Polnjenje podatkovnih skladišč v realnem času

Maja Ferle
SRC.SI, d. o. o.
Tržaška 116, Ljubljana
maja.ferle@src.si
Viljan Mahnič
Univerza v Ljubljani, Fakulteta za računalništvo in informatiko
Tržaška 25, Ljubljana
viljan.mahnic@fri.uni-lj.si

Povzetek

Podatkovna skladišča se v številnih podjetjih uveljavljajo kot skupna osnova za izvajanje analiz podatkov in poročanje iz različnih virov podatkov. Zaradi zahtev sodobnega tempa poslovanja je vedno več teženj k izgradnji podatkovnih skladišč, ki bi nudila podatke v realnem času. Najobsežnejši in tehnično najbolj zahteven del gradnje podatkovnih skladišč je prenos podatkov iz virov v podatkovno skladišče, imenujemo ga tudi proces ETL (angl. *Extract-Transform-Load*). V klasičnem smislu poteka proces ETL v obliki paketnih obdelav, v primeru podatkovnega skladišča v realnem času pa je treba proces ETL prilagoditi zahtevam po sprotnem prenosu podatkov. V prispevku obravnavamo proces ETL v primeru polnjenja podatkov v realnem času in ilustriramo polnjenje podatkov v realnem času na primeru podatkovnega skladišča operaterja mobilne telefonije.
Ključne besede: podatkovno skladišče, ETL, sprotno polnjenje podatkov, podatkovno skladišče v realnem času

Abstract

Real-time streaming ETL

Data warehouses are used in many organizations as unified platforms for data analyses and reporting from diverse data sources. Due to current high-paced business environments it is becoming increasingly important to build data warehouses with real-time data. The most time consuming and technically challenging part of building a data warehouse is the process of extracting, transforming and loading (ETL) data from source systems into the data warehouse. In a classical sense the ETL process is batch oriented while for the purposes of implementing real-time data warehouses the process must be altered to support real-time data loading. In this article we explain real-time ETL in the form of real-time streaming data and illustrate the implementation of a real-time data warehouse for a mobile telecommunications operator.
Keywords: data warehouse, ETL, real-time streaming ETL, real-time data warehouse

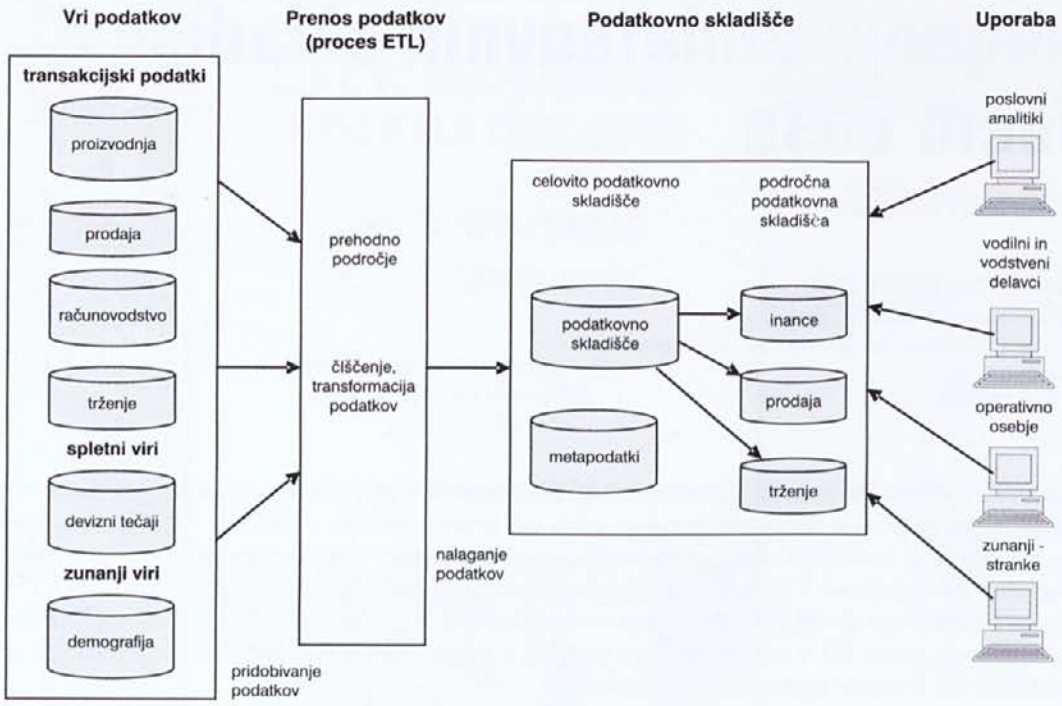
1 Uvod

Podatkovna skladišča se vedno bolj uveljavljajo in so v številnih podjetjih obvezni del informacijske podpore poslovanju. V osnovi predstavljajo podatkovna skladišča skupno podatkovno bazo, v katero se prenašajo podatki iz različnih podatkovnih virov v podjetju in drugih (tudi zunanjih) virov podatkov, tako da tvorijo enovito celoto, ki predstavlja skupni vir podatkov za izdelavo analiz podatkov, poročil in podlago za sprejemanje poslovnih odločitev.

V podatkovno skladišče se podatki prenašajo ob vnaprej predpisanih časovnih intervalih, največkrat v obliki paketnih prenosov. Tak pristop k prenosu podatkov uporabnikom zagotavlja, da so podatki v podatkovnem skladišču stabilni, da se medtem ko uporabniki izvajajo analize podatkov, ti ne spreminjajo. To je ena izmed poglavitnih zahtev za podatkovno skladišče, ki jo je postavil Bill Inmon [6], v svetu znan kot prvi, ki je formalno definiral podatkovno skladišče.

Slika 1 prikazuje arhitekturo skladiščenja podatkov z vsemi sestavnimi deli [11], ki vključujejo vire podatkov, prenos podatkov iz virov v enotno podatkovno skladišče (proces ETL), podatkovno bazo, v kateri je podatkovno skladišče s področnimi podatkovnimi skladišči, namenjenimi posameznim skupinam uporabnikov, in uporabo podatkovnega skladišča s pomočjo različnih orodij in programskih rešitev za analizo podatkov in poročanje.

Najobsežnejši in tehnično najbolj zahteven del gradnje podatkovnega skladišča je prenos podatkov iz virov v podatkovno skladišče [7], imenujemo ga tudi proces ETL. V tem procesu pridobimo podatke iz



Slika 1: **Arhitektura skladiščenja podatkov**

virov (Extract), jih prenesemo ter po potrebi preoblikujemo in prečistimo (Transform), na koncu pa jih naložimo v podatkovno skladišče (Load).

Podatkovno skladišče, v katerega prenašamo podatke ob vnaprej določenih časovnih intervalih, ne zadošča več zahtevam sodobnega tempa poslovanja in vedno večje konkurence med podjetji. Uporabniki potrebujejo informacije za podporo odločanju takoj, ko so nastale. Zato je vedno več teženj k izgradnji podatkovnih skladišč, ki bi podporala odločanje v realnem času. Imenujemo jih podatkovna skladišča v realnem času (angl. *real-time data warehouse*).

2 Podatkovno skladišče v realnem času

Razlogi, zakaj uporabniki potrebujejo podatkovna skladišča v realnem času, so povezani z načinom poslovanja podjetja. Tako kakor že v osnovi velja, naj podatkovna skladišča omogočajo prave informacije pravim osebam ob pravem času, to še bolj velja za uporabo podatkov v realnem času. Poslovne potrebe za podatke v realnem času so različne, v nadaljevanju je naštetih nekaj značilnih primerov uporabe.

2.1 Poslovne potrebe za podatke v realnem času

Najpomembnejša poslovna potreba za podatkovno skladišče v realnem času, ki se pojavlja v zadnjih letih, je podpora **upravljanju odnosov s strankami** (CRM), saj uporabniki potrebujejo vse podatke o stranki zbrane na enem mestu v trenutku, ko imajo opravka s stranko. Značilen primer uporabe je v klicnih centrih, kjer prejemnik klica potrebuje trenutni vpogled v vse podatke poslovanja stranke. Razen vpogleda v podatke o stranki lahko podatkovno skladišče daje še druge informacije v zvezi s stranko, na primer uvrstitev v segmentacijske razrede, vrednost stranke za podjetje in morda verjetnost, da bo stranka zapustila podjetje ali da bo poskusila zlorabo.

Druga potreba po podatkovnem skladišču v realnem času je podpora **upravljanju učinkovitosti poslovanja** ali s tujo kratico BPM (Business Performance Management), ki predstavlja spremljanje poslovnih procesov, namenjenih uspešnemu izvajanju poslovne strategije podjetja [4]. Cilj BPM je omogočiti posameznikom, da imajo pri roki vse informacije, ki jih potrebujejo za učinkovito opravljanje dela, za katerega so odgovorni. Zlasti v pogojih ostre globalne konkurence je pri številnih podjetjih ključno, da se znajo zelo hitro odzvati na dogajanje na trgu in na poteze konkurence. Če učinkovito opravljanje dela pomeni, da potrebujejo informacije v realnem času, morajo imeti zato ustrezno tehnološko podporo, med drugim tudi v obliki podatkovnega skladišča v realnem času.

2.2 Omejitve

V idealnem primeru bi podatkovno skladišče omogočalo odgovore na poslovna vprašanja v realnem času, kar pomeni, da bi bil podatek že v trenutku svojega nastanka v podatkovnem skladišču in na voljo za podporo pri odločanju. Vendar obstajajo za tako rešitev nekatere omejitve.

2.2.1 Tehnološke omejitve

Današnja tehnologija v obliki strojne opreme, sistemov za upravljanje podatkovnih baz, rešitev za prenos podatkov in podobnih prvin pri izdelavi podatkovnih skladišč je sicer dovolj zmogljiva, da obdela zelo velike količine podatkov v zelo kratkem času, vendar je še vedno omejena z odzivnimi časi. Tehnologije, potrebne za izvedbo rešitev v realnem času, zlasti rešitve za povezovanje podatkovnih virov (angl. *enterprise information integration* ali s kratico EII) ali pomnilniške podatkovne baze (angl. *in-memory database*) [12], so novejše in manj uveljavljene, zato so tudi bolj tvegane za uporabo in vsebujejo več neznank v primerjavi z bolj uveljavljenimi tehnologijami, potrebnimi za izvedbo klasičnih skladišč podatkov.

Če naj bodo podatki v podatkovnem skladišču na voljo v realnem času, mora infrastruktura omogočati, da so podatki resnično na voljo v realnem času. To je mogoče le, če je infrastruktura izjemno zanesljiva, razpoložljiva in deluje zelo hitro, sicer ves postopek prenosa podatkov v skladišče v realnem času nima smisla [1]. Za doseganje take hitrosti, zanesljivosti in razpoložljivosti je treba sprejeti tudi določene kompromise, zlasti je treba popustiti pri zahtevah za čiščenje in transformacijo podatkov, ki so lahko pri klasičnih podatkovnih skladiščih časovno zahtevni postopki, za katere pa v realnem času žal ni na voljo dovolj časa.

Prenašanje podatkov v podatkovno skladišče v realnem času dodatno obremenjuje sistem, kar velja za oba sistema, transakcijskega, iz katerega se sproti črpajo podatki med običajnim obratovanjem, in za podatkovno skladišče, v katero se podatki prenašajo. V klasičnem načinu uporabe podatkovno skladišče namreč služi le poizvedovanju, podatki pa se vanj prenašajo medtem, ko ga uporabniki ne uporabljajo. V okviru sedanjih tehnoloških možnosti so rešitve podatkovnih skladišč v realnem času tehnično najbolj učinkovito izvedljive s pomočjo tehnologije relacijske baze podatkov [9], zato se bomo v tem članku omejili le na to vrsto podatkovnih baz.

Sodobne relacijske podatkovne baze so zelo primerne tako za transakcijske sisteme, saj zmorejo v kratkem času zapisati veliko število podatkov, kakor tudi za podatkovna skladišča, saj nudijo hitre odzivne čase tudi za poizvedovanje v zelo veliki količini podatkov. Pri podatkovnih skladiščih v realnem času pa je način uporabe kombiniran, saj je treba istočasno vpisovati podatke in tudi poizvedovati po njih. Oboje naenkrat pa je težko realizirati. V transakcijskih sistemih je na primer navadno manj indeksiranja zaradi čim hitrejšega vpisovanja podatkov, medtem ko v podatkovnih skladiščih večje število indeksov omogoča hitrejše odzivne čase pri poizvedovanju.

2.2.2 Metodološke omejitve

Podatkovno skladišče naj bi po definiciji predstavljalo statično sliko podatkov v nekem časovnem trenutku [6]. Če podatki vanj prihajajo sproti, v realnem času, uporabniki nimajo več na voljo statičnih podatkov, saj se ti nenehno spreminjajo.

Zaradi različnih potreb uporabnikov podatkovnega skladišča, od katerih nekateri uporabniki analizirajo podatke in pregledujejo trende v preteklosti v klasičnem smislu, druga skupina uporabnikov pa potrebuje trenuten vpogled v podatke, je treba podatkovno skladišče razdeliti v več delov z različnimi svežinami podatkov. Del podatkovnega skladišča lahko vsebuje trenutne podatke, od tam pa se podatki počasneje prenašajo v del podatkovnega skladišča, ki ustreza klasični definiciji po Inmonu [6].

2.2.3 Omejitve zaradi načina poslovanja podjetja

Odločitev za izdelavo podatkovnega skladišča v realnem času je tako kakor pri izdelavi klasičnih podatkovnih skladišč predvsem odvisna od uporabniških potreb in zahtev. Vodstvo podjetja bi si morda želelo vpogled v trenutno stanje kazalnikov poslovanja podjetja v poljubnem trenutku, vendar je vprašanje, ali resnično sprotno spremljajo kazalnike. Podobno velja tudi za druge vrste uporabnikov: kljub temu da si morda želijo imeti trenutne podatke, se je treba pri implementaciji tovrstnih rešitev vprašati, ali poslovni procesi v podjetju zmorejo odziv v realnem času ali vendarle zadostuje določena časovna zakasnitev podatkov.

Če namreč poslovni procesi ne morejo slediti poslovnim odločitvam v realnem času, izgradnja rešitve v realnem času ni smiselna [3]. Na primer, če podjetje naroča izdelke enkrat dnevno, jim podatek,

da je sredi dneva zmanjkalo zaloge določenega izdelka, prav nič ne koristi oziroma nima prave poslovne vrednosti. Pri definiciji podatkovnega skladišča v realnem času gre bolj za to, da imajo uporabniki na voljo informacije dovolj hitro, da so se še sposobni odzvati nanje. Zaradi tega se ob pojmu podatkovno skladišče v realnem času uveljavlja tudi pojem »*right-time data warehouse*« ali v prevodu podatkovno skladišče ob pravem času [13].

3 Realizacija sprotnega polnjenja podatkov

Načrtovanje podatkovnega skladišča s podatki v realnem času se v nekaterih podrobnostih razlikuje od načrtovanja klasičnega podatkovnega skladišča, zlasti pri načrtovanju prenosa podatkov. Prenos podatkov v klasično podatkovno skladišče se načrtuje tako, da se izvaja takrat, ko uporabniki ne uporabljajo podatkov v skladišču, navadno ponoči ali ob vikendih. Takrat je na voljo dovolj časa, da se podatki prenesejo iz izvornih sistemov, prečistijo in preoblikujejo ter zapišejo v podatkovno skladišče, nakar se izdelajo še potrebni agregati in druge vnaprej pripravljene podatkovne strukture.

Kadar polnimo podatke sproti, v realnem času, si ne moremo privoščiti izključitve uporabnikov. Čas, ko uporabniki najbolj intenzivno uporabljajo podatkovno skladišče (denimo sredi dneva), zelo verjetno sovpada s časom, ko podatki najintenzivneje prihajajo v skladišče, saj se takrat tudi ustvarjajo. Zato je treba za potrebe podatkovnega skladišča v realnem času proces ETL načrtovati in izvesti tako, da je mogoče obratovanje podatkovnega skladišča za uporabnike tudi med polnjenjem podatkov. Dva primera mehanizmov sprotnega polnjenja podatkov v tabelo dejstev sta tekoče sprotno polnjenje (angl. *trickle feed*) in sprotno polnjenje z menjavo (angl. *trickle & flip*) [9]. Mehanizma sta podrobneje opisana v nadaljevanju.

3.1 Tekoče sprotno polnjenje

Če v podatkovnem skladišču resnično potrebujemo podatke v realnem času, je treba te podatke polniti sproti, torej trenutno [8]. Na prvi pogled bi bilo treba polniti podatke neposredno v tabele dejstev v podatkovnem skladišču. Vendar bi v tem primeru motili tiste uporabnike, ki izvajajo analize podatkov za daljša časovna obdobja in jih trenutni podatki ne zanimajo.

Eden izmed izhodiščnih razlogov za pojav podatkovnih skladišč je bila namreč ravno potreba po ločitvi podatkovnih baz, namenjenih zajemu transakcijskih podatkov, in podatkovnih baz za analize, saj se sicer transakcijske in analitične operacije na podatkih med seboj ovirajo. V primeru, da bi podatke, ki jih prenašamo v podatkovno skladišče, zapisovali neposredno v tabelo dejstev, bi morali uporabnikom omogočiti, da hkrati tudi poizvedujejo po njih. To pa bi povzročilo vrsto tehnoloških težav, zato mehanizem zapisovanja podatkov v realnem času v tabele dejstev ni primeren za uporabo v praksi.

3.2 Sprotno polnjenje z menjavo

Ta mehanizem rešuje problem hkratnega zapisovanja in branja podatkov iz tabele dejstev oziroma njene particije. V začasnem področju se naredi nova tabela dejstev, ki ima povsem enako zgradbo kot tabela dejstev v podatkovnem skladišču. V tej tabeli se zbirajo podatki, ki prihajajo v realnem času, njena vsebina pa je omejena na časovno obdobje, npr. na tekoči dan.

Ob vnaprej predvidenih časovnih intervalih (npr. vsako uro, lahko pa tudi vsako minuto, če je taka poslovna zahteva) se naredi kopija začasne tabele dejstev, ki se potem v trenutku zamenja z ustrezno particijo v podatkovnem skladišču, kar pomeni, da je tabela dejstev v podatkovnem skladišču v trenutku posodobljena. V praksi se je pokazalo [8], da je najučinkovitejše, če se zamenjava zgodi na vsakih 5–10 minut. Testirati je treba na realnih količinah podatkov, da ugotovimo, kako zmogljivo strojno opremo potrebujemo, da je tako kratek časovni interval še smiseln (da se v tem času prepiše začasna tabela).

3.3 Posebnosti načrtovanja podatkovnih skladišč v realnem času

Zaradi sprotnega polnjenja podatkov v podatkovno skladišče v realnem času je treba pri načrtovanju podatkovnega modela podatkovnega skladišča upoštevati nekaj posebnosti, ki jih zahteva sprotno polnjenje. Te so predstavljene v nadaljevanju.

3.3.1 Sinhronizacija agregatov s tabelami merljivih dejstev

V podatkovnih skladiščih pogosto uporabljamo agregate, to so dodatne tabele, v katerih so podatki vnaprej izračunani, npr. sešteti na višje ravni hierarhije, kar omogoča uporabnikom hitrejše odzivne čase pri poizvedovanju. Če se tabela dejstev nenehno spreminja, ker se v njej sproti polnijo podatki v realnem času, to pomeni, da se vnaprej pripravljeni agregati v podatkovnem skladišču ne ujemajo s podatki v

obstoječih tabelah dejstev, saj agregatov navadno ne osvežujemo v realnem času. Ker bi bilo osveževanje agregatov v realnem času časovno zelo zahtevno, se raje odločimo za rešitev, pri kateri uporabniki, ki poizvedujejo po podatkih v realnem času, poizvedujejo neposredno v tabelo dejstev in ne uporabljajo agregatov, uporabniki, ki uporabljajo podatkovno skladišče v klasičnem smislu in ne potrebujejo podatkov v realnem času, pa izvajajo poizvedbe iz agregatov.

Upoštevati je treba tudi podatke v začasnem pomnilniku *(cache)*, sodobnejše podatkovne baze namreč shranijo rezultate poizvedbe v pomnilniku zato, da jih lahko kasneje ponovno uporabijo, hkrati pa ne vedo, da se podatki spreminjajo v realnem času, kar lahko povzroči, da je rezultat poizvedbe neusklajen z dejanskimi podatki. Priporočljivo je, da se pri vpogledu v podatke v realnem času ne upošteva podatkov v pomnilniku [8].

3.3.2 Različne ravni svežine podatkov

Pri načrtovanju podatkovnega skladišča v realnem času je treba misliti na vse uporabnike: tako na tiste, ki potrebujejo vpogled v trenutne podatke, kakor tudi na tiste, ki le analizirajo podatke in ne potrebujejo trenutnega vpogleda. Različne skupine uporabnikov imajo lahko različne zahteve za svežino podatkov, npr. stanje danes zjutraj, stanje do nedelje do polnoči, mesečno stanje po zaključku obračunskega obdobja, ali druge posebne zahteve, npr. nabor izbranih vrst poslov, upoštevati pa je treba še skupino uporabnikov, ki analizirajo podatke v smislu dolgoročnih trendov in nočejo imeti vpogleda v trenutne podatke. Vsaki skupini uporabnikov je treba zagotoviti primerno rešitev za analizo podatkov glede na potrebe po svežini podatkov.

3.3.3 Različne vrste merljivih dejstev

Tabele dejstev v podatkovnem skladišču največkrat vsebujejo transakcijske podatke, ki so aditivni, zato je dodajanje novih zapisov v realnem času preprosto, saj se zapisi z novimi transakcijami preprosto dodajajo v tabelo.

Če pa tabela dejstev vsebuje presek stanja v določenem časovnem trenutku (npr. stanje na bančnih računih na določen dan ali uro), ti podatki niso aditivni prek časovne dimenzije. Vzdrževanje take tabele v realnem času je vprašljivo, saj bi bilo treba v vsakem trenutku izračunati novo stanje, če bi želeli imeti vpogled v trenutno stanje, kar pa predstavlja veliko časovno obremenitev. Zato je najbolje pustiti, da tabele s presekom stanj ohranijo značaj klasičnih tabel dejstev in jih ne vzdržujemo v realnem času.

3.3.4 Vzdrževanje dimenzij v realnem času

Predvideti je treba tudi, kako vzdrževati dimenzije v realnem času. Če se podatki polnijo v tabele dejstev v realnem času, bi bilo treba zagotoviti, da so tudi vse dimenzije osvežene v realnem času. Pri dimenzijah je teže ugotoviti spremembe v izvornih sistemih, saj mnogokrat nimajo časovnega ključa, da bi lahko iskali spremembe od določenega datuma ali ure dalje. Ta težava se pojavlja tudi pri klasičnih podatkovnih skladiščih, pri podatkovnih skladiščih v realnem času pa je še bolj izrazita.

Rešitev, ki je smiselna, je vpeljava tako imenovanih ogrodij ali začasnih zapisov v dimenzijskih tabelah.[1] Namreč, če se v tabeli dejstev znajde zapis o transakciji, za katero še ni vseh podatkov v ustreznih dimenzijah, se ta zapis kljub vsemu zapiše v tabelo dejstev, v manjkajočo dimenzijo pa se vpiše nov zapis, ki vsebuje samo šifro, vsa druga polja pa so prazna. Postopek, ki osvežuje dimenzije, lahko kasneje dopolni manjkajoče podatke [7]. Tako zagotovimo, da se vse transakcije sproti zapišejo v tabelo dejstev in so na voljo za takojšnje vpoglede, ne glede na to, da se dimenzije osvežujejo kasneje.

4 Primer podatkovnega skladišča telefonskih klicev

Kot primer prenosa podatkov v podatkovno skladišče v realnem času bomo obravnavali podatkovno skladišče operaterja mobilne telefonije [5] in na njem prikazali realizacijo enega od možnih pristopov sprotnega polnjenja podatkov.

Operater mobilne telefonije svojim uporabnikom, tako naročniškim kakor tudi predplačniškim, ponuja več storitev, npr. možnosti opravljanja telefonskih klicev, pošiljanja kratkih sporočil SMS, pošiljanja multimedijskih datotek (slike, filmi, zvok) in uporabe posebnih storitev (branje novic, rezervacija kinovstopnic, prebiranje malih oglasov, horoskopa, igranje igric ...). Uporabniki storitev mobilnega operaterja upo-

[1] *Kimball obravnava problematiko usklajevanja podatkov v dimenzijah s podatki v tabelah dejstev kot early arriving facts ali late arriving dimensions.*

rabljajo storitve ves čas, podnevi in ponoči. Uporaba storitev se zapisuje v več sistemih: telefonski klici naročniških uporabnikov v telefonski centrali naročniškega sistema, telefonski klici predplačniških uporabnikov v telefonski centrali predplačniškega sistema, podatki o gostovanju (uporaba storitev v tujih omrežjih) v datotekah, ki jih pošiljajo tuji operaterji mobilne telefonije prek klirinških hiš, uporaba posebnih storitev (novice, oglasi, igrice ...) v namenski podatkovni bazi, podatki o izdanih računih za opravljene storitve in prejetih plačilih pa se beležijo ločeno v sistemu za obračun (*billing*).

Trenutna dinamika polnjenja podatkov v podatkovno skladišče poteka paketno z nočnimi polnjenji podatkov iz vseh virov, tako da imajo uporabniki, ki analizirajo podatke, vsako jutro na voljo podatke prejšnjega dne do časa, ko se je izvajalo polnjenje posamezne skupine podatkov. Ker gre za razmeroma velike količine podatkov, traja polnjenje podatkov v skladišče večino noči, vnaprej pripravljeni agregati pa se izračunavajo v zgodnjih jutranjih urah. Včasih traja priprava agregatov toliko časa, da so uporabniki podatkovnega skladišča zjutraj že na delovnih mestih, pa agregati še niso dokončani, zato morajo čakati na podatke za analize. Če bi se v prihodnosti bistveno povečalo število uporabnikov mobilne telefonije ali če bi ti opravljali bistveno več telefonskih klicev, obstaja nevarnost, da bi se redno dogajalo, da podatkovno skladišče ne bi bilo na voljo zjutraj, ko pridejo uporabniki v službo. V tem primeru bi morda lahko nadgradili strojno in programsko opremo, tako da bi bil sistem zmogljivejši. Druga možna rešitev pa je vpeljava sprotnega polnjenja podatkov v skladišče, zato da se podatki vsaj deloma napolnijo že čez dan in se razbremeni del noči.

4.1 Potreba po podatkih v realnem času

Zaradi boljših analiz si operater mobilne telefonije želi imeti sprotne podatke o uporabi storitev in opravljenih telefonskih klicih, da bi lahko bolj tekoče spremljal obremenjenost telefonskih central, preverjal morebitne zlorabe in spremljal uspešnost raznih prodajnih akcij, zlasti prve dni po uvedbi akcij že prek dneva, ko poteka akcija.

Po drugi strani nekateri podatki niso zanimivi za analizo v realnem času oziroma ni smiselno, da se polnijo v podatkovno skladišče v realnem času. Primeri takih podatkov so podatki o gostovanju v tujih omrežjih, ki so na voljo z zamikom več dni po opravljeni storitvi, takrat ko jih pošljejo tuji operaterji mobilne telefonije. Prav tako niso zanimivi podatki o izdanih računih in prejetih plačilih v realnem času, saj se računi izdajajo nekajkrat mesečno v obliki paketne obdelave, podatke o plačilih pa posredujejo banke praviloma enkrat dnevno, pa še takrat z eno- do dvodnevno zamudo glede na datum plačila.

Zanimivo bi bilo uvesti rešitev za ugotavljanje zlorab, ki bi delovala v realnem času in bi lahko v vsakem trenutku zaznala morebitno zlorabo. Takšna rešitev se uvede tako, da se pri mobilnem operaterju vnaprej izoblikuje značilen profil uporabe storitev vsakega posameznega uporabnika storitev mobilne telefonije z metodami iskanja zakonitosti v podatkih. Potem se v realnem času preverja, ali storitev, ki se trenutno uporablja z dane telefonske številke, ustreza vnaprej izračunanemu profilu. Če ne, bi se vključil alarm, prek katerega bi lahko preverili, ali je sumljiva uporaba resnična ali gre za zlorabo. Ker taka rešitev potrebuje specializirane programske rešitve za odkrivanje zakonitosti v podatkih, pri obravnavanem mobilnem operaterju še ni v uporabi.

4.2 Polnjenje podatkovnega skladišča na klasični način

Podatkovno skladišče je narejeno v relacijski podatkovni bazi Oracle, podatki se polnijo ponoči v obliki paketnih obdelav, ki so narejene s procedurami v jeziku Oracle PL/SQL. Po končanem nočnem polnjenju podatkov, ki traja večji del noči, se v zgodnjih jutranjih urah pripravijo agregirani podatki za poročanje v obliki materializiranih pogledov. Uporabniki, ki analizirajo podatke, uporabljajo orodje Oracle Discoverer, s katerim neposredno poizvedujejo po podatkovnem skladišču. Zaradi velikih količin podatkov in posledično dolgih odzivnih časov pri poizvedbah, so uporabniške poizvedbe večinoma omejene na uporabo materializiranih pogledov (angl. *materialized view*), ki nudijo bolj zgoščene informacije za učinkovitejše analize. V relacijski podatkovni bazi Oracle se materializiran pogled definira na podoben način kakor navaden pogled, razlika je v tem, da je navadni pogled logična podatkovna struktura in se pri poizvedbah podatki berejo iz osnovnih tabel, ki so vsebovane v definiciji pogleda, medtem ko je materializirani pogled fizična struktura, saj se podatki fizično prepišejo v posebno tabelo in so tako podatki za poizvedovanje v materializiranem pogledu že vnaprej pripravljeni, kar omogoča hitrejše odzivne čase.

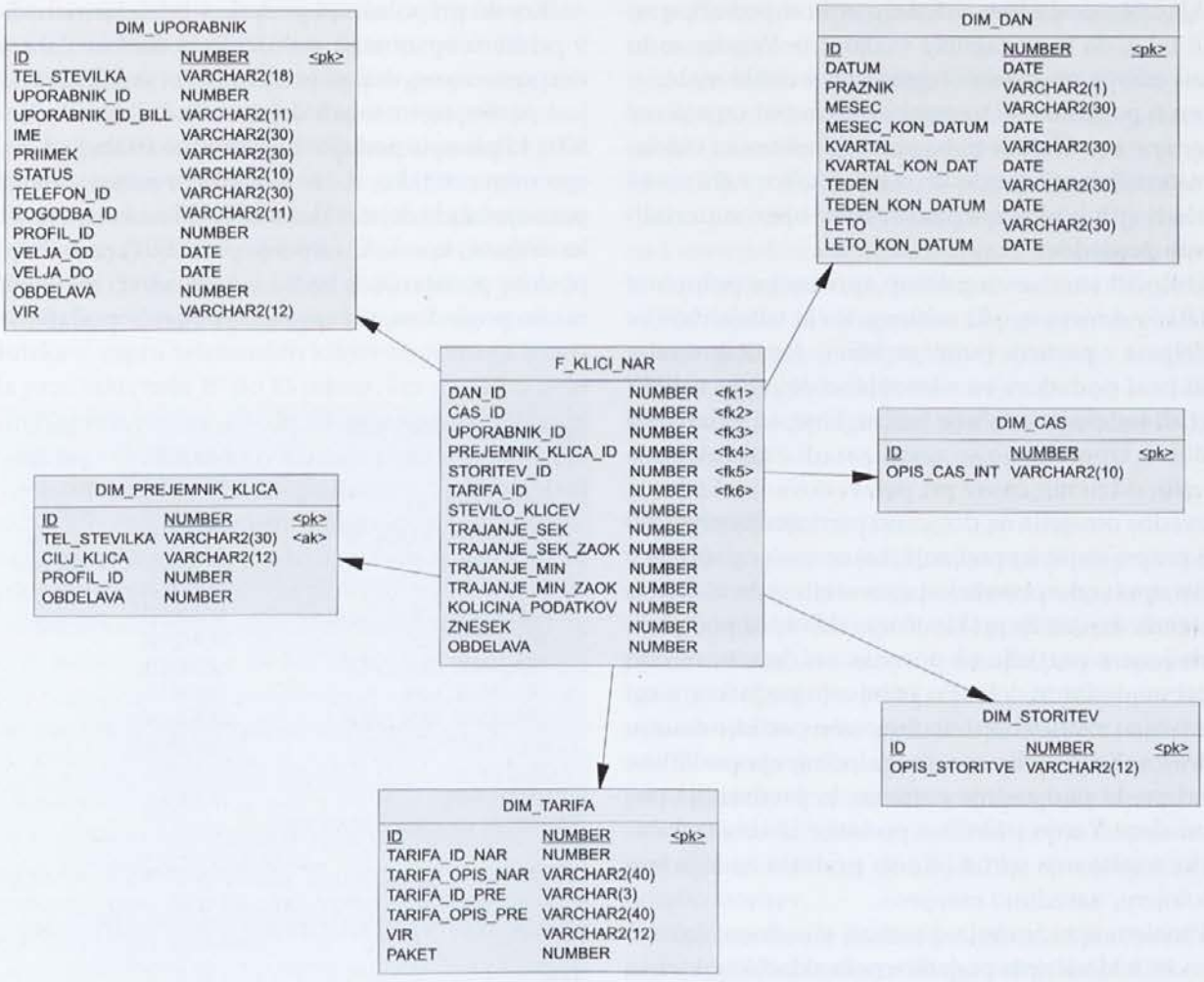
Za ilustracijo si bomo ogledali podatkovni model izseka iz podatkovnega skladišča, ki vsebuje tabelo

dejstev F_KLICI_NAR, v kateri so zabeleženi telefonski klici naročniških uporabnikov mobilnega operaterja (slika 2). Tabela dejstev je povezana z dimenzijami DIM_DAN, v kateri so datum in ustrezne hierarhije (mesec, leto), DIM_CAS, v kateri so polurni časovni intervali znotraj dneva, DIM_STORITEV, v kateri je šifrant storitev, ki jih uporabljajo uporabniki (npr. telefonski klic, sporočilo SMS idr.), in DIM_TARIFA, v kateri so tarife oz. cene, po katerih telefonirajo uporabniki glede na naročniško pogodbo, ki so jo sklenili z mobilnim operaterjem. Zadnji dve dimenziji sta DIM_UPORABNIK, ki predstavlja uporabnika storitev mobilne telefonije, ki je sklenil pogodbo z mobilnim operaterjem, in DIM_PREJEMNIK_KLICA, ki predstavlja prejemnika klica ali storitve. Ta je lahko hkrati tudi uporabnik storitev istega mobilnega operaterja, lahko pa je uporabnik storitev drugega telefonskega omrežja.

Polnjenje tabele dejstev F_KLICI_NAR s klasičnim postopkom ETL poteka prek začasnega področja v več zaporednih korakih, ki so opisani v tabeli 1.

4.3 Sprotno polnjenje podatkov

Za ilustracijo si bomo ogledali sprotno polnjenje podatkov o telefonskih klicih v realnem času. Na težavo naletimo že takoj na začetku, saj se izkaže, da telefonska centrala, v kateri se zapisujejo telefonski klici naročniških uporabnikov, dostavi zapisane podatke v datoteko le enkrat vsako uro. Če bi želeli resnično trenutne podatke, bi bilo treba najprej nastaviti telefonsko centralo tako, da bi pogosteje dostavljala podatke o zapisanih telefonskih klicih. Ker pa uporabniki ne



Slika 2: **Izsek iz dimenzijskega modela podatkovnega skladišča telefonskih klicev**

Tabela 1: **Koraki pri polnjenju podatkov o telefonskih klicih iz datoteke v tabelo dejstev – klasični ETL**

Korak	Ime procedure	Opis
100	nalozi_datoteko	Naloži naslednjo datoteko, ki je na vrsti na izvoru, v tabelo v začasnem področju ZP_KLICI_NAR – naložijo se vsi zapisi v datoteki brez omejitev
200	nastavi_parametre	Nastavi parametre (zaklepanje procesa, da se ne more ponovno zagnati, medtem ko ta poteka)
300	prenesi_klice	Prenesi podatke o opravljenih klicih v vmesno tabelo VM_KLICI_NAR – izberi le tiste zapise, ki predstavljajo odhodne telefonske klice
400	nove_dimenzije	Vstavi ogrodja v dimenzije
500	polni_f_klici_nar	Prenesi podatke v tabelo dejstev v podatkovnem skladišču F_KLICI_NAR

potrebujejo resnično trenutnih podatkov, so se odločili, da so podatki z eno uro zamude dovolj dober približek trenutnega časa za njihove potrebe, saj so doslej imeli podatke na voljo z enodnevno zamudo.

Najpreprostejši način polnjenja podatkov bi bil tak, da bi paketne obdelave, ki se v klasičnem polnjenju skladišča podatkov zaženejo enkrat ponoči, nastavili tako, da bi se zagnale vsako uro. Vendar se tu pojavi težava pri izdelavi agregatov v obliki materializiranih pogledov, ki vzamejo več časa in trajajo več kakor eno uro. Zato je treba poiskati rešitev za izdelavo materializiranih pogledov v manj kakor v eni uri ali poiskati sploh alternativno rešitev brez materializiranih pogledov.

Odločili smo se za pristop sprotnega polnjenja podatkov z menjavo, ki zahteva, da je tabela dejstev razdeljena v particije (angl. *partition*). Particije v relacijski bazi podatkov so namenjene deljenju velikih tabel ali indeksov v ločene fizične kose, ki so laže obvladljivi. Uporabljajo se zlasti zaradi zagotavljanja hitrejših odzivnih časov pri poizvedovanju, kadar je poizvedba omejena na določeno particijo. Omogočajo tudi preprostejše upravljanje, saj se vsaka particija z vidika upravnika obnaša kot samostojna tabela. Ker so bile tabele dejstev že pri klasičnem skladišču podatkov razdeljene v particije za posamezen dan, tu ni bilo potrebno dodatno delo. Pri polnjenju podatkov v realnem času v tabeli dejstev ohranimo particije na ravni dneva. Za potrebe sprotnega polnjenja podatkov naredimo kopijo zadnje particije, ki predstavlja trenutni dan. Vanjo polnimo podatke iz izvorne datoteke vsako uro sproti. Ko so podatki zadnje ure napolnjeni, naredimo menjavo.

Problem osveževanja dimenzij v realnem času je rešen že v klasičnem podatkovnem skladišču, kjer se uporablja pristop z vpisom ogrodij, zato ga pri sprotnem polnjenju podatkov nismo na novo postavljali.

Tabela dejstev F_KLICI_NAR je particionirana na ravni dneva. Skripta, ki definira tabelo dejstev in njene particije v relacijski podatkovni bazi Oracle, je prikazana v nadaljevanju (slika 3). Vse particije niso narejene za vse dni vnaprej, saj se dinamično dodajajo glede na datum polnjenja.

Koraki pri polnjenju podatkov o telefonskih klicih v primeru sprotnega polnjenja podatkov ostanejo nespremenjeni, dodati je treba le korake, ki upravljajo s particijami v tabeli dejstev. Zadnji korak (korak 500), ki prenese podatke neposredno v tabelo dejstev, spremenimo tako, da se podatki prenesejo v kopijo particije tabele dejstev (korak 510) in na koncu postopka dodamo korak, ki zamenja particijo (korak 520). Za potrebe poročanja se bomo morali odreči materializiranim pogledom, saj sprotno polnjenje podatkov ne

```
create table F_KLICI_NAR              (
   DAN_ID                             NUMBER       not null,
   CAS_ID                             NUMBER       not null,
   UPORABNIK_ID                       NUMBER       not null,
   PREJEMNIK_KLICA_ID                 NUMBER       not null,
   STORITEV_ID                        NUMBER       not null,
   TARIFA_ID                          NUMBER       not null,
   STEVILO_KLICEV                     NUMBER,
   TRAJANJE_SEK                       NUMBER,
   TRAJANJE_SEK_ZAOK                  NUMBER,
   TRAJANJE_MIN                       NUMBER,
   TRAJANJE_MIN_ZAOK                  NUMBER,
   KOLICINA_PODATKOV                  NUMBER,
   ZNESEK                             NUMBER,
   OBDELAVA                           NUMBER       not null)
PARTITION BY RANGE (DAN_ID)
(
   PARTITION DAN20041201 VALUES LESS THAN (20041201),
   PARTITION DAN20041202 VALUES LESS THAN (20041202),
   PARTITION DAN20041203 VALUES LESS THAN (20041203),
   ...
);
```

Slika 3: **Definicija tabele dejstev s particijami**

Tabela 2: **Koraki pri polnjenju podatkov o telefonskih klicih iz datoteke v tabelo dejstev – sprotni prenos podatkov**

Korak	Ime procedure	Opis
100	nalozi_datoteko	Naloži naslednjo datoteko, ki je na vrsti na izvoru, v tabelo v začasnem področju ZP_KLICI_NAR – naložijo se vsi zapisi v datoteki brez omejitev
200	nastavi_parametre	Nastavi parametre (zaklepanje procesa, da se ne more ponovno zagnati, medtem ko ta poteka)
300	prenesi_klice	Prenesi podatke o opravljenih klicih v vmesno tabelo VM_KLICI_NAR – izberi le tiste zapise, ki predstavljajo odhodne telefonske klice
400	nove_dimenzije	Vstavi ogrodja v dimenzije
510	polni_f_klici_nar_part	Prenesi podatke v kopijo dnevne particije tabele dejstev v skladišču podatkov F_KLICI_NAR_PART
520	zamenjaj_part	Zamenjaj kopijo particije s particijo v tabeli dejstev F_KLICI_NAR

dopušča časa za njihovo osveževanje, zato ni koraka za osveževanje materializiranih pogledov. Opis korakov pri sprotnem polnjenju podatkov je v tabeli 2.

Podrobnejši opis koraka 520 z izseki iz programske kode sledi v nadaljevanju (slika 4).

Primerjajmo čas, ki je potreben za polnjenje podatkov o telefonskih klicih v časovnem razponu 24 ur. V obeh primerih – tako v obliki klasičnega nočnega polnjenja podatkov kakor v obliki sprotnega polnjenja – je treba v podatkovno skladišče prenesti podatke iz 24 datotek z zapisi telefonskih klicev, ki jih vsako uro pripravi telefonska centrala. Polnjenje posamezne datoteke traja 10 do 15 minut, kar znese pri klasičnem postopku ETL skupaj 4 do 6 ur vsako noč za vseh 24 datotek. Sprotno polnjenje podatkov iz posamezne datoteke z zapisi telefonskih klicev enournega intervala prav tako traja 10 do 15 minut, ker pa se podatki polnijo sproti, so časi 10- do 15-minutnega polnjenja razpršeni prek dneva in noči, zato prihranimo 4 do 6 ur časa nočnega polnjenja podatkov. Transakcijski sistem zaradi tega ni dodatno obremenjen, saj telefonska centrala pripravi datoteko s podatki vsako uro, ne glede na to, ali se podatki polnijo v podatkovno skladišče ponoči ali čez dan. Materializirani pogled za prejšnji dan se naredi ponoči oz. v zgodnjih jutranjih urah, kakor pri polnjenju s klasičnim postopkom ETL. Za tekoči dan ni materializiranega pogleda, marveč se uporablja navaden pogled, ki vsebuje vse podatke iz materializiranega pogleda prejšnjega dne in podrobne podatke trenutnega dne v zadnji particiji, ki pa še niso agregirani.

```
- - zamenjaj particijo v tabeli dejstev z začasno tabelo
ALTER TABLE F_KLICI_NAR
EXCHANGE PARTITION DAN20041203 WITH TABLE F_KLICI_PART;

- - sprazni začasno tabelo in jo pripravi za nadaljevanje polnjenja
TRUNCATE TABLE F_KLICI_NAR_PART;

- - prenesi podatke zadnje particije tabele dejstev v začasno tabelo
INSERT INTO F_KLICI_NAR_PART
SELECT * FROM F_KLICI_NAR PARTITION DAN20041203;
```

Slika 2: **Zamenjava particije in priprava pomožne tabele za polnjenje podatkov**

Materializirani pogled, ki omogoča učinkovitejšo analizo podatkov, je narejen na ravni dneva (brez časovnih intervalov) in na ravni profila skupine prejemnikov klicev. Uporabniki, ki potrebujejo trenutne podatke, poročajo iz navadnega pogleda, ki je narejen nad materializiranim pogledom, v katerem so podatki do prejšnjega dne, in hkrati nad tekočo particijo tabele dejstev tekočega dne. Drugi uporabniki, ki ne potrebujejo trenutnih podatkov, poročajo iz materializiranega pogleda, ki je bil narejen ponoči in vsebuje le podatke prejšnjega dne, tako da se zanje podatki ne spreminjajo.

4.4 Primerjava

S klasičnim procesom ETL traja polnjenje podatkov v skladišče večji del noči, uporabniki pa vsako jutro vidijo podatke prejšnjega dne. Ti podatki zadostujejo za potrebe analiz, kot so npr. tedensko poročanje in kontroling. S sprotnim polnjenjem podatkov pa dobijo uporabniki na voljo podatke, ki so stari največ eno uro. Polnjenje podatkov poteka sproti čez dan, ponoči pa se naredijo le materializirani pogledi. Uporabniki, zlasti v oddelkih trženja in CRM, imajo tako na voljo podatke trenutnega dne in lahko sproti spremljajo porabo storitev.

Najzanimivejše analize podatkov, ki jih uporabniki lahko izvajajo zaradi sprotnega polnjenja podatkov, so spremljanje uspešnosti novih prodajnih akcij, zlasti prve dni po njihovem začetku. Uporabnike v oddelku trženja

namreč že čez dan zanima, kako poteka akcija in kako so dosežena pričakovanja, zato da bi lahko z morebitnim hitrim ukrepanjem, npr. z dodatnim oglaševanjem ali s spremembo ponudbe, spodbudili večje zanimanje kupcev. S klasičnim podatkovnim skladiščem so rezultate akcije lahko spremljali le za pretekli dan, ko je bilo lahko za hitre poteze že prepozno.

V uporabniškem vmesniku, ki je realiziran z orodjem Oracle Discoverer, imajo uporabniki ločen dostop glede na potrebe za analizo podatkov in poročanje. Tisti, ki gledajo le zgodovinske podatke, imajo v uporabniškem vmesniku dostop le do materializiranega pogleda, tako da se zanje kljub sprotnemu polnjenju podatkov nič ne spremeni. Uporabniki, ki gledajo trenutne podatke, pa imajo omogočen dostop do pogleda, v katerem so podatki z zamikom največ ene ure.

Če bi uporabniki želeli imeti še bolj sprotne podatke z zamikom manj od ene ure (ob predpostavki, da bi telefonska centrala dostavljala podatke pogosteje), se ves proces sprotnega polnjenja podatkov ne bi spremenil, le polnjenje bi potekalo pogosteje. Pri tem velja omejitev, da čas, potreben za nalaganje ene datoteke telefonskih klicev, ne sme biti daljši od intervala med dvema nalaganjema dveh zaporednih datotek.

5 Sklep

Skladiščenje podatkov v realnem času si šele utira pot v prakso, zato na tem področju še ni uveljavljenih standardov in značilnih metodologij. Orodja in tehnologije, ki bi se lahko uporabljali v ta namen, še niso zreli za uporabo, saj imajo vsak svoje pomanjkljivosti. Zato še velja, da je najbolj zanesljivo implementirati polnjenje podatkov v realnem času z uveljavljenimi tehnologijami, ki se uporabljajo za klasični proces ETL, le prilagoditi jih je treba, tako da omogočajo sprotni prenos in vpogled v podatke.

Pri tem je treba sprejeti določene kompromise, zlasti je treba popustiti pri obsegu čiščenja in transformaciji podatkov, saj je to časovno zahteven postopek, za katerega v realnem času ni dovolj časa. Tudi pri orodjih za končne uporabnike se pričakujejo izboljšave, zlasti pri odzivnih časih, saj so pri podatkih v realnem času smiselne le poizvedbe, ki se izvedejo zelo hitro, po možnosti prav tako v realnem času.

Zaradi vedno večjih zahtev po vpogledu v trenutne podatke se postavlja vprašanje, ali je sploh smiselno prenašati podatke v podatkovno skladišče v realnem času, saj bi lahko izvajali poizvedbe neposredno na transakcijskih podatkih, ki so trenutni. Kljub temu je smiselno graditi podatkovno skladišče, saj še vedno obstajajo potrebe tudi po klasični uporabi takega skladišča za zgodovinske preglede, analize trendov, transformacije podatkov in njihovo združevanje iz različnih virov. Te osnovne potrebe za podatkovno skladišče, ki ustreza klasični Inmonovi definiciji – kljub zahtevam po podatkih v realnem času –, ne bodo izginile.

6 Literatura

[1] BROBST Stephen A., *Establishing Service Levels for a Data Warehouse*, Business Intelligence Journal, The Data Warehousing Institute, letnik 6, št. 1, zima 2001, str. 37–45.

[2] BROBST Stephen A., *Real-Time Data Warehousing*, The Data Warehousing Institute, World Conference, jesen 2004, 304 strani.

[3] BURDETT John, SINGH Sanjay, *Challenges and Lessons Learned from Real-Time Data Warehousing*, Business Intelligence Journal, The Data Warehousing Institute, letnik 9, št. 4, jesen 2004, str. 31–39.

[4] ECKERSON Wayne, *Best Practices in Business Performance Management: Business and Technical Strategies*, The Data Warehousing Institute Report Series, marec 2004, 32 strani.

[5] FERLE Maja, *Pot do boljšega razumevanja uporabnikov mobilne telefonije*, INFO SRC.SI, SRC.SI d.o.o., št. 38, leto 2004, str. 39–41.

[6] INMON William, *Building the Data Warehouse*, Wiley 2002, 356 strani.

[7] KIMBALL Ralph, CASERTA, Joe: *The Data Warehouse ETL Toolkit*, Wiley 2004, 491 strani.

[8] LANGSETH Justin, *Real-Time Data Warehousing: Challenges and Solutions*, DSSResources.COM, april 2005.

[9] RADEN Neil, *Real Time: Get Real, Part II*, uredil Ralph Kimball, DM Review, junij 2003, Thomson Media, http://www.dmreview.com.

[10] THOMPSON Robert, *The Crumbling Foundations of the Data Warehouse*, DM Review, december 2004, Thomson Media, http://www.dmreview.com.

[11] WATSON Hugh J., *Recent Developments in Data Warehousing*, Communications of the AIS, št. 8, 2001, str. 1–25.

[12] WHITE Colin, *The Federated Data Warehouse*, DM Review, marec 2000, Thomson Media, http://www.dmreview.com.

[13] WHITE Colin, *Now is the Right Time for Real-Time BI*, DM Review, september 2004, Thomson Media, http://www.dmreview.com.

Maja Ferle se več kakor deset let ukvarja z izgradnjo podatkovnih skladišč in rešitev za poslovno obveščanje. Napisala je več strokovnih člankov in predavala na strokovnih konferencah doma in v tujini. Zaposlena je kot svetovalka v podjetju SRC.SI, d. o. o. Študira na magistrskem podiplomskem študiju informacijski sistemi in odločanje na Fakulteti za računalništvo in informatiko Univerze v Ljubljani.

Viljan Mahnič je izredni profesor in prodekan za pedagoško delo na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Ukvarja se z razvojem programske opreme za računalniško podprte informacijske sisteme s posebnim poudarkom na visokošolskih informacijskih sistemih. Od leta 1996 je predstavnik Slovenije v evropski organizaciji za univerzitetne informacijske sisteme EUNIS, od leta 2002 pa tudi član njenega sveta direktorjev. Na Univerzi v Ljubljani vodi projekt izgradnje spletnega informacijskega sistema in podatkovnega skladišča za področje študijske informatike.