

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Luka Kovačič

**Sodobna podpora odločanju v velikem  
poslovnem sistemu**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana 2013



UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Luka Kovačič

**Sodobna podpora odločanju v velikem  
poslovnem sistemu**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE  
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Matjaž Kukar

Ljubljana 2013



Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavlanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja. <sup>1</sup>

*Besedilo je oblikovano z urejevalnikom besedil  $\LaTeX$ .*

---

<sup>1</sup>Zaradi sledenja smernicam o varovanju informacij podjetja so v diplomskem delu vsi prikazi realnih podatkov ustrezno zamegljeni, prav tako pa podjetje nikjer v diplomskem delu ni omenjeno ali navedeno.





Št. naloge: 00361/2012

Datum: 06.12.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **LUKA KOVAČIĆ**

Naslov: **SODOBNA PODPORA ODLOČANJU V VELIKEM POSLOVNEM  
SISTEMU**

**MODERN DECISION SUPPORT IN A LARGE CORPORATION**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:


Prikažite postopek prenove klasičnega poročevalnega sistema, ki bo služil kot sodobna podpora odločanju v velikem poslovnem sistemu. Analizirajte obstoječe poročevalne sisteme v podjetju in predlagajte postopke za prehod na nov sistem, ki temelji na podatkovnem skladiščenju. Pripravite načrt funkcij, ki ob nastavljenih intervalih skrbijo za prenos podatkov iz transakcijske podatkovne baze v podatkovno skladišče. S pomočjo platforme Decisyon 4 pripravite prenovljeni sistem za podporo odločanju, ki vsebuje ekvivalentna poročila, ter omogoča enostavno pripravo novih. Po prenovi ovrednotite prednosti novega sistema, tako glede obremenitve računalniških zmogljivosti, kot uporabniške izkušnje.

Mentor:

  
doc. dr. Matjaž Kukar



Dekan:

  
prof. dr. Nikolaj Zimic



## IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Luka Kovačić, z vpisno številko **63070295**, sem avtor diplomskega dela z naslovom:

*Sodobna podpora odločanju v velikem poslovnem sistemu*

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Matjaža Kukarja,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 20. maj 2013

Podpis avtorja:



# Kazalo

Seznam uporabljenih kratic in simbolov

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Podatkovne baze</b>	<b>3</b>
2.1	Sistem za upravljanje s podatkovno bazo . . . . .	4
2.2	Relacijski model . . . . .	4
2.3	Sistem za upravljanje z relacijsko podatkovno bazo . . . . .	5
<b>3</b>	<b>Sistemi za podporo odločanju</b>	<b>7</b>
3.1	Podatkovno skladišče . . . . .	7
3.2	Proces ETL . . . . .	8
<b>4</b>	<b>Sprotno analitično obdelovanje</b>	<b>11</b>
4.1	OLAP . . . . .	11
4.2	OLAP kocka . . . . .	12
4.3	Operacije v OLAP-u . . . . .	16
4.4	Kategorije orodij OLAP . . . . .	17
<b>5</b>	<b>Uporabljena orodja</b>	<b>19</b>
5.1	Decisyon 4 . . . . .	19

## KAZALO

5.2	Oracle . . . . .	23
<b>6</b>	<b>Prenova sistema za podporo odločanju v izbranem podjetju</b>	<b>27</b>
6.1	Analiza izbranih obstoječih poročanj . . . . .	27
6.2	Prenos podatkov na nov sistem . . . . .	29
6.3	Priprava poročila v orodju Decisyon . . . . .	35
<b>7</b>	<b>Sklepne ugotovitve</b>	<b>47</b>

# Slike

2.1	Razlika med podatkom in metapodatkom. . . . .	3
2.2	Relacijski model - tabela. . . . .	5
2.3	Povezovanje relacijskih tabel. . . . .	6
3.1	Proces ETL. . . . .	9
4.1	Arhitektura OLAP. . . . .	12
4.2	Primer OLAP kocke. . . . .	13
4.3	Primer časovne dimenzije. . . . .	14
4.4	Primer zvezdne sheme. . . . .	15
5.1	Funkcionalnosti platforme Decisyon 4 [6]. . . . .	20
5.2	Arhitektura platforme Decisyon 4 [6]. . . . .	21
5.3	Arhitektura okolja Oracle. . . . .	24
6.1	Izbirna polja obstoječega poročevalnega sistema. . . . .	28
6.2	Graf izbranega obstoječega poročanja. . . . .	28
6.3	Primer nastavitve vira podatkov v orodju Decisyon PWD. . .	30
6.4	Data Integration Designer - Edit Source/Target Datasource. .	31
6.5	Modul "Schedules Designer". . . . .	32
6.6	Modul "Refresh Designer". . . . .	33
6.7	Create dbms_job - PL/SQL stavek. . . . .	34
6.8	Create dbms_job - PL/SQL čarovnik. . . . .	34
6.9	Modul "Dimension Wizard" - 1. in 2. korak. . . . .	36
6.10	Modul "Dimension Wizard" - 3. in 4. korak. . . . .	37

6.11 Modul "Time Dimension Wizard". . . . .	38
6.12 Modul "Cube Wizard" - 1. korak. . . . .	39
6.13 Modul "Cube Wizard" - 2. in 3. korak. . . . .	40
6.14 Modul "Cube Wizard" - 4. in 5. korak. . . . .	41
6.15 Decisyon Web - izbira podatkovnega modela. . . . .	42
6.16 Decisyon Web - urejevalnik poročila. . . . .	43
6.17 Decisyon Web - dodajanje metrike. . . . .	43
6.18 Decisyon Web - modul za kreiranje filtrov. . . . .	44
6.19 Decisyon Web - prikaz poročila. . . . .	45
6.20 Decisyon Web - prikaz poročila po korakih. . . . .	45

# Seznam uporabljenih kratic in simbolov

API	Application Programming Interface
BI	Business intelligence
DBMS	Database Management System
DDL	Data Definition Language
DML	Data Manipulation Language
DOLAP	Desktop OLAP
EMŠO	Enotna matična številka občana
ETL	Extract, Transform and Load
GUI	Graphical User Interface
HOLAP	Hybrid OLAP
HTML	Hyper Text Markup Language
MOLAP	Multi-dimensional OLAP
OLAP	OnLine Analytical Processing
PL/SQL	Procedural Language/Structured Query Language
RAM	Random Access Memory
RDBMS	Relational Database Management System
ROLAP	Relationall OLAP
SGA	System Global Area
SQL	Structured Query Language
XML	Extensible Markup Language



# Povzetek

Cilj diplomske naloge je prikazati postopek prenove poročevalnega sistema, ki služi kot sodobna podpora odločanju v velikem poslovnem sistemu. V začetku dela so teoretično opisane metode in tehnologije, ki so potrebne za doseg zastavljenega cilja, prav tako pa je obširno opisana platforma Decisyon 4, ki je bila s strani podjetja vpeljana kot temelj za prenovo obstoječih poročevalnih sistemov. Delo je v nadaljevanju sestavljeno iz praktičnih primerov projekta, ki ga krojijo trije ključni koraki. V prvem je opravljena analiza obstoječih poročevalnih sistemov in priprava podatkov za njihov prenos na nov sistem. Sledi kreiranje načrtovanih oz. planiranih funkcij, ki ob nastavljenih intervalih skrbijo za prenos podatkov iz transakcijske podatkovne baze v podatkovno skladišče, v zadnjem koraku pa so s pomočjo platforme Decisyon 4 pripravljena zelena poročila. S prenovljenimi poročevalnimi sistemi bo v prihodnosti razbremenjenost transakcijske podatkovne baze prineslo hitrejše in zanesljivejše delovanje tekočih poslovnih procesov v podjetju, vodilnim osebam pa bo omogočeno lažje identificiranje ključnih podatkov pri sprejemanju pomembnih poslovnih odločitev.



# Abstract

The goal of the thesis was to demonstrate the process of renovation of the reporting system, which serves as a modern decision-making support system in a large enterprise. First part of the thesis contains theoretical descriptions of methods and technologies that are required to reach the target and also details of Decisyon 4 platform, which was implemented by the company as the basis for the renovation of existing reporting systems. Thesis is further composed of practical examples of project, which is built with three key steps. The first was an analysis of existing reporting systems and preparation of the data for their transfer to the new system. This was followed by the creation of planned functions that trigger at predetermined intervals and take care of transferring data from transactional database into the data warehouse, and finally preparation of reports using the Decisyon 4 platform. Renewed reporting system will bring faster and more reliable operation of current business processes in the company, while the leading persons of the company will be able to easily identify critical data for making important business decisions.



# Poglavje 1

## Uvod

Za vsako podjetje so poslovne odločitve ključnega pomena. Z uporabo različnih orodij, katerih namen je podpora poslovnemu odločanju, “zožimo” možnosti za sprejemanje napačnih odločitev. Taka orodja se uporabljajo predvsem za namene poročanja in analize (npr. pregled poslovanja v preteklem letu, pregled produktivnosti itd.), preko katerih dobimo informacije potrebne za nadaljnje odločitve v podjetju.

Namen oz. cilj (in rezultat) diplomske naloge je pregled, analiza in posodobitev poročevalnih sistemov v velikem poslovnem sistemu, ki za konkurenčnost na trgu potrebuje sistem, ki bi mu v sedanjosti in tudi v prihodnosti omogočal prave poslovne odločitve.

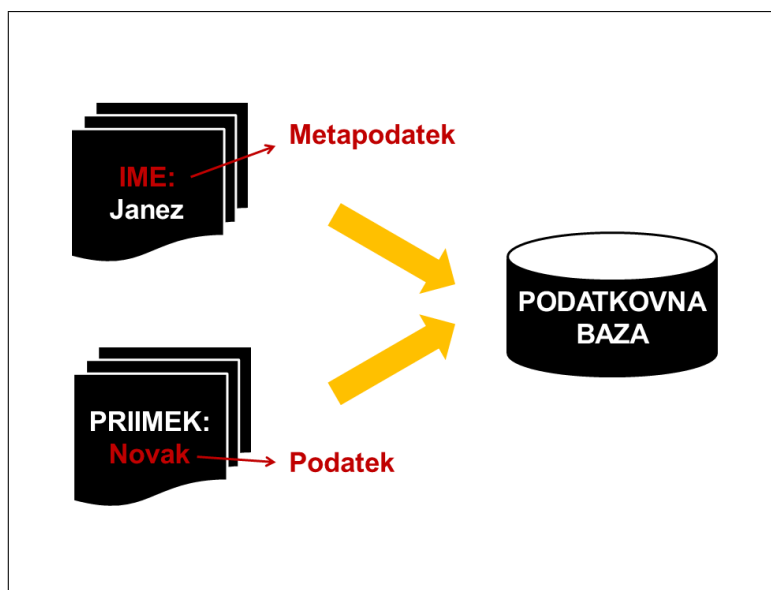
Za doseg cilja je bilo potrebno analizirati obstoječa poročanja, pripraviti načrt za prenos na orodje Decisyon in ta načrt tudi izvršiti. Obstoječa poročanja so temeljila na podpori transakcijske podatkovne baze (Oracle), kar je pomenilo konstantno obremenitev ob izpisovanju poročil. V fazi načrtovanja je bilo zato potrebno pripraviti procedure za prenos podatkov iz transakcijske podatkovne baze v podatkovno skladišče, ki je optimizirano za poizvedovanje in analiziranje. To je zahtevalo uporabo procesa ETL, ki predstavlja polnjenje podatkovnega skladišča. Sledila je uporaba tehnologije OLAP, ki je namenjena večdimenzionalnemu analiziranju podatkov v podatkovnem skladišču (OLAP kocke). Zadnja faza je bila priprava poročil

(grafična upodobitev pridobljenih podatkov) v orodju Decisyon s pomočjo OLAP kock.

V nadaljevanju diplomske naloge so opisani vsi omenjeni koraki, orodja, tehnologije in rezultati prenove, vključno z uporabniško izkušnjo.

## Poglavje 2

# Podatkovne baze



Slika 2.1: Razlika med podatkom in metapodatkom.

Podatkovna baza je integrirana računalniška struktura, ki hrani zbirke podatkov pomembnih za končnega uporabnika in metapodatke oz. podatke o podatkih, preko katerih se upravlja s podatki [2].

Metapodatki vsebujejo informacije o karakteristikah podatkov in relacijah, ki povezujejo podatke znotraj podatkovne baze. Nekaj takih informa-

cij so ime podatka, tip podatka (število, besedilo, datum,...), obveznost podatka in tako naprej. Metapodatki nam nudijo širšo sliko o podatkih v podatkovni bazi, do katerih pridemo s pomočjo sistema za upravljanje s podatkovno bazo [2].

## 2.1 Sistem za upravljanje s podatkovno bazo

DBMS (*angl. Database Management System*) je zbirka programov, ki upravljajo s podatkovnimi strukturami in kontrolirajo dostop do podatkov shranjenih v podatkovni bazi. Služi kot vmesnik med uporabnikom in podatkovno bazo. Podatkovna struktura je shranjena kot zbirka datotek, edini način da iz teh datotek izluščimo podatke pa je s pomočjo sistema za upravljanje s podatkovno bazo. Kreiranje podatkovnih struktur je omogočeno preko **DDL** (*angl. Data Definition Language*), upravljanje s podatki pa preko **DML** (*angl. Data Manipulation Language*). Glavna prednost sistema za upravljanje s podatkovno bazo je ta, da je podatkovno bazo možno uporabljati z različnimi aplikacijami. V preteklosti so imele aplikacije vsaka svojo “podatkovno bazo” imenovano tudi datotečni sistem. Želja po boljšem upravljanju s podatki je vodila k razvoju različnih podatkovnih modelov. Ti določajo kako smejo biti podatki v podatkovni bazi organizirani oz. strukturirani. Poznamo hierarhični, mrežni, relacijski, objektni in večdimenzionalni podatkovni model (podatkovna skladišča) [2].

## 2.2 Relacijski model

Relacijski model je predstavil Edgar F. Codd (IBM) leta 1970 in s tem sprožil revolucijo pri uporabnikih in razvijalcih podatkovnih baz. Temelji na matematični teoriji relacij. Da bi se izognili kompleksnosti matematične teorije si lahko relacijo predstavljamo kot dvodimenzionalno tabelo z vrsticami in stolpci, kjer vsak stolpec predstavlja atribut relacije, vrstica pa n-terico relacije [2].

The diagram shows a table with three columns: 'Ime', 'Priimek', and 'Starost'. The 'Starost' column is highlighted with a yellow box and labeled 'Atribut'. The second row of the table is highlighted with a yellow box and labeled 'N-terica'.

Ime	Priimek	Starost
Janez	Novak	32
France	Kovač	46
Marija	Zupan	65
Ivan	Horvat	54

Slika 2.2: Relacijski model - tabela.

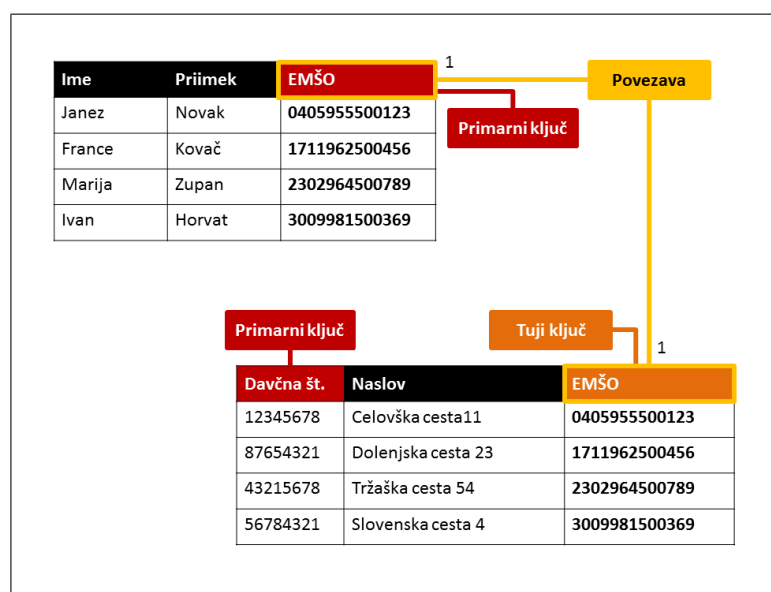
Iz Slike 2.2 lahko razberemo, da atribut predstavlja poimenovanje stolpca oz. lastnost relacije, n-terica pa vsebuje vrednosti teh atributov. Relacijske podatkovne baze so zaradi svoje enostavnosti postale standard za upravljanje s podatki. Posledično so se z razvojem relacijskega modela razvili sistemi za upravljanje z relacijskimi podatkovnimi bazami.

## 2.3 Sistem za upravljanje z relacijsko podatkovno bazo

RDBMS (*angl. Relational Database Management System*) je visoko razvit sistem preko katerega je implementiran relacijski model. RDBMS poleg vseh funkcionalnosti, ki olajšajo razumevanje in upravljanje relacijskega podatkovnega modela, nudi tudi vse osnovne funkcije sistemov za upravljanje s podatkovnimi bazami. Ena večjih prednosti sistema za upravljanje z relacijskimi podatkovnimi bazami je zmožnost prikrivanja kompleksnosti relacijskega modela končnemu uporabniku. To pomeni, da RDBMS upravlja z

vsemi fizičnimi podrobnostmi relacijske baze, uporabnik pa jo vidi kot skupek tabel, ki hranijo podatke. Z na videz enostavnimi in logičnimi akcijami lahko manipulira, išče in vzdržuje podatke v tabelah [2].

Tabele so med seboj povezane preko “skupnih” atributov (vrednosti atributov), ki si jih delijo. V Sliki 2.3 smo za primer vzeli številko EMŠO, ki lahko enolično identificira posamezno osebo. Ta atribut (oz. vrednost atributa) povezuje 2 tabeli, kjer hranimo različne podatke [2].



Slika 2.3: Povezovanje relacijskih tabel.

Atribut EMŠO je zaradi enoličnosti tudi izbrani kandidat za **primarni** ključ. Poleg primarnega ključa poznamo tudi **sekundarni** ključ (**unikaten** ključ - *angl. Unique key*), ki olajša ali pospeši dostop do posamezne vrstice v tabeli ter **zunanji** ali tuji ključ, ki je primaren v drugi relaciji in se uporablja za povezovanje posameznih relacij v podatkovni bazi. Tipi povezav so lahko **1:1** (ena proti ena), **1:M** (ena proti mnogo) in **M:N** (mного proti mnogo). Za poizvedovanje po podatkih v relacijski podatkovni bazi uporabljamo strukturiran poizvedovalni jezik (*angl. SQL – Structured Query Language*) [2].

## Poglavje 3

# Sistemi za podporo odločanju

Sistemi za podporo odločanju so računalniško podprti informacijski sistemi, ki služijo kot pomoč pri poslovnih odločitvah. Njihov glavni namen je, da bi olajšali proces sprejemanja ključnih odločitev v podjetju. Sistemi za podporo odločanju lahko podatke prikazujejo tako (npr. grafično), da so prilagojeni potrebam končnega uporabnika (zaposleni, nadrejeni, vodja uprave itd.)[2].

### 3.1 Podatkovno skladišče

Poslovno odločanje zahteva vedno večjo informacijsko podporo, za katero operativni sistemi nikoli niso bili načrtovani. Zaradi vse večjih količin podatkov in zahtev po analiziranju so se razvila podatkovna skladišča. Namen podatkovnih skladišč je preoblikovanje preteklih podatkov v poslovni vir znanja, kar pomeni enoten pogled na skupne podatke podjetja ne glede na njihov izvor (različna orodja, tehnologije, ...). Bill Inmon, ki mu pravijo tudi oče podatkovnega skladišča, ta strokovni izraz razlaga kot integrirano, entitetno usmerjeno (subject-oriented), časovno odvisno (time-variant) in nespremenljivo (nonvolatile) zbirko podatkov, ki služi podpori odločitvenim procesom. Za lažje razumevanje definicije si pogledjmo kaj predstavljajo omenjene lastnosti[2]:

- **Integriranost.** Podatkovno skladišče je centralizirana, urejena podat-

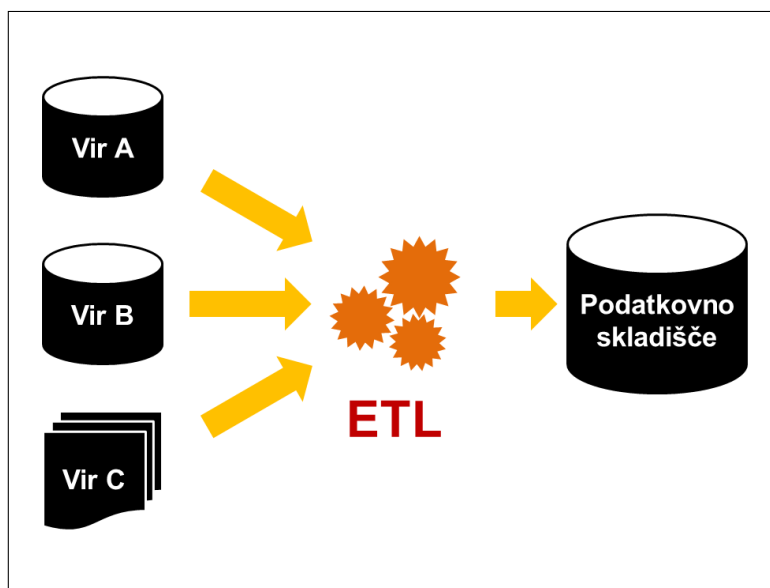
kovna baza, ki hrani podatke različnih formatov, pridobljenih iz več virov znotraj organizacije. Cilj integracije je, da omogoči konsistenten in enoten pogled na podatke.

- **Entitetna usmerjenost.** Podatkovno skladišče je organizirano in urejeno po glavnih enotah podjetja, kot so marketing, proizvodnja, skladiščenje, itd. Za vsako enoto podatkovno skladišče hrani določene entitete (npr. izdelek, regija, stranka, ...).
- **Časovna odvisnost.** Za razliko od operativnih sistemov, ki se osredotočajo na trenutne transakcije (tj. “žive” podatke), podatkovna skladišča predstavljajo podatke v preteklem času. Podatki se v podatkovno skladišče prenašajo periodično (npr. enkrat dnevno) in so lahko prikazani v različnih časovnih formatih (dan, teden, mesec, ...).
- **Nespremenljivost.** Podatki v podatkovnem skladišču predstavljajo zgodovino podjetja, zato niso namenjeni spremembam aktivnih aplikacij in se nikoli ne brišejo. Ob osveževanju se podatki iz operativnih sistemov (“živi” podatki) dodajajo v podatkovno skladišče.

Podatkovno skladišče je običajno namenjeno samo za branje in je optimizirano za poizvedovanje in analiziranje. Preden se podatki dodajo v podatkovno skladišče jih moramo integrirati, kar pomeni, da morajo skozi “podatkovni filter”, ki mu pravimo ETL [2].

## 3.2 Proces ETL

ETL (*angl. Extract, Transform and Load*) je proces, ki predstavlja polnjenje podatkovnega skladišča (Slika 3.1). Ta proces je v večini primerov precej kompleksen, zato je pred pripravo pomembna podrobna analiza podatkov, ki se bodo prenašali v podatkovno skladišče. ETL proces vključuje pridobivanje (*angl. Extract*), preoblikovanje (*angl. Transform*) in nalaganje podatkov v podatkovno skladišče (*angl. Load*) [4, 10].



Slika 3.1: Proces ETL.

**Pridobivanje** podatkov je prva faza procesa ETL. Pridobivanje večinoma poteka iz več različnih sistemskih virov, ki lahko uporabljajo različno strukturo in format podatkov. V tej fazi je potrebno pridobljene podatke preveriti in pretvoriti v predpisano enotno obliko. Podatkovne baze v izvornih sistemih so lahko relacijske ali ne-relacijske, kar pomeni dodatno težavnost pri pripravi podatkov za fazo transformacije (preoblikovanja). Pridobljeni podatki so na koncu zbrani v stolpcih, ki jih potrebujemo za obdelavo [4, 10].

**Preoblikovanje** je faza, ki s pomočjo pravil ali funkcij zagotovi nalaganje izbranih pridobljenih podatkov v ciljni sistem. Kako zahteven je proces preoblikovanja je odvisno predvsem od posameznega podatkovnega vira ter poslovnih zahtev. V nekaterih primerih je lahko zahtevano tudi več spodaj navedenih kombinacij obdelave [4, 10]:

- Nalaganje samo določenih stolpcev podatkov
- Čiščenje in prevajanje podatkov
- Združevanje podatkov iz različnih virov

- Preračunavanje novih vrednosti
- Razcepljanje v več stolpcev

**Nalaganje** je zadnja faza znotraj procesa ETL. V tej fazi naložimo izbrane in preoblikovane podatke v podatkovno skladišče. Obseg nalaganja je odvisen od velikosti in zahtev podjetja. Nekatera podatkovna skladišča stare podatke v določenem časovnem intervalu nadomeščajo z novimi (npr. enkrat dnevno, mesečno), druga podatkovna skladišča pa nove podatke dodajajo k obstoječim. Na ta način lahko spremljajo njihove spremembe skozi čas. Prenos podatkov iz transakcijske podatkovne baze v podatkovno skladišče je prikazan po korakih v Podpoglavju 6.2 [4, 10].

Podatkovno skladišče po končanem procesu ETL vsebuje kakovostne in urejene podatke, ki so tako pripravljene za opravljanje analiz in poročanj. Pri tem si najbolje pomagamo s tehnologijo sprotnega analitičnega obdelovanja, ki nam ponuja pogled na zbirko podatkov z različnih zornih kotov.

# Poglavje 4

## Sprotno analitično obdelovanje

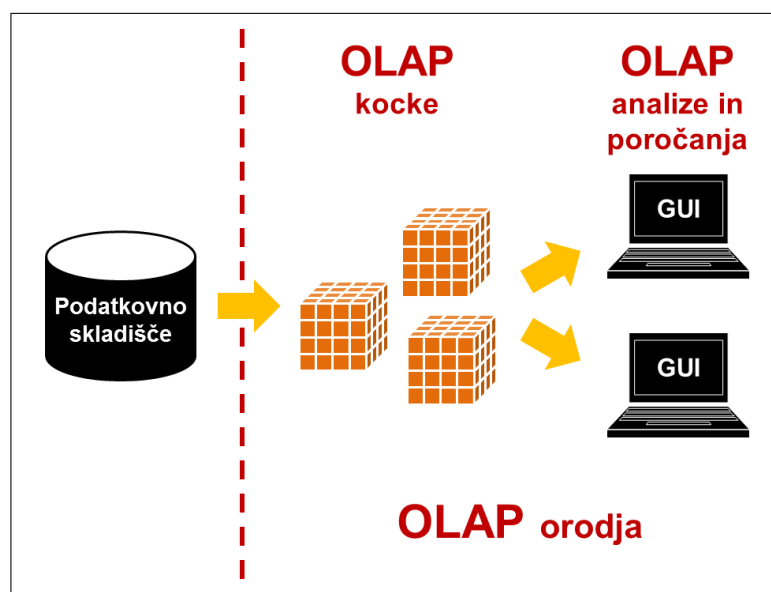
Popolna arhitektura podatkovnega skladišča vključuje hranjenje podatkov za podporo odločanju, filter za njihovo pridobivanje in preoblikovanje (ETL) ter vmesnik, ki omogoča podrobne in natančne predstavitve teh podatkov. Ena izmed najbolj pogostih arhitekturnih oblik za podporo odločanju je **OLAP** [1, 2].

### 4.1 OLAP

Sprotno analitično obdelovanje ali OLAP (*angl. OnLine Analytical Processing*) je tehnologija, ki se uporablja za (večdimenzionalno) analiziranje podatkov v podatkovnem skladišču. OLAP s hitrimi, zanesljivimi in interaktivnimi dostopi do različnih pogledov na podatke uporabnikom omogoča temeljitejše razumevanje in znanje o njihovi podjetju. Sistemi OLAP imajo štiri ključne lastnosti [1, 2]:

- uporabljajo večdimenzionalne tehnike analiziranja podatkov,
- nudijo napredno podporo podatkovnim bazam (dostop do različnih podatkovnih virov, napredne operacije za delo s podatki, hitri odzivni časi itd.),

- nudijo uporabniški vmesnik (*angl. GUI - Graphical User Interface*), ki je enostaven za uporabo in
- podpirajo arhitekturo odjemalec/strežnik (*angl. Client/Server*).



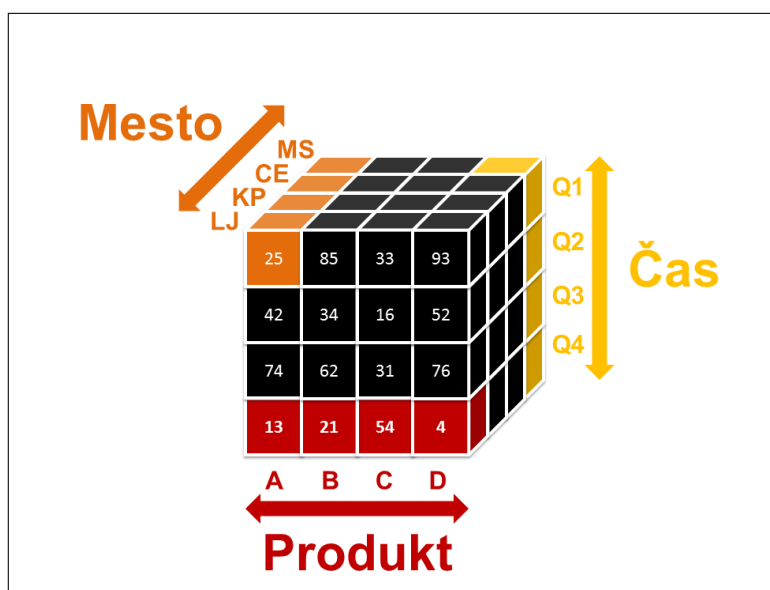
Slika 4.1: Arhitektura OLAP.

Najbolj značilna lastnost tehnologije OLAP je možnost hitrega analiziranja podatkov, v katerem so podatki obdelani in prikazani kot del večdimenzionalne strukture (celote). Podatkovna baza, konfigurirana za OLAP, uporablja večdimenzionalen podatkovni model, ki omogoča izvajanje zapletenih analitičnih poizvedb. Takemu podatkovnemu modelu pravimo **OLAP kocka**. Arhitektura OLAP je prikazana v Sliki 4.1 [1, 2].

## 4.2 OLAP kocka

OLAP kocka je večdimenzionalna podatkovna baza, optimizirana za aplikacije, ki temeljijo na podatkovnih skladiščih in tehnologiji OLAP. Kocka predstavlja način shranjevanja podatkov v večdimenzionalni obliki, definira

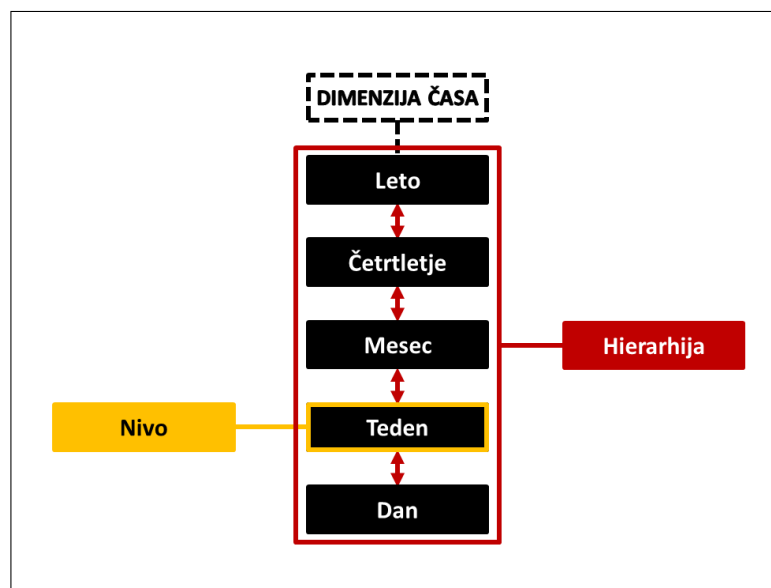
pa jo množica dimenzij in meritev. **Dimenzije** so stranice kocke in predstavljajo kategorije analiziranja, **meritve** oz. merjene vrednosti pa pravimo vrednostim podatkov, ki jih analiziramo. V Sliki 4.2 vidimo primer kocke s tremi dimenzijami in eno meritvijo, t.j. število prodanih produktov [2, 5, 9].



Slika 4.2: Primer OLAP kocke.

Dimenzija (*angl. Dimension*) je strukturna lastnost kocke, predstavljena kot dimenzijska tabela. Ta vsebuje podatke, ki se ujemaajo po tipu in vsebini. Na primer, vsi *dnevi, tedni, meseci, četrletja itd.*, tvorijo časovno dimenzijo (Slika 4.2). Dimenzija v večdimenzionalni strukturi nastopa kot indeks za identifikacijo vrednosti podatkov. Dimenzijo definirajo **atributi**, kot so na primer *ime produkta, tip produkta, kratica mesta itd.* Dimenzije nudijo jedrnat in intuitiven način organiziranja in izbiranja podatkov za potrebe analiziranja in poročanja [2, 5, 9].

Za dimenzije pravimo, da so hierarhično strukturirane, saj so večinoma sestavljene iz več nivojev. To nam omogoča, da se znotraj dimenzije “spuščamo” in “dvigamo” oz. si dimenzijo ogledujemo na različnih nivojih. Dimenzije so lahko **privatne**, uporabne znotraj samo ene kocke, ali **deljene**,



Slika 4.3: Primer časovne dimenzije.

deljene s strani več različnih kock [2, 5, 9].

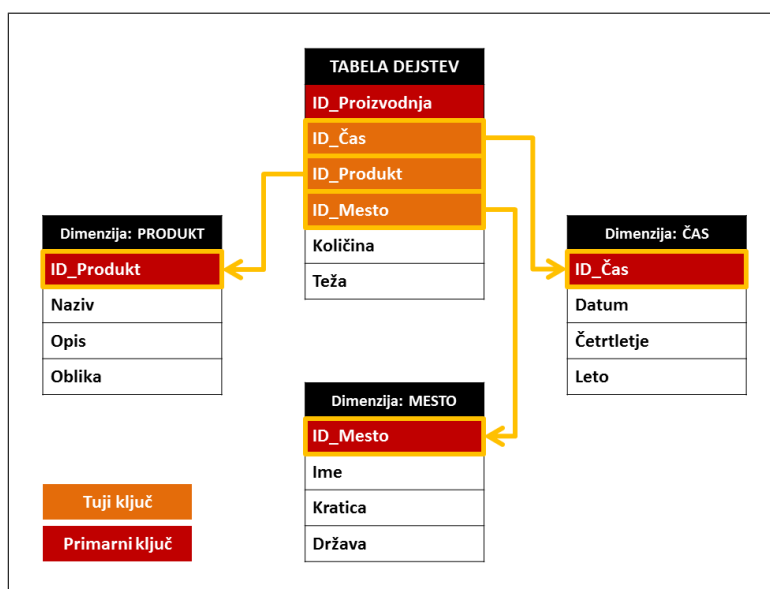
Meritve oz. merjene vrednosti (*angl. Facts, Measures*) predstavljajo podatke, ki jih analiziramo z orodji OLAP. So vrednosti atributov, ki definirajo dimenzijo. Če za primer vzamemo prej omenjeno *Časovno dimenzijo* in atribut *Leto*, so meritve različne letnice (2008, 2009, 2010...). Poznamo dve vrsti meritev [2, 5, 9]:

- **Shranjene meritve**, za katere nam že ime pove, da so shranjene v kocki in se kot take ne spreminjajo.
- **Izračunane meritve**, za katere so definirane funkcije, ki določajo kako se izračunajo oz. agregirajo iz shranjenih meritev (npr. iz nižjega nivoja dimenzije). Funkcije se izvajajo ob prikazovanju različnih poročil ali poizvedovanjih.

**Schema kocke** je večdimenzionalen (relacijski) model, ki je zgrajen iz množice povezanih tabel v podatkovnem skladišču. Kocka je vedno sestavljena iz tabele dejstev in najmanj ene dimenzijske tabele. Najbolj znani

obliki sta zvezdna shema in snežinkasta shema [2, 5].

**Zvezdna shema** (Slika 4.4) je oblika, ki se najbolj uporablja v sprotnem analitičnem procesiranju. V sredini sheme se nahaja tabela dejstev (*angl. Fact table*), okoli nje pa dimenzijske tabele. Te vsebujejo podatke o dimenzijah, medtem ko tabela dejstev vsebuje podatke o merjenih vrednostih in ključe dimenzijskih tabel [2, 5].



Slika 4.4: Primer zvezdne sheme.

**Snežinkasta shema** je izpeljanka zvezdne sheme. Tabela dejstev se uporablja na enak način, dimenzijske tabele pa so normalizirane. To pomeni, da se ponavljajoči podatki prenesejo v dodatne tabele, ki so z dimenzijsko povezane preko ključa. Ta oblika je za obdelovanje kompleksnejša, poizvedbe podaljšujejo čas izvedbe in zmanjšujejo učinkovitost orodij OLAP. Prednost snežinkaste sheme se pokaže pri zelo obsežnih sistemih, kjer je potrebno paziti na velikost podatkovnih skladišč [2, 5].

### 4.3 Operacije v OLAP-u

Orodja OLAP so večinoma narejena tako, da jih lahko uporabljajo tudi uporabniki, ki nimajo strokovnega predznanja. Za izdelavo prvega poročila z orodji OLAP morajo poznati le najosnovnejše operacije, ki služijo pridobivanju zelenih informacij za potrebe analiziranja [3]:

- **Vrtanje v globino** (*angl. Drill down*) nam omogoča “spuščanje” po hierarhiji dimenzije. Na primer iz nivoja *Leto* v nivo *Mesec* (Slika 4.4).
- **Zvijanje navzgor** (*angl. Drill up*) je operacija, ki deluje podobno kot vrtanje v globino, le da deluje v obratno smer. Kadar z “vrtanjem” zaidemo pregloboko nam omogoča “vzpenjanje” po hierarhiji dimenzije, hkrati pa združuje podatke na podlagi funkcije, definirane v višjem nivoju (izračunane meritve).
- **Rezanje** (*angl. Slice and Dice*) je zelo uporabna operacija sestavljena iz dveh delov. “Slice” nam omogoča pogled na dimenzijo kocke (npr. Produkt), “dice” pa nam omogoča, da tej dimenziji dodamo še eno dimenzijo (npr. Čas). Na tak način kreiramo dvodimenzionalen pogled (prerez) kocke.
- **Vrtenje** (*angl. Pivoting*) je operacija, ki jo uporabljamo za prikazovanje poročila. Z njo obračamo podatke do te mere, da nam bo poročilo ustrezalo. Operacija vrtenje ne spreminja podatkov po vsebini. Za primer vzemimo dimenzije iz Slike 4.4 in si zamislimo tabelo s produkti v vrsticah in mesti v stolpcih (prikaz produktov po mestih), ki bi jo z operacijo vrtenja obrnili tako, da bi v tabeli videli prikaz produktov po letih.
- **Vrtanje skozi** (*angl. Drill through*) je operacija, ki je predvsem namenjena za pridobivanje podrobnih podatkov. To pomeni, da vrtamo po hierarhiji tako dolgo, dokler ne pridemo do podatkov, ki predstavljajo eno vrstico v tabeli dejstev. Operacija vrtenje skozi pride prav, ko naletimo na podatek, ki je “sumljiv” oz. močno izstopa od celote.

- **Primerjanje, izračuni in agregiranje** (*angl. Comparing, Calculating, Aggregating*) so tri ločene operacije, ki velikokrat delujejo v povezavi. Operacija primerjanja nam omogoča, da podatke primerjamo po različnih dimenzijah (največkrat po časovni dimenziji). Pri zahtevnih analizah, ko želimo prikazati kar se da veliko informativnih in izpeljanih podatkov, nam prav prideta operaciji izračunavanja in agregiranja podatkov. Z njima lahko poljubno preoblikujemo oz. preračunavamo že pridobljene podatke (npr. prikaz izdelanih produktov po letih z dodanim stolpcem, ki vsebuje primerjavo v odstotkih).

Glavni namen opisanih operacij je, da z njihovo pomočjo hitro ugotovimo, kako uspešno (ali neuspešno) je naše poslovanje. Predvsem z njihovo pomočjo identificiramo nenavadna odstopanja podatkov od celote in na podlagi rezultatov usmerjamo pot nadaljnjega poslovanja. Platforma Decisyon 4 nam omogoča uporabo vseh naštetih operacij.

## 4.4 Kategorije orodij OLAP

Orodja OLAP so kategorizirana glede na arhitekturo shranjevanja in obdelovanja večdimenzionalnih podatkov. Poznamo štiri glavne kategorije orodij OLAP [1, 9]:

- **MOLAP** (*angl. Multi-dimensional OLAP*) ali večdimenzionalni OLAP je sistem, ki podatke hrani v večdimenzionalni strukturi, OLAP kocki. Polni jih s podatki iz transakcijskih baz ali iz podatkovnih skladišč. Za potrebe analiz in poročanj tako ne potrebuje več različnih podatkovnih virov, temveč samo kocke, ki jih je kreiral. MOLAP sistemi omogočajo hitro izvajanje zahtevnejših poizvedb.
- **ROLAP** (*angl. Relational OLAP*) ali relacijski OLAP je sistem, ki temelji na relacijski podatkovni bazi. Pri tako zasnovanem sistemu ni potrebno izdelovati podatkovnega skladišča, saj se podatki lahko

hranijo v transakcijskih (“živih”) podatkovnih bazah. Za uspešno delovanje potrebujejo orodja ROLAP dobro (t.j. drago) strojno opremo, ki je sposobna graditi virtualne večdimenzionalne podatkovne modele in hitro procesirati zahtevne poizvedbe.

- **HOLAP** (*angl. Hybrid OLAP*) ali hibridni OLAP (tudi HROLAP) je kombinacija obeh zgoraj omenjenih sistemov. Shranjene meritve hrani v relacijski podatkovni bazi (ROLAP), izračunane meritve pa hrani v večdimenzionalni podatkovni bazi (MOLAP).
- **DOLAP** (*angl. Desktop OLAP*) ali “namizni” OLAP je sistem, ki podatke hrani na uporabnikovem računalniku. Primeren je predvsem za analize manjšega obsega in zahtevnosti. Imajo manj funkcionalnosti kot ostali sistemi OLAP, vendar so zaradi svoje cene in enostavnosti zelo priljubljeni pri manjših podjetjih.

Kakšen tip orodja izbrati je odvisno predvsem od kompleksnosti poslovnega sistema. V velikih poslovnih sistemih posegajo za HOLAP orodji, ki zagotavljajo hitro izvajanje, hkrati pa z uporabo podatkovnih skladišč ne obremenjujejo transakcijske podatkovne baze. Za manjše sisteme sta primerna MOLAP, kjer želijo hitro izvajanje in ROLAP, kjer transakcijska baza ni močno obremenjena. Za podjetja z majhno količino podatkov in brez potrebe po kompleksnih poročanjih je najprimernejše orodje DOLAP.

# Poglavje 5

## Uporabljena orodja

Za doseganje ciljev zastavljenih v tem projektu (oz. diplomski nalogi) je bila v organizaciji potrebna vpeljava novega orodja za podporo odločanju, ki se imenuje **Decisyon**. Transakcijske (oz. operativne) baze delujejo preko sistema za upravljanje z relacijskimi podatkovnimi bazami (RDBMS) **Oracle**. Ob teh dveh ključnih orodjih so bili pri izdelavi diplomske naloge potrebni še PL/SQL Developer, Microsoft Office Excel itd. V nadaljevanju so opisana uporabljena orodja.

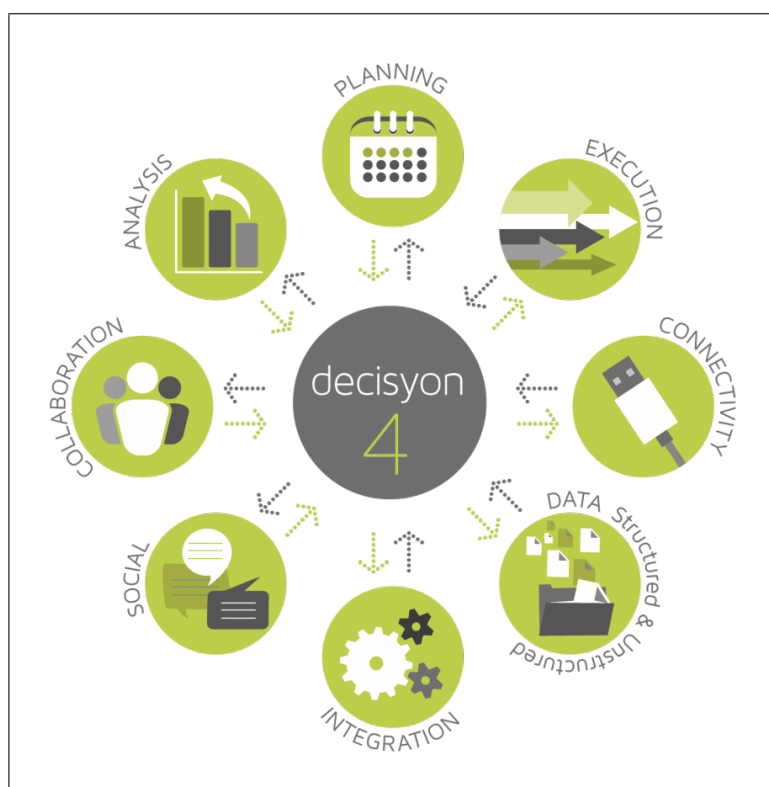
### 5.1 Decisyon 4

Poslovna inteligenca (*angl. BI - Business Intelligence*) je zbirka teorij in metod, ki z zbiranjem, obdelavo in analizo podatkov izboljšujejo procese poslovnega odločanja. Orodja poslovne inteligence (*angl. BI tools*) sestavljajo podatkovna skladišča, procesi za prenos in obdelavo podatkov (ETL), večdimenzionalne kocke (OLAP) in grafični vmesnik za napredni vizualni prikaz podatkov [2].

Decisyon Collaborative Decision Making & Execution je BI platforma, ki vsebuje vse ključne gradnike poslovne inteligence. Funkcije, ki se nahajajo v platformi omogočajo podporo analizi in načrtovanju procesov z uporabo enotnega integriranega sistema (Slika 5.1) [6].

Glede na kontekst nudi platforma tudi aktiviranje specifičnih mehanizmov sodelovanja, s čimer združuje informacije, odločitve in ukrepe [6].

### 5.1.1 Funkcionalnosti platforme Decisyon 4



Slika 5.1: Funkcionalnosti platforme Decisyon 4 [6].

Zaradi edinstvenih funkcij platforme, je mogoče na enostaven, vsestranski in hiter način podpreti procese odločanja usmerjenih v doseganje strateških in operativnih ciljev. Poleg podpore za analizo in načrtovanja procesov je Decisyon 4 sposoben odločitve, sprejete v času analize podatkov in času aktivnosti načrtovanja, propagirati na poslovne procese. Ponujene funkcije je mogoče logično razdeliti v štiri glavne komponente [6]:

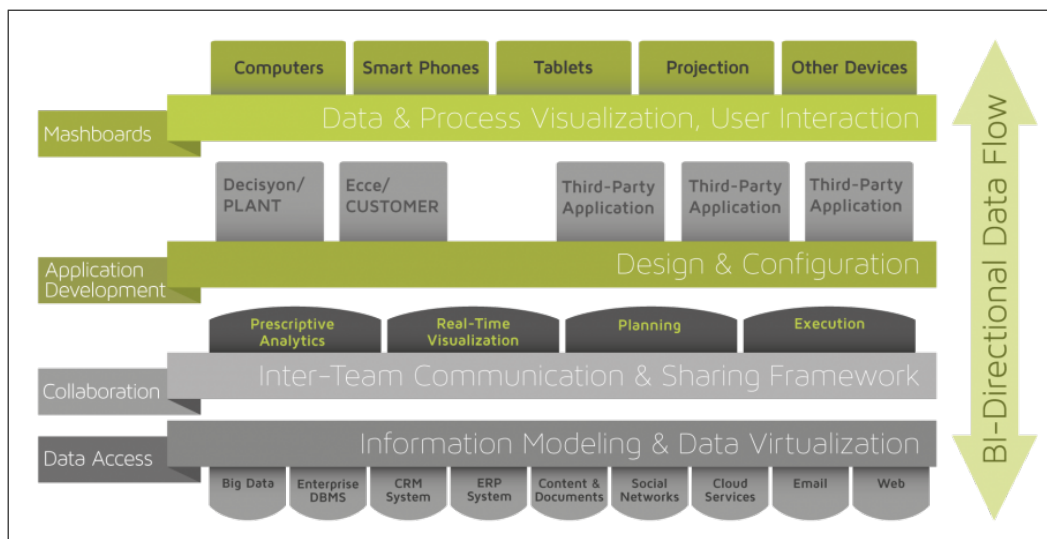
- **Analysis** (slov. analiziranje) - simuliranje alternativnih predpostavk,

vrtanje v globino podatkov, opazovanje podatkov z vseh zornih kotov, dostopnost vsem uporabnikom (tudi tistim brez tehničnega znanja).

- **Planning** (slov. načrtovanje) - strateško in operativno načrtovanje, podpora skoraj vsem možnim metodam poslovnega načrtovanja.
- **Collaboration** (slov. sodelovanje, kolaboracija) - v arhitekturi platforme predstavljeno kot celota, ki nudi različne funkcije za kolaboracijo aktivnosti analiziranja in načrtovanja.
- **Execution** (slov. izvedba) - izvajanje definiranih dogodkov oz. procesov, prikazovanje zelenih poizvedovanj.

### 5.1.2 Arhitektura platforme Decision 4

Arhitektura platforme Decision 4 je razdeljena na štiri večje celote, ki obsegajo vse ponujene funkcionalnosti. V Sliki 5.2 lahko vidimo kako si celote sledijo, v nadaljevanju pa so predstavljene njihove ključne sposobnosti oz. zmožnosti [6].



Slika 5.2: Arhitektura platforme Decision 4 [6].

- **Data Access** (slov. dostop do podatkov) - dostop do relevantnih podatkov in storitev (*angl. Services*), zmožljiv dostop do podatkovne baze s pomočjo HOLAP tehnologije, povezava s storitvami v oblaku preko odprtega vmesnika za programiranje aplikacij (*angl. API - Application Programming Interface*).
- **Collaboration** (slov. sodelovanje, kolaboracija) - usmeritev sodelovanja okoli informacij in procesov (ne samo ljudi), analiziranje in posredovanje rezultatov v sodelovalnem okolju, povezovanje relevantnih informacij in diskusij s specifičnimi odločitvami (in te k ukrepom), oskrbeti uporabnike s primernimi orodji za sodelovanje (opombe, obvestila, sporočila, izmenjava dokumentov, upravljanje nalog itd.).
- **Application Development** (slov. razvoj aplikacij) - s pomočjo prej omenjenih funkcionalnosti (analiziranje, načrtovanje in izvedba) nudi podporo pri kreiranju in oblikovanju poslovnih procesov.
- **Mashboards** (slov. "namizna plošča") - prikazovanje kompleksnih informacij (poročanj), povezav med podatki in statusov različnih procesov s pomočjo naprednega grafičnega vmesnika.

### 5.1.3 Orodja v platformi Decisyon 4

Namen platforme Decisyon 4 je, da organizaciji nudi možnost implementiranja novih rešitev na podlagi dejstev oz. informacij, ki jih prikaže na zeleni način. Za pripravo poročanj ali drugih poslovnih rešitev potrebujemo različna orodja, ki združujejo platformo Decisyon 4 [6]:

- **Decisyon Power Designer** - namenjen razvijalcem in naprednim uporabnikom, ki skrbijo za odkrivanje in administriranje poslovno inteligenčnih rešitev (*angl. Business Intelligence Solutions*). Orodje Decisyon Power Designer nam omogoča upravljanje sistema (povezovanje z zunanji podatkovnimi viri), načrtovanje logičnega modela (kreiranje in povezovanje meritev, dimenzij, hierarhij in večdimenzionalnih

kock), pripravo poročil za poslovno inteligenčne analize (kreiranje metrik, filtrov, poročil, indikatorjev itd.), izdelavo sodelovalnih procesov (pošiljanje nalog, sporočil in opozoril uporabnikom) in razvijanje izvajalnih modulov (kreiranje aplikacij).

- **Decisyon WEB** - z enostavnim (in prijaznim) uporabniškim vmesnikom namenjen predvsem končnim uporabnikom, ki prejemaajo objavljena poročanja. Orodje Decisyon WEB nam omogoča kreiranje poročil in uporabo sodelovalnih funkcij, kot so sporočila, komentarji, pošiljanje nalog itd.
- **Decisyon Mobile** - končnemu uporabniku omogoča dostop do podatkov z brezžičnimi napravami.
- **Decisyon Metadata** - podatkovna baza, ki hrani sistemske metapodatke.

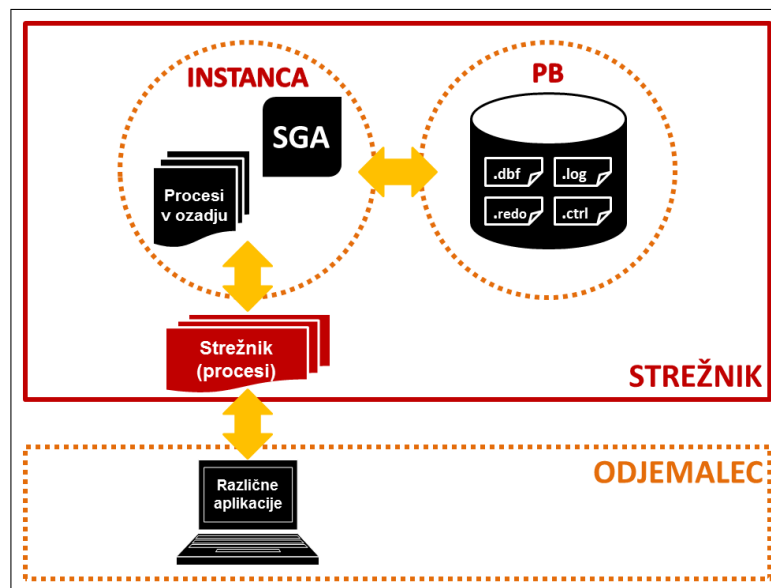
Za pripravo poročil smo v našem primeru uporabili orodja Decisyon Power Designer in Decisyon WEB. Praktični primeri so predstavljeni v naslednjem (6.) Poglavlju.

## 5.2 Oracle

Podatkovna baza Oracle je sistem za upravljanje z relacijsko podatkovno bazo (Oracle RDBMS) razvit s strani organizacije Oracle Corporation. Leta 1977 je Larry Ellison s kolegi razvijalci predstavil prvo različico orodja Oracle, ki je danes eno najbolj pogostih in najbolj uporabljenih sistemov za upravljanje z relacijsko podatkovno bazo. Okolje Oracle je sestavljeno iz podatkovne baze in instance (Slika 5.3) [7, 8].

Podatkovna baza vsebuje fizične podatkovne datoteke, kontrolne datoteke in dnevniške (*angl. Redo log*) datoteke. Podatkovne datoteke hranijo dejanske podatke in indekse. Dnevniške datoteke hranijo vse zapise o spremembah, ki so bile storjene v preteklosti. Kontrolne datoteke vsebujejo pomembne

metapodatke Oraclea, ki vsebujejo vse informacije potrebne za dostop do podatkovne baze s strani instance [7, 8].



Slika 5.3: Arhitektura okolja Oracle.

Instanca je kombinacija nabora fizičnega pomnilnika (*angl. RAM - Random Access Memory*) dodeljena Oracleu in procesov, ki jih Oracle kreira v ozadju za porabo tega pomnilniškega prostora. Temu pomnilniku v Oracleu pravimo SGA (*angl. System Global Area*) in vsebuje vse informacije potrebne za delovanje instance. SGA je torej del fizičnega pomnilnika, ki si ga delijo vsi procesi znotraj ene instance in se instanci dodeli ob zagonu podatkovne baze. Ko podatkovno bazo ugasnemo se SGA prostor sprosti [7, 8].

Ključna značilnost Oraclea je, da je njegova arhitektura razdeljena na logični in fizični koncept hranjenja podatkov. Ta struktura pomeni, da je za obsežno porazdeljeno (mrežno) procesiranje, znano tudi kot “Grid Computing”, lokacija podatkov nepomembna ter transparentna za uporabnika, tako da je možno dodajati in urejati fizične strukture, ne da bi to vplivalo na delovanje podatkovne baze, podatkov v njej ali njenih uporabnikov [7, 8].

Grid computing je tehnologija, ki za zahtevne, obsežne računalniške ob-

---

delave uporablja porazdeljeno informacijsko infrastrukturo. Informacijska struktura je v tem primeru omrežje, v katerem si več računalnikov deli ne-uporabljeno moč procesorjev in/ali prostor na diskih. Na ta način se lahko prilagaja zmogljivost sistema brez zmanjševanja funkcionalnosti (storitev). Okvara na posameznem računalniku v omrežju ne ogroža delovanja celotne podatkovne baze zato je tak sistem trden [7, 8].

Skozi leta so se razvijale dodatne funkcionalnosti in s tem nove verzije prvotnega sistema Oracle, aktualna pa je verzija **Oracle Database 11g**, ki jo uporabljamo tudi v našem podjetju [7, 8].



## Poglavje 6

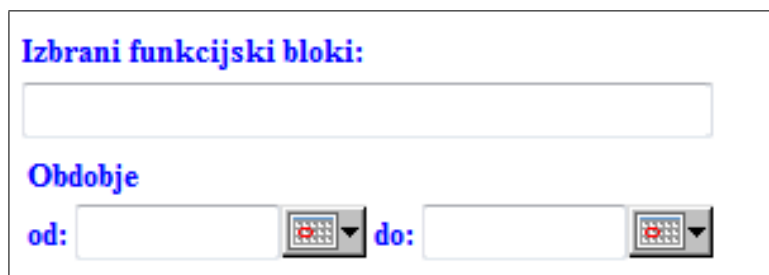
# Prenova sistema za podporo odločanju v izbranem podjetju

V podjetju se za podporo odločanju uporablja veliko število različnih poročanj. Nekatera so namenjena napovedovanju bodočih rezultatov, druga pa pregledovanju preteklih realizacij. Na podlagi pridobljenih informacij podjetje lažje sprejema pomembne poslovne odločitve (nadgradnje sistemov, večanje delovne sile ipd.). Ker so nekatera trenutna poročanja relativno zastarela in počasna, so se v podjetju odločili za nakup novega orodja, ki služi podpori poslovnemu odločanju. Želja je tudi, da bi odkrili rezultate, ki jih stari poročevalni sistem niso bili zmožni prikazovati. Ideja in cilj je seveda čim hitrejši prenos vseh obstoječih poročanj na novi enoten sistem, za kar je potrebno analizirati obstoječa poročanja, pripraviti načrt za njihov prenos ter ta načrt tudi realizirati. Delo je v nadaljevanju razčlenjeno na ta tri ključna podpoglavja vezana na poročanja, ki so zajeta v sklopu projekta (diplomska naloga).

### 6.1 Analiza izbranih obstoječih poročanj

Ključ analize je priprava podatkov za načrtovanje OLAP kocke in prenosa na nov sistem. Obstoječa poročanja so že nekoliko zastarela in potrebujejo

veliko časa za svoje izvajanje. Trenutno delujejo na principu poseganja v transakcijsko bazo. Na podlagi izbranih atributov (datum, lokacija ipd.) se nam prikaže zeleno poročanje (Slika 6.1).

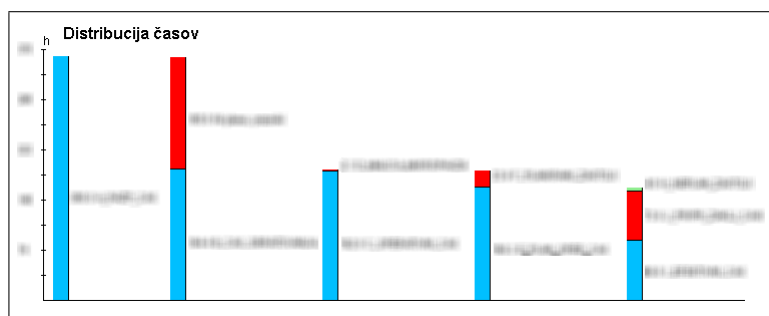


The screenshot shows a web-based form with the following elements:

- A blue heading: **Izbrani funkcijski bloki:**
- A large empty text input field below the heading.
- A blue heading: **Obdobje**
- The label **od:** followed by a date input field and a calendar icon.
- The label **do:** followed by a date input field and a calendar icon.

Slika 6.1: Izbirna polja obstoječega poročevalnega sistema.

Problem pri takem delovanju je, da se ob zagonu poročanja v ozadju zažene veliko število procedur in funkcij, ki podatke preračunavajo ter polnijo prav tako veliko število začasnih tabel iz katerih se ob koncu izriše graf (Slika 6.2). Za primer smo prikazali distribucijo časov posameznih dogodkov glede na attribute izbrane v prejšnjem koraku.



Slika 6.2: Graf izbranega obstoječega poročanja.

Glavni cilj je torej identificiranje tistih tabel v transakcijski bazi, ki bi jih prenašali v podatkovno skladišče. Prav tako je potrebno identificirati dimenzije in meritve, ki bodo predstavljale OLAP kocko. Izbrana poročanja za svoje delovanje skupno uporabljajo 73 procedur. Nekatere procedure se ponavljajo v različnih poročanjih, zato je dejansko število različnih procedur

56. Potrebno je bilo ugotoviti po kakšnem zaporedju se izvajajo in katere tabele ob tem potrebujejo. Procedure večinoma podatke berejo iz transakcijskih tabel, prilagojene oz. normalizirane podatke pa zapisujejo v začasne (*angl. Temporary*) tabele. Tako smo odkrili 18 tabel iz transakcijske baze in 33 začasnih tabel, ki vsebujejo podatke primerne za podatkovno skladišče.

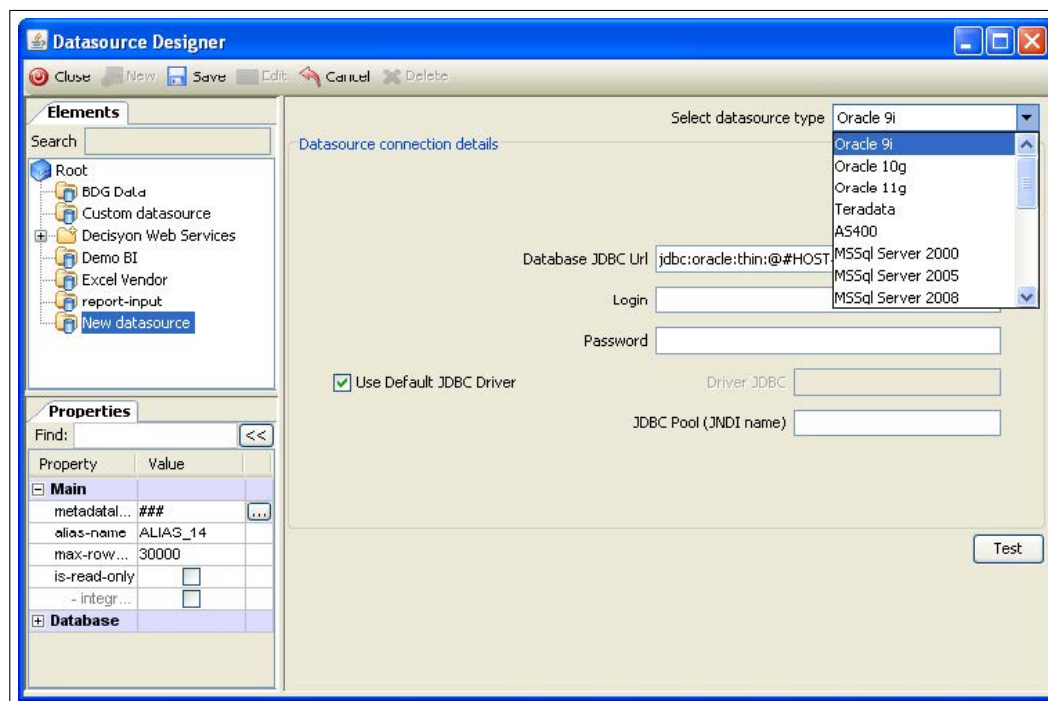
Vsako poročanje ima na koncu tudi svojo funkcijo, ki na podlagi normaliziranih podatkov v začasnih tabelah pripravi datoteko XML (*angl. Extensible Markup Language*). Grafično prikazovanje poročanje se izvaja s pomočjo HTML (*angl. Hyper Text Markup Language*), označevalnega jezika za gradnjo spletnih dokumentov, datoteka XML pa predstavlja njihov vir podatkov.

## 6.2 Prenos podatkov na nov sistem

Po končani analizi smo dobili seznam vseh procedur, ki smo jih morali primerno preurediti in pripraviti za prenos v podatkovno skladišče tako, da bi se sistem v prihodnje čim manj obremenjeval. Na podlagi analiz smo pripravili tabelo, v kateri se zapisani časi za avtomatično izvajanje procedur, ki s podatki polnijo podatkovno skladišče. Prenos podatkov lahko izvajamo na dva načina, z uporabo orodja “**Decisyon Power Designer**” ali pa z uporabo funkcionalnosti Oracle “**DBMS JOB**”.

### 6.2.1 Decisyon Power Designer

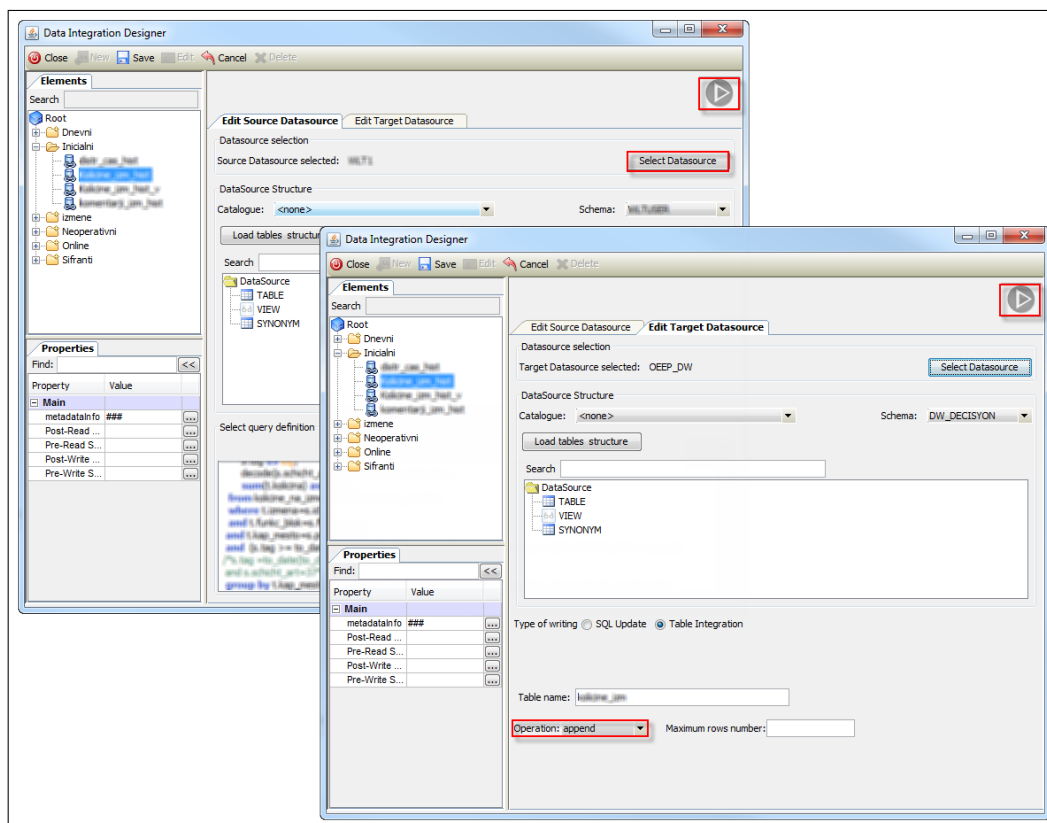
Decisyon PWD je orodje, ki poleg vseh funkcionalnosti omenjenih v 5. Poglavju (upravljanje sistema, načrtovanje logičnega modela, pripravljanje poročil itd.), omogoča tudi prenos podatkov iz transakcijske podatkovne baze v podatkovno skladišče (“ETL” vmesnik). V orodju smo morali pred prenosom najprej nastaviti vir in cilj podatkov preko modula “**Datasource Designer**”. Decisyon se lahko hkrati poveže z velikim številom različnih zunanjih podatkovnih virov (Oracle, MySQL, Microsoft Access, Microsoft Excel, XML itd.). V Sliki 6.3 tako lahko vidimo nastavitve podatkovnega vira, ki je v našem primeru transakcijska podatkovna baza Oracle.



Slika 6.3: Primer nastavitve vira podatkov v orodju Decisyon PWD.

Po uspešno nastavljenem podatkovnem viru in cilju smo s pomočjo modula “Data Integration Designer” nadaljevali z nastavljanjem povezave med njima. Modul je sestavljen iz dveh delov, “Edit Source Datasource” (Slika 6.4), kjer definiramo poizvedbo za podatke, ki jih želimo prenašati in “Edit Target Datasource” (Slika 6.4), kjer nastavimo ciljno tabelo, v katero se bodo shranjevali želeni podatki.

Preko gumba “Select Datasource” se nam prikaže seznam vseh virov, ki smo jih kreirali z modulom “Datasource Designer”. Po izbranem viru se nam prikažejo tabele in pogledi (*angl. View*) določene sheme (*angl. Schema*). V polje “Select query definition” vpišemo poizvedbo, ki nam bo prikazala podatke, katere želimo prenašati. Na drugem zavihku zatem izberemo enega od tipov operacij, ki so na voljo (**create** - kreira novo tabelo, **drop\_create** - izbriše tabelo z enakim imenom in nato zažene operacijo “create”, **append** - doda podatke k obstoječi tabeli in **delete\_append** - izbriše podatke v ciljni



Slika 6.4: Data Integration Designer - Edit Source/Target Datasource.

tabeli in nato zažene operacijo “append”).

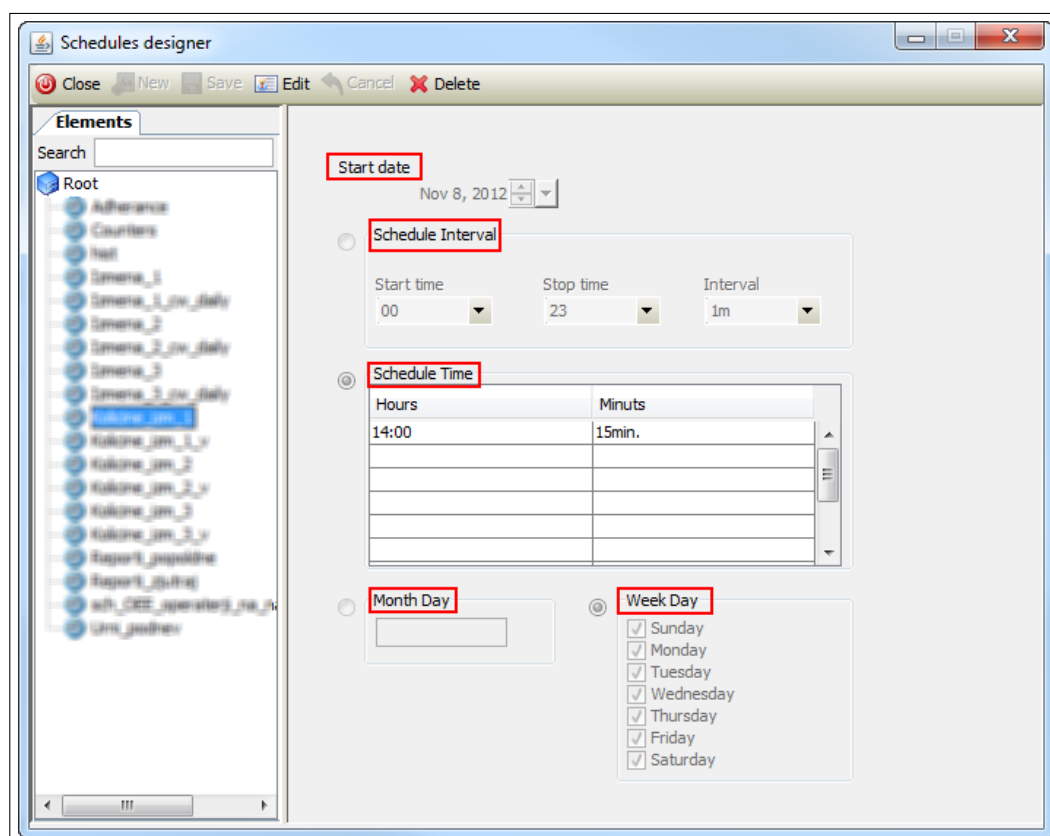
Ko zaključimo z vsemi potrebnimi nastavitvami lahko proces “Data Integration” zaženemo neposredno s klikom na gumb v desnem zgornjem kotu ali pa ga dodamo na seznam (*angl. Schedule*) načrtovanih procesov z uporabo modula “Refresh Designer”.

Da bi lahko uspešno uporabljali “Refresh Designer” moramo pripraviti seznam načrtovanih procesov (razpored), zato potrebujemo modul “Schedules Designer” (Slika 6.5). Ta nam omogoča:

- nastavitev začetnega datuma (*angl. Start date*),
- nastavitev nadaljnjega razporeda izvajanja procesa, kjer lahko nastavimo interval (*angl. Schedule Interval*) ali pa posamezne čase izvajanja

(*angl. Schedule Time*) in

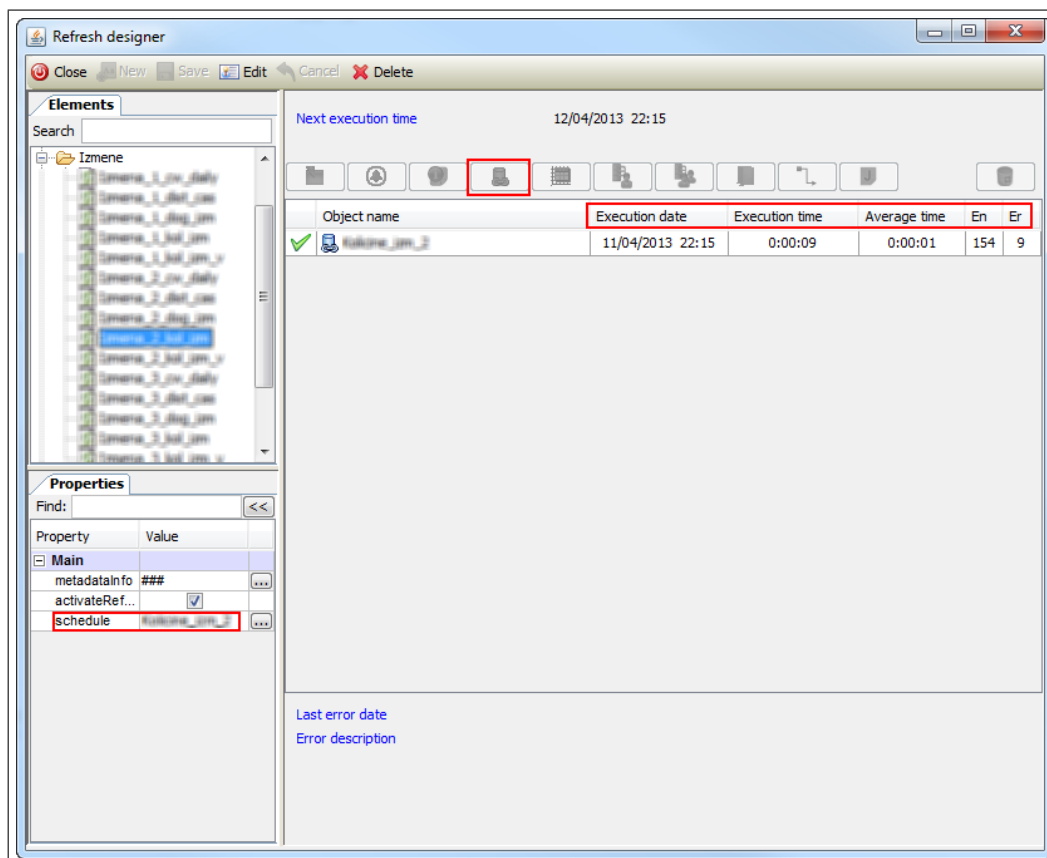
- nastavitve izvajanja na določen dan v mesecu (*angl. Month Day*) ali tednu (*angl. Week Day*).



Slika 6.5: Modul “Schedules Designer”.

“Refresh Designer” nam omogoča izvajanje različnih procesov po enem od razporedov pripravljenih v modulu “Schedules Designer”. Za nas je predvsem pomemben proces s katerim prenašamo podatke iz transakcijske podatkovne baze v podatkovno skladišče. Ob kreiranju novega smo zato izbrali shranjeno proceduro, katero smo jo kreirali z modulom “Data Integration Designer” in ji določili raspored, ki smo ga pripravili v modulu “Schedules Designer”. V Sliki 6.6 vidimo, da lahko v modulu “Refresh Designer” tudi nadzorujemo

izvajanje posamezne procedure (uspešnost izvedbe, datum zadnje izvedbe, čas zadnjega izvajanja, čas povprečnega izvajanja itd.).



Slika 6.6: Modul “Refresh Designer”.

Med ostale procese, ki jih lahko izvajamo s tem modulom spadajo tudi poročanja (*angl. Report*), različna opozorila (*angl. Alarms, Notifications*), zaganjanje različnih programov (*angl. Program Execution*) itd.

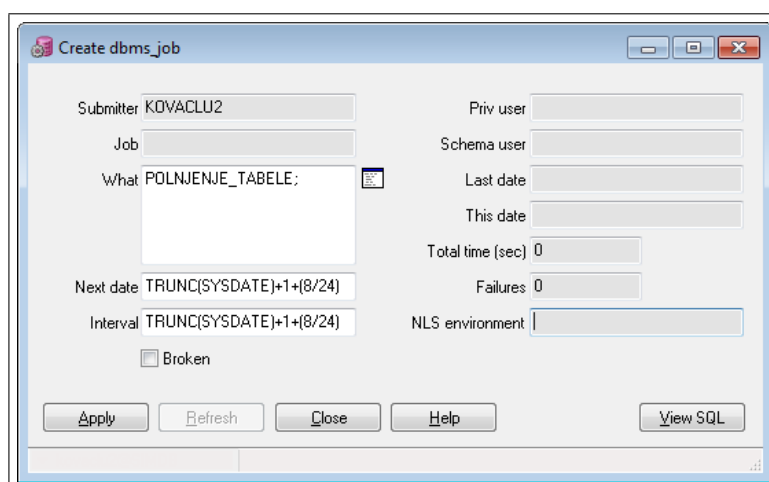
V začetku podpoglavja smo že omenili, da lahko prenos podatkov izvajamo na dva načina, prvi je z uporabo orodja “**Decisyon Power Designer**”, drugi pa z uporabo **Oracle “DBMS JOB”**. Slednji je primeren predvsem takrat, ko ne potrebujemo normaliziranih podatkov, saj je celoten proces hitrejši.

### 6.2.2 Oracle “DBMS JOB”

“DBMS JOB” je paket v orodju Oracle, ki nudi zbirko “planiranih” oz. “načrtovanih” (*angl. Scheduling*) funkcij, s katerimi lahko sprožamo procedure ob določenih intervalih, podobno kot smo to storili z orodjem “Decision Power Designer”. “DBMS JOB” lahko kreiramo s PL/SQL stavkom, kjer nastavimo katera procedura naj se zažene, kdaj naj se prvič zažene in kakšen naj bo nadaljnji interval. V proceduri mora biti med drugim zapisano iz katere tabele (vir) se podatki “črpajo” in v katero tabelo (cilj) se podatki “prenašajo”. V Sliki 6.7 imamo primer takega “DBMS JOB” stavka.

```
begin
  sys.dbms_job.submit(
    job      => :job,
    what     => 'POLNJENJE_TABELE;',
    next_date => 'TRUNC(SYSDATE)+1+(8/24)',
    interval => 'TRUNC(SYSDATE)+1+(8/24)';
  commit;
end;
/
```

Slika 6.7: Create dbms\_job - PL/SQL stavek.



Slika 6.8: Create dbms\_job - PL/SQL čarovnik.

“DBMS JOB” lahko namesto PL/SQL stavka kreiramo tudi s funkcionalnostjo orodja PL/SQL, ki se imenuje “Create dbms.job”. Ko v orodju izberemo to možnost, se nam prikaže novo okno (Slika 6.8), kjer nastavimo vse potrebne attribute.

## 6.3 Priprava poročila v orodju Decision

Po prenosu podatkov smo morali v orodju Decision kreirati dimenzije, OLAP kocke in ustrezne meritve. Po vsem tem smo lahko pripravili ustrezno poročilo. V nadaljevanju so razloženi postopki kako v orodju kreirati vse prej omenjene gradnike za pripravo poročila.

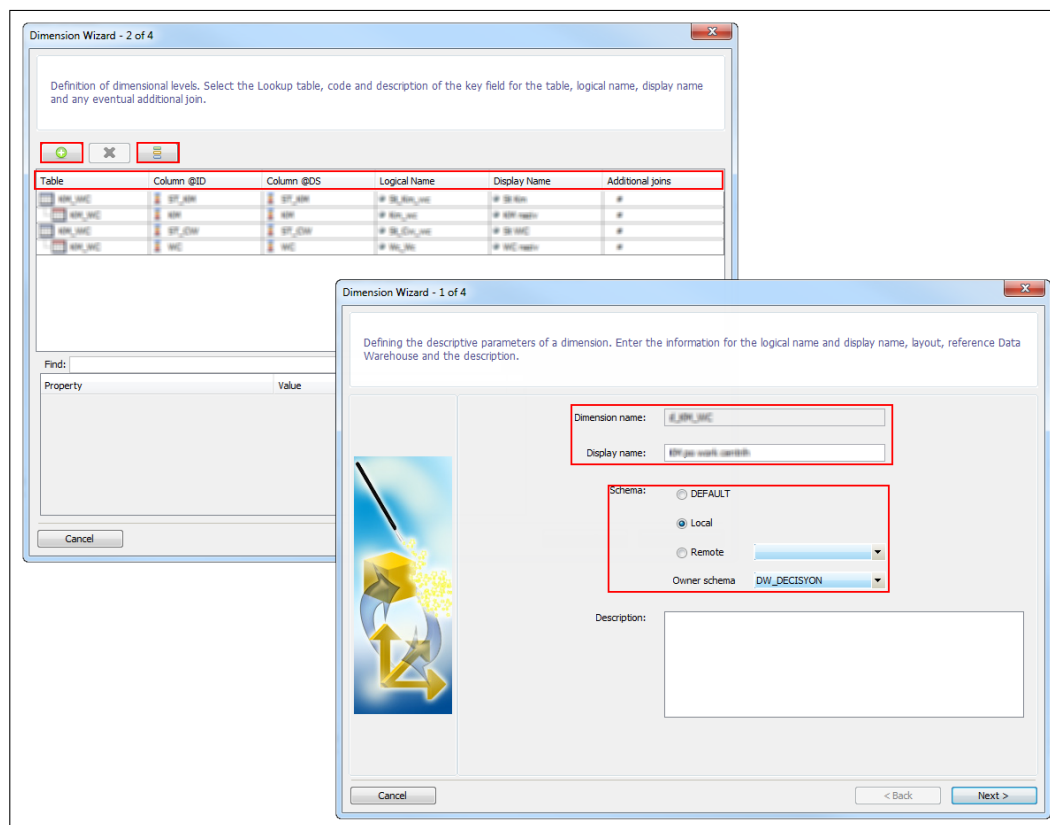
### 6.3.1 Dimenzije

Dimenzije smo teoretično predstavili v Podpoglavju 4.2, v tem pa bomo prikazali, kako jih kreiramo v orodju “Decision Power Designer”. Dimenzije so ključne pri sprotnem analitičnem obdelovanju podatkov, saj brez njih ne moremo definirati OLAP kock. Za kreiranje dimenzij uporabimo modul “Dimension Wizard”, ki je sestavljen iz štirih korakov.

V prvem koraku opišemo dimenzijo in izberemo podatkovni vir. Izbira “Local” nam izbere tisti podatkovni vir na katerega smo trenutno povezani z orodjem “Decision Power Designer” (Slika 6.9).

Nato kreiramo nivoje dimenzije in jih logično povežemo s tabelami v podatkovnem skladišču (Slika 6.9). Ena vrstica predstavlja posamezen nivo, stolpci pa predstavljajo lastnosti oz. nastavitve tega nivoja. “Table” tako predstavlja tabelo, kjer se nahajata ključ in ostali atributi tabele, “Column @ID” je rezerviran za nastavitve primarnega ključa, ki ga potrebujemo za hierarhično povezovanje dimenzij (4. korak), “Column @DS” uporabljamo za opis nivoja, “Logical Name” je unikatno ime nivoja, ki se ne sme ponavljati znotraj podatkovnega modela ter “Display Name”, v katerem je ime nivoja, ki ga želimo uporabljati v poročilih.

Vsakemu posameznemu nivoju lahko dodelimo attribute, ki se bodo v



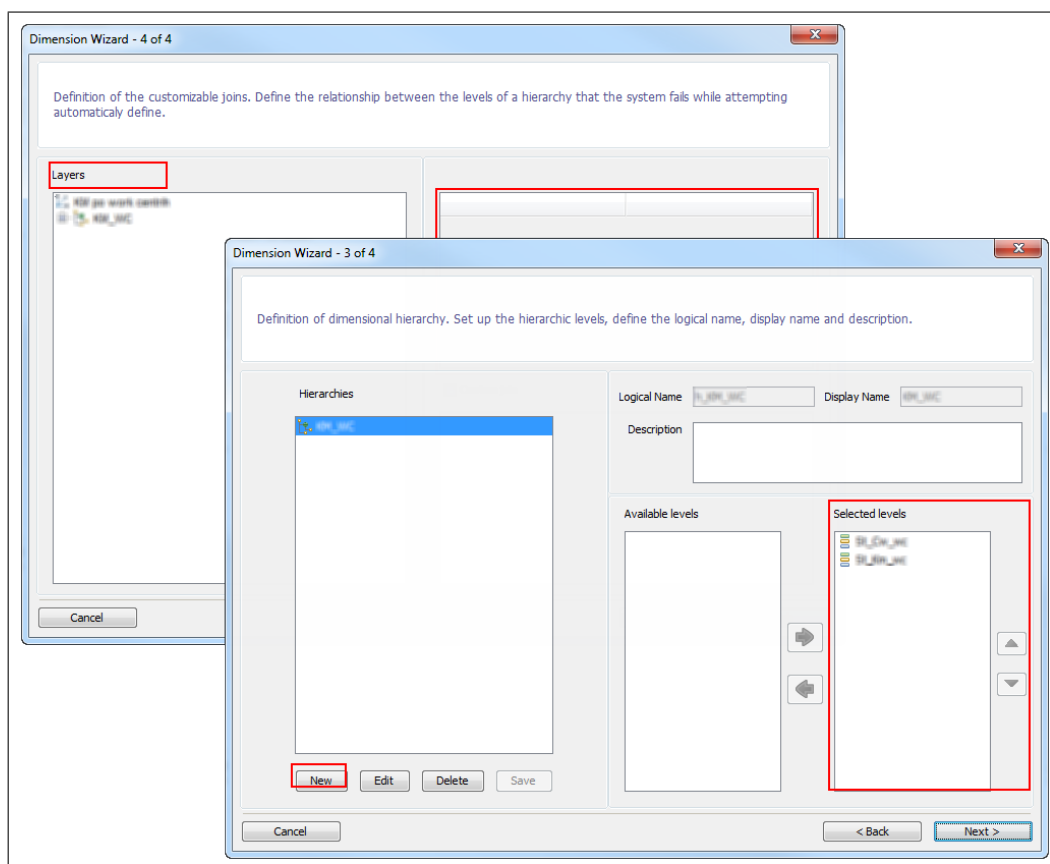
Slika 6.9: Modul “Dimension Wizard” - 1. in 2. korak.

poročanju uporabljali podobno kot nivoji, vendar nimajo vpliva ob definiranju hierarhije, kar nam olajšala samo izgradnjo dimenzije. Za primer bi lahko vzeli nivo “Oseba”, ki bi ji dodelili attribute “Ime”, “Priimek”, “Naslov”, medtem ko bi bil primarni ključ nivoja “Oseba” unikaten identifikator (npr. EMŠO) zapisan v polju “Column @ID”.

V tretjem koraku sledi definiranje hierarhij dimenzije. S klikom na gumb “New” se nam odprejo polja za kreiranje nove hierarhije v katero prenašamo posamezne nivoje, ki smo jih kreirali v prejšnjem koraku. Nivoji se med seboj povezujejo preko ključev definiranih v polju “Column @ID”.

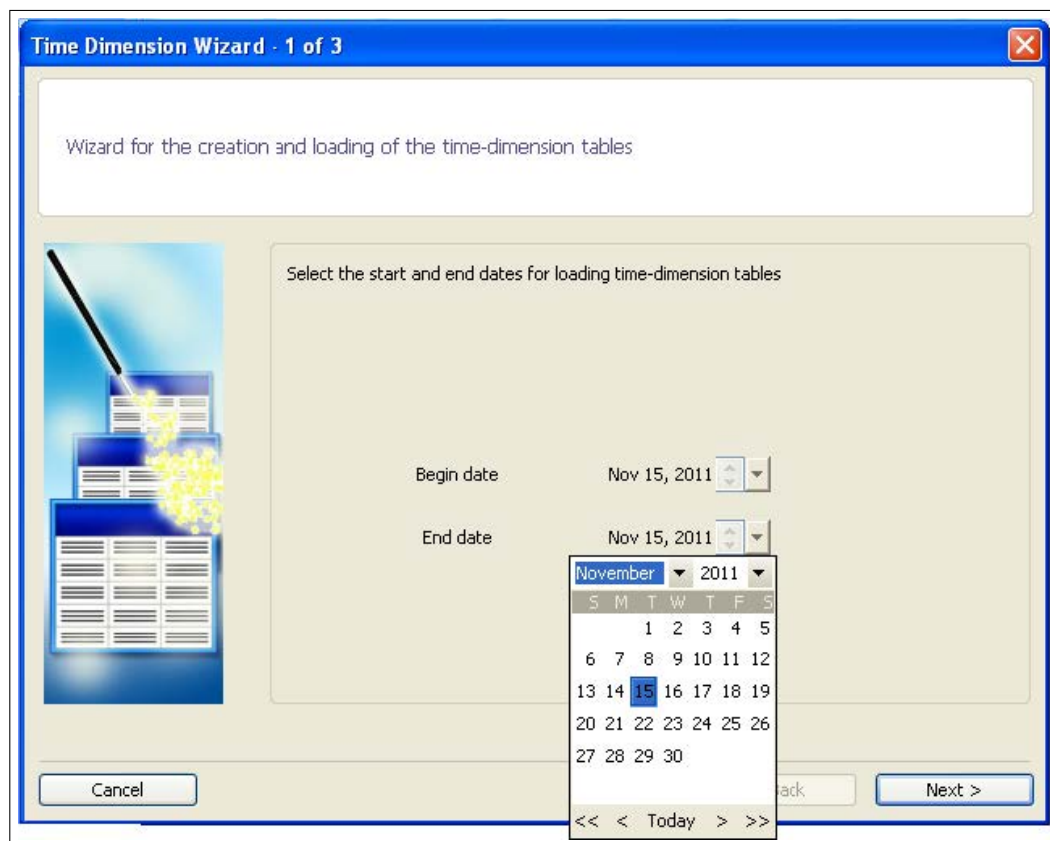
Kadar imamo dimenzijo narejeno na različnih tabelah moramo v zadnjem koraku ročno definirati povezave med nivoji. Na levi strani okna se nam v “drevesni strukturi” izrišejo hierarhije, dimenzije in nivoji, kreirani

v prejšnjih korakih, medtem ko se nam na desni prikažejo primarni ključni povezanih nivojev. Mi smo vse dimenzije kreirali na posameznih tabelah, zato smo lahko ta korak preskočili in s klikom na gumb “Finish” zaključili s procesom kreiranja dimenzije. Tretji in četrti korak kreiranja dimenzije sta prikazana v Sliki 6.10.



Slika 6.10: Modul “Dimension Wizard” - 3. in 4. korak.

Orodje “Decision Power Designer” vsebuje tudi modul “Time Dimension Wizard” (Slika 6.11), ki nam precej olajša kreiranje časovne dimenzije. V samem modulu izberemo časovni termin, na podlagi katerega se kreirane tabele napolnijo z ustreznimi datumi. Tako narejeno časovno dimenzijo lahko uporabljamo v vseh kockah, ki so narejene na enakem podatkovnem modelu.



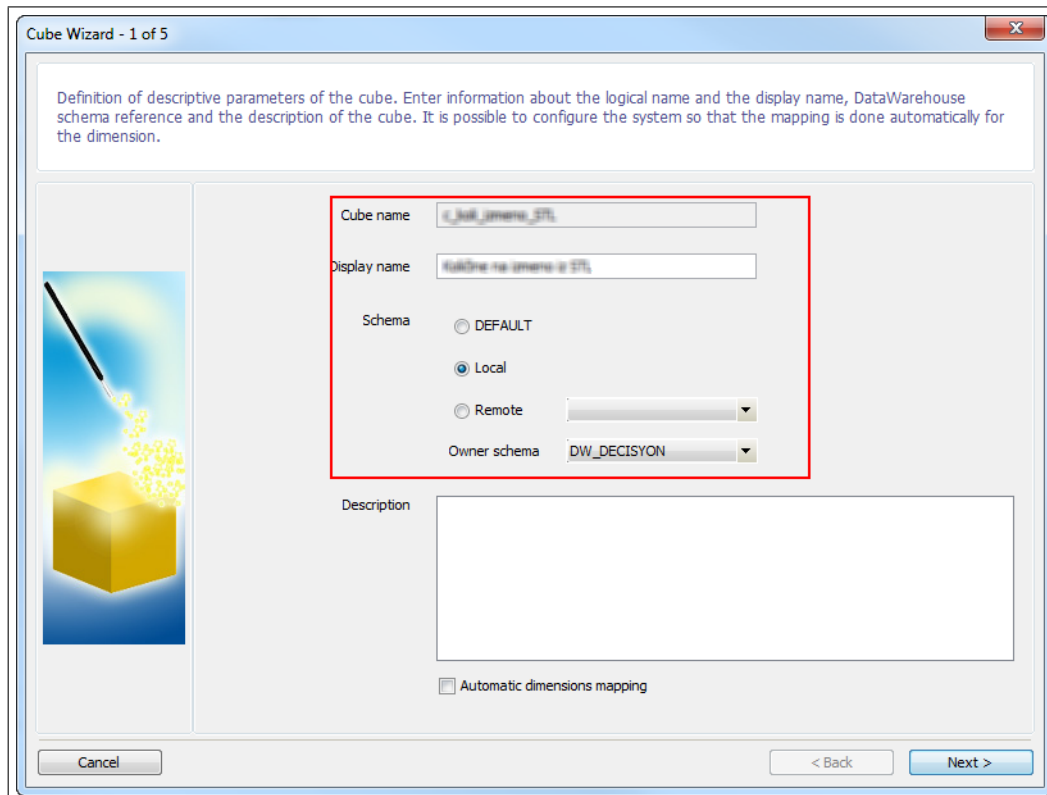
Slika 6.11: Modul “Time Dimension Wizard”.

### 6.3.2 Kocke in meritve

Kot smo že omenili v Podpoglavju 4.2, so OLAP kocke večdimenzionalne strukture, ki jih definirajo množice dimenzij in meritev. Po uspešnem kreiranju dimenzij v orodju “Decision Power Designer” smo lahko nadaljevali s kreiranjem kock s pomočjo novega modula, “Cube Wizard”. Ta je sestavljen iz petih ključnih korakov, po zaključku pa bomo lahko začeli s pripravo poročil. Da bi uspešno kreirali kocko, moramo narediti povezavo s tabelo dejstev v podatkovnem skladišču, izbrati ustrezne dimenzije in definirati meritve.

Tako kot pri ostalih, do sedaj uporabljenih modulih, je tudi v tem potrebno definirati opis (npr. ime kocke) in izbrati podatkovni vir za izdelavo

kocke (Slika 6.12). Polje “Automatic dimension mapping” je dodatna funkcionalnost čarovnika, ki avtomatsko poveže kasneje izbrane dimenzije in tabelo dejstev.



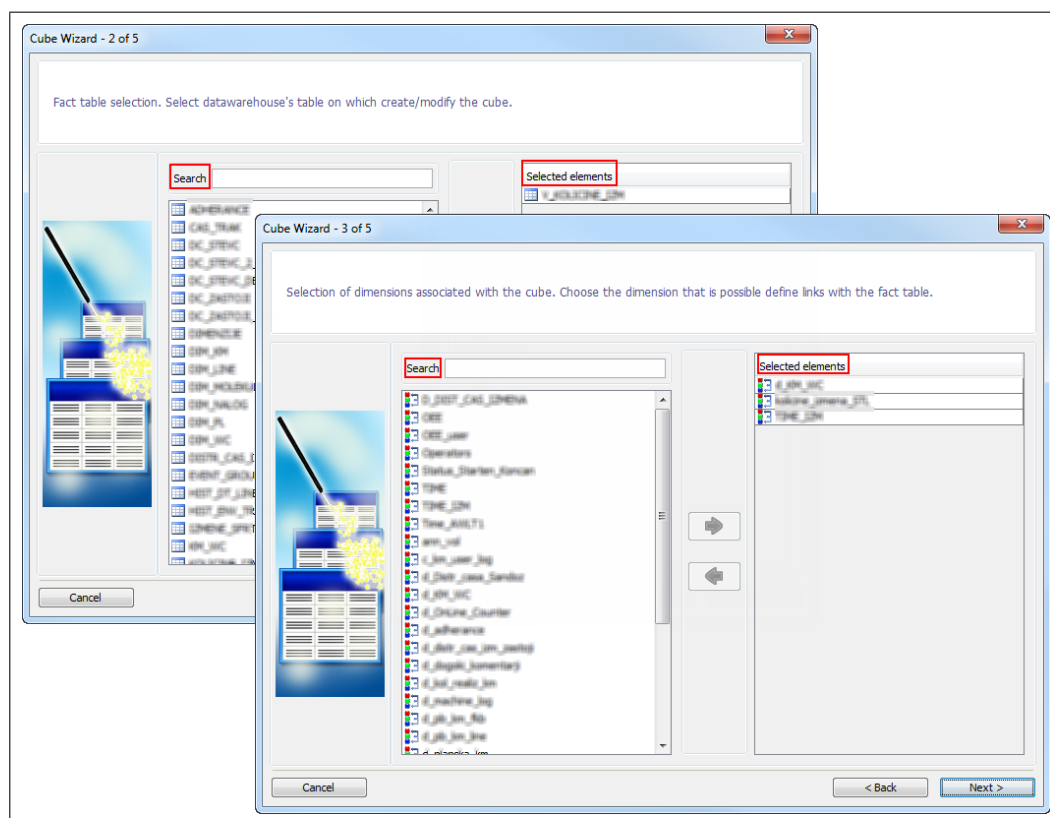
Slika 6.12: Modul “Cube Wizard” - 1. korak.

V drugem koraku se nam na levi strani okna prikažejo vse tabele, ki se nahajajo v izbranem podatkovnem viru. Med njimi izberemo tisto, ki bo predstavljala tabelo dejstev v kocki.

Temu sledi dodajanje ustreznih dimenzij, katere bomo v nadaljevanju povezovali s tabelo dejstev. Oba koraka sta prikazana v Sliki 6.13.

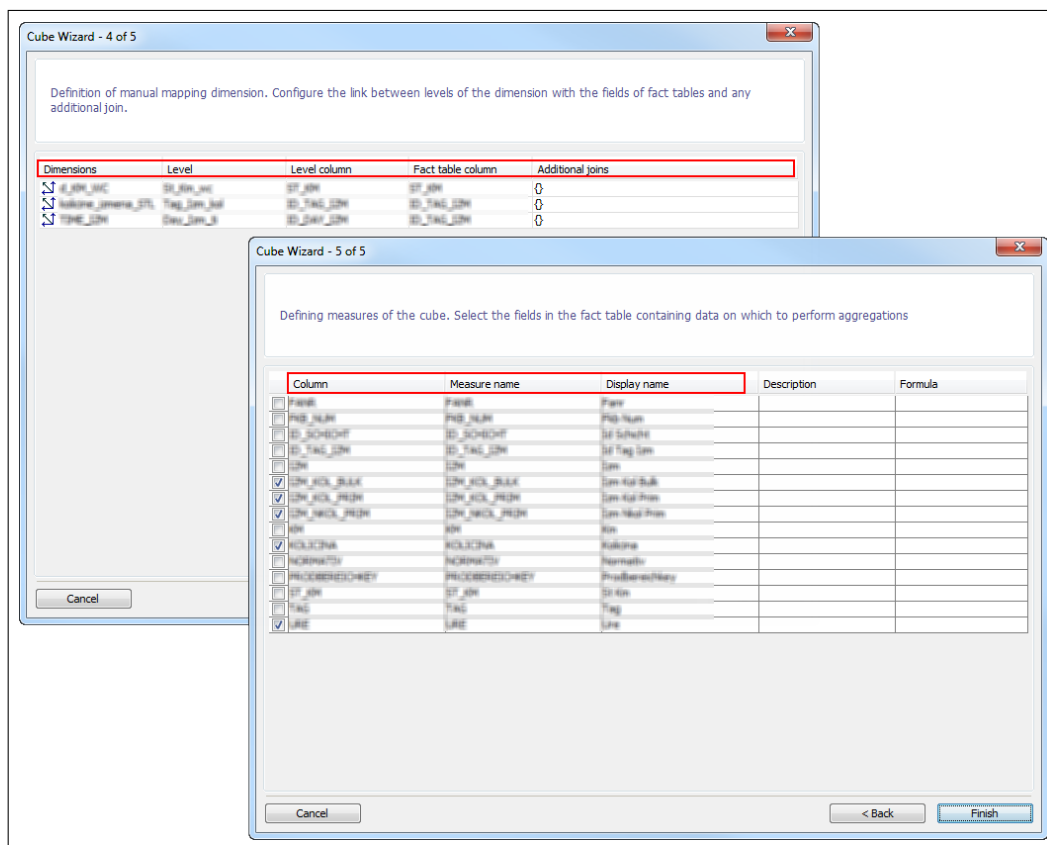
Potem, ko smo izbrali tabelo dejstev in ustrezne dimenzije, sledi 4. korak, ki predstavlja gradnjo zvezdne sheme OLAP kocke (razloženo v Podpoglavju 4.2). Vrstice v čarovniku nam prikazujejo izbrane dimenzije, stolpci pa različne povezovalne nastavitve (Slika 6.14). Pomembna sta predvsem

“Level column” in “Fact table column”, ki vsebujeta ključe za povezovanje dimenzij s tabelo dejstev. Če bi v prvem koraku izbrali možnost “Automatic dimension mapping”, bi čarovnik ta korak avtomatsko preskočil.



Slika 6.13: Modul “Cube Wizard” - 2. in 3. korak.

V zadnjem koraku izbiramo meritve, ki bodo dokončno definirale našo kocko (Slika 6.14). Čarovnik nam prikaže polja tabele dejstev, katero smo izbrali v 2. koraku. Naša naloga je, da s tega seznama izberemo tista polja, ki bodo predstavljala meritve. Ob meritvah, ki se že nahajajo v tabeli dejstev, lahko definiramo tudi meritve “po naročilu” (*angl. Custom Measures*). Te lahko izračunavamo znotraj kocke s polji iz tabele dejstev (meritvami) in nivoji iz kreiranih dimenzij (atributi).



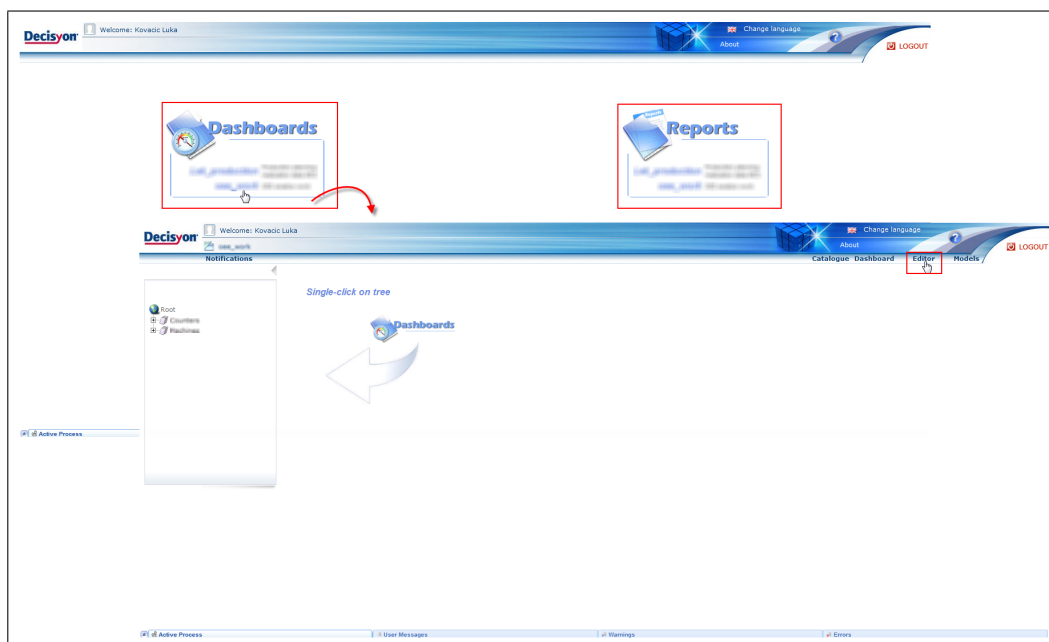
Slika 6.14: Modul “Cube Wizard” - 4. in 5. korak.

### 6.3.3 Decision Web

Z vsemi dosedanjimi postopki smo si pripravili temelje za gradnjo poročila, ki bo naš končni rezultat. Decision nam ponuja dve različni orodji s katerima lahko kreiramo poročila, “Decision Power Designer” in “Decision Web”. Prvo orodje smo spoznali z vsemi do sedaj omenjenimi moduli v tem poglavju, zato smo se odločili gradnjo poročil prikazati z orodjem “Decision Web”. Uporabnik se lahko preko tega orodja poveže na podatkovni model, katerega smo urejali v orodju “Decision Power Designer”. Tam za posameznega uporabnika nastavimo do katerih kock in podatkovnih modelov bo imel dostop.

Za vstopom v “Decision Web” se nam na levi strani prikažejo vsi po-

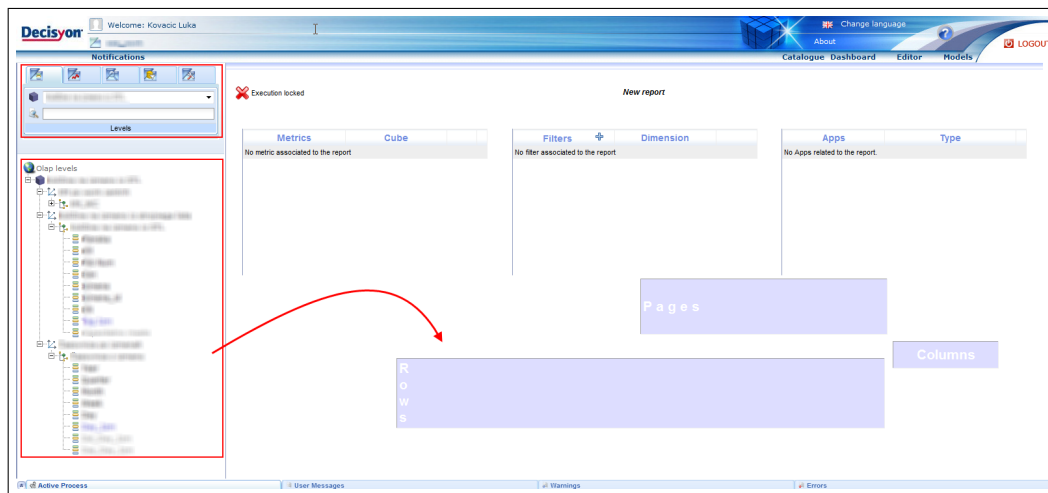
datkovni modeli do katerih imamo urejen dostop, na desni pa vsa shranjena poročila (Slika 6.15).



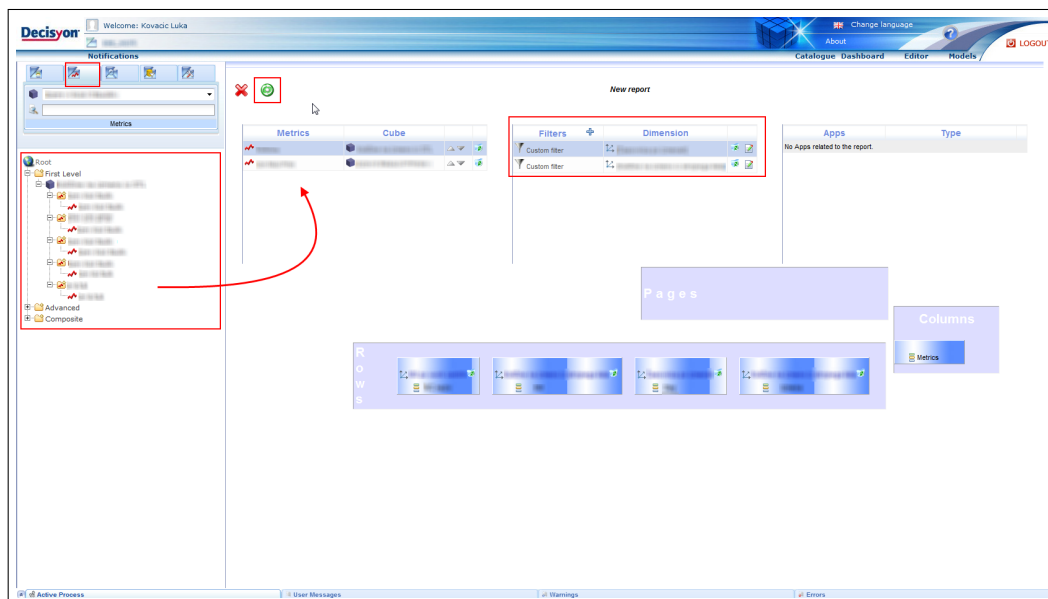
Slika 6.15: Decisyon Web - izbira podatkovnega modela.

Po izbiri podatkovnega modela se nam odpre nova stran na kateri aktiviramo urejevalnik za kreiranje ali urejanje poročil (*angl. Editor*). Urejevalnik je sestavljen iz dveh delov, na levi imamo meni, na katerem izbiramo kocke, dimenzije, meritve itd., na desni pa imamo polja, s katerimi definiramo poročilo (Slika 6.16).

Za izbrano poročilo smo tako prenesli ustrezne nivoje dimenzij v polja na desni strani urejevalnika. Uredili smo jih po zaporedju, ki nam bo pravilno strukturiralo poročilo. Na levi strani smo izbrali tudi ustrezno **metriko**, ki jo je Decisyon avtomatično pripravil na podlagi izbranih meritev ob kreiranju kocke v orodju “Decisyon Power Designer” (Slika 6.17). Metrike služijo obdelovanju kvantitativnih vrednosti v podatkovnem skladišču (npr. seštevanje, filtriranje, izračunavanje meritev “po naročilu” itd.), ki jih kasneje prikazujemo v poročanjih.



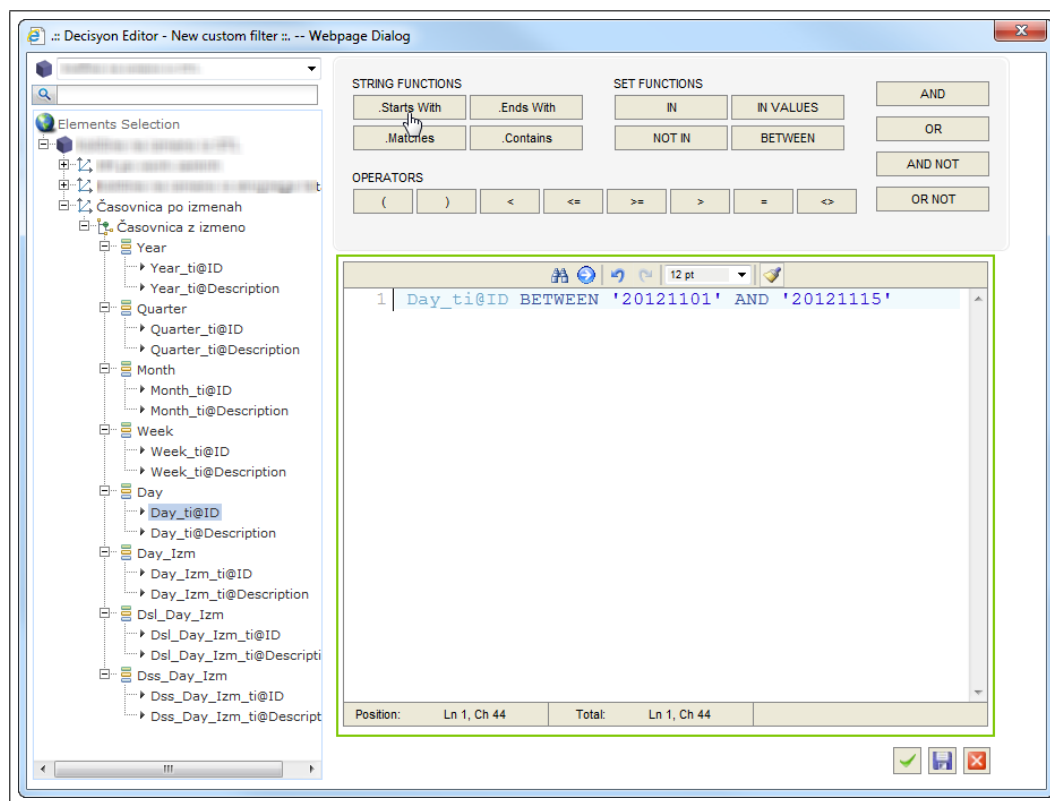
Slika 6.16: Decision Web - urejevalnik poročila.



Slika 6.17: Decision Web - dodajanje metrike.

Če želimo poročilo omejiti na določeno časovno obdobje ali po katerih drugih kriterijih, moramo kreirati filter. To storimo tako, da na desni strani pritisnemo na majhen plus (simbol za dodajanje), kar nam odpre novo po-

javno okno (Slika 6.18).

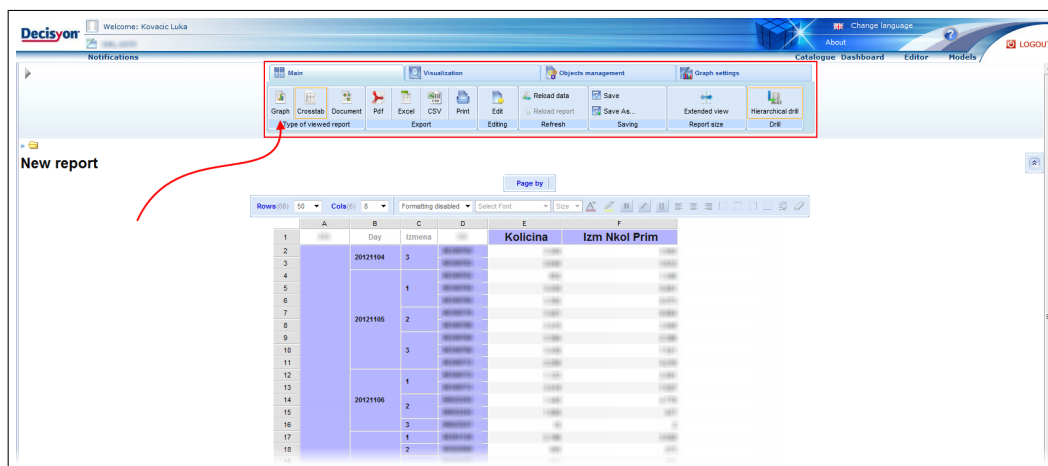


Slika 6.18: Decisyon Web - modul za kreiranje filtrov.

Okno je, podobno kot prejšnje, razdeljeno na dva dela. Na levi strani je seznam s kockami in pripadajočimi dimenzijami, na desni pa urejevalnik za filtre. Po uspešno kreiranem filtru, lahko zaženemo prikaz poročila s pritiskom na zeleni gumb v levem zgornjem kotu urejevalnika.

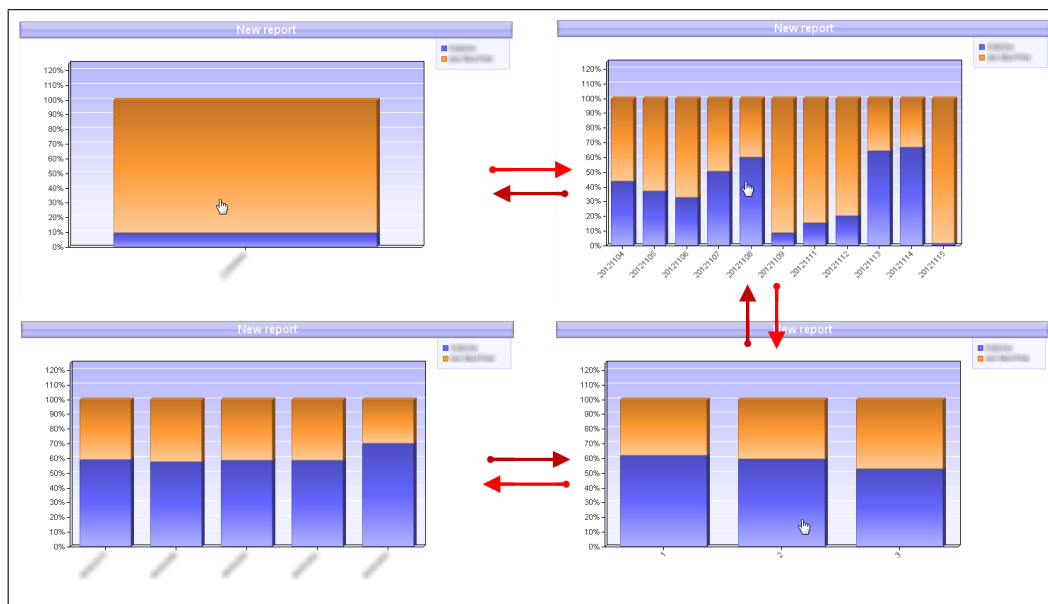
Po prikazu poročila lahko opazimo, da se nam je prikazala tudi nova orodna vrstica (Slika 6.19). S klikom na gumb “Graph” hitro spremenimo prikaz poročila. Na ostalih zavihkih v orodni vrstici lahko izbiramo na kak način naj se nam podatki prikazujejo (*angl. Visualization*), dodajamo nove filtre, sortiranja, metrike itd. (*angl. Objects management*) in spreminjamo nastavitve prikazovanja poročila kot grafa (*angl. Graph settings*).

Za končni rezultat smo izbrano poročila nastavili tako, da se prikazuje kot



Slika 6.19: Decisyon Web - prikaz poročila.

graf, po katerem se lahko “spuščamo” in “dvigujemo” (angl. “Drill down”, “Drill up”). Izbrani meritvi sta prikazani v medsebojnem razmerju v odstotkih. V Sliki 6.20 lahko vidimo korake spuščanja po nivojih v grafu. Slika “roke” na posameznem nivoju nam prikazuje izbrano polje na grafu.



Slika 6.20: Decisyon Web - prikaz poročila po korakih.

Naše izbrano poročilo je sestavljeno iz štirih nivojev. V prvem nivoju vidimo izbrani funkcijski blok. S klikom nanj se nam prikažejo izbrane meritve razdeljene po dnevih - drugi nivo. Izbira posameznega dneva nam v tretjem nivoju prikaže izbrane meritve razdeljene po izmenah, v zadnjem nivoju pa lahko glede na izbrano izmeno vidimo meritve razdeljene po delovnih nalogih.

# Poglavje 7

## Sklepne ugotovitve

Poročevalni sistemi so v velikem poslovnem sistemu ključnega pomena. S prikazovanjem podatkov na različne načine imamo možnost odkrivanja tistih, ki so v danem trenutku lahko odločilni za pravo poslovno odločitev. Ob vsem tem ne smemo zanemariti dejstva, da so lahko poročanja zelo “požrešna” (v računalniškem smislu) ob svojem delovanju.

V diplomski nalogi je opisan proces prenove poročevalnega sistema od začetka (analiza obstoječih poročanj) do konca (priprava poročil v novem orodju) s pomočjo platforme Decisyon 4. Opisali smo tudi njene ključne funkcionalnosti s katerimi nudi sodobno podporo odločanju.

S prenovo poročevalnega sistema smo v poslovnem sistemu močno zmanjšali obremenjenost transakcijske podatkovne baze in na ta način omogočili boljšo varnost in zanesljivost ob izvajanju tekočih poslov, obenem pa imajo nova precej hitrejša odzivna časa.

Transakcijsko podatkovno bazo sedaj “obremenjujemo” le ob proženju nastavljenih procedur za prenašanje podatkov v podatkovno skladišče. To se dogaja trikrat dnevno, po koncu vsake posamezne izmene. Sama poročila se v platformi Decisyon 4 prikazujejo ekstremno hitro v primerjavi s starimi poročanji. Tako lahko za enako poročilo sedaj porabimo le 1-2 sekundi, prej pa so se procedure v ozadju izvajale tudi po 30 sekund preden se nam je to poročilo prikazalo.

Ker smo še vedno v fazi prenašanja poročanj s starega na nov sistem, nimamo kriterijev s katerimi bi lahko realno ocenili uporabniške izkušnje. V proces prenašanja so vključeni le izbrani uporabniki, ki so potrebni zaradi podajanja morebitnih novih (dodatnih) uporabniških zahtev in testiranja prenovljenih poročanj. Njihovi dosedanji odzivi so bili zelo pozitivni, predvsem kar se tiče hitrosti izvajanja in grafičnih prikazovanj poročil.

V prihodnosti je naš cilj vsa obstoječa poročanja prenesti na platformo Decisyon 4, tako z vidika možnosti upokojitve starih sistemov kot tudi z vidika uporabnikov, ki bodo imeli vse potrebne informacije dosegljive na enem mestu.

Glede na razvoj informacijske tehnologije in predvsem mobilnih naprav je naša želja ta, da bomo s prenosom na Decisyon 4 pripravljeni na preskok, ko bodo uporabniki za svoje delo potrebovali le mobilne naprave (npr. tablične računalnike). S tem bodo imeli lažji dostop in možnost hitrejšega odkrivanja morebitnih poslovno pomembnih informacij.

# Literatura

- [1] T. M. Connolly, C. E. Begg. *Database Systems, A Practical Approach to Design, Implementation and Management, Fourth Edition*, Pearson Education, 2005.
- [2] C. Coronel, S. Morris, P. Rob. *Database Systems: Design, Implementation and Management, Ninth Edition*, Cengage Learning, 2011.
- [3] D. Jagarinec. "Operacije v OLAP-u", *Moj mikro*, Delo Revije, dec. 2005, str. 80-81.
- [4] R. Kimball, J. Caserta. *The Data Warehouse ETL Toolkit*, Wiley Publishing, 2004.
- [5] (2007) OLAP. Dostopno na: <http://wiki.fmf.uni-lj.si/wiki/OLAP.pdf>
- [6] (2013) Decisyon 4 Collaborative Decision Making & Execution. Dostopno na: <http://www.decisyon.com/solution/decisyon-4>
- [7] (2013) Introduction to Oracle Database Server. Dostopno na: [http://docs.oracle.com/cd/B28359\\_01/server.111/b28318/intro.htm](http://docs.oracle.com/cd/B28359_01/server.111/b28318/intro.htm)
- [8] (2013) Introduction to Oracle Database. Dostopno na: [http://docs.oracle.com/cd/E11882\\_01/server.112/e16508/intro.htm](http://docs.oracle.com/cd/E11882_01/server.112/e16508/intro.htm)
- [9] (2013) J. Krsnik, Sprotno analitično procesiranje - OLAP. Dostopno na: [http://www.bf.uni-lj.si/fileadmin/users/1/zootehnika/center\\\_za\\\_strokovno\\\_delo/objave/olap.pdf](http://www.bf.uni-lj.si/fileadmin/users/1/zootehnika/center\_za\_strokovno\_delo/objave/olap.pdf)

- [10] (2013) Tehnologija, komponente sistemov poslovne inteligence. Dostopno na: [http://www.add.si/poslovne\\_resitve/add\\_bi/tehnologija.pdf](http://www.add.si/poslovne_resitve/add_bi/tehnologija.pdf)