

REDUKCIJA ŠEME RBP

Suština ovog vida restrukturiranja, koje se sprovodi kao prvo, jeste u smanjenju broja šema relacija u RBP u cilju smanjenja broja i složenosti operacija, i to:

- ✓ Smanjenje/uklanjanje operacija spajanja.

Redukcija se sprovodi tako što se od više šema relacija formira jedna primenom određenih pravila. Pri tome mora biti ispoštovan zahtev reverzibilnosti koji se sastoji u tome da se pogledima nad redukovanom šemom RBP uvek može rekonstruisati polazna šema. U suprotnom, postojao bi gubitak podataka.

Cena koja se plaća pri redukciji jeste povećanje potrebnog memorijskog prostora pošto će u redukovanim šemama relacija postojati NULL vrednosti i dodatna indikatorska svojstva.

Postoje tri pravila za redukciju šeme RBP:

- ✓ Redukcija veze sa kardinalnošću učešća (0,1);
- ✓ Redukcija odnosa specijalizacije;
- ✓ Redukcija odnosa zavisnosti sa kardinalnošću uslovljavanja (0,1)

Redukcija odnosa veze sa kardinalnošću učešća (0,1)

Suština ovog pravila je da se za veze kod kojih bar jedan učesnik ima kardinalnost učešća (0,1) pri formiranju šema relacija primeni pravilo koje važi za kardinalnost učešća (1,1), kada ne nastaje posebna šema relacije veze. Ako su A i B objekti-učesnici, V veza a <Ox> neki ostali atributi, iz polazne situacije

A (IDA, <OA>) V (IDA, IDB, <OV>) B (IDB, <OB>)

dobija se

Am (IDA, <OA>, IDB, <OV>) B (IDB, <OB>)

Reverzibilnost je očuvana, pošto ažurabilni pogledi

```
CREATE VIEW A ( IDA, <OA> )
AS SELECT IDA, <OA>
FROM Am;      CREATE VIEW V ( IDA, IDB, <OV> )
AS SELECT IDA, IDB, <OV>
FROM Am
WHERE IDB IS NOT NULL;
```

rekonstruišu ranije šeme relacija A i V.

U praksi se za dopunjenu šemu relacije zadržava raniji naziv (A umesto Am).

U slučaju veze stepena većeg od dva i sa više učesnika sa kardinalnošću učešća (0,1), za šemu relacije koja će se dopuniti podacima veze bira se ona za koju se očekuje da bi kod izveštavanja bila najčešće spajana sa šemom relacije veze.

Redukcija odnosa specijalizacije

Sušтина ovog pravila je da se od svih šema relacija koje čine jedan odnos specijalizacije formira jedna šema relacije tako što se šema relacije generalnog entiteta dopunjuje svojstvima svih šema relacija specijalnih entiteta i dodatnim selektorskim svojstvima. Postoje dva slučaja.

Ekskluzivna specijalizacija

Kod ekskluzivne specijalizacije svaka generalna instanca može biti najviše jedna specijalna instanca (ni jedna ili jedna kod parcijalne i tačno jedna kod totalne specijalizacije). Za slučaj specijalizacije stepena N se umesto N+1 šema relacija

```
A ( IDA, <OA> )      A1 ( IDA, <OA1> )      ...      AN ( IDA, <OAN> )
```

dobija

```
Am ( IDA, <OA>, Ind, <OA1>, ..., <OAN> )
```

Ind je indikatorsko svojstvo koje kod totalne specijalizacije uzima jednu od N vrednosti val(Ind)_i, a kod parcijalne specijalizacije uz to i vrednost NULL za one generalne instance koje ne specijaliziraju.

Reverzibilnost je očuvana pomoću sledećih ažurabilnih pogleda koji rekonstruišu šemu relacije generalnog slučaja i šemu relacije bilo kog specijalnog slučaja i

```
CREATE VIEW A ( IDA, <OA> )      CREATE VIEW Ai ( IDA, <OAi> )  
AS SELECT IDA, <OA>  
FROM Am;  
AS SELECT IDA, <OAi>  
FROM Am  
WHERE Ind=val (Ind) i ;
```

Svojstvo Ind je neophodno kao univerzalno rešenje, pošto u opštem slučaju osnov specijalizacije može biti učešće u nekom odnosu a da pri tome ne postoje specijalna svojstva.

Inkluzivna specijalizacija

Kod inkluzivne specijalizacije stepena N svaka generalna instanca može biti najviše N specijalnih instanca (ni jedna ili do N kod parcijalne i jedna ili do N kod totalne specijalizacije). Redukcijom se umesto N+1 šema relacija

```
A ( IDA, <OA> )      A1 ( IDA, <OA1> )      ...      AN ( IDA, <OAN> )
```

dobija

```
Am ( IDA, <OA>, Ind, Ind1, ..., IndN, <OA1>, ..., <OAN> )
```

gde je Ind neophodno samo ako je specijalizacija parcijalna, kada ne-NULL vrednost označava da generalna instanca specijalizira a NULL obratno. Ind_i ima isto značenje na nivou i-tog specijalnog slučaja.

Reverzibilnost je očuvana pomoću sledećih ažurabilnih pogleda koji rekonstruišu šemu relacije generalnog slučaja i šemu relacije bilo kog specijalnog slučaja i

```
CREATE VIEW A ( IDA,<OA> )      CREATE VIEW Ai ( IDA,<OAI> )
  AS SELECT IDA,<OA>           AS SELECT IDA,<OAI>
  FROM Am;                    FROM Am
                               WHERE Indi IS NOT NULL;
```

Napomene na nivou pravila

U praksi se za preostalu šemu relacije zadržava raniji naziv (A umesto Am).

U slučaju hijerarhijskog lanca specijalizacije (specijalni slučajevi dalje specijaliziraju) navedena pravila se primenjuju od dole na gore.

U slučaju da neki specijalni slučaj učestvuje u više od jednoj specijalizaciji, bira se jedna po kojoj će se vršiti redukcija - ona koja je primarna, odnosno koja ima kardinalnost učešća u specijalizaciji (1,1).

Redukcija odnosa zavisnosti sa kardinalnošću uslovljavanja (0,1)

Suština ovog pravila je da se od dve šeme relacija kod kojih jedna u odnosu zavisnosti uslovljava drugu sa kardinalnošću uslovljavanja (0,1) formira jedna, i to dopunjavanjem šeme relacije uslovitelja. Pri tome je neophodno da je kardinalnost uslovljenosti zavisnog entiteta (1,1). Postoje dva slučaja.

Egzistencijalna zavisnost

Kod ove vrste zavisnosti uslovljeni entitet ima sopstvenu potpunu identifikaciju preko svog identifikacionog stvojsstva. Za slučaj uslovitelja A i uslovljenog B od dve šeme relacije

```
A ( IDA,<OA> )          B ( IDB,<OB>,IDA )
```

dobija se

```
Am ( IDA,<OA>, IDB,<OB> ) ili Am ( IDA,<OA>, Ind,<OB> )
```

Ako svojstvo IDB ima neko značenje koje je potrebno, koristi se prva varijanta pri čemu je IDB kvazi-ključ - dozvoljena je NULL vrednost i to sa ponavljanjem a za ne-NULL vrednosti važi ograničenje unikatnosti. Ako IDB nije od značaja koristi se druga varijanta gde NULL vrednost Ind označava da uslovljena instanca ne postoji a ne-NULL obrnuto.

Reverzibilnost je striktno očuvana samo kod prve varijante, pomoću sledećih ažurabilnih pogleda koji rekonstruišu polazne šeme relacija

```
CREATE VIEW A ( IDA,<OA> )      CREATE VIEW B ( IDB,<OB>,IDA )
  AS SELECT IDA,<OA>           AS SELECT IDB,<OB>,IDA
  FROM Am;                    FROM Am
                               WHERE IDB IS NOT NULL;
```

