

UDK 519.863

Brane Semulič
Cinkarna Celje

Povzetek

Članek opisuje predlog pristopa k snovanju računalniško podprtih informacijskih sistemov, ki naj bi rešil probleme konsistentnosti sistemov, ki nastopajo zaradi tega, ker obstoječi pristopi običajno ločijo projektiranje baze podatkov od projektiranja programov, ki te podatke uporabljajo.

Summary

In this paper an approach to computer supported information system design is presented. With described approach we can solve the problems of system consistency which occur in case of usual design when in separated ways is approached to the data base design and to the programs design.

1 U V O D

Predlagan pristop se nanaša na izgradnjo tistega dela poslovnih informacijskih sistemov, ki so determinirani in jih je možno detajlno strukturirati. Pri tem izhajamo iz dognanj teorije in prakse baz podatkov.

Pri snovanju večjih kompleksnih programskih proizvodov se srečujemo s problemi, ki se nanašajo na konsistentnost celotnega sistema, ker obstoječi pristopi v veliki meri ločijo projektiranje baze podatkov od projektiranja programov, ki te podatke uporabljajo, kar pomeni, da ne upoštevajo časovne povezanosti med samimi programi, kakor tudi med programi in podatki. Rezultat vsega tega se kaže v nerešenih problemih sinhronizacije.

Razrešitev opisanih problemov lahko dosežemo z uporabo pristopa, ki ga opredeljujeta sledeči fazi (6):

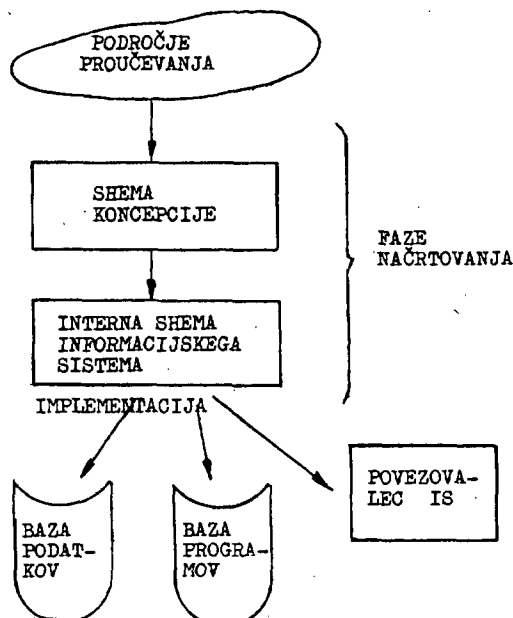
- izdelava sheme koncepcije informacijskega sistema in
- izdelava interne sheme informacijskega sistema.

Rezultat prve faze je "HEMA KONCEPCIJE INFORMACIJSKEGA SISTEMA", ki zagotavlja:

- celovito,
- konsistentno,
- neredundantno in
- jasno

semantično predstavitev proučevanega pojava.

Naloge in prednosti sheme koncepcije informacijskega sistema so enake kot pri shemi in podshemi baze podatkov, vendar s to razliko, da daje navedena shema kompleksnejši prikaz obravnavanega realnega pojava, ker upošteva tudi element časa. V naslednji fazi, ki obsega izdelavo "interne sheme informacijskega sistema", pa dopolnimo rešitve, ki smo jih dobili z izvedbo prve faze, s tehničnimi karakteristikami in parametri, ki smo jih prej izpustili. Rezultat interne sheme informacijskega sistema nato uporabimo za implementacijo programskega proizvoda, ki ga v tem kontekstu zaokrožuje struktura podatkov, ki je zbrana v bazi podatkov, zbirka programov in ustreznih transakcij v bazi programov ter zbirka kontrolnih ukazov, ki jih aktiviramo s pomočjo povezovalca informacijskega sistema, ki sproža programe in transakcije med posameznimi podatki. Na sliki 1 imamo ilustrativen prikaz postopka izgradnje tako zasnovanega programskega proizvoda.



Slika 1

Vlogo povezovalca informacijskega sistema lahko prevzamejo različni akterji, kot so na primer: ročni postopki, za to posebej izdelani programi, kombinacija obeh omenjenih sistemov za upravljanje z bazo podatkov (DBMS) ipd.

V nadaljevanju se bomo omejili na prikaz prve faze predlaganega pristopa, ki obsega izdelavo sheme koncepcije informacijskega sistema.

2 IZDELAVA SHEME KONCEPCIJE INFORMACIJSKEGA SISTEMA

2.1 Izhodišča izdelave sheme koncepcije informacijskega sistema

Pri izdelavi sheme koncepcije informacijskega sistema veljajo enake zakonitosti, kot pri izdelavi logične sheme baze podatkov, vendar s to razliko, da je shema koncepcije delovanja informacijskega sistema obravnavana s širšega vidika. Pri izdelavi omenjene sheme upoštevamo vse vidike realnega pojava in ne samo statični vidik organizacije kot je to slučaj pri bazi podatkov. Tukaj imamo v mislih predvsem transakcije, ki vsakokrat povzročijo vzpostavitev novega konsistentnega stanja znotraj sistema baze podatkov. Zaradi opisanih lastnosti transakcij jim pravimo, da predstavljajo "vedenjske enote" baze podatkov. S transakcijami generiramo torej novo stanje baze podatkov, ki ustreza dejanskemu stanju evolucije realnega pojava.

Pri opazovanju realnega pojava zasledimo, da le ta reagira na vsak dogodek, s tem, da s pomočjo izvedbe določenih aktivnosti vzpostavi novo stanje. Podobno situacijo imamo pri sistemih podatkov, ko s transakcijami vzpostavimo novo stanje, ki zrcali stanje zasledovanega realnega pojava.

Da se izognemo nekoordiniranosti delovanja sistema baze podatkov glede na delovanje realnega pojava, morajo biti transakcije, s katerimi spreminjamo stanje baze podatkov, preslikava aktivnosti zasledovanja realnega pojava.

Na osnovi zapisanega lahko podamo sledeči ugotovitvi, in sicer da (5):

- s podatki ponazarjamo podatkovne pojme (entitete) realnega pojava in da
- s transakcijami ponazarjamo vedenje proizvedenega realnega pojava.

Tako razširjeno gledanje nam je osnova za pristop k izdelavi sheme koncepcije informacijskega sistema, ki ga izdelamo brez upoštevanja kakršnihkoli tehničnih parametrov, ki jih je potrebno upoštevati za implementacijo modela na konkretnem računalniškem sistemu.

V našem gledanju modela koncepcije informacijskega sistema imamo torej formalizirano integracijo treh vidikov, in sicer:

- statične strukture elementov modela informacijskega sistema,
- transakcije med elementi modela informacijskega sistema in
- časovni vidik relacij med posameznimi elementi modela informacijskega sistema.

Če proučujemo realni pojav z opisanih treh vidikov, pridemo do sledečih ugotovitev (4):

- da obstajajo tri osnovne kategorije pojava, ki jih je možno opisati glede na njihove lastnosti kot:
 - objekte
 - dogodke in
 - operacije.
- da je možno dinamične odnose med naštetimi kategorijami izraziti s POVEZAVAMI, ki jih glede na vrsto povezave delimo na:
 - modifikacijo (povezava: OPERACIJA-OBJEKT)
 - spoznanje (povezava: OBJEKT-DOGODEK)
 - sprožitev (povezava: DOGODEK-OPERACIJA)

V nadaljevanju si bomo opisane kategorije in njihove medsebojne povezave nekoliko boljše ogledali.

2.2 Opisi kategorij realnega pojava in njihove povezave

Videli smo, da obstajajo tri osnovne kategorije realnega pojava, in sicer:

- objekt-predstavlja trajno konkretno ali abstraktno proučevano komponento organizacije, ki jo lahko boljše opredelimo,
- operacija-predstavlja nalogo, ki jo je potrebno realizirati v organizaciji ob določenem času,
- dogodek-predstavlja spoznanje o spremembi stanja enega ali več objektov po izvedbi operacij

Med posameznimi opisanimi kategorijami obstajajo povezave, ki jih opredelimo sledeče:

- spoznanje-je povezava, ki združuje dogodek in enega ali več objektov. Izraža spremembo stanja objektov, ki je povzročena s pojavom nekega dogodka,
- sprožanje-je povezava, ki združuje dogodek in eno ali več operacij. Pove, da določen dogodek sproži eno ali več operacij,
- modifikacija-predstavlja zvezo med operacijami in objekti. Izraža kako operacija preoblikuje objekt.

Iz opisanega je moč razbrati, da so dogodki tista kategorija, s katero opredeljujemo dinamiko realnega pojava.

2.3 Opredelitev dinamike realnega pojava

Dinamiko realnega pojava opredeljujejo dogodki, ki jih ločimo na (7):

- pasivne dogodke in
- aktivne dogodke.

Pasivni dogodki povzročijo samo novo stanje proučevanega objekta, medtem, ko aktivni dogodki sprožijo izvajanje ene ali več operacij.

Opisana dinamika realnega pojava temelji na jasnem razlikovanju med:

- stanjem,
- spremembo stanja, ki jo povzročijo pasivni dogodki in
- spremembo stanja, ki jo povzročijo aktivni dogodki.

Stanje nekega objekta je lahko trajno, medtem, ko je sprememba stanja dogodek in je trenutna. Pasivni dogodek izraža prehod iz enega stanja v drugo. Aktivni dogodek se razlikuje od pasivnega po tem, da spremembo stanja povzroči tudi izvajanje ene ali več operacij.

2.4 Formalna opredelitev realnega pojava, njegovih kategorij in njihovih lastnosti

2.4.1 Opredelitev na nivoju realnega pojava

Reslen pojav, ki ga v našem primeru zaokrožuje temeljni proces poslovnega sistema, bomo skušali opredeliti na sledečih nivojih (7):

- nivoju realnega pojava
- nivoju pojmovnega modela in
- nivoju podatkovnega modela

Pri proučevanju temeljnega procesa na nivoju realnega pojava ugotovimo, da ga je možno opredeliti z množico veličin, s katerimi opišemo njegove karakteristike. Te veličine označujemo kot objekte (OB_p). Objekt (OB_p) proučevanja v temeljnem procesu poslovnega sistema je lahko "kupec", "naročilo", "proizvod" itd. Dogodki (DG_p) pa povzročajo spremembe stanja opazovanih objektov (S_p^{OB}), ki po dogodku preidejo v neko novo stanje.

2.4.2 Opredelitev na nivoju pojmovnega modela

Preslikava temeljnega procesa na nivo pojmovnega modela nam služi za prikaz našega gledanja temeljnega procesa, ki abstrahira za nas nepomembne vidike gledanja tega procesa in prikazuje samo tiste elemente in njihove medsebojne odnose, ki so za nas pomembni. S ciljem takšne opredelitve modela temeljnega procesa (TP_p) smo ugotovili, da je v osnovi sestavljen iz treh temeljnih gradnikov - kategorij in sicer: objektov (OB_p), dogodkov (DG_p) in operacij (OP_p). To lahko zapišemo takole (7):

$$TP_p = (OB_p, DG_p, OP_p) \quad (1.1)$$

pri čemer je:

$$OB_p = \{OB_p^i; i = 1, 2, 3, \dots, n\} \quad (1.2)$$

$$DG_p = \{DG_p^i; i = 1, 2, 3, \dots, m\} \quad (1.3)$$

$$OP_p = \{OP_p^k; k = 1, 2, 3, \dots, l\} \quad (1.4)$$

Povezave, ki nastopajo med posameznimi kategorijami, pa napišemo takole:

• modifikacija (MO_p) predstavlja preslikavo o operacije (OP_p) v objekt (OB_p)

$$MO_p : OP_p \rightarrow OB_p \quad (1.5)$$

• spoznanje (SZ_p) predstavlja preslikavo objekta (OB_p) v dogodek (DG_p)

$$SZ_p : OB_p \rightarrow DG_p \quad (1.6)$$

• sprožitev (SP_p) predstavlja preslikavo dogodka (DG_p) v operacijo (OP_p)

$$SP_p : DG_p \rightarrow OP_p \quad (1.7)$$

Stanje (Sp) proučevanega pojava (TP_p) ugotovimo tako, da zasledujemo opisane kategorije in njihove medsebojne povezave (FV_p). Pri tem je potrebno izpostaviti, da nas zanimajo samo tista stanja, ki so dogodki. Formalno izrazimo to takole (7):

$$S_p = (TP_p, PV_p) \quad (1.8)$$

pri čemer je:

$$TP_p = (OB_p, DG_p, OP_p) \text{ in } \quad (1.9)$$

$$PV_p = (MO_p, SZ_p, SP_p) \quad (1.10)$$

Spremembo stanja (Sp) povzroči izvedba transakcij, ki predstavljajo osnovne elemente poslovanja v temeljnem procesu poslovnega procesa. Transakcijo sproži dogodek (DG_p), ki povzroči izvajanje ene ali več operacij (OP_p), s katerimi se izvede sprememba na ustreznih objektih (OB_p). V proučevanem pojavu (TP_p) se lahko izvaja sprožanje dogodkov (DG_p):

• v kronološki odvisnosti (KO_p) ali pa

• v pogojni odvisnosti (PO_p).

Dogodek DG_p^i je v kronološki odvisnosti (KO_p) z dogodkom DG_p^j takrat,

kadar dogodek DG_p^i nepogojno sproži operacijo OP_p^k , ki povzroči spremembo stanja objekta OB_p^j , kar spoznamo z dogodkom DG_p^j .

Dogodek DG_p^j je v pogojni odvisnosti (PO_p) z dogodkom DG_p^i takrat, kadar dogodek DG_p^i pogojno sproži operacijo OP_p^k ali kadar sprememba stanja, povzročena z operacijo OP_p^k ne povzroči vedno dogodka DG_p^j .

2.4.3 Opredelitev na nivoju podatkovnega modela

Proučevanje realnega pojava (TP_p) z nivoja podatkovnega modela predstavlja preslikavo pojmovnega modela z vidika opredelitve in analize potrebnih podatkov. Pri proučevanju kategorij realnega pojava (OB_p , DG_p in OP_p) ugotovimo, da pripadajo oziroma, da so povezani s posameznimi podatkovnimi razredi (R_p). Značilnosti, ki združujejo posamezne kategorije realnega pojava znotraj podatkovnega razreda, lahko formalno zapišemo takole (7):

$$(\forall R_p^i) (\forall x) (K_p^n) (x \in R_p^n \Rightarrow x \in R_p^i \wedge L(x))$$

Se pravi, da pri danem podatkovnem razredu (R_p^i) in opredeljeni lastnosti (L), ki je smiselna za attribute (x) podatkovnega razreda (R_p^i), eksistira kategorija (K_p^n), ki ima za attribute natanko tiste attribute podatkovnega razreda (R_p^i), ki imajo lastnosti (L).

V našem primeru imenujemo podatkovni model "shema koncepcije informacijskega sistema". Da bi zadostili potrebam izdelave predlagane sheme koncepcije in v formalni obliki opredelili prej opisane karakteristike in lastnosti proučevanega pojava, moramo najti ustrezen način formalnega prikazovanja. To dosežemo z (5):

- uporabo tipiziranega modela preslikav
- uvedbo tipov preslikav, s katerimi ponazorimo različne kategorije in
- uvedbo dejavnika časa.

Povezavo med podatkovnimi razredi in opisanimi kategorijami dobimo tako, da:

- vsak podatkovni razred in vse vrste povezav prikažemo z eno ali več preslikavami,
- lastnosti podatkovnega razreda prikažemo z atributi preslikav in
- uporabimo za prikaz podatkovnega razreda množico tipiziranih preslikav, ki jih označimo kot:
 - R-objekt,
 - R-dogodek in
 - R-operacija (6).

Navedene tipe preslikav izrazimo v tretji normalizirani obliki grupiranja podatkov v stavke.

Za potrebe dekompozicije podatkovnega razreda na elementarne podatkovne zapise se poslužujemo tipov preslikav. V nadaljevanju bomo podali kratke definicije posameznih tipov omenjenih preslikav:

- preslikava "R-objekt" ($R-OB_D$) je permanentna preslikava, kjer je vsak atribut v permanentni odvisnosti od ključa preslikave. "R-objekt" predstavlja elementarni položaj opazovanega podatkovnega razreda v sistemu. Ta preslikava nam omogoča, da prikažemo elemente proučevanega pojava in njihovo strukturo. Če bolj precizno povedano, ta preslikava nam omogoča časovno konsistentni vidik opazovanega realnega pojava.

Permanentna odvisnost med dvema atributoma A in B je elementarna, direktna in keno-nična odvisnost, kjer je vsak dogodek a, ki pripada atributu A, odvisen od dogodka b, ki pripada atributu B. Dogodka a in b imata enako življensko dobo. Zapisano trditev lahko izrazimo tudi takole (7):

$$A \rightarrow B = \{ (a,b) : (\forall a) (\forall b) (a \in A \wedge b \in B \wedge a \wedge b) \} \quad (1.12)$$

Atribut A implicira atribut B, pri čemer je a dogodek atributa A in b dogodek atributa B. Za vsak dogodek a in b velja, da sta elementa množice dogodkov atributa A oziroma B in da imata lastnost L.

- preslikava "R-operacija" ($R-OP_D$) je permanentna preslikava, ki se nanaša na določen podatkovni razred in je v tesni povezanosti s preslikavo "R-objekt" ($R-OB_D$). S to preslikavo opišemo spremembo stanja (TIP SPREMEMBE), ki se izvrši na določenem "R-objektu".

Normaliziran zapis relacije "R-operacija" mora odgovarjati naslednjim zahtevam (6):

$$R-OP_D \rightarrow R-OB_D \quad (1.13)$$

$$R-OP_D \rightarrow TIP \text{ SPREMEMBE} \quad (1.14)$$

$$R-OP_D \rightarrow TEKST \text{ OPERACIJE} \quad (1.15)$$

Spremembe se lahko pojavijo kot kreacija, destrukcija ali modifikacija. S tekstom operacije je mišljen kratek opis vsebine operacije.

Iz zapisa (1.13) vidimo, da se R-operacija nanaša na točno določen R-objekt in da z njo točno opredelimo, s kakšno vrsto stanja imamo opraviti (1.14 in 1.15).

- preslikava "R-dogodek" ($R-DG_D$) je permanentna preslikava. Njena normalizacija mora zadostiti sledečim pogojem:
 - da se nanaša na točno določen R-objekt (1.16)
 - da identificira nastalo spremembo izmed možnih tipov sprememb, ki smo jih opisali že pri opredelitvi preslikave "R-operacija" (1.17)
 - da opiše nastalo spremembo (1.18) in
 - da identificira operacije, ki se sprožijo z opisanim dogodkom (1.19)

Navedene pogoje je možno zapisati tudi sledeče (6):

$$R-DG_D \rightarrow R-OB_D \quad (1.16)$$

$$R-DG_D \rightarrow TIP \text{ SPREMEMBE} \quad (1.17)$$

$$R-DG_D \rightarrow OPREDELITEV \text{ SPREMEMBE} \quad (1.18)$$

$$R-DG_D \rightarrow R-OP_D \quad (1.19)$$

Sprožanje operacije, ki jo povzroči dogodek, je lahko "pogojeno" ali "iterativno". V prvem primeru se odgovarjajoča operacija sproži samo v slučaju, ko so izpolnjeni določeni pogoji, ki jih poprej definiramo. Pri iterativnem sprožanju pa se sproži izvajanje operacije ob vsakem pojavu ustreznega dogodka.

2.5 Statični in dinamični vidik sheme koncepcije informacijskega sistema

V predhodnih izvajanjih smo prikazali izhodišča in osnove za pristop k izdelavi sheme koncepcije informacijskega sistema. Temeljni proces poslovnega sistema smo opredelili na nivoju realnega pojava, nivoju pojmovnega modela in nivoju podatkovnega modela. V procesu snovanja informacijskega sistema nas zanima predvsem nivo podatkovnega modela, ki predstavlja nivo opredelitve predlagane sheme koncepcije informacijskega sistema.

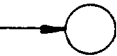
Samo shemo koncepcije informacijskega sistema delimo na dva dela, ki ju ločimo glede na vidik prikazovanja in obravnave na:

- statično podshemo (logična shema baze podatkov) in
- dinamično podshemo (shema interakcij informacijskega sistema)

Statična podshema oz. logična shema baze podatkov zajema opredelitev preslikav objektov ($R-OB_D$) realnega pojava ter njihovih lastnosti. Pri tem smatramo objekte kot podatkovne pojme, ki jih opredeljujejo ključni njihove identifikacije in atributi opisa njihovih lastnosti. S pomočjo sintakse, ki jo predvideva izdelava logične sheme baze podatkov, izrazimo povezave med tako opredeljenimi podatkovnimi pojmi.

Dinamična podshema oziroma shema interakcij informacijskega sistema pa obsega popis dogodkov ($R-DG_D$) in operacij ($R-OP_D$), s katerimi ponazarjamo dinamiko informacijskega sistema ter ilustrativen prikaz informacijskega sistema.

Za potrebe grafičnega prikaza uporabljamo sledeče simbole:

SIMBOL	POMEN
	R-objekt ($R-OB_D$)
	R-operacije ($R-OP_D$)
	R-dogodek ($R-DG_D$)
	iterativno sprožanje R-operacije ($R-OP_D$)
	pogojno sprožanje R-operacije ($R-OP_D$)
	"spoznanje", kot relacija med R-objektom ($R-OB_D$) in R-dogodkom ($R-OP_D$)
	"sprožanje", kot relacija med R-dogodkom ($R-DG_D$) in R-operacijami ($R-OP_D$)
	"modifikacija", kot relacijo med R-operacijo in R-objektom

Slika 3: Simboli, ki se uporabljajo pri izdelavi dinamične podsheme delovanja informacijskega sistema

V nadaljnjem procesu snovanja informacijskega sistema nam dinamična podshema (shema interakcij) služi kot osnova za izdelavo ustreznega programskega proizvoda, medtem, ko nam statična podshema služi kot logična shema baze podatkov, ki je izhodišče za načrtovanje ustrezne fizične zasnove baze podatkov na konkretnem računalniškem sistemu.

3 ZAKLJUČKI

Pri proučevanju realnega pojava in njegove dinamike pridemo do spoznanja, da ga opredeljujejo tri kategorije: objekti, dogodki in operacije, ki jih je možno ustrezno opredeliti na nivoju realnega pojava, nivoju pojmovnega modela in nivoju podatkovnega modela.

Če podatkovnim zapisom na nivoju podatkovnega modela, ki so normalizirani v tretji normalni obliki (3NF), dodamo še časovno omejitve, dobimo elementarne gradnike informacijskega sistema, ki jih opredelimo kot preslikave objektov ustreznih podatkovnih razredov, ki smo jih označili kot R-objekte.

Z kategorijami R-objekt, R-dogodek in R-operacija je na nivoju podatkovnega modela možno izdelati grafični prikaz delovanja informacijskega sistema. Pri tem ima takšen prikaz podobno vlogo kot logična shema baze podatkov in ga imenujemo dinamična shema koncepcije informacijskega sistema. Takšen prikaz se od logične sheme baze podatkov razlikuje v tem, da omogoča prikaz dinamike delovanja informacijskega sistema.

Opisani pristop sodi v družino sodobnih metod systemske analize, ki temeljijo na analizi podatkov. Zaradi konsekvencnih pravil, ki veljajo pri izdelavi dinamične in statične podsheme informacijskega sistema, je možno paralelno delo več strokovnjakov pri izvajanju systemske analize kompleksnih informacijskih sistemov, ki ob zaključku dela svoje rezultate brez problemov vskladijo.

LITERATURA

1. Martin J.: Computer Data Base Organization, Prentice Hall, New Jersey, 1977
2. Prijatelj N.: Matematične strukture I, Mladinska knjiga, Ljubljana, 1971
3. Rolland C., Albin P.: Management of Real Time System Projects, Proceedings of the 7. INTERNET World Congress, Copenhagen, 1982
4. Rolland C., Benci G.: Management of Information System Projects, Proceedings of the 6. INTERNET World Congress, Garmisch-Partenkirchen, 1979
5. Rolland C., Richard C.: Transactions Modeling, Université de Paris I, Pantheon-Sorbonne, Paris, 1982
6. Semolič B.: Magistersko delo, VEKŠ Maribor, 1986