

SKLADIŠTA PODATAKA - Primer: BANKA - Punjenje

Operativna baza podataka

GRAD (IDGra, Naziv)
FILIJALA (IDFil, Naziv, IDGra)
KOMITENT (IDKom, Naziv, IDGra)
RACUN (IDRac, Naziv, TotalUplata, TotalIsplata, IDKom, IDFil)
PROMET (IDPro, DatumVreme, Uplata, Isplata, IDRac)
PROMET_INTERNI (IDPro, IDRacKa)

Baza podataka skladišta

PROMET_INTERNI (IDVre, IDKomOd, IDKomKa, IDGraOd, IDGraKa, Iznos)
PROMET_EKSTERNI (IDVre, IDKom, IDGra, IznosUplata, IznosIsplata)
KOMITENT (IDKom, Naziv, IDPodDel)
PODDELATNOST (IDPodDel, Naziv, IDDel)
DELATNOST (IDDel, Naziv)
GRAD (IDGra, Naziv, IDReg)
REGIJA (IDReg, Naziv)
VREME (IDVre, DatumSat)
VREME_DAN (IDDan, Datum)
STANJE (IDDan, IDKom, IDGra, Iznos)

Zadatak:

- a Dopuniti operativnu bazu podataka za potrebe inkrementalnog paketnog punjenja.
- b Razrešiti paketno inkrementalno punjenje skladišta podataka.

Dopuna operativne baze podataka - opšti postupak

1.

U operativnoj bazi podataka treba razvrstati tabele na tabele fakata i tabele prometa.

2.

U tabelama fakata i prometa treba da postoje direktne ili indirektne reference na sve osnovne dimenzije od interesa, pa u tom smislu treba eventualno uspostaviti veze tabela fakata sa osnovnim dimenzijama. To se može ostvariti na više načina:

- a uvođenjem paralelne tabele naziva *Tabela_DIMENZIJA* koja sadrži ID-ove iz Tabela i ID-ove dimenzija;
- b dopunom Tabela sa ID-ovima dimenzija.

Pri tome radi referencijalnog integriteta treba uvesti i tabele osnovnih dimenzija od kojih svaka ima za naziv naziv dimenzije a kao attribute ID dimenzije i eventualno njen naziv.

Eventualne hijerarhije dimenzija se razrešavaju unutar skladišta podataka.

3.

Treba obezbediti mehanizam prepoznavanja izmena koje su nastale u operativnoj bazi podataka od trenutka poslednjeg inkrementalnog punjenja skladišta podataka. To se može ostvariti na više načina:

- a uvođenjem paralelne tabele naziva *Tabela_NOVO* koja je iste strukture kao i Tabela;
- b uvođenjem paralelne tabele naziva *Tabela_NOVO* koja sadrži samo ID-ove iz Tabela;
- c dopunom Tabela sa atributom DatumKreiranja (paketni interval nije kraći od dana).

U varijantama a i b treba obezbediti i okidače koje će za INSERT u Tabela vršiti INSERT u *Tabela_NOVO*.

U varijantama a i b, sve tabele *Tabela_NOVO* prazne se nakon svakog paketnog inkrementalnog punjenja skladišta podataka

Dopuna operativne baze podataka - rešenje

U konkretnom primeru, razvrstavanje tabela po koraku 1 daje:

- a tabele fakata: GRAD, KOMITENT, FILIJALA, RACUN;
- b tabele prometa: PROMET, PROMET_INTERNI.

Primenom varijante 2a i varijante 3b dobija se (dopune su navedene crveno i iskošeno):

GRAD (IDGra,Naziv)

FILIJALA (IDFil,Naziv,IDGra)

KOMITENT (IDKom,Naziv,IDGra)

RACUN (IDRac,Naziv>TotalUplata>TotalIsplata,IDKom,IDFil)

PROMET (IDPro,DatumVreme,Uplata,Isplata,IDRac)

PROMET_INTERNI (IDPro,IDRacKa)

KOMITENT_PODELATNOST (IDKom,IDPodDel)

PODELATNOST (IDPodDel,Naziv)

KOMITENT_NOVO (IDKom)

PODELATNOST_NOVO (IDPodDel)

Primenom varijante 2b i varijante 3a dobija se

GRAD (IDGra,Naziv)
FILIJALA (IDFil,Naziv,IDGra)
KOMITENT (IDKom,Naziv,IDGra,*IDPodDel*)
RACUN (IDRac,Naziv>TotalUplata>TotalIsplata,IDKom,IDFil)
PROMET(IDPro,DatumVreme,Uplata,Isplata,IDRac)
PROMET_INTERNI (IDPro,IDRacKa)
PODDELATNOST (IDPodDel,Naziv)
KOMITENT_NOVO (IDKom,Naziv,IDGra,IDPodDel)
PODDELATNOST_NOVO (IDPodDel,Naziv)

Primenom varijante 2b i varijante 3c dobija se

GRAD (IDGra,Naziv)
FILIJALA (IDFil,Naziv,IDGra)
KOMITENT (IDKom,Naziv,IDGra,*IDPodDel*,DatumKreiranja)
RACUN (IDRac,Naziv>TotalUplata>TotalIsplata,IDKom,IDFil)
PROMET(IDPro,DatumVreme,Uplata,Isplata,IDRac)
PROMET_INTERNI (IDPro,IDRacKa)
PODDELATNOST (IDPodDel,Naziv,DatumKreiranja)

U ovom primeru za punjenje će se kao ilustrativno koristiti kombinovano rešenje:

GRAD (IDGra,Naziv)
FILIJALA (IDFil,Naziv,IDGra)
KOMITENT (IDKom,Naziv,IDGra,*IDPodDel*,DatumKreiranja)
RACUN (IDRac,Naziv>TotalUplata>TotalIsplata,IDKom,IDFil)
PROMET(IDPro,DatumVreme,Uplata,Isplata,IDRac)
PROMET_INTERNI (IDPro,IDRacKa)
PODDELATNOST (IDPodDel,Naziv)
PODDELATNOST_NOVO (IDPodDel,Naziv)

Paketno inkrementalno punjenje - Rešenje

Paketno inkrementalno punjenje je na kraju svakog dana.

Na početku perioda između dva punjenja tabele `PODDELATNOST_NOVO` je prazna. Održavanje te dodatne tabele u operativnoj bazi podataka se može razrešiti na dva načina:

- Okidačem koji reaguje na INSERT u odgovarajuću matičnu tabelu. Ovo se radi Ineprekidno tokom perioda između dva punjenja, primer

```
Okidac      NovaPodDelatnost:
Vreme:      AFTER
Promena:    INSERT U PODDELATNOST
Opseg:      ROW
Koristi:    NEW ROW N
Uslov:      -
Akcija:     N.IDPodDel,N.Naziv
            -----
            -----
            N.IDPodDel >
            N.Naziv >
            INSERT PODDELATNOST_NOVO
```

- upitom koji u tabelu `PODDELATNOST_NOVO` ubacuje one redove iz tabele `PODDELATNOST` koji ne postoje u odgovarajućoj tabeli skladišta podataka.

Nakon što je tabela `PODDELATNOST_NOVO` popunjena novim sadržajem, vrši se ažuriranje sadržaja tabele dimenzija u skladištu podataka:

```
INSERT INTO DW.PODDELATNOST ( IDPodDel,Naziv,NULL )
SELECT IDPodDel,Naziv
FROM   PODDELATNOST_NOVO

INSERT INTO DW.KOMITENT ( IDKom,Naziv,IDPodDel )
SELECT IDKom,Naziv,IDPodDel
FROM   KOMITENT
WHERE  DatumKreiranja > DatmPoslednjegPunjenja
```

Kao finalni korak ažuriranja, manuelno se ,preko odgovarajućih UPDATE naredbi, vrši u tabeli `DW.PODDELATNOST` zamena NULL vrenosti IDDel sa odgovarajućim vrednostima.

Sa ovim je završeno punjenje tabele dimenzija u skladištu podataka.

Nakon toga, vrši se punjenje tabela fakata, pri čemu se kao izvorište koriste osnovne (bez dopunskih) tabele operativne baze podataka. Konkretno:

Tabela fakata PROMET_INTERNI:

```
FOR EACH Red IN PROMET
  IF DATE ( Red.DatumVreme ) > DatumPrethodnogPunjenja
    IF Red.IDPro EXISTS IN PROMET_INTERNI
      Red.DatumVreme -f1> vIDVre
      Red.DatumVreme -f2> vDatumSat
      Red.IDPro > PROMET_INTERNI > vIDRacKa
      vIDRacKa > RACUN > vIDKomKa
      vIDKomKa > KOMITENT > vIDGraKa
      Red.IDRac > RACUN > vIDKom
      vIDKom > KOMITENT > vIDGra
      INSERT INTO DW.VREME
        VALUES (vIDVre,vDatumSat)
      INSERT INTO DW.PROMET_INTERNI
        VALUES (vIDVre,vIDKom,vIDKomKa,vIDGra,vIDGraKa,Red.Uplata)
    ENDIF
  ENDIF
ENDFOR
```

Tabela fakata PROMET_EKSTERNI:

```
FOR EACH Red IN PROMET
  IF DATE ( Red.DatumVreme ) > DatumPrethodnogPunjenja
    IF Red.IDPro NOT EXISTS IN PROMET_INTERNI
      Red.DatumVreme -f1> vIDVre
      Red.DatumVreme -f2> vDatumSat
      Red.IDRac > RACUN > vIDKom
      vIDKom > KOMITENT > vIDGra
      INSERT INTO DW.VREME
        VALUES (vIDVre,vDatumSat)
      INSERT INTO DW.PROMET_EKSTERNI
        VALUES (vIDVre,vIDKom,vIDGra,Red.Uplata,Red.Isplata)
    ENDIF
  ENDIF
ENDFOR
```

Tabela fakata STANJE:

```
CURRENT_DATE -f2> vIDDan
INSERT INTO DW.VREME_DAN
  VALUES (vIDVre,CURRENT_DATE)

FOR EACH Red IN RACUN
  Red.IDKom > KOMITENT > vIDGra
  INSERT INTO DW.STANJE
    VALUES (vIDDan,Red.IDKom,vIDGra,Red.Uplata-Red.Isplata)
ENDFOR
```

Na kraju se prazni tabela PODDELATNOST_NOVO i ažurira tabela skladišta koja sadrži datumvreme prethodnog punjenja.