

## 30. Transakce – koncept a vlastnosti transakcí (ACID).

- Poskytují:
  - nezávislý přístup více uživatelům naráz do DB
  - rezistenci DB vůči systémovým poruchám
- Komponenty
  - **souběžné zpracování** (concurrency control) – zabezpečuje, aby každý uživatel viděl jenom konzistentní stavy databáze bez ohledu na to, že s ní pracují i další uživatelé
  - **zotavení z chyb** (error recovery) – zajišťuje, aby stav databáze nebyl narušen v případě systémových poruch
- Databáze je v konzistentním stavu, když splňuje všechny IO definované ve schématu databáze.
- **Transakce je jistá postupnost jedné nebo více SQL operací se kterou se zachází jako s celkem**

### Vlastnosti transakcí – ACID

- **Atomicity** = atomicita – transakce se tváří jako celek, buď se provede celá, nebo se neprovede vůbec
- **Consistency** = konzistence – transakce transformuje DB z jednoho konzistentního stavu do jiného konzistentního stavu
- **Isolation** = izolovanost, nezávislost) – dílčí efekty transakce nejsou viditelné jiným transakcím, jakoby transakce běžela izolovaně
- **Durability** = trvanlivost – efekty úspěšné transakce jsou uloženy do databáze (logování)

## Izolovanost

- musí platit **serializovatelnost** - operace se můžou vykonávat současně, ale výsledek musí stejný jako výsledek některé sekvence postupného vykonávání transakcí.
- zabezpečeno pomocí **uzamykacího** mechanismu
  - transakce by měla být rychle uskutečnitelná (nečekat např. na data od uživatele)
- Když  $U_1 \{T_{11}, T_{12}\}$  a  $U_2 \{T_{21}, T_{22}\}$ , kolik sekvencí postupného vykonávání transakcí (**sériový rozvrh**) existuje?
- U „Různé výsledky (2)“ serializovatelnost garantuje 2 konzistentní výsledky a eliminuje jeden výsledek nerefluktující skutečnost

- **Trvanlivost**
  - Pokud nastane porucha po úspěšné transakci, DBMS garantuje, že změny vykonané transakcí jsou uloženy.
  - zabezpečeno pomocí **logovacího** mechanismu
- **Atomicita**
  - Pokud nastane porucha počas transakce, DBMS garantuje, že se zruší všechny dosavadní změny vykonané transakcí.
  - zabezpečeno pomocí **rollback** mechanismu

$R(A) \subseteq R\{(5), (6)\}$

T1: UPDATE R SET A=A+1;

T2: UPDATE R SET A=2\*A;

Jaké jsou možné výsledky?