daný cílový okamžik



*úkol*

# Úkol 1: Záloha databáze



# Úkol 1: Záloha databáze

1. Přepněte databázi AdventureWorks do Full Recovery režimu
2. Ověřte integritu této databáze
3. Vytvořte první plnou zálohu této databáze
4. Spust'te následující SQL dotaz:

```
UPDATE Person.Contact  
SET FirstName = 'Carl'  
WHERE FirstName = 'John'
```

# Úkol 1: Záloha databáze (2)

1. Proved'te diferenciální zálohu této databáze
2. Proved'te zálohu transakčního logu této databáze
3. Porovnejte velikost diferenciální zálohy a zálohy transakčního logu
4. Proved'te 2. plnou zálohu do jiného souboru se zapnutou kompresí a určete v % kolik místa se ušetřilo

*úkol*

# Úkol 2: Obnova databáze

# Úkol 2: Obnova databáze

1. Zazálohujte databázi všemi 3 druhy záloh v pořadí Full, Differential a Transaction Log
2. Vytvořte ještě jednu zálohu transakčního logu
3. Obnovte databázi AdventureWorks do stavu 10s před provedením poslední zálohy transakčního logu



# Úkol 2: Obnova databáze

4. Obnovte databázi AdventureWorks do původní podoby
5. Zazálohujte databázi všemi 3 druhy záloh v pořadí Full, Differential a Transaction Log
6. Vytvořte ještě jednu zálohu transakčního logu
7. Proveďte operaci v **MARKED TRANSACTION**
8. Obnovte databázi AdventureWorks do stavu před provedením označené transakce

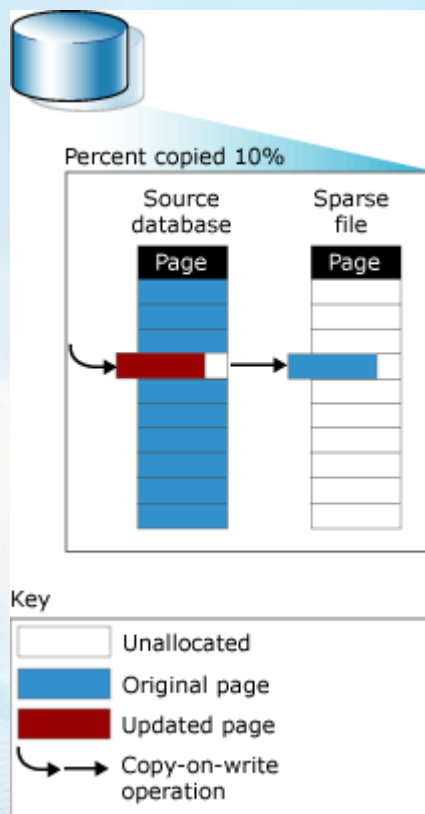
# 4. Obnova z databázového snapshotu

The background of the slide features a bright blue sky with wispy white clouds. In the foreground, there are several overlapping, flowing, abstract shapes in various shades of blue, creating a sense of movement and depth. The overall aesthetic is clean, modern, and professional.

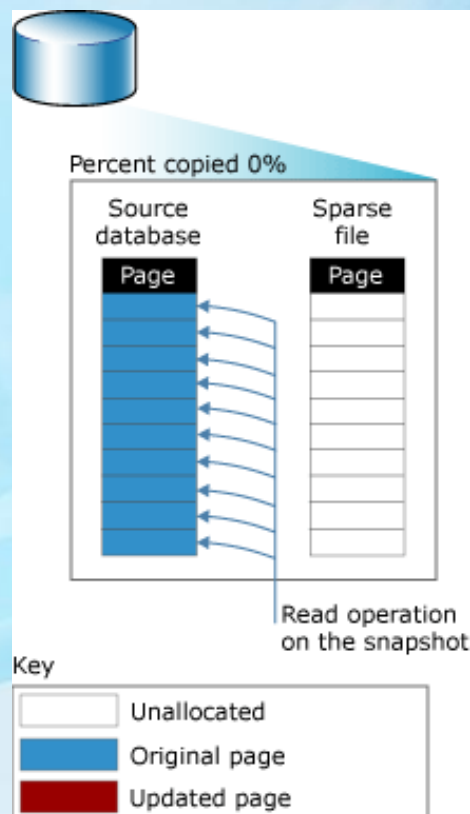
# Databázový snapshot

- ➞ Read-only pohled na databázi v konkrétním stavu
- ➞ Typické použití:
  - ➞ Tvorba reportů nad databázemi v určitém okamžiku
  - ➞ Ochrana před chybou uživatele/správce
  - ➞ Rychlá obnova dat

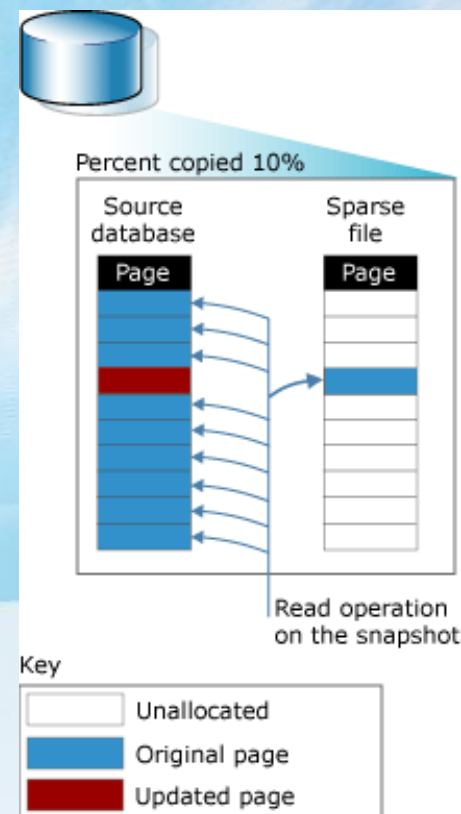
# Jak funguje snapshot



**Zápis: Copy-on-write**



**Čtení databázového snapshotu**





*ukázka*

Databázový snapshot

# 5. Automatizace záloh a údržby



# SQL Server Agent

- Komponenta SQL Serveru sloužící jako časovač
- Vytváříme Job, tvořen kroky (Job Step)
- Umožňuje spustit:
  - T-SQL skript
  - PowerShell skript
  - Externí aplikace
  - SSIS balíček
- Spouštění je řízeno časovým plánem

# SQL Server Agent

- Umožňuje definovat Alerty a Operátory
- Využití:
  - Když dojde k selhání některého kroku je o tom informován emailem operátor co má právě službu
- Vyžaduje nakonfigurovaný Database Mail



# Maintenance Plans

- Postaveny nad SSIS
- Umožňují **vizuální** sestavení plánů údržby databází v dané instanci formou **workflow**
- Jsou spouštěny SQL Agentem v daných časech

*ukázka*

**Tvorba Maintenance Plans**

# Úkol 3: Automatizace údržby

*úkol*

# Úkol 3: Automatizace údržby

1. Vytvořte Maintenance plán, který pokrývá kompletní údržbu databáze a obsahuje i upozornění pro operátory