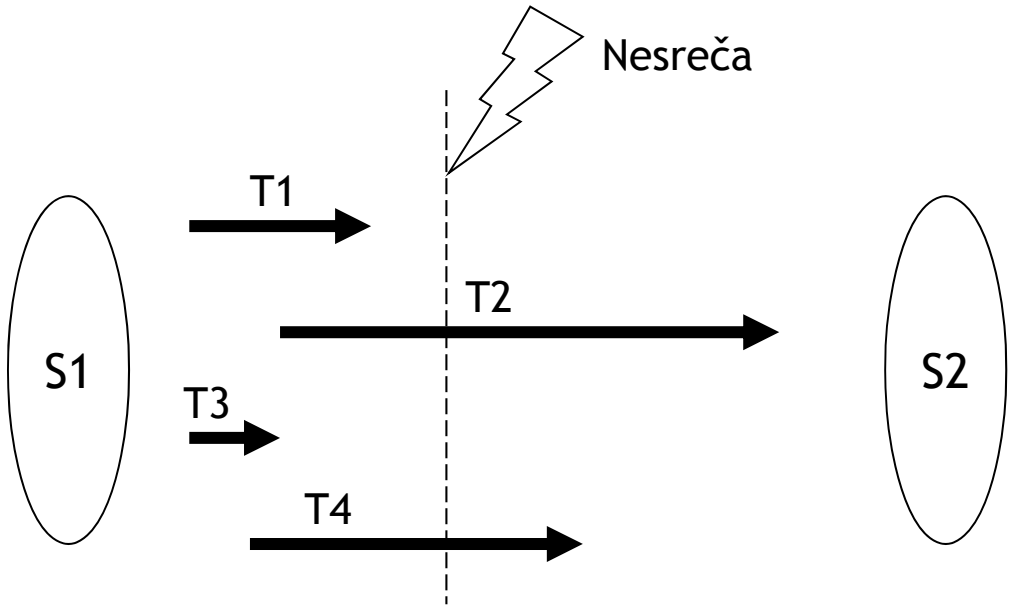
Podatkovna varnost

Podatkovna varnost

- Potreba po obnovljivosti
- Transakcije in obnovljivost
- Komponente SUPB za obvladovanje obnovljivosti
- Tehnike obnovljivosti

Kaj je obnova podatkov po nesreči?

- Proces vzpostavljanja podatkovne baze v zadnje veljavno stanje, ki je veljalo pred nastopom nesreče.



Potreba po obnovljivosti...

- Shranjevanje podatkov se običajno navezuje na štiri različne tipe medijev za shranjevanje podatkov, z naraščajočo stopnjo zanesljivosti:
  - glavni pomnilnik (neobstojni pomnilnik): podatki v njem ne preživijo sistemskih nesreč,
  - magnetni disk ("online" obstojni pomnilnik): zanesljivejši in cenejši od glavnega pomnilnika, vendar tudi počasnejši,
  - magnetni trak ("offline" obstojni pomnilnik): še zanesljivejši in cenejši od diska, vendar tudi počasnejši, omogoča samo zaporedni dostop,
  - optični disk: najzanesljivejši od vseh, še cenejši od traku, hitrejši od traku, omogoča neposredni dostop do podatkov.

Potreba po obnovljivosti…

- Obstaja več vrst nesreč, od katerih je potrebno vsako obravnavati na drugačen način.

- Nesreča lahko prizadane podatke tako v glavnem, kot v sekundarnem pomnilniku.

Potreba po obnovljivosti...

- Vzroki za nesreče:
  - odpoved sistema: zaradi napak v strojni ali programski opremi; posledica je izguba podatkov v glavnem pomnilniku,
  - poškodbe medija: zaradi trka glave diska ob magnetno površino postane medij neberljiv; posledica so neberljivi deli sekundarnega pomnilnika,
  - programska napaka v aplikaciji: zaradi logične napake v programu, ki dostopa do podatkov v PB, pride do napak v eni ali več transakcijah,
  - neprevidnost: zaradi nenamernega uničenja podatkov s strani administratorjev ali uporabnikov,
  - sabotaža (namerno oviranje dela): zaradi namernega popačenja ali uničenja podatkov, uničenja programske ali strojne opreme.

Potreba po obnovljivosti

- Ne glede na vrsto napake, vedno smo pri nesrečah soočeni z dvema bistvenima problemoma:
  - izguba podatkov v glavnem pomnilniku (vključno s podatki v medpomnilniku),
  - izguba podatkov na sekundarnem pomnilniku.

- V nadaljevanju:
  - pregled tehnik za lajšanje posledic nesreče in
  - tehnike za obnavljanje po nesreči.

Transakcije in obnovljivost…

- Transakcija predstavlja osnovno enoto obnovljivosti.

- Za obnovljivost skrbi upravljavec za obnovljivost (recovery manager), ki mora v primeru nesreče zagotavljati dve od štirih lastnosti transakcij (ACID):
  - atomarnost in
  - trajnost.

Transakcije in obnovljivost…

- Naloga upravitelja obnovljivosti je, da pri obnovitvi PB po nesreči zagotovi:
  - da se vse spremembe, ki so bile v PB izvedene v okviru posamične transakcije uveljavijo v celoti ali pa
  - da se ne uveljavi nobena sprememba.

- Problem je kompleksen, ker pisanje v PB ne predstavlja atomarne akcije → transakcija lahko izvede COMMIT (uveljavitev sprememb), vendar se spremembe v PB ne zabeležijo, ker enostavno ne "dosežejo" PB (nastop nesreče).

Transakcije in obnovljivost…

- Podatkovni vmesniki so področje v glavnem pomnilniku, v katerega se pri prenašanju podatkov iz/v sekundarnega pomnilnika podatki pišejo ali iz njega berejo.
- Prenos vsebine podatkovnih vmesnikov v sekundarni pomnilnik (trajne spremembe) se sproži samo v primeru izvedbe posebnih ukazov:
  - COMMIT ali
  - avtomatično, ko postanejo podatkovni vmesniki polno zasedeni (eksplicitno zapisovanje vsebine podatkovnih vmesnikov v sekundarni pomnilnik označujemo kot prisilno zapisovanje (force-writing)).

Transakcije in obnovljivost…

- Če se nesreča pripeti med pisanjem v pod. vmesnike ali med prenosom podatkov iz pod. vmesnikov v sek. pomnilnik, mora upravitelj za obnovljivost ugotoviti status transakcije, ki je izvajala pisanje v času nesreče:
  - če je transakcija izvedla ukaz COMMIT, mora upravitelj za obnovljivost zaradi zagotavljanja lastnosti trajnosti zagotoviti ponovno izvajanje transakcije (Roll-forward ali Redo),
  - če transakcija ni izvedla ukaza COMMIT, mora upravitelj za obnovljivost zaradi zagotavljanja lastnosti atomarnosti izvesti razveljavljanje posodobitev, ki jih je do tedaj transakcija izvedla (Rollback ali Undo).

Transakcije in obnovljivost...

- Če je potrebno razveljaviti samo eno transakcijo govorimo o parcialnem razveljavljanju (partial undo). Ta se izvaja tudi pri sočasnem dostopanju do podatkov zaradi uporabe protokolov za nadzor sočasnosti.

- Če je potrebno razveljaviti vse v času nesreče aktivne transakcije, govorimo o globalni razveljavitvi (global undo).

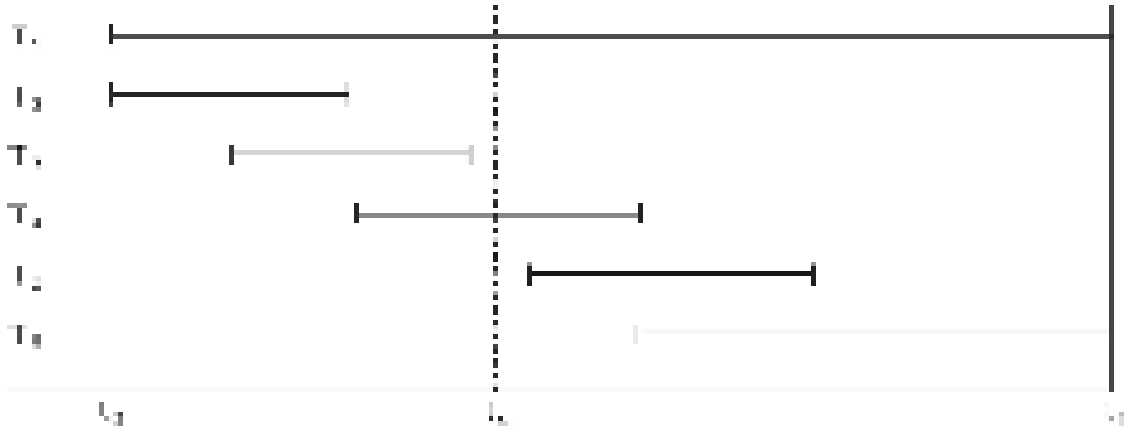
Kontrolne točke

- Kontrolna točka: točka sinhronizacije med PB in diskom (tudi kar se tiče dnevnika). Izvede se zahteva po izpisu vseh podatkovnih vmesnikov na disk.

  – Tako smo prepričani, da so bile transakcije, ki so bile zaključene pred izpisom vmesnikov, zanesljivo uveljavljene ali razveljavljene v PB na disku.

PRIMER: uporaba UNDO/REDO

- Transakcije T_1 do T_6 se izvajajo sočasno, SUPB začne delovati ob t_0, t_c je kontrolna točka, nesreča pa nastopi ob t_f:



- T_2 in T_3 izvedeta COMMIT in spremembe se uveljavijo v PB.

PRIMER: uporaba UNDO/REDO

- T_1 in T_6 ne izvedeta ukaza COMMIT do trenutka nesreče, zato jih upravitelj za obnavljanje pri ponovnem zagonu razveljavi (UNDO).
- Za T_4 in T_5 ni jasno, do katere mere so se njune spremembe uveljavile v PB - ali je bila vsebina podatkovnih vmesnikov zapisana v sekundarni pomnilnik ali ne.
  - Ker nimamo na razpolago nobene dodatne informacije o stanju transakcij, je upravitelj za obnavljanje prisiljen ponoviti (REDO) transakcije T_2, T_3, T_4 in T_5.

Komponente SUPB za obnovljivost

- V okviru SUPB so za obnavljanje podatkov po nesreči na voljo naslednje komponente:
  - mehanizem za izdelavo varnostnih kopij, ki periodično kreira kopije PB,
  - dnevnik, ki hrani podatke o trenutnem stanju transakcij in spremembah v PB,
  - mehanizem za izvajanje kontrolnih točk, ki omogoča da se posodobitve, ki jih izvajajo transakcije v PB, ohranijo (zahteva po izpisu vseh datotečnih vmesnikov na disk),
  - upravljalec za obnovljivost, komponenta SUPB, ki omogoča obnoviti podatkovno bazo v zadnje konsistentno stanje, ki je veljalo pred nastopom nesreče.

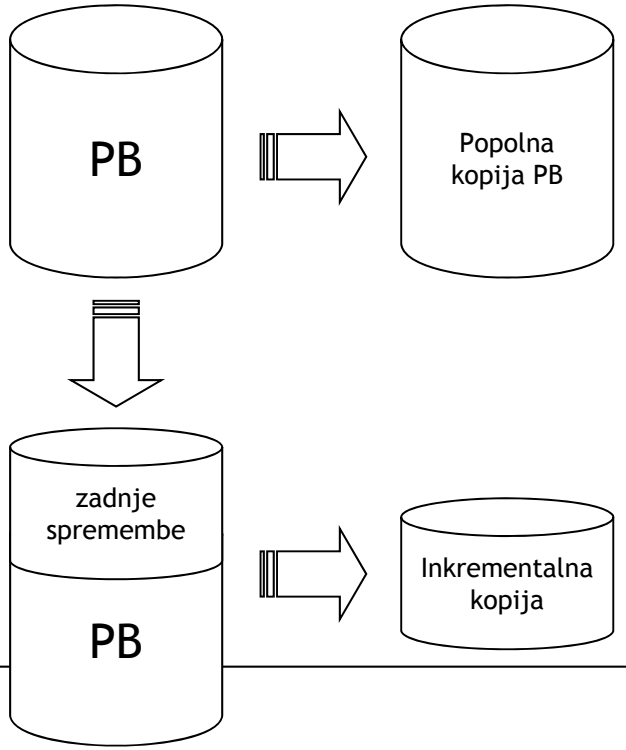
Mehanizem za izdelavo varnostnih kopij...

- Mehanizem mora omogočati izdelavo varnostnih kopij PB in dnevnika v določenih intervalih, ne da bi pred tem bilo potrebno prekiniti delovanje PB .

- Kopijo PB se uporabi v primeru poškodb PB ali njenega uničenja.
- Varnostna kopija se običajno hrani na magnetnem traku.

Mehanizem za izdelavo varnostnih kopij

- Varnostna kopija je lahko:
  - popolna kopija PB ali
  - inkrementalna kopija, ki vsebuje samo spremembe izvedene od zadnje popolne ali inkrementalne kopije PB.



Dnevnik...

- V dnevnik se zapisujejo vse spremembe, ki jih transakcije izvedejo v PB.
- Dnevnik lahko vsebuje naslednje podatke:
  - transakcijske zapise, kjer je dnevniški zapis sestavljen iz:
    - identifikatorja transakcije,
    - tipa dnevniškega vpisa (začetek tr., insert, update, detele, abort, commit),
    - identifikator podatka, na katerega se nanaša operacija (operacije: insert, delete, update) v okviru transakcije,
    - predhodna vrednost podatka: vrednost podatka pred ažuriranjem (samo za operacije update in delete),
    - vrednost podatka po ažuriranju (samo za operacije insert in update),
    - podatki potrebni za upravljanje dnevnika: kazalec na prejšnji in naslednji dnevniški zapis, ki pripada določeni transakciji.
  - zapise kontrolnih točk.

Dnevnik...

- Primer segmenta dnevniške datoteke, ki prikazuje tri sočasne transakcije T_1, T_2 in T_3. Stolpca pPtr in nPtr predstavljata kazalce na predhodni in naslednji dnevniški vpis transakcije.

Tid	Time	Operation	Object	Before image	After image	pPtr	nPtr
T1	0612	START				0	2
T1	0613	UPDATE	STAFF SL21	(old value)	(new value)	1	5
T3	0614	START				0	4
T3	0616	INSERT	STAFF SL27		(new value)	3	7
T2	0617	DELETE	STAFF SL5	(old value)		0	8
T2	0617	UPDATE	DISTRICT PG16	(old value)	(new value)	0	9
T3	0618	START				0	11
T1	0618	COMMIT				2	12
	0619	CHECKPOINT	T2, T3				
T2	0619	COMMIT				6	13
T3	0620	INSERT	PHONE LIST PG21		(new value)	7	15
T3	0621	COMMIT				11	14

Tehnike obnovljivosti...

- Uporaba posamezne procedure za obnavljanje podatkov v PB po nesreči je odvisna od obsega nastale škode. Razlikujemo dva primera:
- Obsežne poškodbe PB:
  - vzrok: npr. diskovna nesreča.
  - posledica nesreče: uničena podatkovna baza.
  - podatke se obnovi z uporabo kopije PB in dnevnika; podatki iz dnevnika služijo za ponovitev (redo) uveljavljenih transakcij.
  - ta način obnavljanja predvideva, da dnevnik ni bil poškodovan; dnevnik naj se torej nahaja na disku, ki je ločen od podatkovnih datotek.

Tehnike obnovljivosti...

- Manjše poškodbe; PB ni fizično poškodovana:
  - vzrok: odpoved sistema med izvajanjem transakcij.
  - posledica nesreče: PB preide v neveljavno – nekonsistentno stanje.
  - transakcije, ki so se prekinile je potrebno razveljaviti, ker so postavile PB v nekonsistentno stanje.
  - lahko se tudi zgodi, da je nekatere transakcije potrebno ponoviti, če njihove spremembe niso "dosegle" sekundarnega pomnilnika.
  - v tem primeru za obnavljanje ne potrebujemo kopije PB, ampak zadostujejo predhodne in posodobljene vrednosti podatkov, ki se nahajajo v dnevniških vpisih (glej primer izseka iz dnevnika).

Tehnike obnovljivosti...

- Tehnike obnovljivosti podatkov po nesrečah, ki privedejo PB v nekonsistentno stanje:
  - odloženo ažuriranje,
  - sprotno ažuriranje.
- Odloženo in sprotno ažuriranje se ločita po načinu zapisovanja posodobljenih podatkov v PB, obe pa uporabljata dnevnik.
- Obnovitvene tehnike morajo biti za uporabnika transparentne!

Odloženo ažuriranje...

- Pri protokolu za odloženo ažuriranje se podatki (posodobljeni) ne zapisujejo neposredno v PB.
- Vsa ažuriranja v okviru transakcije se najprej shranijo v dnevnik. Pri uspešnem zaključku transakcije se izvede dejansko ažuriranje PB.
- V primeru nesreče:
  - če se transakcija prekine, v PB ni potrebno razveljaviti nobene spremembe, ker se te nahajajo samo v dnevniku,
  - pred nesrečo uspešno zaključene transakcije je potrebno ponoviti (redo), ker se njihova ažuriranja lahko še niso dejansko zapisala v PB. V tem primeru se uporabi zapise shranjene v dnevniku.

Sprotno ažuriranje...

- Pri uporabi protokola sprotnega ažuriranja transakcija izvaja neposredno spreminjanje podatkov v PB še preden se uspešno zaključi.
- V primeru nesreče je poleg ponavljanja (redo) uspešno zaključenih transakcij, potrebno razveljaviti (rollback) vse transakcije, ki so bile aktivne v času nesreče.

Primerjava odloženega in sprotnega ažuriranja

- Z vidika učinkovitosti:
  - Odloženo ažuriranje je učinkovitejše, če se v povprečju izvede več neuspešnih transakcij (ni potrebno spreminjati PB).
  - Sprotno ažuriranje je učinkovitejše, če se v povprečju izvede več uspešnih transakcij (ni potrebno veliko popravljati podatkov v PB).