

# Reliacinės Duomenų Bazės, ACID garantijos, BASE

2021

**Atomicity** - transakcija įvykdoma nedalimai, “viskas arba nieko”

**Consistency** - visais laiko momentais duomenų būseną vieną, nepažeidžianti jai taikomų apribojimų

**Isolation** - lygiagrečiai vykstančios transakcijos apsaugomos nuo įtakos viena kitai

**Durability** - jei transakcija įvyko, jos efektai išlieka sistemoje

Izolacija

# Izoliacijos lygiai

Izoliacijos Lygis / Anomalija	Dirty Reads	Lost Updates	Non-repeatable Reads	Phantom Reads	Serialization Anomaly (Write Skew)
Read Uncommitted					
Read Committed					
Repeatable Read				**	
Snapshot*					
Serializable					

\* Nėra SQL standarto dalis

\*\* Pavyzdžiui PostgreSQL Repeatable Reads apsaugo ir nuo Phantom Reads fenomeno

# Dirty Reads

Transakcija mato kitos transakcijos pakeitimus iškart.

Mato naują  
reikšmę nors T2  
dar neužbaigta.

T1	T2
SELECT value FROM data WHERE id = 1;	
	UPDATE data SET value = 15 WHERE id = 1;
SELECT value FROM data WHERE id = 1;	
	COMMIT;

# Lost Updates

Dviems transakcijoms atnaujinant tą pačią reikšmę, vienos rezultatas niekada nebus matomas.

T1	T2
<code>SELECT value FROM data WHERE id = 1;</code>	<code>SELECT value FROM data WHERE id = 1;</code>
	<code>UPDATE data SET value = 15 WHERE id = 1;</code>
<code>UPDATE data SET value = 16 WHERE id = 1;</code>	
	<code>COMMIT;</code>
<code>COMMIT;</code>	

Abi transakcijos vienu metu keitė vieną reikšmę, T1 “nugalėjo”.

# Non-Repeatable Reads

Transakcijos vyksmo metu, kitai transakcijai pasibaigus matomi, kitos transakcijos pakeisti duomenys.

Mato naują  
reikšmę.

T1	T2
<code>SELECT value FROM data WHERE id = 1;</code>	
	<code>UPDATE data SET value = 15 WHERE id = 1;</code>
	<code>COMMIT;</code>
<code>SELECT value FROM data WHERE id = 1;</code>	
<code>COMMIT;</code>	

# Phantom Reads

Vienai transakcijai pasibaigus, naujai įterpti elementai matomi dar vykstančiai.

Matomos naujai  
įterptos reikšmės.

T1	T2
SELECT value FROM data;	
	INSERT INTO data VALUES (15, 125);
	COMMIT;
SELECT value FROM data;	
COMMIT;	

Lygiai taip pat ir su ištrintomis reikšmėmis.



# Serialization Anomaly (Write Skew)

T1	T2
SELECT value FROM data WHERE id = 1;	
	SELECT value FROM data WHERE id = 2;
	UPDATE data SET value = v WHERE id = 1;
UPDATE data SET value = v WHERE id = 2;	
	COMMIT;
COMMIT;	

Transakcijų pabaigoje tiek id: 1, tiek id: 2 elementai turi turėti tą pačią reikšmę. Šiuo atveju, reikšmės bus sukeistos.

# Serializacija

Užtikrina, kad lygiagrečiai veikiančių transakcijų rezultatas atitiks kažkuriai transakcijų tvarkai, lyg jos būtų vykdomos nuosekliai.

Praeitame pavyzdyje, serializacija garantuotų, kad reikšmės būtų sukeistos. T.y. vieną iš dviejų:

T1 po to T2

arba

T2 po to T1

# O kitame pasaulyje

**Basic Availability** - svarbiausia pasiekiamumas

**Soft state** - suderinamumas tarp atskirų sistemos mazgų nėra garantuojamas; vieno kliento matoma būsena gali kisti net nesant kitų klientų veiksmas

**Eventual consistency** - duomenys pasiekia suderinamą būseną ateityje

# Galutinis vientisumas (*angl. Eventual consistency*)

Išskirstytai sistemai, nustojus priimti įrašus, jos mazgai galų gale pasieks vieną bendrą būseną.

Sistemos, kurios pasižymi šią savybe:

- negarantuoja, kad tuo pat metu atiekamos identiškos užklauskos pateiks tą patį atsakymą
- bet gali tęsti darbą, net kai dalis sistemos yra nepasiekiamą
- pasižymi didesniu našumu (mažesne uždelsa klientui), nei stipresnio vientisumo sistemos