

*Autor:*

*Mia Knežević RA242-2017*

## Konkurentni pristup resursima u bazi

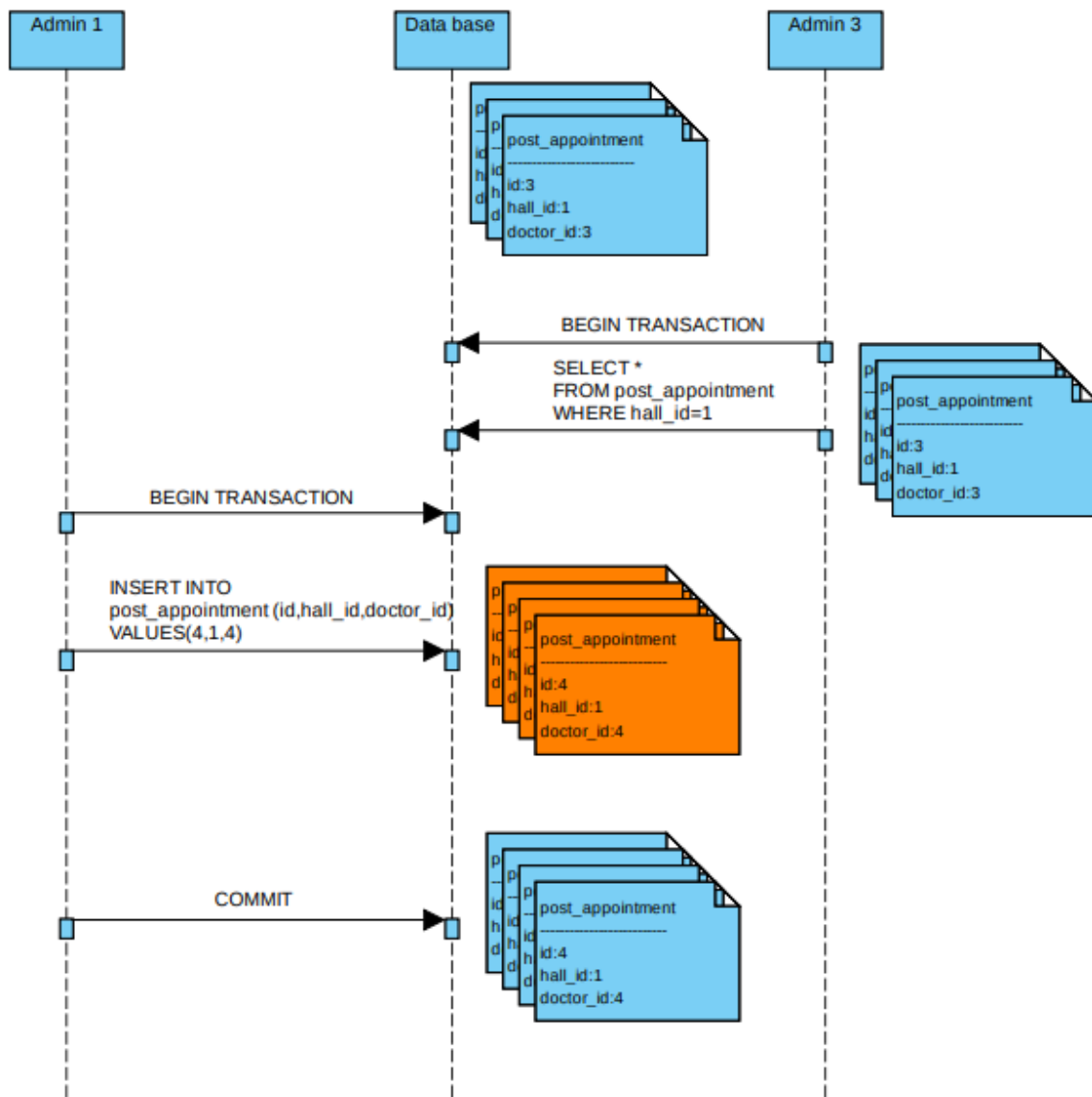
*Rešavanje konfliktnih situacija- tačke 3.20, 3.21*

### **Tačka 3.20 (Postupak rezervisanja sala za operacije):**

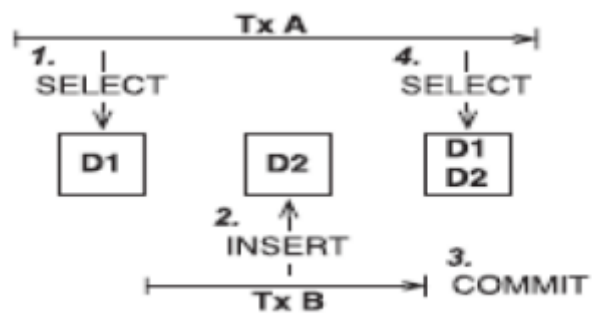
Prilikom rezervisanja sala za operacije, može doći do konfliktne situacije (Dijagram 1.1) ukoliko više admina klinike u istom trenutku pristupi dodeli istih. U tom momentu, admin koji prvi izvrši izmenu u bazi podataka stvara problem drugom adminu, koji u tom trenutku bira slobodan termin u sali kao i lekara koji će izvršiti zahvat. Slobodni termini za zakazivanje novih operacija zavise od toga koliko ima zakazanih pregleda koji zauzimaju odredjenu salu i zauzimaju slobodne termine doktora. Ukoliko dođe do zakazivanja novog pregleda ili operacije dok prvi admin još uvek zakazuje prvi pregled, sužava se opseg slobodnih termina za zakazivanje, a admin o tome nije obavešten. U tom slučaju postoji verovatnoća da dođe do konflikta i zakazivanja istih termina zbog neverodostojne slike o trenutnom stanju u bazi podataka. Verodostojnu sliku o trenutnoj količini zakazanih pregleda i operacija i rezervisanim terminima, admin može imati samo ako ponovo pokrene proces zakazivanja.

Ovakav problem sa pristupanjem bazi podataka spada u **PHANTOM READ** grupu problema. Jedna transakcija tek prilikom drugog čitanja dobija verodostojnu sliku o stanju unutar baze, usled izmena koje je u međuvremenu uvela druga transakcija. Eliminisanje ovog konflikta se radi uvođenjem **SERIALIZABLE** nivoa izolacije, koji sprečava da transakcije čitaju modifikovane podatke, kao i commit-ovanje izmena od strane drugih transakcija paralelno, sve dok se prva transakcija u potpunosti ne izvrši. Sprečen je unos novih podataka u bazu, kao i njihova izmena.

Na dijagramu 1.2 prikazano je paralelno dešavanje dve transakcije TxA i TxB. Na samom početku TxA ima uvid u jedne podatke koje TxB u međuvremenu menja, dodaje nove. Priikom sledećeg pristupa istim podacima, TxA ima uvid u ažurirane podatke.



Dijagram 1.1



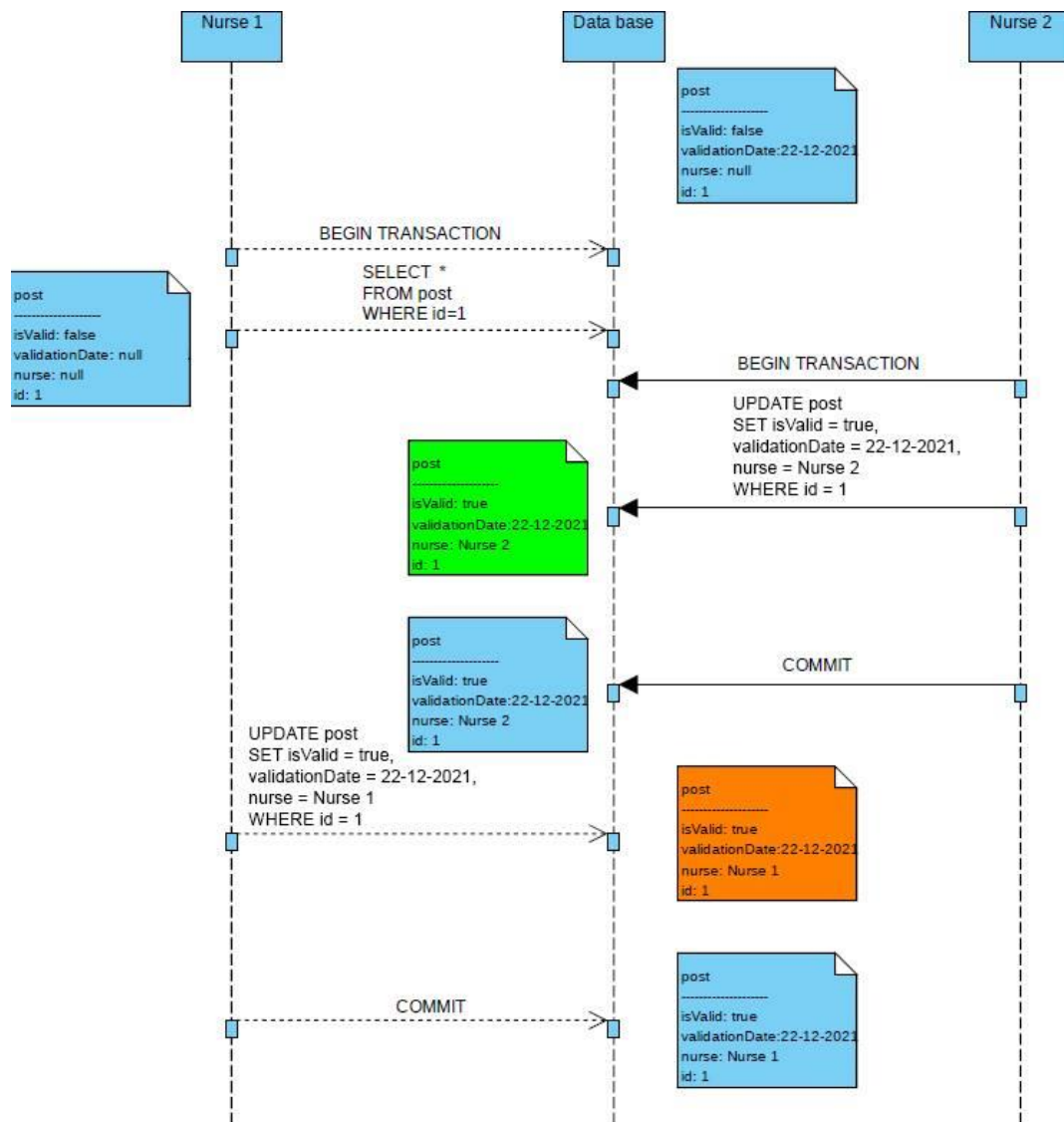
Dijagram 1.2

### Tačka 3.21 (Postupak overe recepta):

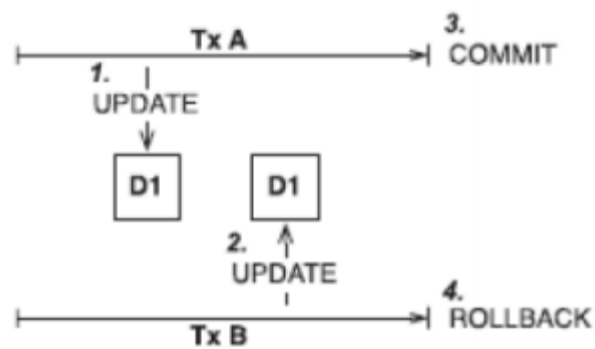
Prilikom overe bilo kog recepta koji je prepisan od starane doktora (Dijagram 2.1), iz kompletne liste koja je dostupna svim medicinskim sestrama, može doći do konflikta ukoliko dve medicinske sestre overe isti recept neposredno jedna nakon druge. Prilikom procesa overe, nakon klika na dugme „overi“, prva sestra definiše recept kao overen čime se u isti upisuje njeno ime, datum overe i potvrda da recept više nije neoveren, da bi nakon iste akcije od strane druge sestre, svi ovi podaci bili redefinisani, čime se gubi prvobitna validna informacija o kojoj strana koja je prva validno overila recept nema nikakvu informaciju.

Gore opisani problem - **LOST UPDATE**, nastaje kada dve različite transakcije menjaju isti podatak bez zaključavanja, što posledično izaziva nesvesan gubitak podataka. Međutim, odabirom nivoa izolacije transakcija za svaku konekciju, ovaj problem može da se eliminiše. Za eliminisanje Lost Update problema koristi se **READ\_UNCOMMITTED** nivo izolacije. Obezbeđeni su mehanizmi zaključavanja koji sprečavaju da dođe do konflikta ove prirode. Uvažava se isključivo transakcija koja je prva okupirala resurs, dok druga, paralelna transakcija biva povučena i svaka naknadna izmena se poništava.

Ovaj opisani način zaštite podataka je prikazan na dijagramu 2.2, gde su TxA i TxB dve različite transakcije (u našem slučaju započete od strane 2 medicinske sestre) koje pokušavaju istovremeno da pristupe istom resursu D1, u našem slučaju neoverenom receptu. TxA vrši prva izmenu, stoga se ona uvažava, a izmene od strane TxB se poništavaju. Time su podaci konzistentni i validni, a anomalije u bazi podataka sprečene.



Dijagram 2.1



Dijagram 2.2