

Datové a procesní modely

Relační databáze

Přednáška 4 b Sou

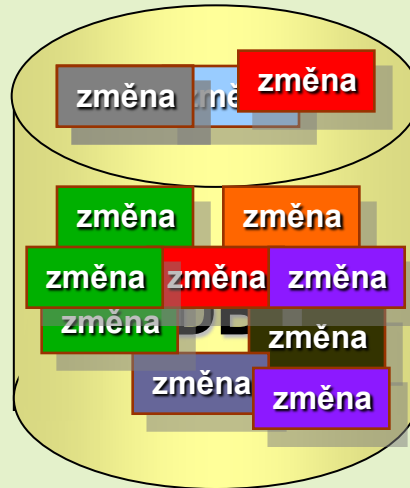
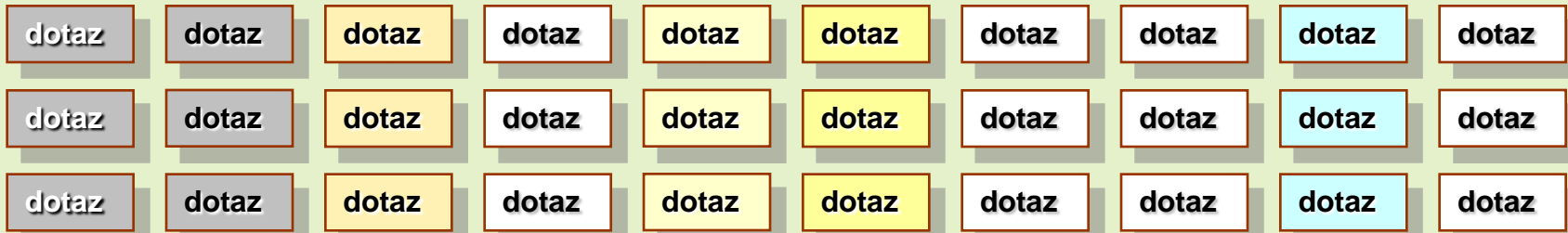
Marian Kamenický

Synte software group a.s.
marian.kamenicky@syntea.cz

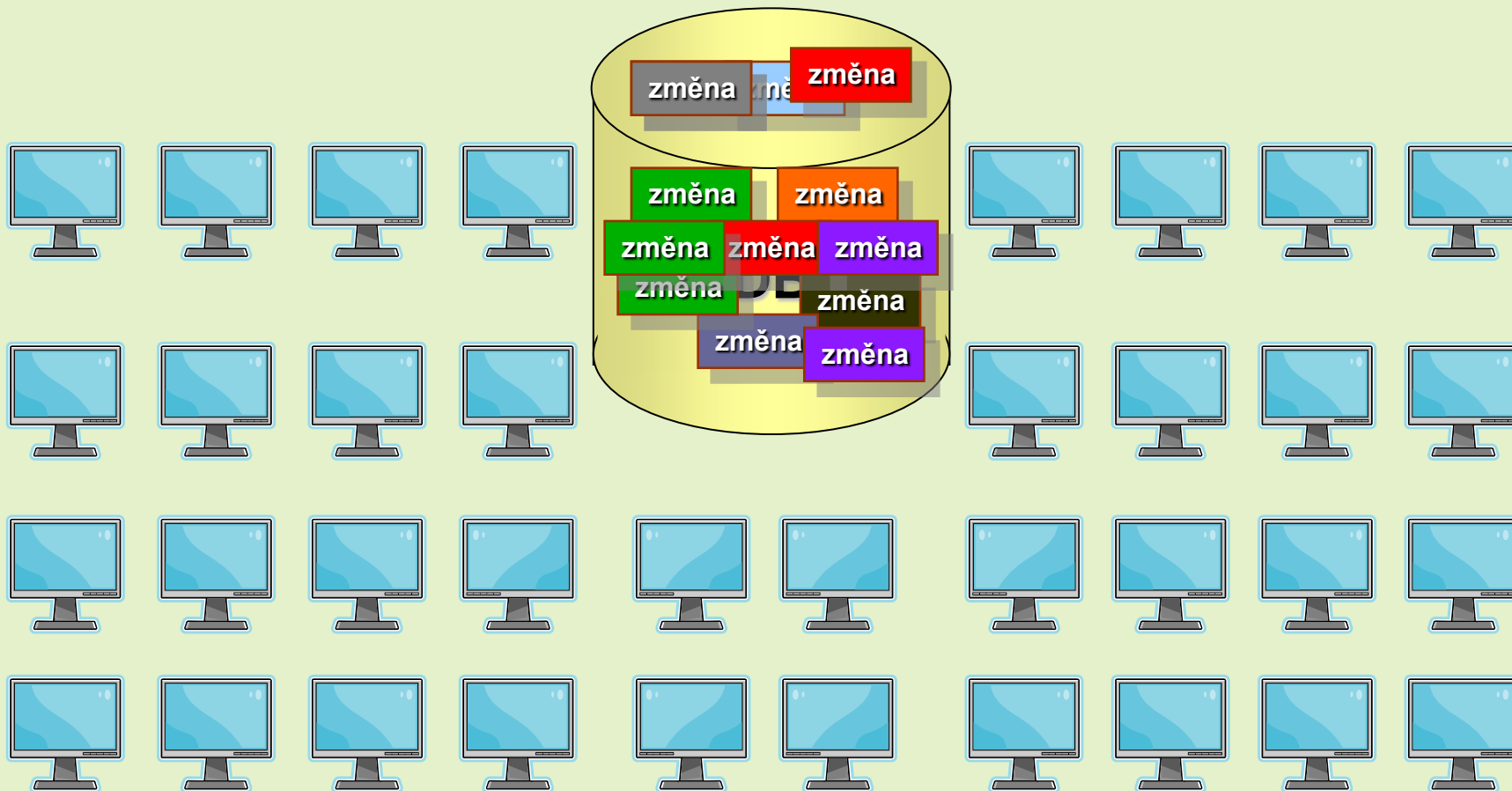
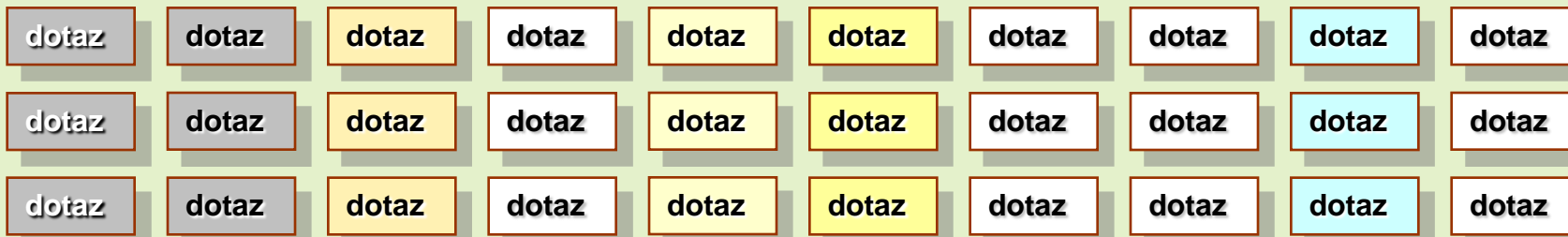
MFFUK Praha

2019/20

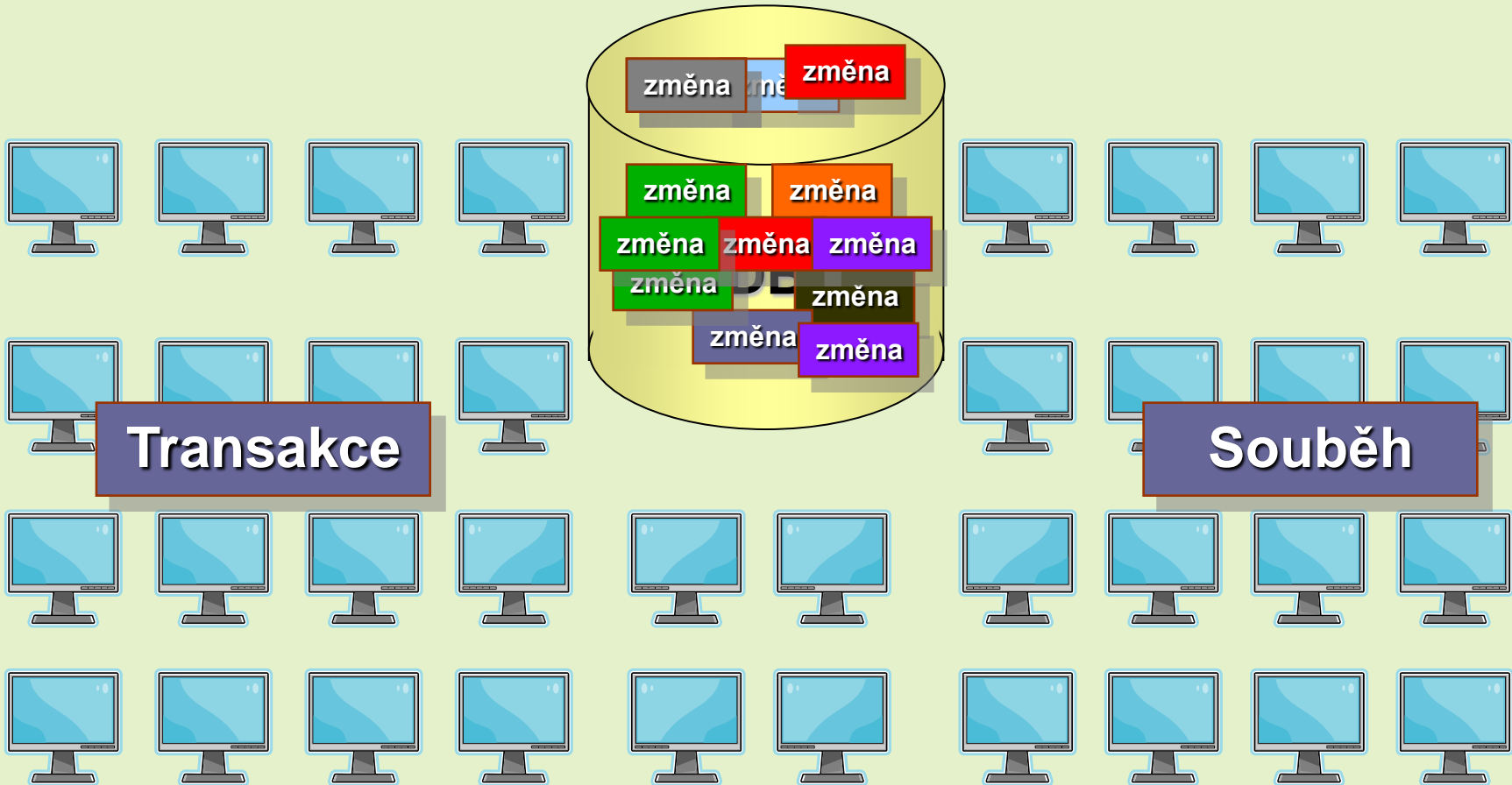
Brutální prostředí provozní databáze



Brutální prostředí provozní databáze



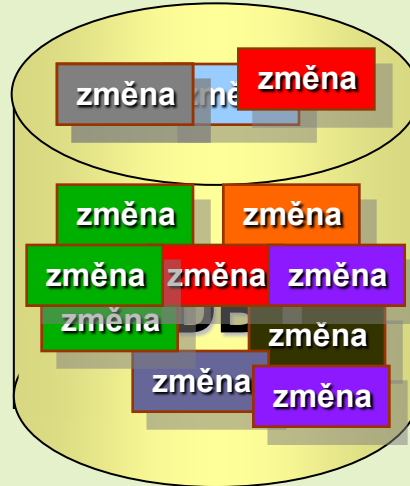
Brutální prostředí provozní databáze



Brutální prostředí provozní databáze



Jak to zvládnout ?



Transakce

Souběh

Transakce - víceuživatelské prostředí – souběh



Pepa

Zbozi	Mnoz
lampa A	10



Tom

```
SELECT Mnoz .. FROM.. WHERE.. -> 10
```

Objednávka na 8 10-8 ----> 2

```
UPDATE .. SET Mnoz = 2
```

```
SELECT Mnoz .. FROM.. WHERE.. -> 10
```

Objednávka na 6 10-6 ----> 4

```
UPDATE .. SET Mnoz = 4
```

Zbozi	Mnoz
lampa A	2

Zbozi	Mnoz
lampa A	4

Problém ztracené aktualizace

Aktualizace jsou prováděny bez vzájemného ohledu

Transakce - víceuživatelské prostředí – souběh



Pepa

Zbozi	Mnoz
lampa A	10



Tom

```
SELECT Mnoz .. FROM.. WHERE.. -> 10  
Objednávka na 8    10-8    ----> 2  
UPDATE .. SET Mnoz = 2
```

Pro zákazníka moc drahé - **Storno**
ROLLBACK WORK

Zbozi	Mnoz
lampa A	2

```
SELECT Mnoz .. FROM.. WHERE.. -> 2  
Objednávka na 6  
Odmítnutí objednávky na 6
```

Zbozi	Mnoz
lampa A	10

Problém nepotvrzených dat (nečisté čtení)

Změny v DB se vrátí zpět poté, co se projeví jejich účinky

Transakce - víceuživatelské prostředí – souběh



Pepa

Zbozi	Mnoz
lampa A	10



Tom

```
SELECT Mnoz .. FROM.. WHERE.. -> 10  
Objednávka na 8      10-8      ----> 2  
UPDATE .. SET Mnoz = 2  
COMMIT
```

```
SELECT Mnoz .. FROM.. WHERE..->10  
Jiné SELECTy
```

10

Zbozi	Mnoz
lampa A	2

```
SELECT Mnoz .. FROM.. WHERE..-> 2
```

2

- v rámci jedné transakce dva různé výsledky téhož dotazu
- externí události zasahují do konzistentního podledu na DB

Problém nekonzistentních dat

(neopakovatelné čtení)

- jedna akce je ovlivněna částečným výsledkem jiné akce
- SUM (sloupec) při sčítání se jednotlivé sloupce mění – nejasný výsledek
- statistiky se dějí na proměňujících se datech

Transakce - víceuživatelské prostředí – souběh



skladník

Zbozi	Mnoz
lampa A	10
lampa C	10



Tom

INSERT ... ('lampa B',15)

Zbozi	Mnoz
lampa A	10
lampa B	15
lampa C	10

SELECT SUM(Mnoz)... FROM... -> 20 **20**
SELECT count(*) ... FROM... -> 2 **2**

SELECT SUM(Mnoz)... FROM... -> 35 **35**
SELECT count(*) ... FROM... -> 3 **3**

Problém příznačných dat

(vlození přeludu)

- byl přidán řádek, který tam před tím nebyl (INSERT)
- varianta nepotvrzených dat
- při čtení dle primárního klíče je u vložení příznaku výsledek vždy týž
- u nepotvrzených dat se mohou výsledky různit

Transakce - víceuživatelské prostředí – souběh

- to je stejný konflikt jako ten předchozí



- není
- výsledek dotazu na stejný řádek bude tentýž
 - i v případě, že se řádek změnil
- technicky se to dá zařídit například tak
- že si stroj pamatuje
 - na co jste se [v transakci] ptal
- ale nepamatuje si
 - na co jste se ještě neptal
- do součtu nový řádek zařadí
 - neví, že vznikl až potom
 - co jste poprvé sumarizoval



IN

u n

Transakce - víceuživatelské prostředí – řešení souběhů

- problém ztracené aktualizace
 - problém nepotvrzených dat (nečisté čtení)
 - problém nekonzistentních dat (neopakovatelné čtení)
 - problém příznačných dat (vlození přeludu)
- to máme teda móře problémů



Řešení ??

➔ **IZOLACE transakcí**



Izolace transakcí

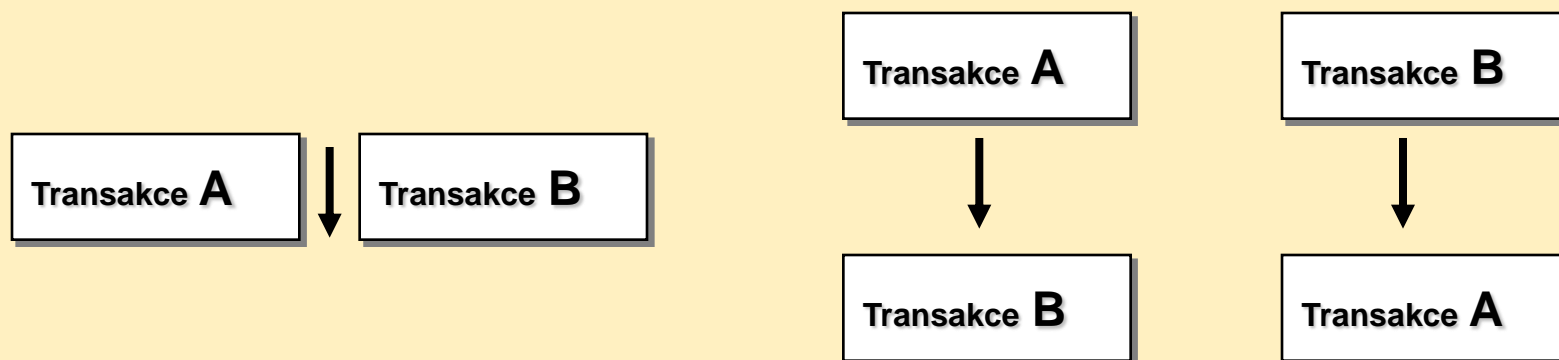
- v průběhu transakce pohled na DB
- zcela konzistentní
- v transakci **nejsou vidět**
- **nepotvrzené změny** jiných uživatelů
- v transakci **nejsou vidět**
- **potvrzené změny** jiných uživatelů
 - potvrzené v průběhu (uprostřed) dané transakce

- jak řešit větev

```
IF Mnozstvi > 0 THEN  
    v průběhu větve Mnozstvi < 0  
END THEN
```

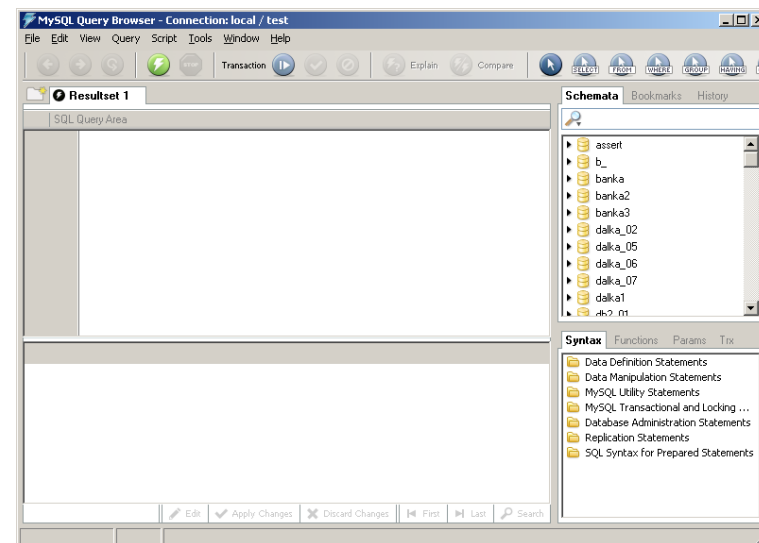
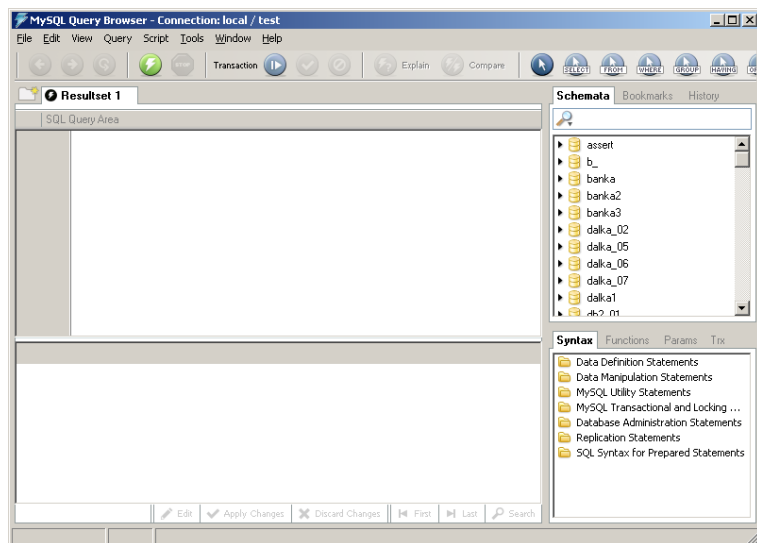
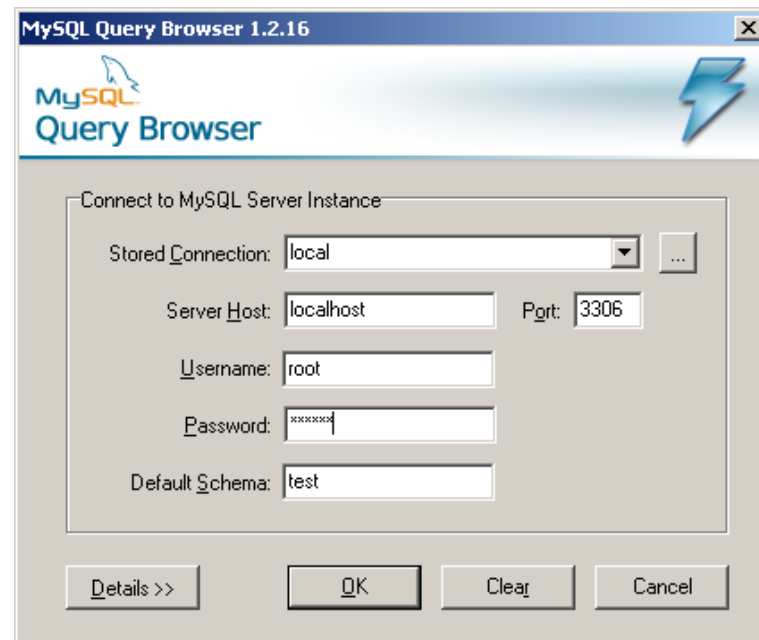
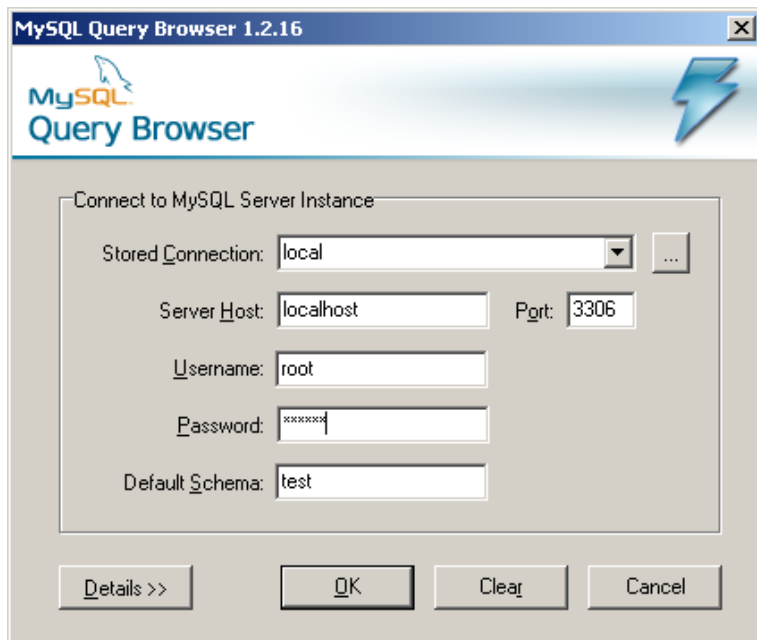
Izolace transakcí - serializací

- serializace transakcí
- transakce A,B spuštěny souběžně
- DBS zajistí
- výsledky budou stejné, jako by
 - transakce A proběhla první a za ní transakce B
 - NEBO transakce B proběhla první a za ní transakce A

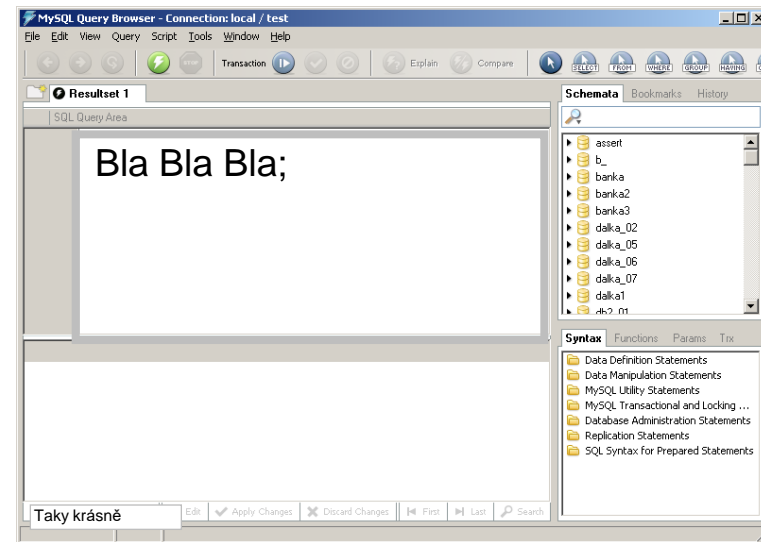
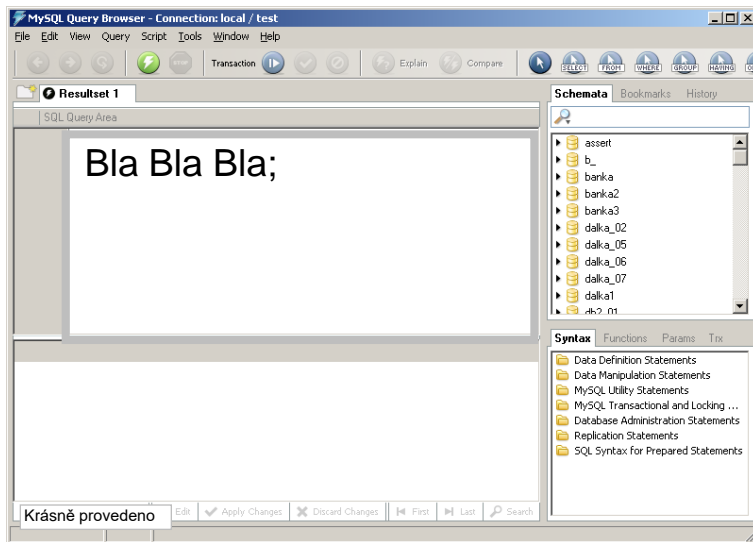


- skrytí nepotvrzených transakcí
- skrytí transakcí
- potvrzených po zahájení mé transakce

MySQL souběh dvou transakcí



MySQL souběh dvou transakcí



MySQL transakce

```
Update WinLa Set Mnoz=8;  
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-2;  
COMMIT;  
Select * From WinLa;
```

WilMyLa	
Dil	Mnoz
Lampa	8

WilMyLa	
Dil	Mnoz
Lampa	6

```
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-5;  
COMMIT;  
Select * From WinLa;
```

WilMyLa	
Dil	Mnoz
Lampa	8

WilMyLa	
Dil	Mnoz
Lampa	1

MySQL Query Browser - Connection: local / test

Resultset 1

SQL Query Area

Schemata

- asset
- bc
- banka
- banka2
- banka3
- dalka_02
- dalka_05
- dalka_06
- dalka_07
- dalka1
- re-2_m

Syntax | Functions | Params | Trx

- Data Definition Statements
- Data Manipulation Statements
- MySQL Utility Statements
- MySQL Transactional and Locking ...
- Database Administration Statements
- Replication Statements
- SQL Syntax For Prepared Statements

1 row affected

MySQL Query Browser - Connection: local / test

Resultset 1

SQL Query Area

Schemata

- asset
- bc
- banka
- banka2
- banka3
- dalka_02
- dalka_05
- dalka_06
- dalka_07
- dalka1
- re-2_m

Syntax | Functions | Params | Trx

- Data Definition Statements
- Data Manipulation Statements
- MySQL Utility Statements
- MySQL Transactional and Locking ...
- Database Administration Statements
- Replication Statements
- SQL Syntax For Prepared Statements

Query is being executed



MySQL transakce

```
Update WinLa Set Mnoz=8;  
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-2;  
COMMIT;  
Select * From WinLa;
```

WilMyLa	
Dil	Mnoz
Lampa	8

WilMyLa	
Dil	Mnoz
Lampa	6

```
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-5;  
COMMIT;  
Select * From WinLa;
```

WilMyLa	
Dil	Mnoz
Lampa	8

WilMyLa	
Dil	Mnoz
Lampa	1

- došlo k serializaci transakcí
- druhá čekala na ukončení [Commit] první
- po ukončení první transakce druhá dokončila Update

1 row affected

Data Definition Statements
Data Manipulation Statements
MySQL Utility Statements
MySQL Transactional and Locking ...
Database Administration Statements
Replication Statements
SQL Syntax for Prepared Statements

Query is being executed

Data Definition Statements
Data Manipulation Statements
MySQL Utility Statements
MySQL Transactional and Locking ...
Database Administration Statements
Replication Statements
SQL Syntax for Prepared Statements



MySQL transakce

```
Update WinLa Set Mnoz=8;  
BEGIN;  
Select * From WinLa;  
Update WinLa Set Mnoz=Mnoz-2;
```

WilMyLa	
Dil	Mnoz
Lampa	8

```
BEGIN;  
Select * From WinLa;  
Update WinLa Set Mnoz=Mnoz-5;
```

WilMyLa	
Dil	Mnoz
Lampa	8

- druhá transakce čeká na ukončení první
- počkáme chvíli s Commit
- až do vypršení času timeout

1 row affected

Query is being executed



MySQL transakce

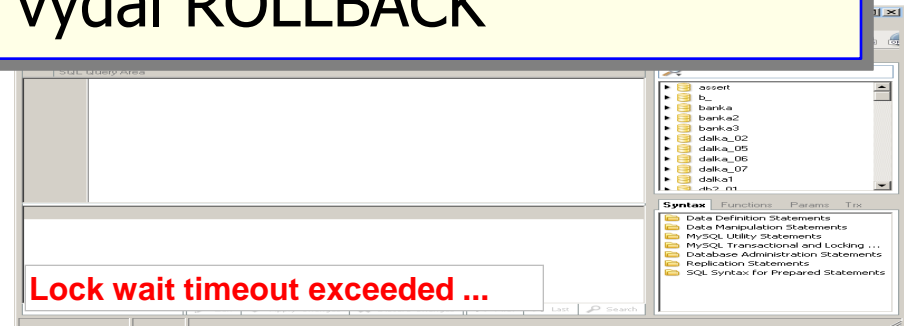
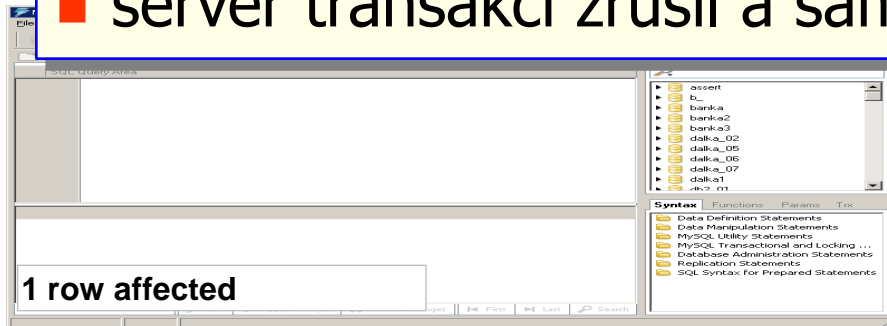
```
Update WinLa Set Mnoz=8;  
BEGIN;  
Select * From WinLa;  
Update WinLa Set Mnoz=Mnoz-2;
```

WilMyLa	
Dil	Mnoz
Lampa	8

```
BEGIN;  
Select * From WinLa;  
Update WinLa Set Mnoz=Mnoz-5;
```

WilMyLa	
Dil	Mnoz
Lampa	8

- u druhé transakce vypršel čas
- server transakci zrušil a sám vydal ROLLBACK



MySQL transakce

```
Update WinLa Set Mnoz=8;  
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-2;
```

Commit;
Select * From WinLa;

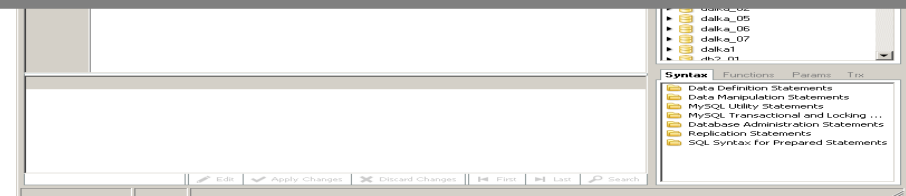
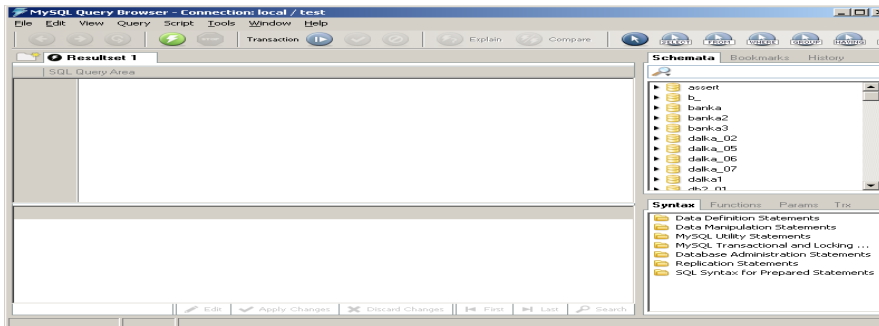
WilMyLa	
Dil	Mnoz
Lampa	6

```
BEGIN;  
  Select * From WinLa;  
  Update WinLa Set Mnoz=Mnoz-5;
```

Commit;
Select * From WinLa;

WilMyLa	
Dil	Mnoz
Lampa	6

- pravá transakce byla zrušena
- server vydal ROLLBACK
- 5 se neodečetlo





dneska máte velkou

Kliku

Navzdory finanční krizi



HVCPRG



První den jako manažer



- co Vás celý den
- tržby !

```
SELECT Current_Time as Cas,  
       Sum (Celkem) as Prodano  
FROM   Trzby  
WHERE  Datum=CURRENT_DATE ;
```

■ jak je to možné ?



CAS	Prodano
8.00	130 000
8.30	130 000
9.00	130 000
....
17.00	130 000

První den jako manažer



- co Vás celý den
- tržby !

```
SELECT Current_Time as Cas,  
       Sum (Celkem) as Prodano  
FROM   Trzby  
WHERE  Datum=CURRENT_DATE ;
```

■ jak je to možné ?

- Vaše transakce je izolována od změn
- je zajištěno opakovatelné čtení
- žádné změny po začátku transakce
 - se neprojeví v součtech
- vyhovuje nám to ??

ne

CAS	Prodano
8.00	130 000
8.30	130 000
9.00	130 000
....
17.00	130 000

První den jako manažer

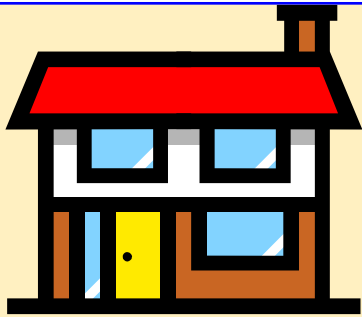


- různé aplikace
 - požadují různou míru izolace
- konkrétní aplikace volí kompromis mezi
 - striktní integrita
 - přijatelná výkonnost

17.00

130 000

Problém lidský, leč pro přednášení nevhodný



1



2



3

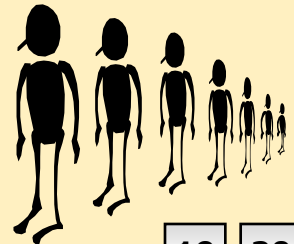
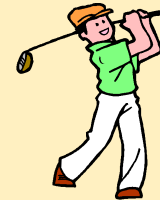


4



5

6

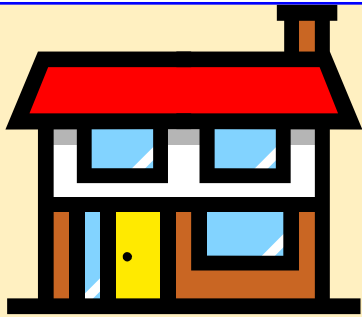


19

20

- operace = maximální izolace
- 1 operace = 15 minut
- 20 účastníků = 5 hodin

Problém lidský, leč pro přednášení nevhodný



6

- maximální izolace souběžných operací
- něco stojí
- je to vždy efektivní - propustné ???
- pro různé situace

- operace volit různou úroveň izolace [propustnosti]
- 1 operace = 15 minut
- 20 účastníků = 5 hodin

20

Úroveň izolace transakcí

určuje

- které druhy konfliktů jsou tolerovány

Úroveň izolace	Ztracená akt.	Dirty read	Neopakovatelné čt	Přelud
READ_UNCOMMITTED	NE	ANO	ANO	ANO
READ_COMMITED	NE	NE	ANO	ANO
REPEATABLE_READ	NE	NE	NE	ANO
SERIALIZABLE	NE	NE	NE	NE

- DB nepřipustí případ ztracené aktualizace
 - **data nelze přepsat dvěma souběžnými transakcemi**
- **SERIALIZABLE** nepřipustí žádný konflikt
 - ale něco to stojí
 - něco se tím musí zabývat a zajistit to

Špinavé čtení

Úroveň izolace

Ztracená akt.

Dirty read

Neopakovatelné čt

Přelud

READ_UNCOMMITTED

NE

ANO

ANO

ANO

- nejnižší úroveň izolace, největší tolerance konfliktů
 - transakce vidí do změn nepotvrzených dat
 - průběh transakce mohou ovlivnit změny načtených dat
 - potvrzené i nepotvrzené
 - lze užít pouze když
 - uživatel toleruje, že data jsou nejasná a nejistá
-
- transakce A mění řádek, B čte změny ještě před potvrzením
 - pokud transakce A změny nepotvrdí (ROLLBACK)
 - transakce B načte řádek, který nikdy neexistoval

Špinavé čtení - transakce vidí do nepotvrzených změn

Úroveň izolace

Ztracená akt.

Dirty read

Neopakovatelné čt

Přelud

READ_UNCOMMITTED

NE

ANO

ANO

ANO



kolik tomu Novákovi přidám

~~5 000~~

to je sakra moc

~~2 000~~

dyť má 3 děti a vyhořel mu barák

~~7 000~~

mám na to - nemám

dám **4 000**

kolikpak dá

hele **5 000**

2 000 - to je škudla

7 000 - to bude Mařka koukat

4 000 - aspoň že tak

Neopakovatelné čtení

Úroveň izolace	Ztracená akt.	Dirty read	Neopakovatelné čt	Přelud
READ_COMITED	NE	NE	ANO	ANO

- úroveň, která postačuje pro většinu běžných transakcí
 - nepotvrzené změny jsou izolovány
 - potvrzené změny jsou viditelné a mohou ovlivnit průběh transakce
-
- transakce A čte řádek
 - transakce B řádek odstraní/změní a potvrdí
 - pokud transakce A čtení zopakuje, dostane jiný výsledek

Opakovatelné čtení

Úroveň izolace

Ztracená akt.

Dirty read

Neopakovatelné čt

Přelud

REPEATABLE_READ

NE

NE

NE

ANO

- druhá nejvyšší izolace, velmi přijatelná
- data, která se potvrzeným způsobem mění
- nejsou viditelná
- může nastat pouze problém přeludu (agregace)

vložení přeludu

- transakce A čte skupinu řádek dle podmínky (WHERE)
- transakce B vloží do skupiny další či změní řádek/y ve skupině
- transakce A zopakuje dotaz dle podmínky (WHERE)
 - dostane jiné výsledky

Serializovaná [úplná] izolace

Úroveň izolace

Ztracená akt.

Dirty read

Neopakovatelné čt

Přelud

SERIALIZABLE

NE

NE

NE

NE

- nejvyšší míra izolace
- transakce je zcela izolována od okolí
- nedojde k žádnému popsanému druhu konfliktu
- schopnost znovu spolehlivě načítat
 - řádky tabulky
 - agregační výsledky
- => **ideální izolace !!!**

Serializovaná [úplná] izolace

Úroveň izolace

Ztracená akt.

Dirty read

Neopakovatelné čt

Přelud

SERIALIZABLE

NE

NE

NE

NE

- **velice nákladná**
- ostatní souběžné transakce mohou trpět
- všechny transakce [dlouhé] s absolutní izolací
- **nic se nedá dělat, všechno se zadře !!!**
- **téměř likvidace konkurenčních přístupů**

Konflikt ztracené aktualizace

- pokud se jakýkoliv program v jakékoliv úrovni izolace
- pokouší přepsat data již přepsaná
 - potvrzená či nepotvrzená
- přepis [update] se nepovede a dojde k odvolání transakce
- **každá transakce musí počítat s tím**
 - **že bude odvolána**



Nastavení úrovně izolace transakcí

SET TRANSACTION ISOLATION LEVEL

{ READ UNCOMMITTED

| READ COMMITTED

| REPEATABLE READ

| SERIALIZABLE }

[.....]

SET TRANSACTION ISOLATION LEVEL

[READ WRITE | READ ONLY]



SET TRANSACTION ISOLATION LEVEL

READ COMMITTED READ ONLY;



Otázka – co je SQL databáze ?

Jednoduchá odpověď

ACID

ACID

- co je to transakce ?
- jednotkna práce v SQL
- několik povelů **[Begin ... Commit]**
- **funkcionalita TRANSAKCE je doleko více**
- **funkcionalita TRANSAKCE je ACID**

ACID

A

Atomicita

C

Concistency Konzistence

I

Izolace

D

Durabilita - trvanlivost