



SQL-MDL

Jiří Zimola

1. Úvod do SQL

- SQL (Structured Query Language) je jazyk pro správu a manipulaci s relačními databázemi.
- Používá se k vytváření, čtení, aktualizaci a mazání dat (CRUD operace).
- Klíčové příkazy:
 - SELECT - výběr dat
 - INSERT - vložení dat
 - UPDATE - aktualizace dat
 - DELETE - odstranění dat
 - CREATE, ALTER, DROP - práce se strukturou databáze
- SQL se používá v různých databázových systémech jako MySQL, PostgreSQL, SQL Server a Oracle.

2. Co je MDL (Meta Data Locking)?

- Mechanismus v MySQL, který chrání metadata databázových objektů (tabulek, indexů, pohledů atd.).
- Zabraňuje změnám metadat, pokud jsou objekty používány jinými dotazy.
- MDL byl zaveden od MySQL 5.5.
- Automaticky se aplikuje a nevyžaduje explicitní správu uživatelem.

Jak MDL funguje?

- Čtení (READ lock): umožňuje souběžné čtení, ale blokuje změny metadat.
- Zápis (WRITE lock): blokuje ostatní operace čtení i zápisu na dobu úprav metadat.
- MDL se aplikuje na tabulky při běžných dotazech (SELECT, INSERT, UPDATE, DELETE) a operacích jako ALTER TABLE.

4. Problematika MDL zámků

- **Deadlock (zámek na mrtvém bodě):** může nastat, pokud více procesů čeká na uvolnění MDL.
- **Blokování ALTER TABLE:** změna struktury tabulky může být blokována dlouhými dotazy.
- **Řešení:**
 - **Použití** pt-online-schema-change nebo gh-ost pro změny struktury bez blokování.
 - **Krátké transakce a optimalizace dotazů.**
 - **Správné nastavení zámků a izolace transakcí.**
 - **Monitorování zámků pomocí** SHOW ENGINE INNODB STATUS a INFORMATION_SCHEMA.

5. Příklady SQL příkazů a MDL chování

INSERT – Vložení dat

- Používá se k přidání nových záznamů do tabulky.

```
INSERT INTO uzivatele (id, jmeno, vek) VALUES (1, 'Petr', 30);
```

```
INSERT INTO uzivatele (id, jmeno, vek) VALUES (2, 'Anna', 25);
```

```
INSERT INTO uzivatele (id, jmeno, vek) VALUES (3, 'Karel', 40);
```

```
INSERT INTO objednavky (id, uzivatel_id, cena) VALUES (1, 1, 250.50);
```

```
INSERT INTO objednavky (id, uzivatel_id, cena) VALUES (2, 2, 125.00);
```

UPDATE – Aktualizace dat

- Používá se k modifikaci existujících záznamů.

UPDATE uzivatele SET vek = 31 WHERE id = 1;

UPDATE uzivatele SET jmeno = 'Pavla' WHERE id = 2;

UPDATE objednávky SET cena = 300.00 WHERE id = 1;

UPDATE uzivatele SET vek = vek + 1 WHERE jmeno = 'Karel';

UPDATE objednávky SET cena = cena * 0.9 WHERE
uzivatel_id = 2;

DELETE – Odstranění dat

- **Používá se k odstranění záznamů z tabulky.**

DELETE FROM uzivatele WHERE id = 2;

DELETE FROM objednávky WHERE cena < 150.00;

DELETE FROM uzivatele WHERE vek > 60;

DELETE FROM objednávky WHERE uzivatel_id NOT IN (SELECT id FROM uzivatele);

DELETE FROM uzivatele WHERE jmeno LIKE 'P%';

6. Příklady MDL chování

Výběr dat (**SELECT**)

- Aplikuje **MDL READ LOCK**.

SELECT * FROM uzivatele;

SELECT jmeno, vek FROM uzivatele WHERE vek > 25;

SELECT * FROM objednávky WHERE cena > 200;

Změna struktury tabulky (**ALTER TABLE**)

- Aplikuje **MDL WRITE LOCK**.

ALTER TABLE uzivatele ADD COLUMN email VARCHAR(100);

ALTER TABLE objednávky MODIFY COLUMN cena
DECIMAL(10,2);

Detekce zámků v MySQL

```
SELECT * FROM performance_schema.metadata_locks;
```

Uvolnění zablokovaných transakcí

```
KILL QUERY <thread_id>;
```

8. Pokročilé techniky správy MDL

- **Monitorování MDL:**
 - Použití **SHOW PROCESSLIST** pro identifikaci blokováných dotazů.
 - **INFORMATION_SCHEMA.INNODB_TRX** ukazuje běžící transakce.
- **Optimalizace výkonu:**
 - Použití indexů ke snížení doby dotazů.
 - Správná izolace transakcí (**READ COMMITTED**, **SERIALIZABLE**).
- **Použití replikace:**
 - Asynchronní replikace může minimalizovat dopady MDL na výkon.

9. Závěr

- SQL je mocný nástroj pro správu databází.
- MDL zajišťuje konzistenci metadat, ale může vést k blokacím.
- Správné řízení transakcí, optimalizace dotazů a pokročilé techniky pomáhají minimalizovat problémy s MDL.
- Monitoring a nástroje jako pt-online-schema-change umožňují efektivnější správu databáze.