

LATVIJAS UNIVERSITĀTE

LAILA NIEDRĪTE

**DATU NOLIKTAVU KONCEPTUĀLO MODEĻU
IZSTRĀDES METODES UN REALIZĀCIJAS
MEHĀNISMI**

Promocijas darbs

Rīga - 2008

LATVIJAS UNIVERSITĀTE

LAILA NIEDRĪTE

**DATU NOLIKTAVU KONCEPTUĀLO MODEĻU
IZSTRĀDES METODES UN REALIZĀCIJAS
MEHĀNISMI**

Promocijas darbs
datorzinātņu doktora (Dr. sc. comp.) zinātniskā grāda iegūšanai

Nozare: datorzinātnes
Apakšnozare: datoru un sistēmu programmatūra

Zinātniskais vadītājs:
profesors, Dr. sc. comp.,
MĀRIS TREIMANIS

R ī g a - 2008

SATURS

1. IEVADS	6
1.1. Pētījuma motivācija un aktualitāte.....	6
1.2. Darba mērķi.....	8
1.3. Promocijas darba galvenie rezultāti.....	8
1.4. Rezultātu publikācijas un prezentācijas konferencēs	9
1.5. Promocijas darba struktūra.....	10
2. DATU NOLIKTAVU PAMATELEMENTI	12
2.1. Nodaļas nolūks.....	12
2.2. Daudzdimensiju modeļa elementi	12
2.3. OLAP operācijas	14
2.4. Datu noliktava kā materializēto skatu kopa	15
2.5. Datu noliktavu konceptuālie modeļi.....	16
2.5.1. Uz E/R modeli balstīti datu noliktavu konceptuālie modeļi	20
2.5.2. Uz UML balstīti datu noliktavu konceptuālie modeļi	22
2.5.3. Neatkarīgi konceptuālie modeļi.....	24
2.6. Datu noliktavu metamodeļu standarts CWM un konceptuālie modeļi	26
3. EKSISTĒJOŠU METOŽU ANALĪZE DATU NOLIKTAVU KONCEPTUĀLO MODEĻU IZVEIDEI	28
3.1. Nodaļas nolūks.....	28
3.2. Metodes jēdziens	28
3.3. Metožu klasifikācija	28
3.4. Datu vadītas metodes	29
3.4.1. DFM metode	30
3.4.2. Inmona metode.....	33
3.5. Prasību vadītas metodes	35
3.5.1. Informācijas prasību vadīta metode.....	35
3.5.2. Procesu vadītas metodes.....	36
3.5.3. Mērķa vadītas metodes.....	37
3.5.4. Lietotāju vadītas metodes.....	37
3.6. Vairāku pieeju kombinēšana.....	39
3.7. Pieeju izvēles problēma datu noliktavas konceptuālā modeļa izveidei.....	40

4. PROCESU MĒRĪŠANA UN KONTROLE	42
4.1. Nodaļas nolūks.....	42
4.2. Mērīšanas pamatjēdzieni un ontoloģijas	42
4.2.1. Informācijas vajadzības un mērāmie procesi un produkti	43
4.2.2. Procesu mērīšanas ontoloģijas.....	45
4.3. Biznesa procesu mērīšanas sistēmas	45
4.4. Metode procesu mērījumu un indikatoru noteikšanai.....	46
4.5. Procesu datu noliktavas jēdziens	48
4.5.1. Ražošanas procesu datu noliktava	49
4.5.2. Darbu plūsmu datu noliktava.....	49
4.5.3. Izpildes pārvaldības sistēma	50
4.5.4. Korporatīvā izpildes mērīšanas sistēma	50
5. PIEDĀVĀTĀS METODES KONCEPTUĀLO MODEĻU IZVEIDEI	52
5.1. Nodaļas nolūks.....	52
5.2. Metodes definēšana	52
5.2.1. Metožu inženierija un jaunu metožu definēšana	53
5.2.2. Promocijas darbā piedāvāto metožu definēšana	56
5.3. Lietotāju vadītā metode.....	57
5.3.1. Lietotāju vadītās metodes procesa modelis	57
5.3.2. Lietotāju vadītās metodes detalizēts apraksts	59
5.3.3. Secinājumi un rezultāti par lietotāju vadīto metodi	72
5.4. Datu vadītā metode	76
5.4.1. Datu vadītās metodes procesa modelis	76
5.4.2. Datu vadītās metodes detalizēts apraksts.....	78
5.4.3. Secinājumi un rezultāti par datu vadīto metodi	100
5.5. Mērķu vadītā metode.....	104
5.5.1. Mērķu vadītās metodes procesa modelis	104
5.5.2. Mērķu vadītās metodes detalizēts apraksts.....	105
5.5.3. Procesu mērīšana un procesu kontrole izmantojot datu noliktavu.....	116
5.5.4. Secinājumi un rezultāti par mērķu vadīto metodi.....	117
5.6. Nodaļas secinājumi.....	119
6. DATU NOLIKTAVAS REALIZĀCIJĀ IZMANTOJAMĀS METODES	121
6.1. Nodaļas nolūks.....	121
6.2. Metode pieejas kontrolei datu noliktavām.....	121
6.2.1. Prasības pieejas kontrolei datu noliktavās	122
6.2.2. Pieejas kontroles modelis datu noliktavai	123
6.2.3. Pieejas kontroles mehānisms, izmantojot dinamiskos skatus	125
6.3. Metode dalīto faktu izmantošanai lēni mainīgu faktu gadījumā	133
6.3.1. Lēni mainīgo faktu jēdziens	133
6.3.2. Faktu dalīšanas un svarošanas pielietošanas situāciju apskats	133

6.3.3.	Piedāvātās metodes apraksts.....	134
6.3.4.	Dalīto un svaroto faktu izmantošanas piemērs	137
6.4.	Nodaļas secinājumi.....	139
	NOBEIGUMS	140
	SAĪSINĀJUMI	143
	LITERATŪRA	145
	PIELIKUMI	154
1.	pielikums Promocijas darba autores publikācijas.....	154
2.	Pielikums, Augstskolu datu noliktavas - situācijas raksturojums.....	155
3.	pielikums. LU datu noliktavas raksturojums.....	158
4.	pielikums. GQ(IM) metodes soļu detalizēts apraksts	161
5.	pielikums. Datu vadītas metodes aprobācijas projekta raksturojums	163
6.	pielikums Indikatoru definēšana ar OCL	167
7.	pielikums. Mērķu vadītas metodes aprobācijas projekta raksturojums	169

1. IEVADS

Pirmās datu noliktavas tika izveidotas deviņdesmito gadu sākumā. Vairāku autoru pētījumi apliecināja, ka tiešsaistes transakciju apstrādi (*on-line transaction processing* - OLTP) un tiešsaistes analītisko apstrādi (*on-line analytical processing* - OLAP) nevar efektīvi realizēt vienas datu bāzes pārvaldības sistēmas vidē [CCS93], [Inm02], jo veicamās operācijas ir izteikti atšķirīgas. OLTP gadījumā tās ir īsas lasīšanas/rakstīšanas operācijas, kur iesaistīti viens vai daži ieraksti, OLAP gadījumā tās ir garas (var būt pat miljoniem ierakstu) lasīšanas operācijas. OLAP vajadzībām tika radītas datu noliktavas.

Tiek uzskatīts [KRRT98],[JLVV03], ka tradicionālās tiešsaistes transakciju apstrādes sistēmas (OLTP) nav piemērotas lēmumu atbalstam. Kā piemērots risinājums tiek piedāvātas datu noliktavas.

Datu noliktavu visbiežāk lietotā definīcija ir: datu noliktava ir "saskaņotu, nemainīgu un vēsturisku datu kolekcija vadības lēmumu pieņemšanas atbalstam" [Inm02].

Datu noliktavas raksturo sekojošas īpašības:

- datu analīzes vajadzību nodrošināšana - datu noliktavas tiek veidotas, lai nodrošinātu speciālistiem informācijas apstrādes jomā ātrākas un labākas lēmumu pieņemšanas iespējas [Inm02], [JLVV03]. Tiek sagaidīts, ka būs pieejama nepieciešamā informācija vajadzīgā laikā un vietā par atbilstošām izmaksām;

- informācijas pieejamības nodrošināšana - datu noliktavas integrē uzņēmuma informāciju no dažādiem uzņēmuma datu avotiem [Inm02], [KRRT98], tajā skaitā no dažādām OLTP sistēmām, integrējot, nepieciešamības gadījumā, arī datus no ārējiem datu avotiem;

- datu noliktavas ir orientētas uz efektīvu lielu datu apjomu apstrādi. Efektivitāti nodrošina gan datu bāze, kuras pamatā ir daudzdimensiju modelis [KRRT98], gan tas, ka datu noliktavu var uzskatīt arī par tehnoloģiju kopu [JLVV03], jo tiek izmantoti gan specifiski izstrādes rīki, gan datu bāzu risinājumi, piemēram, īpaši datu noliktavām piemēroti indeksi, materializētie skati;

- datu noliktavu projekti ir raksturīgi ar to īstenošanas ilgumu [KRRT98], kas nosaka prasības gan kvalitātei, gan izmaiņu pārvaldībai.

Balstoties uz iepriekš minētām īpašībām, var secināt, ka datu noliktava ir stratēģija, kā integrēt dažādu lietotāju grupu interesējošus datus no heterogēniem datu avotiem vienotā konceptuālā modelī un vienotā tehniskā risinājumā. Datu noliktava kalpo par pamatu OLAP lietotnēm.

1.1. Pētījuma motivācija un aktualitāte

Tradicionāli datu noliktavas tiek izstrādātas finansu sektorā, apdrošināšanas kompānijās, tirdzniecībā, veselības aizsardzībā. Tomēr bieži vien datu noliktavas tiek izstrādātas arī citās jomās, tajā skaitā izglītībā. Autore laika posmā no 1998. līdz 2003. gadam piedalījies augstskolu informatīvo sistēmu izstrādē gan Latvijas Universitātei (LUIS), gan citām

augstskolām (LAIS), regulāri piedalījusies Eiropas universitāšu informācijas sistēmu (EUNIS) konferencēs, kurās prezentējusi arī rakstus par augstskolu informācijas sistēmām Latvijā [SNN01],[NN03]. EUNIS konference, kurā katru gadu piedalās vairāk nekā 100 dažādu Eiropas universitāšu pārstāvji un prezentē savu pieredzi, bija viens no faktoriem, kas motivēja promocijas darba autori pievērsties datu noliktavu problēmām un datu noliktavu izstrādei augstskolām. Jāatzīst gan, ka pēdējo 7 gadu laikā ziņojumi par augstskolu datu noliktavām EUNIS konferencē nav bijuši sevišķi daudz, piemēram, 1999. un 2004. gadā - par Slovēnijas (Ljubļanas) [BRK99], [MP04] un Portugāles (Porto) [DR99] universitāšu datu noliktavām, 2001. gadā - par Francijas starpaugstskolu datu noliktavu [Des01]. Atšķirīga situācija ir ASV, kur datu noliktavas ir izstrādātas daudzām augstskolām, dati atrodami gan augstskolu mājas lapās, piemēram, [MIT], gan EDUCAUSE (ASV) [EDUC] gadskārtējo konferenču prezentācijās, gan [THO97] pētījumā. Promocijas darba autore citu augstskolu pieredzi un datu noliktavu iespējamības analīzi augstskolās apkopojusi rakstā [Nie04], kā arī ar pētījuma rezultātiem var iepazīties promocijas darba 2. pielikumā. Laika posmā no 2003. līdz 2007.gadam autore piedalās LU datu noliktavas izstrādē, uzturēšanā un attīstīšanā. Praktiskā darba pieredze datu noliktavas projektā bija otrs motivējošs faktors pievērsties pētījumiem par datu noliktavu problemātiku.

Datu noliktavu jomā eksistē daudz dažādu metožu, kā noskaidrot prasības datu noliktavai, lai tiktu izveidots atbilstošs datu noliktavas konceptuālais modelis, tomēr nav vienprātības par labākajām metodēm, nav vienota standarta. Jāpiebilst, ka pārsvarā literatūrā ir uzsvars uz konceptuālajiem modeļiem un to īpašību formālu definēšanu, tajā pašā laikā tikai dažiem konceptuālajiem modeļiem ir formāli aprakstītas metodes, kā šie modeļi iegūstami. Problēmas aktualitāti apliecina arī datu bāzu konferenču tēmas un atbilstošajā jomā atzītu ekspertu pārskata darbi par aktuālām pētniecības problēmām.

Daudzās datu bāzu jautājumiem veltītās konferencēs, piemēram, VLDB, PODS, SIGMOD parasti ir atsevišķas sesijas par datu noliktavām vai ar tām saistītiem jautājumiem (datizrāci, datu integrāciju u.c.). Tiek organizētas arī specializētas konferences tieši par datu noliktavām - DOLAP (*Data Warehousing and OLAP*) un DaWaK (*Data Warehousing and Knowledge Discovery*), no 1999 līdz 2003. gadam bija arī DMDW (*Design and Management of Data Warehouses*). Lai apzinātu aktuālus virzienus tālākiem pētījumiem, datu noliktavu jomā atzīti zinātnieki laiku pa laikam publicē pārskata rakstus [Wid95], [Vas00], [RALT06] vai uzstājas ar uzaicinātiem priekšlasījumiem [Riz03] konferencēs par situāciju šajā jomā.

Šajos pārskatos analizētas arī konceptuālo modeļu un datu noliktavu izstrādes metožu problēmas. 2000. gadā DMDW konferencē [Vas00] uzsvēra, ka acīm redzami pietrūkst standarta metodes datu noliktavu izstrādei. Tā vietā eksistē rīku piegādātāju vai ekspertu ieteikumi, kā vajadzētu izstrādāt datu noliktavu. [Vas00] atzīst, ka trijās lielajās konferencēs VLDB, PODS, SIGMOD tieši šie jautājumi netiek skarti. 2003. gadā DMDW konferencē pārskata referātā [Riz03] tiek konstatēts, ka no izstrādātāja viedokļa pastāv objektīva nepieciešamība pēc ērti lietojamas standarta metodes datu noliktavu izstrādei. 2006. gadā DOLAP konferencē prezentētajā pārskata rakstā [RALT06] metodes nepieciešamība ir starp aktuālajiem un joprojām neatrisinātajiem jautājumiem. Autori uzskata [RALT06], ka atsevišķām izstrādes fāzēm ir piedāvātas dažas metodes, piemēram, [GMR98a] - konceptuālai modelēšanai, tomēr plašākas

metodes, kas nodrošinātu, ka datu noliktava atbilstu pasūtītāju prasībām, tiek piedāvātas ļoti reti. Par prasību analīzi [RALT06] konstatē, ka datu noliktavu jomā šim jautājumam tiek pievērsts pārāk maz uzmanības, kaut arī prasību analīze ir viena no iespējam samazināt projekta neveiksmīgu galarezultātu. [RALT06] piemin gan datu vadītas, gan prasību vadītas pieejas datu noliktavu izstrādei, bet uzskata, ka pētījumiem daudzsološākā ir dažādu pieeju kombinēšana. Autori uzskata [RALT06], ka nepieciešami turpmāki pētījumi, lai izstrādātājiem varētu piedāvāt metodes, kā noskaidrot informācijas vajadzības un tās pārveidot par datu noliktavu konceptuāliem modeļiem.

1.2. Darba mērķi

Promocijas darba mērķis ir izstrādāt efektīvas metodes datu noliktavu konceptuālo modeļu izveidei un implementācijai un definēt datu noliktavu konceptuālo modeļu izveidei situācijas faktoru kopu, kas nosaka piemērotas metodes izvēli vai jaunas metodes izstrādes nepieciešamību.

Lai sasniegtu mērķi, ir jāatrisina šādi uzdevumi:

- 1) Izpētīt eksistējošās pieejas un tām atbilstošās metodes datu noliktavu konceptuālo modeļu izstrādē.
- 2) Pielietojot metožu inženierijas teoriju, izstrādāt situācijām specifiskas metodes datu noliktavu konceptuālo modeļu izstrādei,
- 3) Izstrādātās konceptuālo modeļu izveides metodes aprobēt datu noliktavu projektos,
- 4) Jauno metožu izstrādes un aprobācijas laikā apkopot un analizēt situāciju raksturojošos faktorus, kas nosaka eksistējošu metožu komponentu izvēli vai adaptāciju, vai jauna metodes komponenta izstrādes nepieciešamību.
- 5) Izstrādāt datu noliktavu realizācijas metodes lietotāju pieejas kontrolei un lēni mainīgu faktū uzkrāšanai. Metodes aprobēt datu noliktavu projektos.

1.3. Promocijas darba galvenie rezultāti

Galvenie promocijas darba rezultāti ir:

- Darbā izpētīti datu noliktavu konceptuālie modeļi un eksistējošās metodes to izveidei. Rezultātā secināts, ka eksistējošām metodēm ir gan trūkumi, gan priekšrocības noteiktās izstrādes situācijās. Tātad pastāv nepieciešamība vērtēt izstrādes situācijas, lai varētu izvēlēties piemērotāko metodi vai nepieciešamības gadījumā izstrādāt jaunu.
- Izstrādātas trīs metodes datu noliktavu konceptuālo modeļu izstrādei - lietotāju vadīta, datu vadīta un mērķu vadīta metode. Metožu izstrādē lietota metožu inženierijas pieeja. Katrai metodei dota formāla reprezentācija un metodes komponentu detalizēts apraksts.
- Katras metodes komponentiem aprakstīti situācijas faktori, kas nosaka jau eksistējošas metodes komponenta lietošanu vai adaptāciju, vai jauna komponenta izstrādi konkrētā situācijā.
- Veikts izstrādāto metožu situācijas faktoru apkopojums un salīdzinājums.
- Izstrādātas metodes sekmīgi aprobētas Latvijas Universitātes datu noliktavas projektos. Katra no metodēm aprobēta atsevišķā projektā atšķirīgās datuvju izstrādes situācijās.

- Izstrādātas divas metodes, kas lietojamas datu noliktavas realizācijā - viena metode - pieejas kontrolei datu noliktavās un otra metode - lēni mainīgu faktu glabāšanai un izmantošanai.

1.4. Rezultātu publikācijas un prezentācijas konferencēs

Promocijas darbs balstās uz pētījumiem, kuru rezultāti ir atspoguļoti publikācijās (skat. atsevišķo 1. pielikumu):

- publikācija par augstskolu datu noliktavām:

[Nie04] L. Niedrite, "Requirements and Options for Data Warehouses at Universities", Latvijas Universitātes raksti, 669. sējums "Datorzinātne un Informācijas tehnoloģijas", 2004, lpp. 117- 125.

- Publikācijas par datu noliktavu konceptuālo modeļu izstrādi [BN04], [SN05], [NSTN07a], [NSTN07b]:

[BN04] J. Benefelds, L. Niedrite, "Comparison of Approaches in Data Warehouse Development in Financial Services and Higher Education", ICEIS, Vol.1, Porto, 2004, pp. 552-557.

[SN05] D. Solodovnikova, L. Niedrite, "Using Data Warehouse Resources for Assessment of E-Learning Influence on University Processes", In: Proceedings of the 9th East-European Conference on Advances in Databases and Information Systems, Tallinn, Estonia , 2005, pp. 233. – 248.

[NSTN07a] L. Niedrite, D. Solodovnikova, M. Treimanis, A. Niedritis, „The Development Method for Process-Oriented Data Warehouse”, In: WSEAS Transactions on Computer Research, Issue 2, Vol. 2, February 2007, pp. 183 – 190.

[NSTN07b] L. Niedrite, D. Solodovnikova, M. Treimanis, A. Niedritis, “Goal-Driven Design of a Data Warehouse-Based Business Process Analysis System”, In: Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering And Data Bases (AIKED ‘07), Corfu, February 2007.

- Publikācijas par datu noliktavu realizācijas metodēm: [NG04], [GN06].

[NG05] L. Niedrite, L. Grundmane, "Controlling access to Data Warehouse data within the database", In: Eds. J.Barzdins and A.Caplinskas. Databases and Information Systems, Selected papers from the 6th International Baltic Conference DB&IS'2004, Riga, Latvia, IOS Press, FAIA (Frontiers in Artificial Intelligence and Applications), vol. 118, 2005, pp. 126-140.

[GN06] L. Grundmane, L. Niedrite, "Splitting facts using weights", In: Proceedings of the 8th International Conference Of Enterprise Information Systems (ICEIS 2006), Paphos, Cyprus – 2006.

Promocijas darba rezultāti prezentēti konferencēs:

- ICEIS 2004. gadā [BN04] un 2006. gadā [GN06],
- Baltic DB&IS 2004. gadā [NG05],
- AIKED 2007. gadā [NSTN07b],
- 63. LU zinātniskā konference, Informācijas tehnoloģiju sekcija, 2005. gada februāris, prezentācija: Niedrīte, L. "Datu noliktavu konceptuālo modeļu izstrādes metodoloģijas",
- 64. LU zinātniskā konference, Informācijas tehnoloģiju sekcija, 2006. gada februāris, prezentācija: Solodovņikova, D., Niedrīte, L., Niedrītis, A. "Procesu datu noliktavas izmantošana Latvijas Universitātē".

1.5. Promocijas darba struktūra

Promocijas darbam ir 171 lappuse, 80 attēli un 29 tabulas. Promocijas darbs ir sakārtots 6 nodaļās. Promocijas darbam ir 7 pielikumi.

Darba 1. nodaļā tiek definēta promocijas darba pētījuma sfēra un dots pamatojums tēmas izvēlei, kā arī tiek aprakstīti iegūtie promocijas darba rezultāti.

Darba 2. nodaļā tiek definēti ar datu noliktavu saistīti jēdzieni, kuri ir būtiski promocijas darbam. Nodaļas mērķis ir sniegt priekšstatu par eksistējošo datu noliktavu konceptuālo modeļu daudzveidību, tiek analizēti dažādi datu noliktavu konceptuālo modeļu tipi.

Darba 3. nodaļā tiek dota metodes definīcija. Nodaļā tiek analizētas eksistējošas datu noliktavu konceptuālo modeļu izstrādes pieejas, kas nosaka prasību noskaidrošanas veidu, un procesu, kā izveidot prasībām atbilstošu datu noliktavu konceptuālo modeli. Nodaļā dota pieeju klasifikācija un dots pārskats katras pieejas eksistējošām metodēm.

Darba 4. nodaļā raksturoti būtiski elementi jaunas metodes definēšanai jaunākajā un mazāk pētītajā pieejā datu noliktavu konceptuālo modeļu izstrādē - mērījumu vadītā pieejā: mērīšanas pamatjēdzieni un metodes procesu mērījumu un indikatoru noteikšanai. Tiek analizētas eksistējošas procesu mērīšanas sistēmas, tajā skaitā procesu datu noliktavas kā viens no iespējamiem procesu mērīšanas sistēmu realizācijas veidiem.

Darba 5. nodaļā tiek piedāvātas trīs jaunas metodes - lietotāju vadīta, datu vadīta un mērķu (mērījumu) vadīta metode. Metodes tiek veidotas atbilstoši metožu inženierijas nostādņēm, novērtējot situāciju raksturojošos faktorus datu noliktavu projektos. Atkarībā no situācijas konteksta tiek izmantoti gatavi vai adaptēti citu autoru metožu komponenti, vai arī izstrādāti jauni situācijai specifiski metožu komponenti. Katrai jaunajai metodei tiek novērtēti metodes konstruēšanā izmantotie komponenti un iegūta situācijas faktoru kopa.

Darba 6. nodaļā tiek piedāvāts risinājums divām problēmām datu noliktavu realizācijā - viena metode ir par pieejas kontroli datu noliktavām, otra metode ir saistīta ar lēni mainīgu faktu glabāšanas un izmantošanas problēmu.

Promocijas darba rezultāti un secinājumi, kas iegūti, izstrādājot promocijas darbu, ir apkopoti darba noslēguma nodaļā.

Promocijas darbam ir 7 pielikumi:

1. pielikums - atsevišķs pielikums - autores publikāciju kopijas;

2. pielikums - Augstskolu datu noliktavas - situācijas raksturojums;
3. pielikums. LU datu noliktavas raksturojums;
4. pielikums. GQ(IM) metodes soļu detalizēts apraksts;
5. pielikums. Datu vadītas metodes aprobācijas projekta raksturojums;
6. pielikums Indikatoru definēšana ar OCL;
7. pielikums. Mērķu vadītas metodes aprobācijas projekta raksturojums.

Pateicības: Pētījums veikts ar Eiropas Sociālā Fonda atbalstu.

2. DATU NOLIKTAVU PAMATELEMENTI

2.1. Nodaļas nolūks

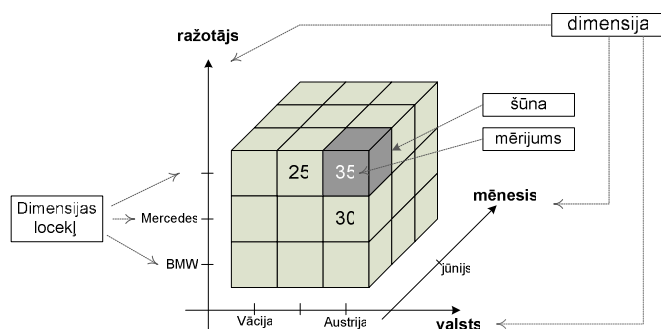
Šajā nodaļā sniegts apskats par datu noliktavu specifiskām īpašībām un uzbūves elementiem, kas nosaka datu noliktavu atšķirību no citām informācijas sistēmām, raksturota šo īpašību ietekme uz datu noliktavu izstrādi.

Nodaļas mērķis ir sniegt priekšstatu par eksistējošo datu noliktavu konceptuālo modeļu daudzveidību. Nodaļā analizēti konceptuālo modeļu tipi, detalizētāk aprakstot dažus eksistējošus konceptuālos modeļus, kuru komponenti promocijas darbā piedāvātajās metodēs tiek izmantoti kā gatavi vai adaptējami komponenti.

2.2. Daudzdimensiju modeļa elementi

Datu noliktavām pamatā ir daudzdimensiju modeļi. Šajā apakšnodaļā tiks sniegtas daudzdimensiju modeļa jēdzienu definīcijas un piemēri.

Lai atspoguļotu daudzdimensiju skatījumu uz datiem, daudzi autori [CD97], [SBHD98], [TPGS01], [JLVV03] izmanto kuba metaforu (skat. 1. att.).



1. att. Daudzdimensiju shēmas vizualizācija ar kuba metaforas palīdzību, pēc [SBHD98]

Viens no daudzdimensiju modeļa pamatjēdzieniem ir fakts. Fakts ir:

- datu analīzes subjekts vai mērķis [SBHD98], [TPGS01],
- uzņēmuma intereses objekts, lieta vai notikums [TPGS01].

Lietojot kuba metaforu, viss kubs kopumā ir fakts. Faktus apraksta faktu atribūtu kopa [TPGS01],[SBHD98], kas tiek saukta par **mērījumiem** vai faktu atribūtiem, tie tiek saukti arī par kvantitatīviem datiem, jo parasti ir skaitliski lielumi. Faktu atribūti jeb mērījumi var būt atomāri vai atvasināti[SBHD98], tie var būt aprēķināti no citiem mērījumiem un/vai dimensiju atribūtiem.

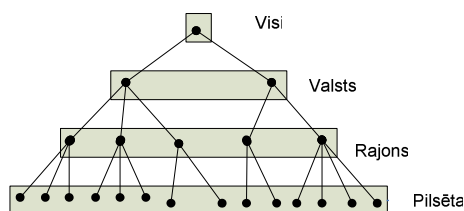
Izmantojot kuba metaforu, kuba šūnas satur mērījumus, kas apraksta fakta, ko reprezentē kubs, īpašības.

Mērījumu kopa ir balstīta uz **dimensiju** kopu. Dimensijas nosaka veidu, kā faktus iespējams analizēt [TPGS01],[SBHD98]. Dimensijas ir aprakstoši dati, tās izmanto, lai atlasītu

faktus un grupētu tos dažādos detalizācijas līmeņos. Kādas dimensijas ietilpst analizējamā fakta dimensiju kopā, izriet no detalizācijas pakāpes, kas izvēlēta fakta reprezentēšanai.

Atbilstoši kuba metaforai, dimensijas ir kuba asis.

Dimensijas tiek aprakstītas ar **dimensiju atribūtiem**. Visus vai daļu no dimensiju atribūtiem var sakārtot hierarhiski, veidojot klasifikācijas **hierarhijas**, kas ir būtiskas datu analīzē. Dimensijas atribūtus, kas veido hierarhijas, sauc par **hierarhiju līmeņiem** (skat. 2. att.). Dažādi līmeņi saskaņā ar [SBHD98] atbilst dažādām detalizācijas pakāpēm un dažādiem klasifikācijas veidiem.



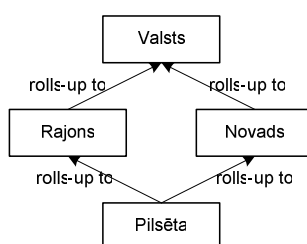
2. att. Hierarhiju līmeņu elementi

Semantiskā nozīme divu hierarhijas līmeņu saistībai ir sekojoša: tiek uzskatīts, ka līmenis L1 ir saistīts ar līmeni L2 agregācijas ("roll-up" - OLAP funkcija) vai detalizācijas ("drill-down"-OLAP funkcija) nolūkos, ja dotās lietotnes kontekstā ir jēga L1 līmeņa elementu klasifikācijai attiecībā pret L2 līmeņa elementiem.

Patvaļīgs līmenis var tikt klasificēts attiecībā pret jebkuru skaitu citu līmeņu, tādējādi vienā dimensijā var būt vairākas hierarhijas, protams, ja eksistē vairāki klasifikācijas veidi.

Hierarhijas līmenis var ietilpt vairākos **alternatīvos hierarhijas ceļos** (skat.3. att.), kas nozīmē, ka starp diviem dažādiem līmeņiem eksistē vairāki "roll-up" ceļi.

Daudzdimensiju skatījums uz datiem ir līdzīgs masīva jēdzienam, tomēr analogija nav precīza, jo masīvs atļauj dimensiju datus sakārtot lineāri kā masīva indeksus. Kuba metafora šo hierarhisko sakārtojumu nevar atspoguļot, tāpēc dimensiju atribūtu hierarhiskais sakārtojums tiek parasti reprezentēts atsevišķi. Dimensijas hierarhijas parasti attēlo kā orientētu, aciklisku grafu.



3. att. Alternatīvas hierarhijas reprezentācija ar modeli

Hierarhijas modelējot, būtiski ir ņemt vērā tādas īpašības kā pilnība un stingrība [TPGS01]. Pilnība nozīmē, ka visām kāda līmeņa instancēm hierarhijas zemākā detalizācijas pakāpē eksistē atbilstošs līmenis augstākā hierarhijas līmenī un ka augstāka līmeņa instance sastāv tikai no šīm zemākā līmeņa instancēm. Stingrība nozīmē, ka zemāka līmeņa objekts ir saistīts tikai ar vienu augstāka līmeņa objektu.

Cits veids, kā var tikt strukturētas dimensijas, ir dimensiju atribūti, kas ir aprakstoši atribūti dimensiju hierarhiju līmeņiem. Šie atribūti nav hierarhiski sakārtoti.

Datu analīzē šāda tipa atribūti dimensijās tiek izmantoti, lai ierobežotu analizējamo faktu kopu [TPGS01].

Sarežģīts daudzdimensiju modelis var sastāvēt no vairākiem kubiem [SBHD98], ja jāanalizē vairāki fakti, vai mērījumiem ir atšķirīgas dimensiju kopas. Šiem kubiem var būt kopīgas dimensijas, kaut arī var būt situācija, ka atšķiras kopīgās dimensijas detalizācija, kas katrā no kubiem tiek lietota.

Tā kā fakts ir viens no galvenajiem daudzdimensiju modeļa elementiem, ir būtiski, kā tiek definēts šis elements un kāda ir tā semantika. Dažādu autoru darbos fakta jēdziena semantika atšķiras. Daži piemēri formālām fakta definīcijām ir apskatīti apakšnodaļā par konceptuālajiem modeļiem, šeit kā piemēru minēšu fakta reprezentāciju [TPGS01], kas faktus uzskata par daudzpret-daudz attiecībām starp visām dimensijām un kā viens - pret -daudz attiecībām starp faktu un katru atsevišķu dimensiju. Jāpiezīmē, ka atsevišķos gadījumos var būt situācija, kad fakts reprezentē daudzpret-daudz attiecību tikai starp divām vai dažām atsevišķām dimensijām, bet ne starp visām faktu raksturojošām dimensijām.

2.3. OLAP operācijas

Lietotāja darbs ar datu noliktavu notiek ar OLAP operāciju palīdzību, tāpēc tālāk dots neliels ieskats OLAP operācijās.

Aprakstot lietotāja darbības ar OLAP lietotni, tiek izmantots navigācijas jēdziens, ar to apzīmējot lietotāja veikto kuba izpētes procesu, veicot interaktīvi dažādas OLAP operācijas, kuras rezultāts, ja izmanto daudzdimensiju kuba metaforu, ir kuba šūna, kuba divdimensiju vai daudzdimensiju apakškubs. Populārākās OLAP operācijas ir [JLVV03]:

- agregācija (konsolidēšana, arī "*roll-up*") ir vaicājums, lai iegūtu datu kopsavilkumus - tas nozīmē, ka, izmantojot hierarhiju līmeņu attiecības tām dimensijām, kas interesē lietotāju, jāatrod mazāk detalizētie dimensiju hierarhijas līmeņi, kā arī jāaprēķina jauns salikts mērījums atbilstoši atrastajiem mazāk detalizētajiem hierarhiju līmeņiem un izvēlētai agregācijas funkcijai (SUM, MAX u.c),

- detalizācija ("*drill-down*", "*roll-down*") ir vaicājums, lai iegūtu vairāk detalizētus datus, tā ir pretēja darbība agregācijai, tā arī izmanto hierarhiju līmeņu attiecības,

- *slicing* atlasa datus atbilstoši izvēlētu dimensiju fiksētām vērtībām. Piemēram, atlasīt 2003. gada pārdošanas datus pa reģioniem un produktu kategorijām (tiek fiksēta Laika dimensija). Dažādos literatūras avotos vēl atsevišķi izdala *dicing*, ja dimensiju skaits ir lielāks par vienu, *slicing* definējot vienai fiksētai dimensijai,

- *screening* ir operācija, kuras laikā dati tiek salīdzināti ar kādu definētu kritēriju, piemēram, "Atlasīt datus par precēm, kuru pārdošanas summas iepriekšējā mēnesī ir lielākas par X",

- *pivoting* (rotēšana) ir operācija, kas maina atskaitēs dimensiju asis vietām.

Jāuzsver, ka OLAP operācijas, īpaši tas attiecas uz agregāciju, ne visos gadījumos ir pieļaujamas, jo var iegūt nepareizus rezultātus. Tādēļ agregācijas iespējamība attiecībā pret dažādiem hierarhiju līmeņiem un dažādiem mērījumiem ir īpašs izpētes objekts un to nepieciešams atbilstoši formalizēt.

Lielākajai daļai fakti jābūt aditīviem [KRRT98], kas attiecīgi nozīmē, ka SUM agregācijas funkcija var tikt pielietota fakta atribūtiem attiecībā pret visām hierarhijām.

Fakta atribūts ir semi-aditīvs, ja tas nav aditīvs attiecībā pret visām dimensijām (piemēram, preču skaits noliktavā nav summējams attiecībā pret laiku, bet ir summējams attiecībā pret preces dimensiju). Ne-aditīvs fakts - nav summējams attiecībā pret visām dimensijām, bet var pielietot citas agregācijas funkcijas. Piemēram, temperatūras mērījumi ir ne-aditīvi fakti.

2.4. Datu noliktava kā materializēto skatu kopa

Lielā mērā datu noliktavas datu bāzes modelis ir atkarīgs no pieejamiem datu avotiem, to modeļiem, kā arī no datu avotu datu kvalitātes.

Viena no pieejām datu noliktavu jomā uzskata datu noliktavu par materializētu skatu kopu [GM95], [AASY97], [ZGMW98]. Skati tiek definēti, izmantojot datu avotus, bet materializācija nozīmē, ka skatu izpildes rezultātā iegūtie dati tiek glabāti, bet atsevišķā vietā.

Materializēto skatu pieejas gadījumā tiek aplūkoti n datu avoti. Tiek pieņemts, ka datu avotu modeļi ir relāciju modeļi. Katrs no modeļiem var sastāvēt no jebkura skaita relācijām, bet, definējot modeli, tiek pieņemts, ka datu avotam i eksistē vismaz viena bāzes relācija R_i , kas tiek izmantota materializētā skata definēšanai. Skata funkcija vienkāršākajā gadījumā tiek definēta ar SPJ-izteiksmi (*select-projection-join*).

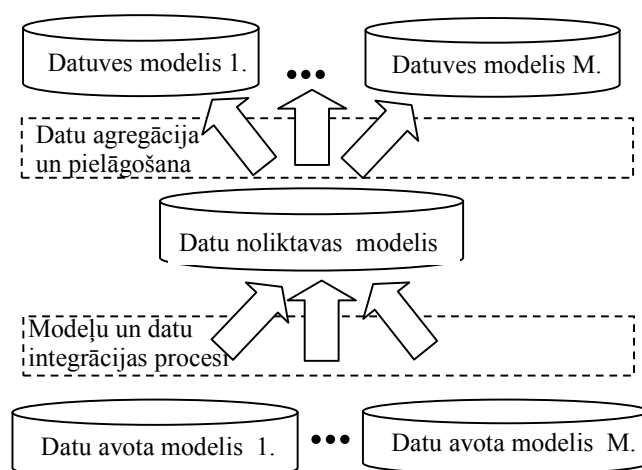
$\{R_1, \dots, R_i, \dots, R_n\}$ ir atbilstošo n datu avotu bāzes relāciju kopa, skats V tiek definēts sekojoši:

$$\prod_{\text{ProjAttr}} \sigma_{\text{SelectCond}} (R_1 \bowtie \dots \bowtie R_i \bowtie \dots \bowtie R_n)$$

Sarežģītākos gadījumos datu noliktavas skats var tikt definēts ar sarežģītākām funkcijām nekā SPJ-izteiksme, piemēram, izmantojot datu agregācijas funkcijas.

Var uzskatīt, ka datu noliktavā materializēto skatu kopa ir izkārtota vairākos līmeņos, kur dati augstākā līmenī tiek iegūti no datiem iepriekšējā līmenī [JLVV03]. Pirmo, zemāko datu līmeni veido datu avoti, kam var būt atšķirīgi datu modeļi, tāpēc ir iespējami arī semantiski konflikti, kur vieni un tie paši dati tiek reprezentēti ar dažādiem modeļiem. Nākošais datu līmenis ir „globālās datu noliktavas”, sauktas arī par „korporatīvo datu noliktavu”, kurai modelis iegūstams transformācijas procesu rezultātā integrējot datu avotu modeļus. Nākošais datu līmenis ir datuves jeb OLAP datu bāzes, kas orientētas dažādu lokālu uzņēmuma problēmu risināšanai, piemēram, departamentam piederoši dati, departamenta funkcijām atbilstoši dati – tās ir neliela apjoma datu noliktavas, kas satur apakškopu no „globālās datu noliktavas” datiem.

Saistību starp datu noliktavas komponentu modeļiem atspoguļo zīmējums (skat. 4. att.).



4. att. **Datu noliktavas komponentu modeļu saistība**

Datu noliktavas projektēšanā, ņemot vērā iepriekš aprakstītās modeļu atkarības, var aplūkot virkni problēmu:

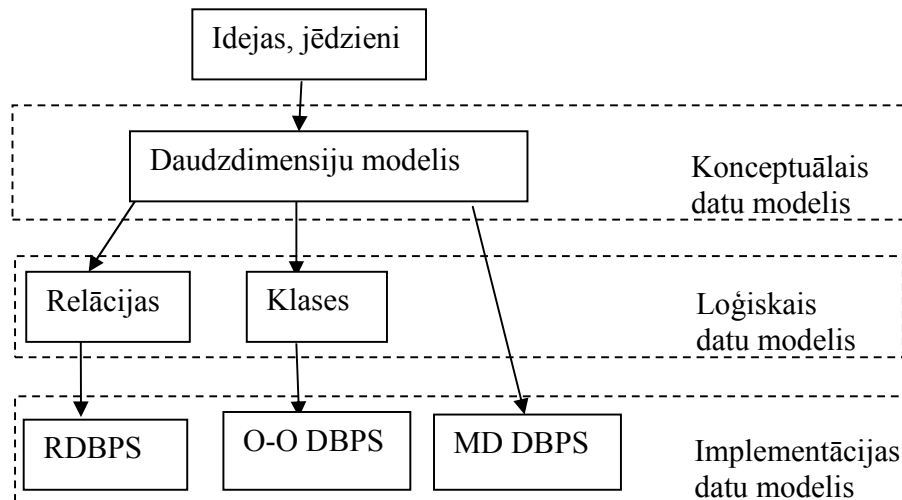
- datu semantikas saskaņošana no dažādiem avotiem un datu noliktavas „globālā modeļa” izveide,
- modeļu izmaiņas – uzņēmuma biznesa likumu maiņa ietekmē gan datu avotu modeļus, gan prasības datu noliktavai, līdz ar to datu noliktavas modeli.

Datu noliktavas projektēšana ir process, kas turpinās, kas iteratīvi tiek veikts visā datu noliktavas dzīves cikla laikā [KRRT98]. Būtiska ir arī pieejas izvēle – “*top-down*” vai “*bottom-up*” [KRRT98] – “*top-down*” pieejā tiek veidots globālais datu modelis, kas integrē datu avotu modeļus un vēlāk arī datus. Pieeja lēna, dārga, bet vēlāk vieglāk realizēt izmaiņas. “*Bottom-up*” pieeja pieļauj vienkāršāku, ātrāku risinājumu – atsevišķu specializētu datuvju izstrādi un vēlāk līdz ar datu noliktavas attīstību, tiek risināti šo datuvju saskaņošanas jautājumi.

Datu noliktavu uzskatīšana par materializēto skatu kopu atspoguļo vienu būtisku datu noliktavu īpašību - datu integrāciju no daudziem avotiem, taču datu analīzes vajadzību nodrošināšanai ir būtiski noskaidrot datu noliktavas modeļa elementu semantiku, kas šajā pieejā netiek ņemta vērā, kā arī analīzes prasības, ko arī nav iespējams izsecināt tikai no datu avotiem.

2.5. Datu noliktavu konceptuālie modeļi

Konceptuālie modeļi ir saprotamāki lietotājam, apraksta daudzdimensiju modeli neatkarīgi no tālākās realizācijas. Loģiskie modeļi apskata daudzdimensiju paradigmas elementus, atkarībā no realizācijai paredzamā datu bāzu pārvaldības sistēmas veida, izmantojot relācijas vai klases. Realizācijas modeļi ir specifiski konkrētai DBPS un apskata informācijas optimālas glabāšanas, indeksēšanas u.c. ar realizāciju saistītas problēmas. Šo modeļu līmeņu savstarpējā saistība redzama attēlā. (skat. 5. att.).



5. att. **Datu noliktavu modeļu līmeņi**

Pastāv uzskats, ko atbalsta datu noliktavu konceptuālo modeļu autori, tajā skaitā, piemēram, [SBHD98], ka eksistējošās konceptuālās modelēšanas metodes, kas tiek lietotas relāciju (piemēram, ER modelis [Chen76]) vai objekt-orientētām sistēmām, nenodrošina pietiekamu atbalstu daudzdimensiju modeļa attēlošanai intuitīvā veidā, kā arī, tās lietojot, tiek zaudēta daļa daudzdimensiju modeļa semantikas. Vajadzīgā semantika tad papildus jāpieraksta neformāli, kas šādus modeļus padara, piemēram, nepiemērotas automātiskai ģenerēšanai. Dažādu autoru konceptuālie modeļi, piemēram, [SBHD98], [TBC99], [LTS02] ir ar dažādām izteiksmes iespējām (*expressiveness*), kā to parāda konceptuālo modeļu salīdzinājumi.

Literatūrā atrodami vairāki daudzdimensiju modeļu salīdzinājumi [BSHD98], [Ped00], [ASS01], kas izmanto dažādus modeļu vērtēšanas kritērijus. Atšķiras arī šajos salīdzinājumos apskatīto modeļu kopas.

[BSHD98] piedāvātais salīdzinājums apskata prasības formālam konceptuālam modelim, lai tas būtu izmantojams OLAP lietotņu modelēšanai:

- kuba struktūras un tā satura tieša nodalīšana,
- sarežģītu dimensiju struktūru modelēšanas iespējas:
 - līmeņi, līmeņu instances, formālistiski līmeņu struktūras aprakstīšanai,
 - dimensiju atribūti, kas neapraksta hierarhijas,
- dimensiju un mērījumu simetrija (pamatideja - katrs dimensijas atribūts var tikt uzlūkots par mērījumu un analizēts),
 - sarežģītāki mērījumi - aprēķinātie mērījumi, to aprakstīšana, mērījumu (faktu) aditivitātes īpašību aprakstīšana,
 - OLAP operāciju aprakstīšanas iespējas (piemēram, algebra).

[Ped00] konceptuālos daudzdimensiju modeļus analizē attiecībā pret pamatprasībām un papildus prasībām. Pamatprasības šajā salīdzinājumā ir līdzīgas [BSHD98] formulētajiem nosacījumiem, bet papildus prasības [Ped00] autors formulējis, balstoties uz pieredzi medicīnas datu datu noliktavu izstrādē:

- nestingras hierarhijas (*Non-strict*) - starp hierarhiju līmeņiem var eksistēt m:n attiecības (pārklājošas klasifikācijas),
- nesabalansēti instanču koki hierarhijās (*Non-onto*) - dažādiem ierakstiem dimensijā ceļš no “saknes” līdz “lapām” ir dažāda garuma, var neeksistēt “apakšējie” līmeņi,
- nepārklājošas hierarhijas (*Non-covering*) - pa vidu starp hierarhijas “sakni” un “lapām” var tikt izlaisti atsevišķi hierarhijas līmeņi – apdzīvoto vietu hierarhija, struktūrvienību hierarhijas,
- jāvar attēlot m:n attiecības starp faktiem un dimensijām,
- jāvar attēlot modelī atribūtu vērtību izmaiņas laikā - literatūrā pazīstamā “lēni mainīgo dimensiju” problēma arī pārstāv šo prasību.

Salīdzinājuma [ASS01] autori apskata trīs datu modeļu līmeņus atkarībā no tajos izmantotajiem jēdzieniem un no tā, kurā datu noliktavas izstrādes fāzē tie tiek izmantoti - konceptuālos, loģiskos un fiziskos. Autori papildus ievieš ceturto modeļu grupu - "formālais līmenis", tajā iedala modeļus, kas apraksta teorētiskos pamatojumus daudzdimensiju modelim, ietverot arī algebras.

Salīdzinājuma [ASS01] autori uzskata, ka daudzdimensiju modeļus var apskatīt trīs detalizācijas līmeņos:

- augstākajā - apskata dimensijas un faktus; iespēju navigēt no vienas shēmas otrā apraksta kopējas dimensijas,
- vidējā - dimensijas tiek detalizētas dimensiju līmeņos, kas veido hierarhijas, un fakti tiek detalizēti faktu šūnās, kas satur datu jeb mērījumu kopu atbilstoši faktu šūnu nosakošajiem dimensiju līmeņiem,
- zemākajā - atspoguļo dimensiju līmeņu atribūtus - klasifikācijas atribūtus un faktu atribūtus - mērījumus.

Autori šo klasifikāciju izveido, lai būtu iespējams salīdzināt daudzus konceptuālos modeļus, kuros daudzdimensiju modeļu jēdzieni apskatīti dažādās detalizācijas pakāpēs.

Pieņemot, ka pēc noklusēšanas modelī jau ir pamatelementi - fakts, mērījums, dimensija, hierarhija, [ASS01] autori vērtē sekojošas jaunas daudzdimensiju modeļa īpašības:

- vai modelī iespējams sīkāk aprakstīt mērījumus;
- vai ir paredzēti klasifikācijas atribūti;
- vai var modelī atspoguļot attiecības starp mērījumiem un klasifikācijas atribūtiem;
- vai dimensijām tiešā veidā tiek modelēti dimensiju līmeņi;
- vai eksistē fakta šūnas jēdziens, t.i., vai mērījumi tiek grupēti;
- vai var modelēt attiecības starp fakta šūnu un dimensiju līmeņiem.

Visos iepriekš apskatītajos modeļu salīdzinājumos [Ped00],[ASS01],[BSHD98] tikai daži modeļi ir analizēti visās klasifikācijās konceptuālā līmenī, piemēram, Cabbibo un Torlones modelis [CT98] un Lehnera modelis [Leh98]. Pārējie aplūkoti modeļi katrā salīdzinājumā ir atšķirīgi. Iemesli šādai salīdzinājumu autoru izvēlēto modeļu dažādībai ir vairāki. Eksistē daudz dažādu, vairāk vai mazāk formalizētu modeļu, kas apraksta daudzdimensiju paradigmas elementus, bez tam [ASS01] analizē jaunākus modeļus nekā abi pārējie salīdzinājumi

[Ped00],[BSHD98]. Svarīgi ir arī tas, ko katras klasifikācijas autors izprot ar konceptuālo modeli. Piemēram, [Ped00] pieskaita Kimbala modeli ar zvaigznes shēmu [KRRT98] pie konceptuāliem modeļiem, bet [ASS01] pieskaita šo modeli loģisko modeļu kategorijai. Savukārt [AGS97] modelis, piemēram, salīdzinājumā [ASS01] pieder formālo modeļu grupai, bet abos pārējos salīdzinājumos [Ped00],[BSHD98] šis modelis ir pieskaitīts konceptuāliem modeļiem.

Secinājumi:

- daudzdimensiju konceptuālie modeļi ir grūti salīdzināmi, jo mēdz apskatīt jēdzienus dažādās detalizācijas pakāpēs, ko [ASS01] mēģina risināt, ieviešot modeļu detalizācijas līmeņus;

- atšķiras autoru interpretācijas, kas ir datu noliktavu konceptuālais modelis, tas attiecas ne tikai uz šo salīdzinājumu autoriem, bet arī uz modeļu autoriem. Tā, piemēram, [CT98] modeli visas apskatītās klasifikācijas pieskaita pie konceptuālajiem modeļiem, bet paši modeļa autori uzskata to par loģisko modeli.

- lielākā daļa daudzdimensiju konceptuālo modeļu apmierina vairākumu no pamatprasībām, [Ped00], [BSHD98], bet papildus prasības pēc [Ped00] tiek apskatītas tikai atsevišķās pieejās un parasti ir tikai netieši izsecināmas;

- pēc [ASS01], [Ped00] un [BSHD98] salīdzinājumu rezultātiem, daudzdimensiju konceptuālie modeļi ir ar dažādu ietverto elementu un īpašību kopu, līdz ar to izmantojot dažus modeļus, var pietrūkt būtiskas īpašības, kas nepieciešamas kādas konkrētas sfēras daudzdimensiju modeļa izveidei. Modeļos ietverto elementu un to īpašību dažādība atspoguļo viedokļu atšķirības jautājumā par to, kādam jābūt daudzdimensiju konceptuālam modelim, un kā sekas, joprojām nav vienprātības par standartu [RALT06];

- pēc [LTS02] autoru domām, iepriekš pieminētās daudzdimensiju modeļa īpašības tiek neprecīzi interpretētas gandrīz visos konceptuālos daudzdimensiju modeļos.

Ir iespējamās divas pieejas konceptuālā modeļa izveidei: veidot modeli no jauna, kas nozīmē arī papildus darbu šī modeļa formālam pamatojumam, vai izmantot kādu esošu modeli un modificēt to tā, lai varētu izmantot daudzdimensiju paradigmas raksturlielumu izteikšanai.

Tādēļ datu noliktavu konceptuālos modeļus pēc izveides tipa var klasificēt sekojošās grupās [RALT06]:

- uz E/R modeli balstītie konceptuālie modeļi - piemēram, ME/R [SBHD98], Star E/R [TBC99];

- uz UML balstītie konceptuālie modeļi - piemēram, izmantojot UML stereotipus [LTS02], [LTV04];

- dažādu autoru piedāvāti neatkarīgi konceptuālie modeļi - piemēram, *Dimensional Fact Model* [GMR98a], [GMR98b], ADAPT [Bul98].

Šajā klasifikācijā pieminētie konceptuālie modeļi, izņemot ADAPT [Bul98], promocijas darbā aplūkoti detalizētāk.

Var definēt grafu $RG=(E,V)$, kur E - visu dimensijas līmeņu $e_1 \dots e_k$ kopa un

$$V = \{ (e_i, e_j) \mid i \neq j \wedge 1 \leq i, j \leq k \wedge e_i \text{ "rolls-up to" } e_j \}.$$

RG ir orientēts, aciklisks grafs. "Roll-up" attiecības vārds apraksta kritēriju klasificēšanai.

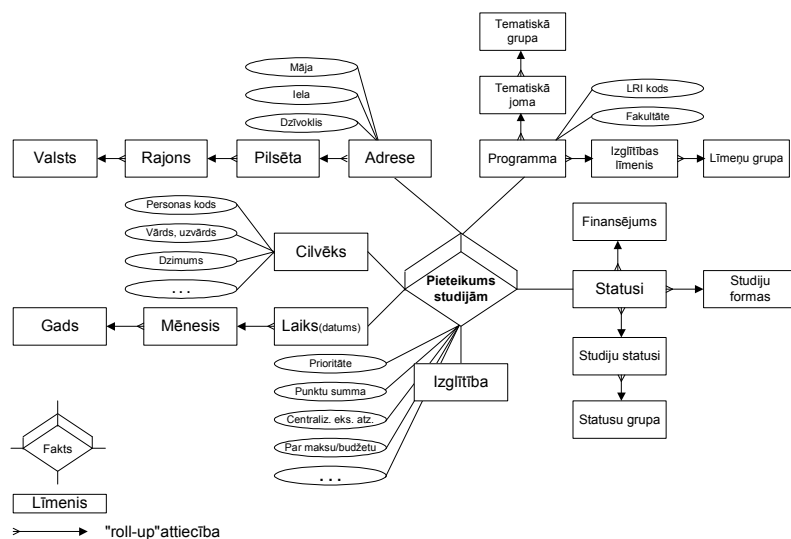
Fakta attiecību kopa ir N -āras attiecības specializācija, kas sasaista N dažādas dimensiju līmeņu entītijas. Tieši faktam piesaistītās dimensiju līmeņu entītijas sauc par atomāriem dimensiju līmeņiem. Fakta attiecību kopas atribūti modelē mērījumus par reālās pasaules faktiem (kvantitatīviem datiem).

Saprotamībai ME/R jaunajiem elementiem tiek ieviesti grafiski apzīmējumi (skat.7. att.).

Dimensijai kā jēdzienam nav tieši atbilstoša elementa ME/R modelī, jo dimensiju var definēt kā dimensiju līmeņu kopu. Fakta attiecību kopa definē N atomāros līmeņus e_{i1}, \dots, e_{in} . Atbilstošā dimensija D_k tad tiek definēta ar $RG(E,V)$ apakšgrafu, ko savukārt definē atbilstošs atomārais dimensijas līmenis.

Ar ME/R var modelēt dimensiju hierarhisko struktūru, vairākas paralēlas hierarhijas, kopējus hierarhijas līmeņus vairākām hierarhijām. Dimensiju atribūti tiek modelēti kā dimensiju līmeņu atribūti. Iespējams arī izveidot vairāku kubu modeli, definējot vairākus faktus un ietverot arī šajā modelī informāciju par detalizācijas pakāpi dimensijām, kādā tās tiek koplietotas. Faktu atribūti var būt arī aprēķināmi mērījumi, tiesa, nav iespējams šajā modelī ietvert aprēķināšanas veidu, jo ME/R, līdzīgi kā ER, atspoguļo statistisku informāciju par modelējamo problēmu apgabalu. Mērījumu aprēķināšana būtu jāiekļauj funkcionālā modelī.

Piemērs ME/R pielietojumam ir augstskolas reflektantu datu konceptuālais modelis (skat. 7. att.):



7. att. ME/R pielietojuma piemērs

2.5.1.2. StarER modelis

StarER modeļa [TBC99] īpatnība ir mazliet atšķirīga fakta semantika. Fakts atspoguļo reālās pasaules faktus, kas var tikt uzlūkoti par procesiem, kas laika gaitā ģenerē datus.

Vēl StarER modeli raksturo faktu atribūtu analīze un modelēšana attiecībā pret īpašību, vai fakti var tikt summēti vai agregēti ar citām funkcijām, lai iegūtu jaunu informāciju. [TBC99] attiecībā pret agregāciju apskata 3 tipu faktu atribūtus: krājums (*stock*), plūsma (*flow*) un relatīvā vērtība (*value-per-unit*).

- *Stock* (S)- raksturo kāda lieluma stāvokli noteiktā laika momentā. Tos var uztvert kā kāda noteikta parametra momentuzņēmumus, ko monitorē datu noliktava. Šāda tipa atribūti ir agregējami attiecībā pret laika dimensiju,

- *Flow* (F)- fiksē kāda monitorējama apkārtējās vides parametra izmaiņu lielumu laika periodā. Šāda tipa faktu atribūti parasti ir aditīvi jeb summējami pret jebkuru dimensiju,

- *Value-per-unit* (U) - lielums līdzīgs kā "*stock*", bet dotā vērtība tiek mērīta attiecībā pret kādu citu lielumu, kas ir mainīgs laikā.

Par pamatu [TBC99] savam StarER modelim izmanto ER modeli. Tiek izmantoti sekojoši StarER modeļa elementi:

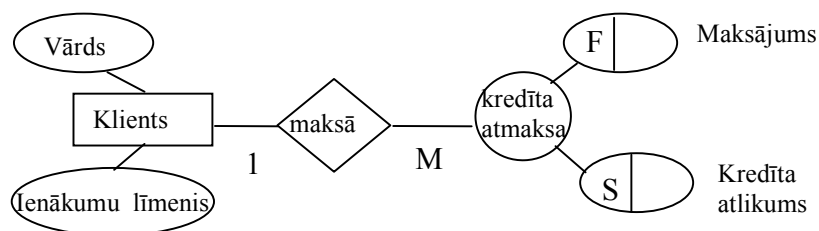
- faktu kopa: reprezentē reālās pasaules faktus, kam ir vienādi faktu atribūti un vienādas īpašības. Semantika - fakts ir process, kas ģenerē datus noteiktos laika momentos; fakti apzīmē ar apli,

- entītiņu kopa: reprezentē reālās pasaules objektus ar līdzīgām īpašībām; apzīmē ar taisnstūri,

- attiecību kopa: reprezentē asociācijas starp entītiņu kopām vai starp entītiņu kopu un faktu kopu.

- atribūti: statistiskas entītiņu kopu, attiecību kopu un/vai faktu kopu īpašības. Atribūtiem var norādīt to īpašības attiecībā pret agregācijas iespēju, apzīmējot atribūta tipu ar S ("*Stock*"), F ("*Flow*") vai V ("*value-per-unit*").

Piemērs StarER notācījas izmantošanai datu noliktavas konceptuālai modelēšanai redzams attēlā (skat.8. att.).



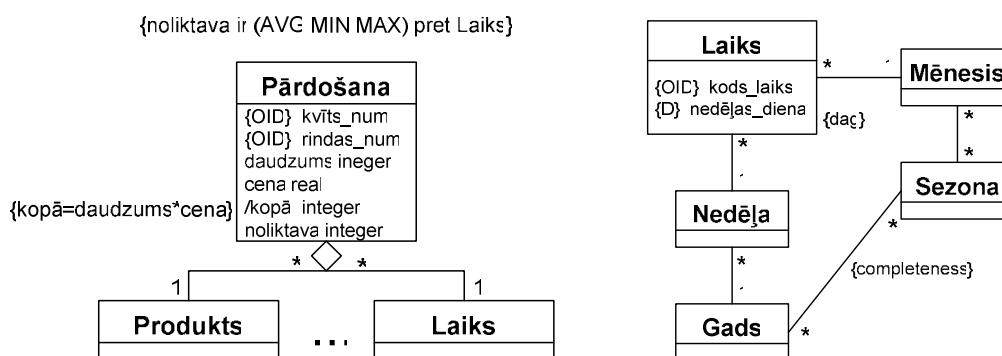
8.att. Piemērs StarER notācījas lietošanai

2.5.2. Uz UML balsīti datu noliktavu konceptuālie modeļi

[LTS02] autori apraksta UML profilu daudzdimensiju modelēšanai, bet pati pieeja klašu diagrammu izmantošanai daudzdimensiju modelēšanai jau iepriekš aprakstīta [TPGS01], kas aplūko būtiskas daudzdimensiju modeļa īpašības konceptuālā līmenī, kā m:n attiecības starp faktiem un dimensijām, deģenerējušās dimensijas, daudzas paralēlas un alternatīvas hierarhijas dimensijās, nestingras un pilnas hierarhijas. Ar OCL tiek aprakstītas elementu īpašības.

[LTS02] un [TPGS01] autoru pieejā daudzdimensiju modeļa strukturālās īpašības tiek modelētas ar UML klašu diagrammas palīdzību. Piemēri pielietojumam redzami attēlā (skat.9.att.). Fakti un dimensijas tiek attēlotas ar faktu klasēm un dimensiju klasēm. Katra faktu klase tiek definēta kā n dimensiju klašu agregācijas attiecība. Ja starp kādu no dimensijām un faktu ir m:n attiecība, tad diagrammā to var attēlot, norādot 1..* kardinalitāti dimensiju klases lomai.

Tiek uzskatīts, ka visi mērījumi faktu klasē ir aditīvi. Neaditīviem mērījumiem tiek definēti likumi, kas tiek ietverti fakta klasē. Ja mērījumi tiek aprēķināti, tad mērījumu apzīmē ar / un aprēķināšanas likumus var aprakstīt blakus fakta klasei, liekot {} iekavās.



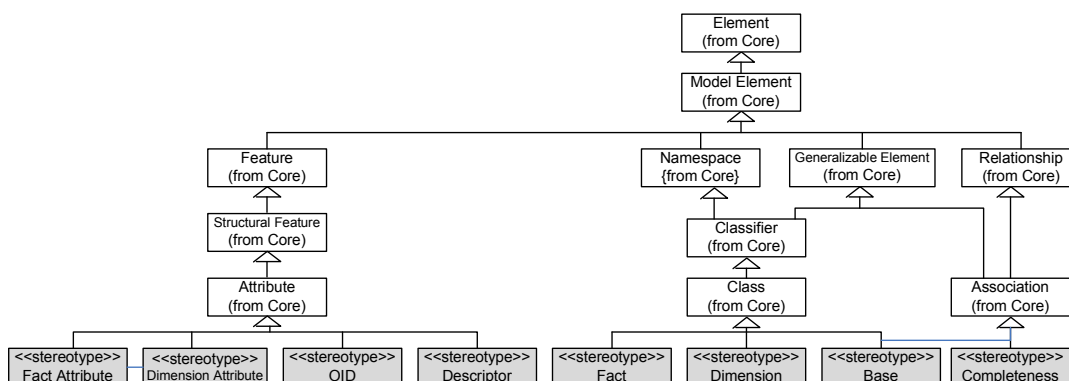
9.att. UML profila lietošanas piemēri daudzdimensiju modelēšanā, pēc [LTS02]

Identificējošiem atribūtiem blakus nosaukumam tiek pierakstīts {OID}, tādejādi faktu tabulai bez mērījumiem parādās jaunas īpašības - var tikt aprakstīta arī deģenerējusies dimensija (dimensija, kuras vienīgais atribūts ir kods) - "Kvīts", ietverot tās atribūtu "Kvīts_num" faktu tabulā, aprakstot to ar {OID}.

Dimensijās katrs hierarhijas līmenis tiek uzdots ar klasi, [LTS02] autori tās sauc par bāzes klasēm. Klašu asociācijas apzīmē attiecības starp diviem hierarhijas līmeņiem. Vienīgais nosacījums ir tāds, ka bāzes klasēm jādefinē orientēts aciklisks grafs (DAG - *Directed Acyclic Graph*), ar dimensijas klasi kā sakni - {dag} tiek pierakstīts klāt katrai dimensijas klasei. DAG ļauj reprezentēt gan alternatīvus ceļus, gan vairākas paralēlas hierarhijas.

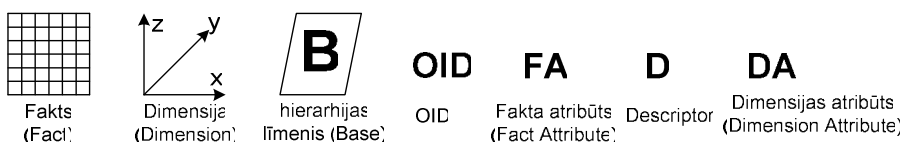
Katram hierarhijas līmenim tiek definēts identifikators {OID} un aprakstošs atribūts {D} (D- *descriptor*). Šis nosacījums gan neattiecas uz daudzdimensiju modeļu īpašību attēlošanu, bet uz modeļa izmantojamību automātiskai ģenerēšanai. Asociācijām norādot klašu lomās kardinalitātes 1 vai 1..*, tiek aprakstītas hierarhiju stingrības īpašība. Definējot {completeness} ierobežojumu asociācijām, tiek aprakstīta hierarhiju pilnības īpašība. Autori pēc noklusēšanas pieņem, ka hierarhijām pilnības īpašība nav spēkā.

[LTS02] autoru piedāvātais UML paplašinājums (skat.10.att.) daudzdimensiju modeļu aprakstīšanai ietver stereotipu (*Stereotype*) un ierobežojumu kopu.



10.att. UML paplašinājums daudzdimensiju modeļu aprakstīšanai, pēc [LTS02]

UML paplašinājumā (skat.10.att.) ir definēti astoņi stereotipi, kuru grafiskie attēli redzami attēlā (skat.11.att.) - fakts, dimensija un bāze, kas ir klases elementa specializācijas, fakta atribūts, dimensijas atribūts, OID un *Descriptor*, kas ir atribūta elementa specializācijas, kā arī definēta viena asociācijas elementa specializācija - *Completeness* (pilnība).



11. att. Daudzdimensiju modeļa elementu grafiskie apzīmējumi pēc [LTS02].

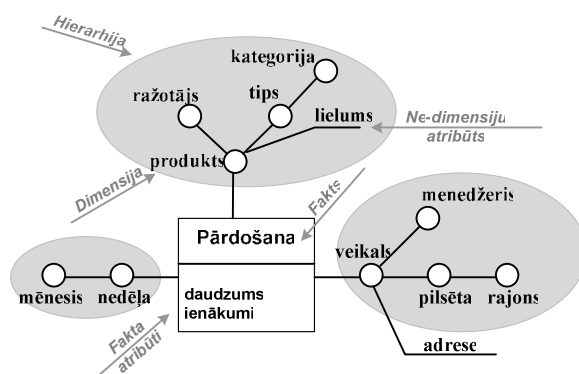
Katrs stereotips tiek definēts, norādot tā bāzes klasi, neformālu aprakstu, grafisko attēlu, kā arī ierobežojumus, lai formalizētu katra stereotipa lietošanas legālas iespējas atbilstoši daudzdimensiju modeļa semantikai. Šim nolūkam autori izmanto OCL. Piemēram, Fakta stereotipam autori apraksta, ka katrs fakts var būt saistīts tikai ar dimensiju klasēm:

```
self.allOppositeAssociationEnds->forAll(participant.ocllsTypeOf(Dimension)).
```

2.5.3. Neatkarīgi konceptuālie modeļi

Šajā apakšgrupā ietilpst lielākā daļa datu noliktavu konceptuālo modeļu, bet kā piemērs tiks apskatīts viens no tiem - DFM (*Dimensional Fact Model*). Datu noliktavas modelis DFM [GMR98b] ir konceptuālā modeļa piemērs, kas ir pilnībā autoru definēts (nav esošu modeļu paplašinājums).

DFM sastāv no vairākām faktu shēmām. Shēmu pamatelementi ir fakti, atribūti, dimensijas un hierarhijas. Šajā modelī ir iespējams attēlot faktu atribūtu aditivitātes īpašības, dimensiju atribūtu obligātumu un atribūtus, kas neveido dimensiju hierarhijas, kā arī daļēji pārklājošas faktu shēmas. Pamatelementi semantikas ziņā atbilst promocijas darba 2.2. apakšnodaļā dotajiem daudzdimensiju paradigmas elementu aprakstiem. Zīmējumā (skat.12.att.) dots daudzdimensiju modeļa piemērs, izmantojot DFM grafisko notāciju. Faktu apraksta ar tā vārdu un parasti ar skaitliskiem atribūtiem, kam periodiski tiek piešķirtas vērtības, kas būtībā nozīmē, ka fakts tiek mērīts no dažādām perspektīvām.

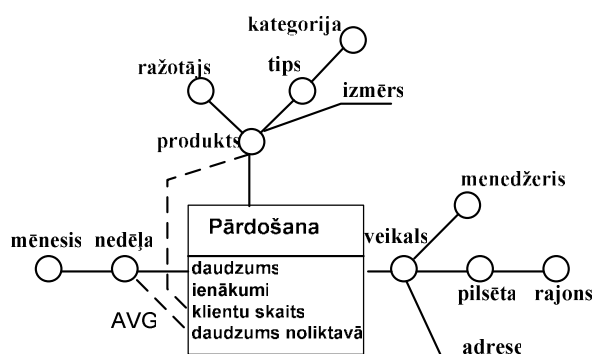


12. att. DFM notācijas lietošanas piemērs pēc [GMR98b]

Faktu shēma ir strukturēta koka veidā, kuras sakne ir fakts. Katra koka virsotne, kas tieši piesaistīta faktam, ir dimensija. Apakškoki, kam saknes ir dimensijas definē hierarhijas, kuru virsotnes (apzīmē ar apli) ir atribūti ar diskretu vērtību kopu, bet šķautnes reprezentē $m:1$ attiecības starp atribūtu pāriem. Ir arī atribūti, ko [GMR98b] sauc par ne-dimensiju atribūtiem, bet citi autori, piemēram, [SBHD98], gluži pretēji - par dimensiju līmeņu atribūtiem. Šos atribūtus apzīmē ar līniju, piemēram, "adrese", un tie satur papildus informāciju par kādu hierarhijas atribūtu, bet netiek izmantota fakta agregācijai, jo tādai agregācijai nav saturīgas nozīmes. Dažas šķautnes hierarhijās var būt apzīmētas ar pārtrauktu līniju, kas nozīmē neobligātu attiecību starp hierarhijas virsotnēm.

Fakts reprezentē $m:n$ attiecību starp dimensijām. Katra dimensiju vērtību kombinācija definē fakta instanci, ko raksturo tieši viena katra fakta atribūta vērtība.

DFM pēc noklusēšanas tiek pieņemts, ka fakta atribūti ir aditīvi attiecībā pret visām dimensijām, semi-aditivitāte tiek definēta tieši norādot ar pārtrauktu līniju (skat.13.att.), kurš fakta atribūts attiecībā pret kuru dimensijas hierarhijas atribūtu nevar tikt summēts, ja var pielietot citu operāciju, to jānorāda.



13.att. Fakta semi-aditivitātes attēlojums modelī, pēc [GMR98b]

DFM formālais modelis tiek definēts sekojoši [GMR98b]: ja $g=(V,E)$ ir orientēts, aciklisks un vāji sakarīgs grafs, tad var teikt, ka g ir kvazi - koks (*quasi-tree*) ar sakni $v_0 \in V$, ja jebkura cita

virсотne $v_j \in V$ var tikt sasniegta no v_0 vismaz pa vienu ceļu. Ar $\text{path}_{ij}(g) \subseteq g$ tiks apzīmēts ceļš (ja tāds eksistē), kura sākuma virсотne ir v_i un beigu virсотne v_j un ar $\text{sub}(v_i) \subset g$ apzīmē kvazi-koku ar sakni virсотnē $v_i \neq v_0$.

[GMR98b] tiek definēta fakta shēma: fakta shēma ir kortežs $f=(M,A,N,R,O,S)$, kur:

- M skaitlisku vai Būla izteiksmju kopa, kas izmanto vērtības no operacionālajām sistēmām (datu avotiem). Katrs $m_i \in M$ tiek saukts par mērījumu,

- A ir identifikatoru kopa, ko sauc par dimensiju atribūtiem. Katram $a_i \in A$ atbilst diskrēta vērtību kopa $\text{Dom}(a_i)$,

- N ir identifikatoru kopa, kas tiek saukti par ne-dimensiju atribūtiem,

- R ir sakārtotu pāru kopa. Katrs pāris (a_i, a_j) , kur $a_i \in A \cup \{a_0\}$ un $a_j \in A \cup N$ ($a_i \neq a_j$), tādi, ka grafs $\text{qt}(f) = (A \cup N \cup \{a_0\}, R)$ ir kvazi-koks ar sakni a_0 . a_0 ir mākslīgs atribūts, kas apzīmē faktu, ap kuru ir centrēta visa shēma. Pāris (a_i, a_j) modelē m:1 attiecību starp atribūtiem a_i un a_j .

Fakta shēmas f "dimensiju komplekts" (*dimension pattern*) ir kopa $\text{Dim}(f) = \{a_i \in A \mid (a_0, a_i) \in R\}$; katrs $\text{Dim}(f)$ elements tiek saukts par dimensiju. Ja grib uzsvērt, ka dimensijas atribūts a_i ir dimensija, to apzīmē ar d_i . Par dimensijas $d_i \in \text{Dim}(f)$ hierarhiju tiek saukts kvazi-koks $\text{sub}(\text{qt}(f), d_i)$.

- $O \subset R$ ir neobligātu attiecību kopa,

- S ir agregācijas apgalvojumu kopa, kur katrs elements ir trijnieks (m_j, d_i, Ω) , kur $m_j \in M$, $d_i \in \text{Dim}(f)$ un $\Omega \in \{\text{'SUM'}, \text{'AVG'}, \text{'COUNT'}, \text{'MIN'}, \text{'MAX'}, \text{'AND'}, \text{'OR'}, \dots\}$ ir agregācijas operators. Ja $(m_j, d_i, \Omega) \in S$, tad tas nozīmē, ka mērījums m_j var tikt agregēts attiecībā pret dimensiju d_i ar grupēšanas operatoru Ω .

Fakta shēmas instances tiek definētas sekojoši [GMR98b]:

- primārās fakta instances ir informācijas pamatvienības, kas attēlotas datu noliktavā, ko raksturo tieši viena vērtība katram mērījumam. Ar $\text{pf}(\alpha_1, \dots, \alpha_n)$ apzīmē primārā fakta instances, kas atbilst dimensiju vērtību kombinācijām $(\alpha_1, \dots, \alpha_n) \in \text{Dom}(d_1) \times \dots \times \text{Dom}(d_n)$;

- sekundārās fakta instances ir primāro fakta instanču agregācijas, kas balstītas uz atļautiem dimensiju agregācijas modeļiem.

DFM autori piedāvā arī pusautomātisku metodi kā modeli uzbūvēt, metodes idejas aprakstu skatīt apakšnodaļā 3.4.

2.6. Datu noliktavu metamodeļu standarts CWM un konceptuālie modeļi

Datu noliktavu metamodeļu standarts *Common Warehouse Metamodel* (CWM) pēc būtības ir vairāku metamodeļu kopa, kas apraksta dažādus datu noliktavu aspektus - daudzdimensiju kubus, ETL (datu sagatavošanas) procesus, datu noliktavu relāciju implementācijas u.c.

CWM ir OMG (*Object Management Group*) platformneatkarīga metamodeļu specifikācija [OMGc], [PCTM03], lai nodrošinātu datu noliktavu metadatu apmaiņu starp dažādiem rīkiem un platformām, tā pamatā ir trīs standarti:

- MOF (*Meta Object Facility*), OMG metamodeļu standarts;
- UML (*Unified Modeling Language*), OMG modelēšanas standarts, kas definē objekt-orientētu modelēšanas valodu;
- XMI (*XML Metadata Interchange*), OMG metadatu apmaiņas standarts, kas nodrošina metadatu apmaiņu XML formātā.

Uz MOF balstītu modeļu, tajā skaitā CWM, attēlošanai var lietot UML. CWM sastāv no 21 pakotnes, katra apraksta kādu no datu noliktavas aspektiem. Pakotnes grupētas līmeņos. Daudzdimensiju modeļa īpašības pamatā apraksta "Analīzes līmeņa" OLAP pakotne.

Eksistē pētījumi, piemēram, [MT02], kur aplūkota iespēja attēlot daudzdimensiju modeli konceptuālā līmenī par semantiski ekvivalentiem CWM metamodeļa elementiem. Kā daudzdimensiju modelis izmantota [TPGS01] aprakstītā objekt-orientētā pieeja. [MT02] sekmīgi definē atbilstības, bet jāpiezīmē, ka viņi apskata tikai daudzdimensiju modeļa pamatelementus.

Metamodeļi veidoti kā vispārīgas, ārējas reprezentācijas kopīgajiem metadatiem ar mērķi nodrošināt datu apmaiņu. Tomēr tie nespēj pilnībā atspoguļot daudzdimensiju konceptuālajos modeļos ietverto semantiku, tāpēc nav iespējama efektīva to izmantošana dažādu datu noliktavu integrēšanai [RALT06].

3. EKSISTĒJOŠU METOŽU ANALĪZE DATU NOLIKTAVU KONCEPTUĀLO MODEĻU IZVEIDEI

3.1. Nodaļas nolūks

Nodaļā tiek dota metodes definīcija. Datu noliktavu konceptuālo modeļu izstrādei eksistē vairākas pieejas, kā noskaidrot prasības datu noliktavas modelim un šo modeli izveidot. Nodaļā dota klasifikācija pieejām un dots pārskats par katras pieejas eksistējošām metodēm. Nodaļā tiek analizēti pieeju trūkumi, lai pamatotu jaunu metožu izstrādes nepieciešamību. Nodaļā analizētie pieeju pozitīvie aspekti un pieejām atbilstošu metožu eksistence pamato promocijas darba ideju par esošo metožu atsevišķu komponentu izmantošanu piemērotā situācijā. Metodes, kuru komponenti ir izmantoti promocijas darbā piedāvāto metožu konstruēšanā, aprakstītas detalizēti.

Nodaļā dotie metožu pārskati izmantoti citu autoru metožu analīzes sadaļās promocijas darba autores publikācijās [BN04], [SN05], [NSTN07a], [NSTN07b].

3.2. Metodes jēdziens

Literatūrā pieejami dažādas metodes (metodoloģijas) definīcijas un dažādu autoru vidū nav vienprātības par terminoloģiju, tiek lietots gan metodoloģijas [AF03], [PMBOK], gan metodes [Bri96], [Rol97] jēdziens vienas un tās pašas parādības apzīmēšanai.

[AF03] metodoloģijas definīcija: "Metodoloģija ir rekomendējama fāžu, procedūru, likumu, tehniku, rīku, dokumentācijas, pārvaldības un apmācības kopums, ko izmanto sistēmas izstrādei." Metodoloģija saskaņā ar [PMBOK] ir tehniku, procedūru, likumu sistēma, ko lieto nozarē strādājošie.

Otra autoru grupa [Bri96], [Rol97] uzskata, ka metodoloģijas jēdziena sākotnējā nozīme "metožu pētīšana"- ir kļuvusi neviennozīmīga, jo tiek lietota kā sinonīms metodes jēdzienam vai lai apzīmētu metožu hierarhijas, šie autori runā par sistēmas izstrādes metodēm.

[Bri96] definē, ka "metode ir pieeja sistēmas izstrādes projekta veikšanai, kas balstīta uz domāšanas veidu un sastāvoša no norādījumiem un likumiem, strukturētiem sistemātiskā veidā izstrādes aktivitātēs, kam atbilst noteikti izstrādes produkti".

Promocijas darbā tiks lietots metodes jēdziens, jo tālāk promocijas darbā tiek izmantotas metožu inženierijas idejas par metožu komponentiem, bet metožu inženierijas kontekstā tiek lietots tieši metodes jēdziens.

3.3. Metožu klasifikācija

Datu noliktavu konceptuālā modeļa izstrādei eksistē vairākas pieejas kā noskaidrot prasības, kas jāietver uzņēmuma vajadzībām visatbilstošākā datu noliktavas modelī. Katrā no pieejām izstrādātas vairākas metodes, kā šo pieeju izmantot datu noliktavas konceptuālā modeļa izveidē.

Prasības datu noliktavām atšķiras no citu sistēmu prasībām. Datu noliktavām var runāt par informācijas prasībām [WS03], [Goe05]. Datu noliktavu sistēmas netiek būvētas, lai tiktu apmierinātas funkcionālas prasības, bet gan lai attēlotu lēmumu pieņemšanai vajadzīgu informāciju.

Pēc [IEEE Std. 610.12-1990] prasība ir A) nosacījums vai spējas, kas vajadzīgas lietotājam, lai atrisinātu problēmu vai sasniegtu mērķi. B) nosacījums vai spējas, kas jāapmierina vai jāpiemīt sistēmai vai sistēmas komponentam, lai apmierinātu līgumu, standartu, specifikāciju vai kādu citu formāli ierobežojošu dokumentu. C) dokumentēta nosacījuma vai spējas reprezentācija atbilstoši A) vai B). Prasības tiek iedalītas funkcionālās un nefunkcionālās prasībās.

Informācijas vajadzības ir minimāli funkcionālas, tāpēc neatbilst iepriekš minētajai definīcijai. Pieņem, ka informācija atbilst prasībām tad, ja saturīgi nepieciešamā informācija tiek piegādāta atbilstoši noteiktām kvalitātes prasībām un noteiktā vizuālā formā [Goe05].

Pēc analogijas ar funkcionālām un nefunkcionālām prasībām [Goe05] ievieš neinformācijas prasību jēdzienu, ar to saprotot, piemēram, drošības un ātrdarbības aspektus.

Pašreizējas datu noliktavas konceptuālā modeļa izstrādes metodes var sadalīt vairākās grupās atkarībā no pieejas veida, dažādiem autoriem lietojot gan dažādas klasifikācijas, gan dažādus vienas un tās pašas pieejas nosaukumus:

- datu vadītās pieejas metodes [Art05], [LBMS02], kuru pamatā ir datu avotu modeļu un datu izpēte, modeļu un datu integrācijas problēmas. Datu noliktavas konceptuālais modelis tiek iegūts no datu avotu modeļiem, tos transformējot un neinteresējoties vai interesējoties minimāli par analīzes vajadzībām organizācijā. Šīs pieejas tiek sauktas arī par piedāvājuma vadītām pieejām [WS03], ar piedāvājumu saprotot esošos datu avotus.

- prasību vadītās pieejas metodes, kuru pamatā ir prasību noskaidrošana dažādos veidos, šī klasifikācijas grupa atrodama [WS03] kā pieprasījuma vadītās pieejas. Citi autori tā vietā apskata sīkākas šīs grupas apakšgrupas, klasificējot pēc dažādiem prasību noskaidrošanas veidiem. [Art05] runā par metriku vai mērījumu vadītām pieejām, [LBMS02] - par lietotāju vadītām un mērķa vadītām pieejām, [BU00]- par procesu vadītām pieejām.

[WS03] minētās pieejas uzskata par pieejām informācijas prasību analīzei datu noliktavu sistēmā, ieskaitot šeit arī datu vadītās pieejas.

Tālāk analizēta katra no pieejām, aplūkojot atbilstošu metožu piemērus, novērtējot to priekšrocības un trūkumus.

3.4. Datu vadītās metodes

Datu noliktavas sistēmas ir datu vadītās salīdzinājumā ar klasiskām sistēmām, kas ir prasību vadītās izstrādes metodes [Inm02]. Datu vadītā pieejā izejas punkts datu noliktavas izstrādei ir eksistējoši dati transakciju sistēmās un transakciju sistēmu modeļi. Modeļi tiek integrēti, dati tiek savākti, attīrīti, integrēti un izmantoti, lai analizētu organizācijas darbību.

Datu vadītās metodes piedāvā vairāki autori [GMR98a], [GMR98b], [Inm02], [BU99], [PD02]. Viens no plašāk zināmajiem ir [GMR98a], [GMR98b] pusautomātiska metode, kas būvē

datu noliktavas konceptuālo modeli “*Dimensional Fact Model*” (DFM) no eksistējošas transakciju apstrādes sistēmas datu bāzes ER modeļa. [Inm02] savukārt piedāvā "pārkārtotu" ūdenskrituma sistēmas izstrādes dzīves ciklu, kurā lietotāju datu analīzes vajadzību noskaidrošana ir pārcelta pēc datu noliktavas modeļa realizācijas.

Datu vadītai pieejai var apskatīt gan priekšrocības, gan trūkumus. Datu vadītas pieejas priekšrocības ir sekojošas:

- pieeja ir precīzi definējama, visi datu noliktavas dati tiek iegūti no datu avotos eksistējošiem datiem;
- vienmēr var iegūt summāros datus, tie tiek aprēķināti no iepriekš eksistējošiem datiem [Art05].

Savukārt par trūkumiem var uzskatīt to, ka:

- datu avoti bieži vien ir neatkarīgi izstrādāti, līdz ar to var pastāvēt semantiski konflikti, kurus ir problemātiski novērst, ja nav skaidri zināms, kāds ir beigu stāvoklis, uz kuru jāpāriet;
- pastāv gnozeoloģiskas problēmas [Art05] – nav skaidrs, ko reprezentē iegūtie summārie dati un vai tie ir patiesi, jo dati būtībā ir vākti vienam mērķim (operatīviem mērķiem), bet lietoti citiem mērķiem (analīzei). Informācija, kas atvasināta no datu kopas, ir patiesa attiecībā pret šo datu kopu, bet apšaubāmi, vai šo informāciju var lietot, lai raksturotu organizāciju;
- piedāvājuma vadītas pieejas tērē resursus nevajadzīgu datu struktūru specificēšanai, nepietiekami iesaista lietotājus datu noliktavu izstrādē [WS03].

Tālāk tiks apskatītas dažas datu vadītas metodes - DFM [GMR98a] kā tipiska datu vadīta metode, kurā lietotāju intervēšana notiek minimāli (par izņēmumiem tiek runāts metodes aprakstā), un Inmona metode [Inm02], kurā jēdziens "datu vadīta" tiek interpretēts savādāk nekā parasti (neveic uzreiz transformācijas no datu avotu ER modeļiem, kā, piemēram, [GMR98a], [GMR98b]), un kurā lietotāju intervēšana notiek, bet ar specifisku mērķi.

3.4.1. DFM metode

DFM autori [GMR98a] formāli definē ne tikai savu konceptuālo modeli (skat. nodaļā 2.5.3), bet apraksta arī metodi pārejai no ER modeļa uz faktu shēmu.

Metode sastāv no sekojošiem soļiem:

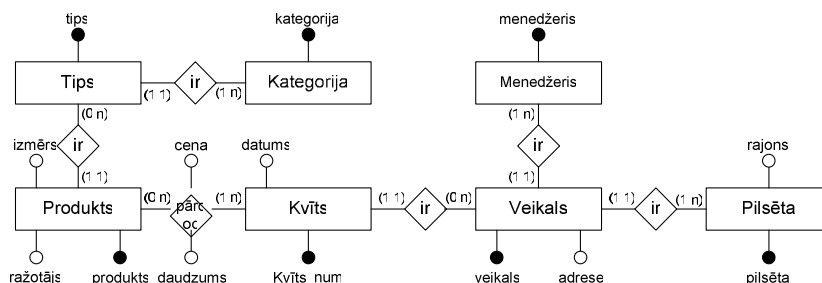
1. solis. Definē faktu;
2. solis. Katram faktam:
 - a) būvē atribūtu koku,
 - b) veic atribūtu koka "apcirpšanu" un "aizstāšanu",
 - c) definē dimensijas,
 - d) definē fakta atribūtus,
 - e) definē hierarhijas.

1. solis. Fakta definēšana

Par faktu datu avota sistēmas ER modelī var kalpot gan entītija F, gan n-āra attiecība starp entītijām $E_1 \dots E_n$. Vienkāršības labad katra n-āra attiecība tiek aizstāta ar entītiju F, kur F ar katru entītiju E_i saista bināra attiecība ar kardinalitāti (1,1) attiecības "F galā" un ar kardinalitāti (m_i ,

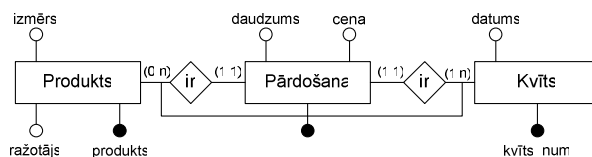
M_i) attiecības " E_i galā", $m_i \in \{0,1\}$ un $M_i \in \{1,n\}$, m_i - minimālā, bet M_i maksimālā iespējamā kardinalitāte.

Ja par faktu izvēlas ER modelī jau eksistējošu entītijai, tai jābūt tādai, kuras atribūtu vērtības bieži mainās. Katrs fakts, kas identificēts ER modelī, kļūst par dažādu faktu shēmu sakni.



14. att. Datu avota sistēmas ER-modelis, pēc [GMR98a]

ER modeli (skat.14.att.) pārveido, ar starpentītijai aizstājot $m:n$ attiecības (skat. 15.att.). Melnā krāsā attēlos ir identificējošo atribūtu apzīmējumi, baltā – neidentificējošie atribūti.



15. att. $m:n$ attiecību pārveidošana, pēc [GMR98a]

2.a. solis. Atribūtu koka veidošana

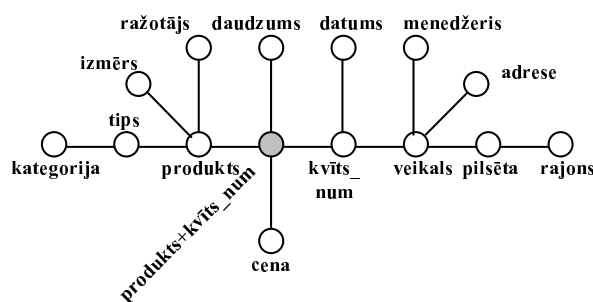
E/R modeļa entītijai F par atribūtu koku sauc koku, kam katra virsotne atbilst kādam atribūtam no ER modeļa, bet koka sakne atbilst F identifikatoram (identificējošam atribūtam).

Kokam tiek pievienotas virsotnes ar rekursīvu algoritmu, kura ideja ir sekojoša:

1) ja atribūtu kokam ir pievienota virsotne, kas ER modelī atbilst kādas entītijas identificējošam atribūtam, tad šai virsotnei ar šķautnēm jāpievieno arī visi pārējie dotās entītijas atribūti.

2) ja starp divām entītijām A un B ir $1:m$ attiecība, tad, ja atribūtu kokam ir pievienota entītijas B identificējošais atribūts b , tad šim atribūtam b pievieno ar šķautni arī entītijas A identificējošo atribūtu a .

Dotajam piemēram, kas redzams zīmējumā (skat.14.att.), atbilstoši aprakstītajam algoritmam ir izveidots atribūtu koks (skat. 16. att.). Par faktu "produkts+kvīts_num" izvēlēta starpentītijai "Pārdošana", ko savukārt raksturo attiecības ar entītijām "Produkts" un "Kvīts".



16. att. Atribūtu koks, pēc [GMR98a]

2.b. solis. Atribūtu koka "apcirpšana" un "aizstāšana"

Atribūtu koks tiek vienkāršots, izmantojot ekspertu zināšanas par to, kādi dati ir nepieciešami datu analīzē. Atribūtu koku var pārveidot, veicot "apcirpšanu" un/vai "aizstāšanu". Apcirpšana tiek veikta, izmetot no atribūtu koka jebkuru apakškoku. Aizstāšana nozīmē, ka kāda no virsotnēm ir jāizmet no atribūtu koka.

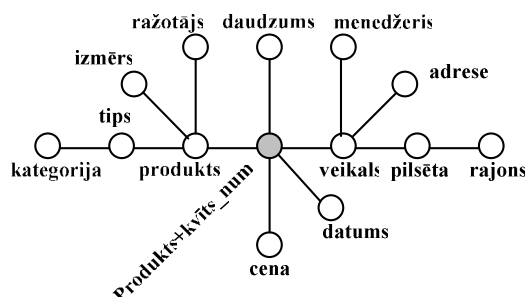
Ja v ir izmetamā virsotne un v' ir tās vecāks, tad aizstāšanu apraksta sekojošs algoritms:

```

aizstāt ( $v$ ):
{ for each  $v''$  |  $v''$  ir  $v$  bērns do
add ( $v', v''$ ); // pievieno  $v''$  virsotnei  $v'$ 
drop  $v$ ;
}

```

Piemērā (skat.17.att.) veikta "aizstāšana" - izmesta virsotne "kvīts_num", virsotnes "datums" un "veikals" pievienotas faktam "produkts+kvīts_num".



17. att. Pārveidots atribūtu koks, pēc [GMR98a]

2.c. solis. Dimensiju definēšana

Dimensijas ir jāizvēlas no atribūtu koka saknes bērniem. Vēl ir būtiski apzināties laika dimensijas nozīmi datu noliktavās, tādēļ, ja laiks nav tieši piesaistīts atribūtu koka saknei, bet ir virsotne kādā no apakškokiem, tad jāapsver iespēja pielietot iepriekš aprakstīto virsotņu aizstāšanu vai arī, ja laiks dotajā ER modelī nav starp atribūtiem, jebkurā gadījumā pievieno laika dimensiju faktam.

2.d. solis. Fakta atribūtu definēšana

Fakta atribūti parasti ir vai nu fakta F instanču skaits vai SUM / AVG / MAX / MIN izteiksmes, kur izmantoti atribūti no atribūtu koka, izņemot tos atribūtus, kas izvēlēti dimensijām.

Lai būtu vieglāk izveidot loģisko modeli, jau šajā fāzē ir noderīgi izveidot vārdnīcu, piesaistot fakta atribūtam atbilstošu aprēķināšanas izteiksmi, piemēram:

```
daudzums = SUM(pārdošana.daudzums)
```

Agregācijas operatori tiek pielietoti visām "Pārdošana" instancēm vienai un tai pašai nedēļai (var tikt izvēlēta cita detalizācija), veikalam, produktam. Dažreiz agregācijas operatori nav jāpielieto, bet viens pret vienu atribūts jāpārnes no ER modeļa uz fakta shēmu, ja tas jau atspoguļo vajadzīgo detalizācijas pakāpi.

2.e. solis. Hierarhiju definēšana

Atribūti hierarhijās jāsakārto tā, lai starp koka virsotnēm būtu $m:1$ attiecība. Par pamatu ņem jau uzbūvēto atribūtu koku, bet vēl var veikt iepriekš jau pielietoto aizstāšanu, var pievienot jaunus līmeņus (piemēram, aprēķinot skaitlisku atribūtu intervālus). Šajā solī tiek arī identificēti, kuri no atribūtu koka atribūtiem netiks izmantoti agregācijai, bet ir tikai aprakstoši.

Nobeigumā jāsecina, ka metode nav pilnībā automatiski realizējama, jo divās situācijās bija nepieciešams eksperts: izejas punkta jeb fakta izvēlē, kā arī atribūtu koka vienkāršošanā.

Bieži šis ierobežojums, ka metode nav pilnībā automatizējama, piemīt arī citu autoru datu vadītu metodoloģiju aprakstos, piemēram, [PD02]. Kā rekomendācijas, ko izvēlēties par faktu, ja nav iespējas pavaicāt ekspertam, ir dotas sekojošas pazīmes [KRRT98]:

- OLTP sistēmu tabulas, kas realizē $m:n$ attiecību, "ne-atslēgas" atribūti,
- lietotāju vaicājumi OLTP sistēmās (ar agregācijas funkciju), piemēram:

```
SELECT SUM(amount) FROM SALES ...
```

Šajā gadījumā summējamais atribūts var kalpot par faktu.

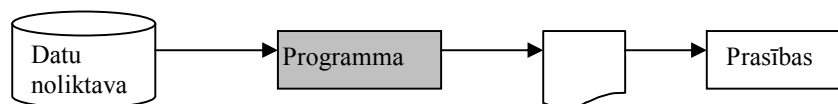
- UPDATE biežs lietojums kādai tabulai – bieži mainīgs skaitlisks atribūts var būt potenciāls fakts,
- vispār jebkurš skaitlisks atribūts var būt fakts.

3.4.2. Inmona metode

[Inm02] aprakstīts pavisam cits datu noliktavas izstrādes dzīves cikls, kas ir gandrīz pretējs sistēmas izstrādes dzīves cikla ūdenskrituma modelim.

Inmona datu noliktavas izstrādes dzīves cikls sākas ar datiem. Programmas tiek rakstītas attiecībā pret datiem, tad programmu rezultāti tiek analizēti un beidzot tiek saprastas analīzes vajadzības, kuras jānodrošina datu noliktavai.

Datu noliktavas izstrādes dzīves cikls pēc [Inm02] redzams attēlā (skat.18.att.).

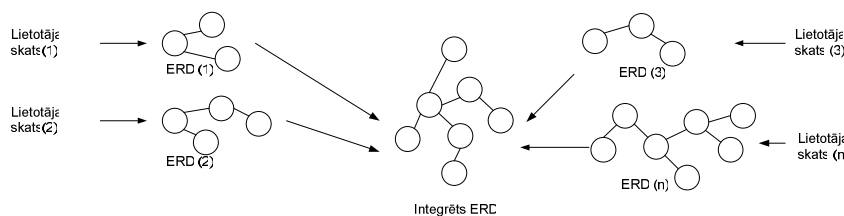


Datu noliktavas izstrādes dzīves cikls:

- Datu noliktavas implementācija
- Datu integrēšana
- Testēšana
- Programmas rakstīšana pēc datiem
- Sistēmas projektējums
- Rezultātu analīze
- Prasību saprašana

18.att. Datu noliktavas izstrādes dzīves cikls, pēc [Inm02]

Inmons datu modelēšanai lieto trīs līmeņus - 1) ER-modeļi (bez atribūtiem), 2) DIS (*Data Item Set*), Inmona ieteikts veids augšējā līmeņa modeļa detalizēšanai un 3) fiziskais modelis – realizācija tabulās.



19.att. Integrēta ER modeļa veidošana, pēc [Inm02]

Inmons datu noliktavas ER-modeļa iegūšanai, lai būtu skaidrs par kādām entītijām interesēties, pirmkārt, ierosina ierobežot pētāmo apgabalu, visām ieinteresētajām pusēm par to vienojoties. Otrkārt, tiek izstrādāti atsevišķi ER-modeļi, kas tiek veidoti pēc dažādu potenciālo lietotāju grupu intervijām. Treškārt, apvienojot šos atsevišķos “lietotāju skatus”, tiek izveidots visa uzņēmuma ER-modelis pētāmā problēmu apgabala ietvaros (skat. 19.att.).

Tālāk tiek izstrādāti DIS un fiziskais modelis, tiek realizēta datu noliktavas datu bāze atbilstoši izstrādātajiem datu modeļiem. Inmons šādu pieeju sauc par “datu bāzētu”, kaut arī šajā gadījumā apzīmējums vairāk attiecas uz datu noliktavas izstrādes dzīves cikla īpatnību.

Autors apgalvo, ka prasības ir pēdējā lieta, ko aplūko lēmumu pieņemšanas atbalsta sistēmu izstrādes gaitā, tās tiek saprastas pēc tam, kad datu noliktavā ir ielādēti dati un vaicājumu rezultātus analizē lietotāji. Kompānijas mērķi netiek atspoguļoti. Lietotāju vajadzības datu noliktavas sistēmā tiek integrētas pēc tam, kad sākotnējais datu bāzes variants datu noliktavai jau ir gatavs.

Datu noliktava ir integrētu un vēsturisku datu krātuve, kas primāri tiek lietota stratēģisku lēmumu pieņemšanai ar tiešsaistes analītiskas apstrādes (OLAP) tehnoloģijas līdzekļiem [Inm02]. Autors par būtiskāko uzskata to, ka datu noliktava ir datu bāze, kur tiek glabāta integrēta, vēsturiska un apkopota informācija, iegūta no vairākiem heterogēniem, autonomiem un atdalītiem informācijas avotiem.

3.5. Prasību vadītas metodes

Ir virkne rakstu, kuru mērķis ir aprakstīt prasību noskaidrošanas metodes datu noliktavām. Lielāko daļu no aprakstītajām metodēm var pieskaitīt kādai no apakškategorijām - procesu, mērķu vai lietotāju vadītām metodēm (tās tiks apskatītas atbilstošos promocijas darba punktos). Viena no prasību vadītajām metodēm [WS03], nav pieskaitāma nevienai no nosauktajām apakškategorijām, jo pieejas pamatā ir informācijas prasības, kuru noskaidrošanai vienlīdzīga loma ir gan lietotājiem, gan arī datiem, kas jau tiek analizēti eksistējošās sistēmās.

3.5.1. Informācijas prasību vadīta metode

Informācijas prasību vadītu metodi pārstāv [WS03]. Autori apskata četru soļu metodi prasību inženierijai datu noliktavām. Autori apskata informācijas prasību noteikšanu. Sākotnēji tiek noskaidrota esošā situācija datu analīzē organizācijā, kādi dati tam nepieciešami. Tā kā datu noliktavām ne tikai jānodrošina lēmumu pieņēmēji ar aktuālajam lēmumu pieņemšanas procesam vajadzīgo informāciju, bet arī jāapmierina arī nākotnes, pašreiz vēl nezināmas informācijas vajadzības [WS03], tad, intervējot potenciālos lietotājus, tiek noskaidrotas vēl neapmierinātās informācijas vajadzības. [WS03] autori uzskata, ka centrālajam jautājumam datu noliktavu izstrādē vajadzētu būt informācijas prasību un pieejamās informācijas piedāvājuma atbilstības definēšanai.

Metodes fāzes ir sekojošas:

1. fāze - Inicializācija

Tiek identificēti galvenie lietotāji un galvenie lietotņu tipi – tas var ietekmēt izstrādājamo modeli (OLAP, datizrades rīks, reportu rīki u.c.).

2. fāze “As Is” analīze

Analīzē esošo informācijas piedāvājumu no regulāri lietoto esošo atskaišu puses, tātad noskaidro esošās analīzes vajadzības un analīzē izmantoto informāciju. Uz šī pamata izveido augsta līmeņa informācijas shēmu. Tiek novērtēta arī modelī ietvertu datu kvalitāte.

3. fāze “To Be” analīze

Analīzē patiesās informācijas vajadzības - biznesa jautājumus, kurus nevar atbildēt vai var atbildēt tikai daļēji, izmantojot eksistējošas informācijas sistēmas. Definē atbilstību starp informācijas prasībām un informācijas shēmu no 2. soļa, lai noskaidrotu objektīvas neapmierinātās informācijas vajadzības. Definē prioritātes neapmierinātajām informācijas prasībām. Izveido detalizētāku datu modeli nekā 2. solī, tiek detalizētāk aprakstīti datu izguves algoritmi, novērsti jēdzienu semantiskie konflikti.

4. fāze “Modelēšana”

Izveido datu modeli. Autori iesaka izmantot kādu no datu noliktavu konceptuālajiem modeļiem, kas ietver datu noliktavu jēdzienu semantiku, piemēram, ME/R, ADAPT, DFM.

[WS03] autori izmantojuši dažas metožu inženierijas idejas, piemēram, 4. fāzē atzīstot, ka jāizmanto gatavs, piemērots, konceptuālais modelis, bet metode ir aprakstīta tikai no metodes procesu soļu secības aspekta, nespecificējot ne tehnikas, ne dokumentu tipus.

3.5.2. *Procesu vadītas metodes*

Procesu vadītas pieejas gadījumā [BU00] autori piedāvā pieeju, kas balstās uz SOM (*Semantic Object Model*) – semantisku objektu modeļu procesu modelēšanas metodi, lai izveidotu sākotnējo datu noliktavas struktūru. Pirmais projektēšanas procesa solis nosaka mērķus un pakalpojumus, ko kompānija sniedz saviem klientiem. Tad biznesa process tiek analizēts, pielietojot SOM mijiedarbības shēmu, kas nosaka klientus un viņu transakcijas. Trešajā solī konceptuālo objektu shēma ir noteikta, ir izvēlēti atribūti, kas nepieciešami analīzei. Pēdējais solis identificē mērījumus un dimensijas.

Procesu vadītas metodes tiek izstrādātas arī tā saukto procesa datu noliktavu kontekstā, piemēram, [KWL01] (skat. apakšnodaļu 4.5.3). [LBMS02] autori apraksta līdzīgu metodi, kas būtībā ir pārveidota un papildināta [KWL01] metode.

Detalizētāk tiks aplūkota [KO04] metode. Pēc [KO04] sākumā tiek apzināta esošā situācija – eksistē vai tiek būvēts (iesaistot lietotājus) procesu modelis „*As Is*” un datu modelis, kas atspoguļo procesu izmantotos datus.

Pielieto lietotāju vadītu metodi (netiek detalizēta, kāda), lai apzinātu analīzes vajadzības. Tad tiek būvēts procesu modelis „*To Be*”, kas ietver datu analīzes procesus, kā arī tiek iegūts jauns ER modelis, kas ietver datu analīzei vajadzīgo informāciju.

Metodes priekšrocības ir lēmumu pieņemšanas procesu modelēšana un tieši lēmumu pieņemšanai vajadzīgo datu iekļaušana ER-modelī, tomēr šīs metodes rezultātā:

1. netiek izveidots dimensiju modelis, tas vēl jāveido, iespējams, pielietojot kādu datu vadītu pieeju, izmantojot lēmumu pieņemšanas funkcijām atbilstošu ER-modeli,
2. nenotiek procesus raksturojošu rādītāju identificēšana, tātad paliek atklāts jautājums, kuri no modelī ietvertajiem atribūtiem būs fakti. Rezultātā aktuāla ir tāda pati problēma, kā datu vadītu pieeju gadījumā.

Par procesa vadītas pieejas variāciju var uzskatīt arī "Metriku vadītu pieeju" [Art05]. Metriku vadīta pieeja datu noliktavu izstrādē sākas ar svarīgāko biznesa procesu identificēšanu, kas jāmēra un kurām jāseko līdzī laikā, lai organizācijas darbs būtu efektīvāks. Dati var eksistēt jau iepriekš, ja nē, tad jāparūpējas par to turpmāku vākšanu.

Priekšrocība metriku vadītai pieejai - datu noliktavu modelis pēc savas būtības atspoguļo biznesa procesus un to izmaiņas laikā. Dimensiju modelis satur faktus, kas reprezentē mērījumus par svarīgākajiem biznesa procesiem (atkarīgais mainīgais, fakts) un dimensijas, kas ir šo faktu atribūti (neatkarīgie mainīgie).

Kaut arī metriku vadīta pieeja vairāk atbilst datu noliktavas modeļa būtībai, šajā pieejā [Art05] saskata virkni pētāmu problēmu:

- Kā noteikt galvenos biznesa procesus?
- Kā noteikt atbilstošus mērījumus šiem identificētajiem procesiem?
- Kā noteikt, ka šie mērījumi ir derīgi?
- Kā noteikt, ka dimensiju modelis korekti atspoguļo neatkarīgos mainīgos (dimensijas)?
- Vai ir iespējams izveidot atbilstošu datu agregācijas teoriju?

3.5.3. Mērķa vadītas metodes

Pieejas, kad kopā tiek piedāvāta uz mērķu analīzi balstīta prasību analīze un metode, kā iegūt datu noliktavas konceptuālo modeli, kas saturētu mērķu sasniegšanas atbalstam nepieciešamos indikatorus, ir apskatīti [GRG05] un [BCCFP01] darbos.

[GRG05] rakstā ir piedāvāta mērķu vadīta pieeja datu noliktavas prasību analīzei. Analīze tiek veikta no divām dažādām perspektīvām: organizācijas modelēšana (*organizational modeling*) koncentrējas uz ieinteresētajiem (*stakeholders*) un lēmumu modelēšana (*decisional modeling*) koncentrējas uz lēmumu pieņēmējiem. Datu noliktavas konceptuālais projektējums tiek izveidots, izmantojot lēmumu modelēšanas laikā iegūto loģisko diagrammu, faktus attēlo ar avota datu modeļa entītijām vai relācijām. *Hierarhijas* katram faktam tiek būvētas pēc $m:n$ attiecībām datu avota modeli līdzīgi kā [GMR98a], [GMR98b] DFM metodē, kas tika apskatīta iepriekš.

[BCCFP01] savā rakstā piedāvā metodi datuvju identificēšanai un būvēšanai, kas pamatojas uz biznesa vajadzībām. Metode sastāv no trim soļiem: *top-down* analīze, *bottom-up* analīze un integrācija. *Top-down* analīzē izmanto *Goal-Question-Metric* (GQM) pieeju, ko izmanto, lai aizpildītu GQM abstrakcijas lapas (*abstraction sheets*). No abstrakcijas lapu informācijas būvē ideālu zvaigznes shēmu katram mērķim. Kvalitātes faktoros (pētāmā objekta īpašības, ko grib analizēt) no abstrakcijas lapām uzskata par faktiem, variācijas faktoru sadaļas mainīgos (t.i., faktoros, kas ietekmē atribūtu vērtības) par dimensijām ideālā zvaigznes shēmā. *Bottom-Up* analīzes laikā tiek identificētas kandidātu zvaigznes shēmas, t.i., zvaigznes shēmas, kas reāli var tikt implementētas no pieejamiem datu avotiem. Sākot no ER modeļa, šajā solī notiek analīze, lai atrastu entītijas, kas varētu būt fakti, un tad veido daudzus grafus, kam centrā ir šīs entītijas. Katrs grafs atbilst potenciālai zvaigznes shēmai. Nākošā fāze ir integrācija, kas specificē, kā ideālas prasības var tikt attēlotas reālā sistēmā.

[LM04] piedāvā uzņēmuma mērījumu sistēmu (*Corporate performance measurement System*), kur procesu izpildes dati tiek integrēti uzņēmuma datu noliktavā. Autori datu noliktavas modeļa izveidei piedāvā metodi, kurā no kompānijas mērķiem tiek noteikti analizējamajiem biznesa procesiem specifiski mērķi, atbilstoši *Wal-Mart Model* [Wes01], kuriem savukārt tiek atrasti biznesa jautājumi, kas kalpo divu lietotāju kategoriju – procesu īpašnieku un struktūrvienību vadītāju vajadzībām. Biznesa jautājumiem tiek piesaistīti atbilstoši mērījumi, norādīti mērījumu ieguves datu avoti. Nav dots formāls algoritms, kā no iepriekš pieminētajiem mērījumiem un jautājumiem būvēt datu noliktavas konceptuālo modeli, no paša modeļa elementiem jāsecina, ka mērījumi ir izmantoti kā faktu tabulu atribūti.

3.5.4. Lietotāju vadītas metodes

Visvairāk aprakstīta ir lietotāju vadīta pieeja. Lietotāju vadītu pieeju izmanto [Wes01], [PR95], [Goe05], kā arī daļēji Kimbals [KRRT98], lai noskaidrotu lietotāju prasības. [PR95] piedāvā katalogu lietotāju interviju uzkrāšanai, lai savāktu gala lietotāju prasības. Tiek rekomendēts intervēt dažādas lietotāju grupas, lai pilnīgāk saprastu biznesu.

3.5.4.1. Kimbala metode prasību noteikšanai

Prasību noskaidrošanai pēc "Kimbala metodes" [KRRT98], [KR02] tiek intervēti biznesa lietotāji, lai saprastu, kādu darbu viņi dara un kā viņi pieņem lēmumus. Paralēli tiek intervēti arī IT speciālisti, lai izpētītu pieejamos datu avotus, vai eksistē analīzei nepieciešamie dati un kāda ir to kvalitāte.

Prasību vākšana balstās uz intervijām, Kimbals piedāvā jautājumu sagataves, ko var izmantot atkarībā no intervējamā. Intervējamo izvēle - būtiski ir pārklāt organizāciju horizontāli, lai pārklātu dažādas iespējamās datu noliktavas lietotāju grupas un lai iegūtu globālu skatījumu par organizācijā lietotajiem terminiem. Sāk veidot vārdnīcu, kas noderēs dažādu datu avotu integrēšanā.

Izvēlētajai biznesa sfērai intervējamo izvēlei jāpārklāj arī organizācija vertikāli, jo vadība nosaka augsta līmeņa stratēģiju, bet vidējā līmeņa vadītāji palīdz šo stratēģiju saprast.

3.5.4.2. Wal-Mart metode

Wal-Mart metode [Wes01] orientēta uz biznesa stratēģiju realizāciju. *Wal-Mart* metode [Wes01] tiek veikta četros soļos:

- biznesa puses pārstāvji definē kompānijas mērķus, nosaka to prioritātes,
- katram biznesa mērķim noskaidro biznesa jautājumus, kuru atbildes palīdz saprast, kā mērķus sasniegt,
- divos atkārtotos piegājienos nosaka jautājumu prioritāšu grupas, tad prioritātes, kā līdzekli izmanto Mērķu-Prioritāšu Grupu - Prioritāšu - Jautājumu tabulu.
- definē biznesa mērķus - t.i., jautājumiem ar visaugstāko prioritāti definē datu elementus - mērījumus un datu hierarhijas, attiecībā pret kurām jāanalizē mērījumi.

3.5.4.3. "Viewpoint" metode datu noliktavām

[Goe05] autors izvirza tēzi, ka informācijas vajadzību analīze ir tikai daļa no prasībām datu noliktavu sistēmām, pētījumu centrā ir informācijas adresāts ar savām vajadzībām, un šo vajadzību formalizēšanai tiek definēts paplašināts prasību jēdziens un prasību pārvaldībai datu noliktavām tiek piedāvāta "*viewpoint*" metode, kuras pamatā ir ideja, ka sistēmas izstrādē ir iesaistītas daudzas personas, kuru dažādās vajadzības ir jāņem vērā. Viņu dažādie viedokļi daļēji un nepilnīgi atspoguļo sistēmu, tas izriet no viņu dažādām lomām un atbildības pakāpes. Viena iesaistītā persona (*Stakeholder*) un viņa viedoklis tad tiek saukts par "*Viewpoint*" (skatījums). Oriģinālā metode, pēc kuras veidota šī metode datu noliktavu specifikai, ir [KS96]. Izstrādātajam šajā gadījumā sistemātiski ir jākrāj un jāanalizē individuālā skatījuma prasības, kas vēlāk jāaskaņo vienotā, nepretrunīgā sistēmas specifikācijā.

Līdzīgi kā [KS96] autors [Goe05] apraksta prasību vākšanas un analīzes procesa soļus atbilstoši „*Viewpoint*“ metodei, bet datu noliktavu gadījumā.

Sākumā tiek identificēti skatījumi, tiešo skatījumu klases (dažādi lietotāji) un netiešo skatījumu klases (organizācijas, sistēmas, apkārtējās vides skatījumi). Datu noliktavu gadījumā, piemēram, datu aizsardzības likumi varētu atbilst netiešajiem ārējās vides likumu skatījumiem.

Skatījumu prasību dokumentēšanā izmanto klasiskajā „Viewpoints“ metodē ieteikto formulāru paņēmieni, kur formulāra uzbūve izveidota tā, lai uzskaitītu un aprakstītu prasības. Metodes autors piedāvā veidot formulāra struktūru atbilstoši viņa aprakstītajām informācijas prasībām un neinformācijas prasībām.

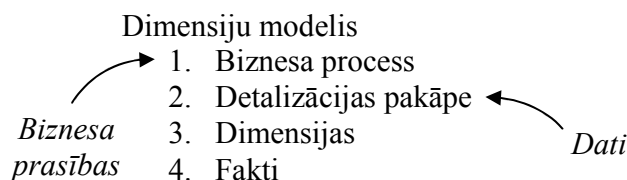
Autors apskata visu prasību kategoriju dokumentēšanu:

- datu noliktavas saturs tiek aprakstīts ar dažādu konceptuālo (uz lietotāju orientēto) modeļu palīdzību,
- vizualizācijas, kvalitātes un neinformācijas aspektu dokumentēšanas nolūkam tiek izmantoti formulāri.

Prasību analīzes un prasību specifikācijas izstrādes solī notiek dažādu skatījumu saskaņošana, kuras laikā nosaka pretrunas un tās novērš, kā arī izveido vienotu datu modeli un prasību specifikāciju. Lai novērstu pretrunas starp pārējo aspektu prasībām (nevis informācijas), tiek veidotas „ietekmju matricas“, atzīmējot dažādu prasību konfliktus vai pozitīvu ietekmi. Tiek veidota arī „prasību“ → „Skatījumu“ matrica, kas palīdz saprast, vai konfliktējošās prasības nāk no viena vai dažādiem skatījumiem. Šīs matricas var palīdzēt diskusijās prioritāšu noteikšanā.

3.6. Vairāku pieeju kombinēšana

Kā piemēru vairāku pieeju apvienojumam var minēt Kimbala metodi. Kimbals [KRRT98], [KR02] piedāvā četru soļu pieeju (skat. 20.att.).



20.att. Četru soļu datu noliktavas izstrādes process

Dimensiju modeļa izveide sākas ar datu noliktavas "Bus Architecture" matricas definēšanu, kas atspoguļo kandidātprocesus un kandidātdimensijas, lai apzinātos dažādu datuvju kopējās dimensijas. Matrica un datu avotu stāvokļa novērtējums palīdz saskaņot informācijas vajadzības ar reālām iespējām. Pēc tam tiek pielietota četru soļu metode.

Pēc četru soļu metodes zvaigznes shēmas projektēšanai pēc kārtas ir jāizdara četras izvēles:

- procesa izvēle – nosaka, kāds būs datuves priekšmetu apgabals;
- jāizvēlas faktu tabulas detalizācijas pakāpe, kas tad nosaka faktu tabulas ieraksta nozīmi;
- izvēlas dimensiju tabulas - faktu tabulas detalizācijas pakāpe nosaka nepieciešamās dimensijas. Dimensiju tabulas detalizācija nevar būt mazāka nekā faktu tabulai;

- izvēlas faktus – faktu izvēle arī ir atkarīga no faktu tabulas detalizācijas, tie jāizsaka atbilstoši iepriekš definētajai datalizācijas pakāpei.

Pirmais solis atbilst lietotāju vadītai pieejai, pārējos trīs nosaka datu avotos pieejamie dati, to izpētes rezultāts.

Jāpiezīmē, ka arī citas iepriekš pieminētās metodes ir vairāku, parasti divu, metožu apvienojumi, bet iepriekš klasificētas pēc primārās no metodēm, piemēram, [GRG05] tiek izmantota mērķu vadīta pieeja, bet [BCCFP01] - datu vadīta pieeja.

3.7. Pieeju izvēles problēma datu noliktavas konceptuālā modeļa izveidei

Gan pēc literatūrā pieejamajiem visu trīs pieeju salīdzinājumiem [LBMS02], [WS03], gan iepazīstoties ar konkrētām metodēm no katras pieejas, var izdalīt metožu trūkumus un priekšrocības.

Priekšrocības:

- lietotāju vadītai pieejai - lietotāju vajadzību noskaidrošana, lietotāju iesaistīšana, kas datu noliktavu projektiem ir būtiski, lai nodrošinātu sekmīgu realizētās datu noliktavas lietošanu,
- datu vadītai pieejai - visātrāk ir iegūstams datu noliktavas modelis,
- procesu (arī mērķu) vadītai pieejai - noskaidrojot būtiskos biznesa procesus un tos raksturojošus lielumus, var veidot modeli tieši šo rādītāju analīzei.

Trūkumi:

- lietotāju vadītai pieejai - lietotājiem ir neskaidri priekšstati par datu noliktavu, par uzņēmuma stratēģiju un uzdevumiem kopumā, viedokļu saskaņošana prasa daudz laika, pastāv problēmas ar prioritāšu noteikšanu;

- datu vadītai pieejai - iegūtie modeļi, kas pēc dažām metodoloģijām tiek iegūti pusautomātiskā veidā, var neatspoguļot uzņēmuma mērķu analīzei nepieciešamos faktus (agregētus datus, atvasinātus datus u.c.), jo modeļu izveides pamatā ir avotu sistēmu datu modeļi, kas nav orientēti uz analīzi. Tiek iegūti datu noliktavas daudzdimensiju modeļi, kas varētu neatbilst datu analīzes vajadzībām. Kādam iegūtie modeļi jāvērtē pēc kvalitātes, pēc atbilstības biznesa analīzes vajadzībām;

- procesu (arī mērķu) vadītai pieejai - modelis atspoguļos augstākās vadības un atsevišķu ekspertu viedokli. Modelis tiek veidots šaurai problēmai, grūti paredzēt visu lietotāju vajadzības, atspoguļo biznesa procesus. Tas nonāk pretrunā ar faktu, ka datu noliktavas paredzētas lēmumu pieņemšanas atbalstam, bet lēmumu pieņemšanas procesi parasti procesu modelī netiek atspoguļoti vai to nav iespējams principā izdarīt, jo katrs lēmumu pieņemšanas gadījums ir pārāk specifisks un grūti aprakstāms, vai arī lēmumu pieņēmēji nevēlas dalīties zināšanās par šo procesu.

Var secināt, ka parasti jāizmanto vairāk nekā viena pieeja, lai iegūtu datu modeli, kas maksimāli precīzi atspoguļotu uzņēmuma datu analīzes vajadzības. Jautājums drīzāk ir, kuru modelēšanas tehniku izvēlēties par primāro.

Izvēloties pieejas, var konstatēt sekojošas problēmas:

- nav ieteikumu, kuru pieeju izvēlēties par primāro, kādos gadījumos;

- nav ieteikumu, kādu modelēšanas tehniku izvēlēties, jo visas prasības konceptuāliem modeļiem neapmierina neviena no tehnikām;
- netiek apskatīts, kā aprakstīt lietotāju vajadzības pēc konkrētas detalizācijas pakāpes atšķirīgu prasību vai tiesību gadījumā;
- nav vispāratzītu ieteikumu konkrētām biznesa sfērām, gan pieejas, gan konceptuālās modelēšanas tehnikas izvēlē;
- publikācijās par izglītības sfērā izstrādātām datu noliktavām var runāt par atsevišķu datuvju realizāciju dažiem atsevišķiem procesiem.

4. PROCESU MĒRĪŠANA UN KONTROLE

4.1. Nodaļas nolūks

Viens no jaunākajiem pieejas veidiem, kas tika apskatīti 3. nodaļā punktā 3.5.2. ir procesa vadītas pieejas variācija - metriku vadīta pieeja. Metriku vadīta pieeja datu noliktavu izstrādē sākas ar svarīgāko biznesa procesu identificēšanu, kas jāmēra un kurām jāseko līdzī laikā, lai organizācijas darbs būtu efektīvāks. Priekšrocība metriku vadītai pieejai - datu noliktavu modelis pēc savas būtības atspoguļo biznesa procesus un to izmaiņas laikā. Šī pieeja ir cieši saistīta ar mērķa vadītu pieeju, jo metriku identificēšana bieži vien sākas ar organizācijas biznesa mērķu noskaidrošanu. Nodaļa raksturo trīs būtiskas lietas jaunas metodes definēšanai šajā pieejā - 1) mērīšanas pamatjēdzienus, 2) eksistējošas procesu mērīšanas sistēmas, tajā skaitā procesu datu noliktavas kā vienu no iespējamajiem to realizācijas veidiem un 3) eksistējošas metodes procesu mērījumu un indikatoru noteikšanai. Nodaļā dotie materiāli izmantoti promocijas darba autores publikācijās [NSTN07a] un [NSTN07a] sadaļās, kas raksturo citu autoru darbus šajā pieejā.

4.2. Mērīšanas pamatjēdzieni un ontoloģijas

Datu noliktavās viens no pamatjēdzieniem ir fakts, ko raksturo fakta atribūti jeb mērījumi. Būtiski ir noskaidrot, kā mērījumus definēt un ko vispār nozīmē mērīt.

Katrs uzņēmums ir ieinteresēts maksimāli efektīvā savas darbības organizācijā. Lai varētu uzlabot biznesa procesus, tie vispirms ir jāizprot. Eksistē daudzas biznesa procesu definīcijas, piemēram, [Har91] definīcija: "Process ir jebkura aktivitāte vai aktivitāšu grupa, kas saņem ievadi, pievieno tai vērtību un nodrošina rezultātu iekšējam vai ārējam klientam. Procesi izmanto organizācijas resursus, lai nodrošinātu noteiktu rezultātu."

Biznesa procesi nodrošina uzņēmuma mērķu sasniegšanu. Lai būtu iespējams noteikt, vai mērķi ir sasniegti, nepieciešams šos procesus gan kvantitatīvi, gan kvalitatīvi mērīt, un iegūtie mērījumi ir jāsalīdzina ar mērķa vērtībām. Pēc salīdzināšanas var izdarīt secinājumus un nepieciešamības gadījumā veikt izmaiņas biznesa procesos. [Har91] apgalvo: "Mērījumi ir galvenais. Ja nav iespējams mērīt, tad nav iespējams kontrolēt. Ja nav iespējams kontrolēt, tad nav iespējams pārvaldīt. Ja nevar pārvaldīt, tad nevar uzlabot". Uz biznesa procesu mērīšanas ideju balstās vairāku autoru darbi par datu noliktavu un biznesa procesu integrāciju, tajā skaitā [LM04], [KWL01].

Mērīšanas procesu var definēt sekojoši [FW95]: "Mērīšana ir process, kurā skaitliskas vērtības vai simboli tiek piešķirtas entītijas atribūtiem reālajā pasaulē tādā veidā, kas raksturo šos atribūtus pēc skaidri definētiem likumiem."

Bieži vien biznesa procesi tiek mērīti tikai procesa beigās, tādā veidā samazinot iespēju tieši ietekmēt procesu gaitu [Har91]. Atbilstoši mērīšanas mērķim un laika momentam var runāt par procesu kontroli un procesu mērīšanu [KHRS05].

Procesu kontrole (*Process monitoring*) nozīmē uzraudzīt situāciju, lai par to izdarītu secinājumus. Procesu kontrole reālā laikā izmanto datus no procesu izpildes sistēmas, piemēram, darba plūsmas sistēmas, žurnālu failiem. Procesu dati tiek vākti atbilstoši iepriekš definētiem indikatoriem, kas attiecas uz pašu procesu izpildi.

Procesu mērīšana (*Process Measurement*) nozīmē konstatēt biznesa procesu darbības rezultātus, izmantojot iepriekš definētu indikatoru kopu. Procesu mērīšana kombinē datus par procesiem ar biznesa datiem.

Procesu mērīšanai un kontrolei [KHRS05] klasificē un apraksta vairākas iespējamās realizācijas: žurnāla (*log*) failu analizatori, datu noliktava, ko var izmantot, ja vajag analizēt datus par ilgāku periodu, ja biznesa dati jau ir pieejami datu noliktavā, kā arī procesu mērīšanas un kontroles rīki.

Biznesa procesu mērīšana parasti tiek apskatīta tās pielietojumos, piemēram, biznesa procesu pārbūvē, procesu mērīšanā un kontrolē un šim nolūkam izstrādātās praktiskās realizācijās. Runājot par biznesa procesu mērīšanu, ar mērīšanu saistītie jēdzieni parasti netiek analizēti. Izņēmums ir programminženierija, kur tiek mērīti programmatūras izstrādes procesi. Ir izstrādātas gan ontoloģijas, gan metodes mērījumu un indikatoru noteikšanai. Programmatūras procesu mērīšanas pamatā ir mērījumu teorija, tomēr gan praktiķi gan pētnieki bieži vien ignorē mērījumu teoriju un tās likumus, kā rezultātā tiek iegūti maldinoši rezultāti [Mun03]. Tajā pašā laikā literatūrā ir pieejami avoti, kas apskata programmatūras procesu mērīšanu tieši no mērījumu teorijas viedokļa [Mun03], [MCJ+01]. Ir arī virkne standartu, kas veltīti mērīšanai, piemēram, [ISO15939], [ISOVIM], kā arī tiek mēģināts izstrādāt atbilstošas ontoloģijas [OM04], [GBC+06].

Lai lēmumu pieņēmēji varētu saprast mērījumus, nepieciešama precīza un saskaņota mērīšanas jēdzienu definēšana un to savstarpējo sakarību aprakstīšana. Turpmāk tiks dotas jēdzienu definīcijas, kas izmantojamas arī raksturojot biznesa procesu mērīšanu.

4.2.1. Informācijas vajadzības un mērāmie procesi un produkti

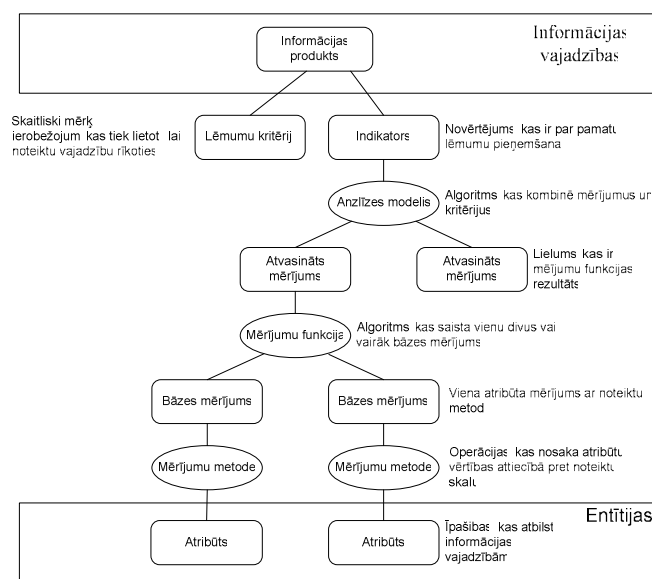
Saistību starp informācijas vajadzībām un potenciāli mērāmajiem procesiem un produktiem apraksta mērīšanas informācijas modelis [MCJ+01]. Mērīšanas informācijas modelī izmantotie jēdzieni ir definēti saskaņā ar starptautisko standartu ISO/IEC 15939 [ISO15939].

Lai tiktu uzsākts mērīšanas process, jāeksistē informācijas vajadzībām, kas nepieciešamas lēmumu pieņemšanas atbalstam. Datus, kas vajadzīgi, lai apmierinātu informācijas vajadzības, iegūst, mērot procesu komponentus un produktu īpašības - tāpat dažādu entītiņu atribūtus.

Mērīšanas informācijas modelī [MCJ+01] detalizētāk aplūkosim mērīšanas konstrukciju, kas apraksta, kā atribūtiem tiek piešķirta vērtība un kā tie tiek pārveidoti par indikatoriem, kas ir pamatā lēmumu pieņemšanai. Mērīšanas konstrukcijas shēma (skat. 21.att.) atspoguļo hierarhisku saistību tālāk apskatītiem pamatjēdzieniem [MCJ+01]:

- mērāms atribūts ir nosakāma entītiņas īpašība, ko kvantitatīvi vai kvalitatīvi identificē cilvēks vai automatizēti līdzekļi,

- bāzes mērījums ir viena atribūta mērījums, kas ir definēts ar kādu iepriekš noteiktu mērījumu metodi. Mērījumu metodes izpilde rada vērtību šim mērījumam. Katru bāzes mērījumu raksturo: mērījumu metode, skala, skalas tips, mērvienība,
- atvasināts mērījums ir mērījums vai lielums, kas ir definēts kā funkcija no diviem vai vairākiem bāzes mērījumiem un/vai atvasinātiem mērījumiem. Tas satur informāciju par vairāk nekā vienu atribūtu. Tādejādi vienkāršas viena atribūta transformācijas nepievieno jaunu informāciju un nerodas jauns atvasināts mērījums,
- mērījumu funkcija ir algoritms vai aprēķini, kas tiek veikti, lai kombinētu divu vai vairāku bāzes mērījumu vērtības un/vai atvasinātus mērījumus,
- indikators ir mērījums, kas nodrošina novērtējumu noteiktiem atribūtiem, kas iegūti atbilstoši analīzes modelim, kas savukārt atbilst informācijas vajadzībām. Indikatori kalpo par pamatu mērījumu analīzei un lēmumu pieņemšanai,
- analīzes modelis ir algoritms vai aprēķini, kas saista divus vai vairākus bāzes mērījumus un/vai atvasinātus mērījumus ar atbilstošiem lēmumu kritērijiem. Analīzes modelis rada novērtējumus atbilstoši informācijas vajadzībām,
- lēmumu kritēriji ir skaitliski mērķi, robežas, ko lieto, lai noteiktu vajadzību darbībām vai tālākai izpētei. Tie palīdz interpretēt mērījumu rezultātus. Tie var būt balstīti uz konceptuālu izpratni par gaidāmo rezultātu vai izrēķināti no datiem.



21.att. Mērīšanas konstrukcijas shēma, pēc [MCJ+01]

Mērīšanas kontekstā būtiski runāt par mērījumu skalām. Mērīšanas skala ir viens no elementiem, kas definē mērījumu. No skalas veida atkarīgas pieļaujamās darbības ar mērījumiem [PGF96], tā piemēram, nominālā skala piešķir vārdu, nosaukumu, kategoriju atribūtam. Ordinālā skala ļauj mērījumu vērtības sakārtot augošā vai dilstošā secībā, mērījumu vērtībām nav pieļaujama vidējo vērtību rēķināšana.

Mērījumu skala un mērījumu metode ietekmē arī analīzes tehniku izvēli vai analīzes modeļu izvēli, lai noteiktu indikatorus. Tā piemēram, aprēķināt vidējo vērtību no datiem, kam vērtības ir atbilstoši nominālai skalai, nav jēgas.

4.2.2. *Procesu mērīšanas ontoloģijas*

Ir daudzi mēģinājumi definēt programmatūras mērīšanas ontoloģijas [OM04], [GBC+06]. Plašs apskats un salīdzinājumi standartiem un pētījumiem, kas ietver programmatūras mērīšanas jēdzienus un terminoloģiju, atrodami [GBC+06]. Šajā darbā tiek piedāvāta SMO (*Software Measurement Ontology*), kas pamatā balstīta uz diviem standartiem: ISO/IEC 15939 [ISO15939], kur jēdzienu definīcijas atbilst iepriekšējā apakšnodaļā apskatītajiem no [MCJ+01] un [ISOVIM], kas ietver 120 dažādus jēdzienus, kas saistīti ar mērīšanu, bet nav specifiski orientēta uz programmatūras mērīšanu. SMO tiek reprezentēts ar metamodeli.

Pamatjēdzienu definīcijas [GBC+06] sakrīt ar iepriekš aprakstīto mērīšanas informācijas modeli no [MCJ+01], tālāk seko jēdzieni, kas SMO adaptēti no [ISOVIM]:

- mērījums (*Measure*) – Definētā mērījumu pieeja un mērījumu skala (vispārinājums bāzes mērījumam, atvasinātam mērījumam, indikatoram);
- mērīšanas pieeja (*Measurement Approach*) - operāciju virkne, kuras mērķis ir noteikt mērījumu rezultātu (vispārinājums mērījumu metodei, mērījumu funkcijai, analīzes modelim);
- mērīšana (*Measurement*) – darbību virkne, lai saistītu mērījumu rezultātu un doto atribūtu, izmantojot mērīšanas pieeju;
- mērīšanas rezultāts (*Measurement result*) – skaitlis vai kategorija, kas noteikta entītijas atribūtam mērīšanas rezultātā;

4.3. **Biznesa procesu mērīšanas sistēmas**

„Ja uzņēmums grib izdzīvot konkurences apstākļos, tad tam ir jāizmanto *mērījumu sistēmas*, kas ir izveidotas atbilstoši uzņēmuma stratēģijām un spējām" [KN96]. Svarīgi arī, kā izvēlēties mērījumus. Atbilstoši *Balanced Scorecard* - darbības efektivitātes mērīšanas metodei [KN96], organizāciju var novērtēt atbilstoši četrām perspektīvām - finansu, klienta, biznesa procesu un mācīšanās un jaunievedumu perspektīvai.

[KN96] apgalvo, ka iekšējo biznesa procesu perspektīva identificē un mēra kritiskos organizācijas procesus, kuros tai jābūt pārākai.

Esošo biznesa procesu mērījumu sistēmu trūkumi ir [KWL01]:

- biznesa procesi netiek mērīti sistemātiski,
- mērījumi notiek pa atsevišķām struktūrvienībām, nevis par organizācijas biznesa procesiem,
- orientējas uz finansu indikatoriem,
- dati nonāk uzņēmuma rīcībā pēc noteikta laika perioda.

Kā risinājumu [KWL01] iesaka veikspējas mērījumu sistēmu (VMS) – sistēmu, kas centralizēti glabā un pārvalda visus ar procesu izpildi saistītus datus, ieskaitot gan finansiālos,

gan citus rādītājus. Galvenais VMS sistēmas darbības mērķis – laicīgi piegādāt visaptverošu informāciju par uzņēmuma biznesu.

Tiek izvirzītas prasības korporatīvajai procesu izpildes mērīšanas sistēmai [KWL01]. Tai jāietver gan finansu, gan ne-finansu rādītāji, jāizmanto daudzi avoti, jānodrošina analīzes iespējas par lieliem laika periodiem, jābūt iespējai izmantot to dažādiem lietotāju līmeņiem.

Jaunākās veikspējas mērījumu teorijas, piemēram *Balanced Scorecard*, praktiski netiek izmantotas datu noliktavu izstrādē [LM04]. Datu noliktavas atspoguļo tradicionālo veikspējas mērīšanu, tas ir, patreiz datu noliktavu uzmanības centrā ir klientu attiecību pārvaldība un finansu rādītāji. Pēc [KN96] aprakstītās *Balanced Scorecard* metodes ir pārklāti finansu un klientu aspekti, bet nav iekšējo biznesa procesu aspektu un izglītošanas un izaugsmes aspektu. [LM04] apgalvo, ka nākošais solis ir integrēt uzņēmuma datu noliktavā iekšējo biznesa procesu aspektu. Šī risinājuma priekšrocība ir esošās datu noliktavas infrastruktūras izmantošana.

Svarīgs ir jautājums, kā noskaidrot, kuri ir būtiskākie mērījumi, kādas ir to prioritātes, vai ir pieejami dati šo mērījumu iegūšanai, vai ir nepieciešami jauni dati, kas līdz šim nav uzkrāti, un beigās, kas nav mazāk svarīgi, kāds būs atbilstošais datu noliktavas modelis.

4.4. Metode procesu mērījumu un indikatoru noteikšanai

GQ(I)M *Goal –Question – (Indicator) – Measure* metode [PGF96] ir modificēta plaši pazīstamā GQM [BR88], [BCR94] metode. Šeit lietotās ar mērīšanu saistītās definīcijas nav pretrunā ar iepriekš apakšnodalā 4.2. dotajām definīcijām, kā arī balstās uz mērījumu teoriju.

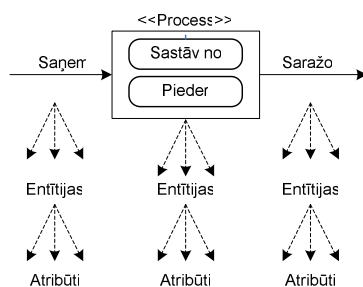
Metodē tiek izmantoti sekojoši mērīšanas procesa elementi: entītijas (intereses objekti), atribūti (entītiju īpašības) un likumi (kopā ar atbilstošām skalām), lai piešķirtu atribūtiem vērtības.

Entītiju veidi, kurus var aprakstīt mērīšanas procesā, var būt, piemēram, produkti, procesi, resursi, organizācijas, vides, ierobežojumi, u.c. Pamatā šīs metodes nozīmi var raksturot citāts [PGF96]: „Problēma un māksla mērījumu procesā ir izvēlēties pareizos atribūtus, lai iegūtu noderīgu priekšstatu par entītiju, ko mēra”.

Mērķu bāzēts mērīšanas process [PGF96] balstās uz 3 pieņēmumiem un sastāv no 10 soļiem. Pieņēmumi:

- mērījumu mērķi tiek iegūti no biznesa mērķiem,
- mērīšanas procesa uzmanības centrā ir mentālie modeļi, kas attīstās,
- GQ(I)M pārveido neformālus mērķus par izpildāmām mērīšanas struktūrām.

Primārais mehānisms, lai pārvērstu biznesa mērķus par jautājumiem un mērījumiem ir mentālie modeļi. [PGF96] tiek izmantots mentālā modeļa jēdziens – abstrakcija, ar kuras palīdzību novērojumus pārvērst lēmumos un darbībās. Formalizēt mentālo modeli var, piemēram, procesu modeļa veidā. Metodē lietoto vispārīgu procesu modeli apraksta shēma (skat. 22.att.).

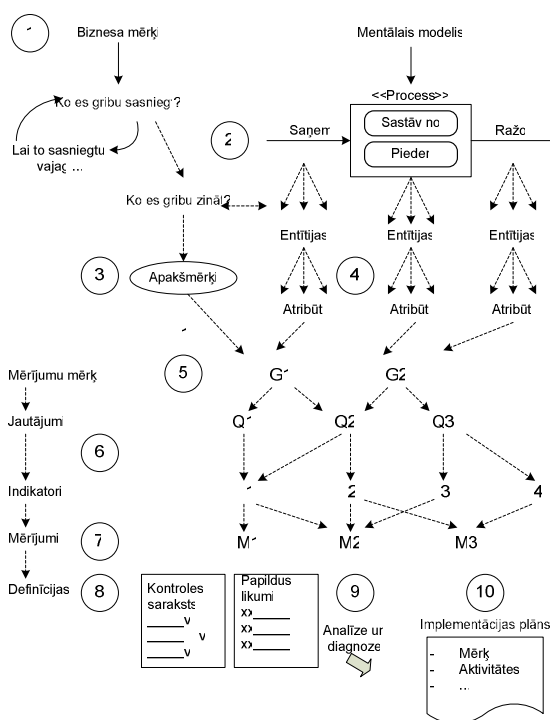


22.att. Vispārīgs procesa modelis pēc [PGF96]

Šāda tipa procesu modelis satur četrus veidu elementus:

- saņemtās „lietas” (ievade un resursi) – tiek izmantotas un pārstrādātas;
- saražotās „lietas” (izvade) – produkti, blakusprodukti, efekti;
- procesa modeļa komponenti – aktivitātes, plūsmas, aģenti;
- „lietas”, kas tiek pārvaldītas un izmantotas (iekšēji artefakti, piemēram, noliktava, darbarīki, dati, zināšanas).

Mērķu bāzēta mērīšanas procesa 10 soļi atspoguļoti attēlā (skat. 23.att.), bet soļu apraksts dots 4. pielikumā.



23. att. Metode mērījumu noteikšanai ar 10 soļiem, pēc [PGF96]

4.5. Procesu datu noliktavas jēdziens

Datu noliktava kā risinājums procesu datu uzkrāšanai ir aprakstīta vairākos rakstos, katrā izmantojot dažādas procesu datu noliktavas definīcijas. Atšķirīgo definīciju un līdz ar to piedāvāto risinājumu pamatā ir vairāki iespējami faktori:

- dažādi datu avoti – darba plūsmas rīku audita failu izmantošana vai neizmantošana, un izmantošanas gadījumā to integrēšana ar citu avotu datiem, atsevišķs gadījums – tikai šī faila datu analīze;

- dažādas biznesa jomas – specifisks gadījums ir automatizēti ražošanas procesi;
- dažāds datu analīzes veids - datu analīzes veidi darba plūsmu audita failu analīzes gadījumam ir minēti [Mue01], bet tie ir spēkā arī vispārīgas procesu datu noliktavas gadījumā:

- dažādi datu analīzes mērķi (tehniskie, biznesa);

Tehnisko mērķu gadījumā analīzi veic, piemēram, sistēmas administratori un darba plūsmu izstrādātāji, šādu mērķu piemēri var būt gan darba plūsmas sistēmu testēšana, gan sistēmas noslodzes un atbildes laika mērīšana.

- atšķirīgs laika periods – īstermiņa analīze jeb monitorings, kas nozīmē aktīvu (tādu, kas izpildās) darba plūsmu datu analīzi un ilgtermiņa analīze jeb darba plūsmu kontrole, kad darba plūsmu audita dati tiek analizēti pēc darba plūsmu izpildes;

Monitorings tehnisko mērķu gadījumā ir, piemēram, noslodzes, aktīvo lietotāju skaita, atbildes laika mērīšana, bet biznesa mērķu gadījumā tas var būt, piemēram, īstermiņā nepieciešamo resursu prognoze. Monitorings vajadzīgs operatīvai reaģēšanai uz problēmsituācijām darba plūsmas izpildes laikā, bet kontrole vajadzīga, lai tālākā perspektīvā uzlabotu darba plūsmu, piemēram, noskaidrotu neefektīvus elementus darba plūsmas uzbūvē.

- atšķirīgi lietotāji – tiek ņemtas vērā individuāla lietotāja vai organizācijas vajadzības;
- dažādas datu analīzes perspektīvas – klientu, darbinieku, vides, attīstības, apmācības, procesu, resursu un citas;
- dažādas potenciālo lietotāju grupas (procesu pārvaldnieki, procesu izpildītāji, IT personāls).

Dažādo iepriekš pieminēto faktoru esamība katrā konkrētā gadījumā nosaka gan vajadzīgos datus, kas jāiekļauj modelī, gan arī šo datu detalizācijas pakāpi.

Runājot par datiem, jāpiemin arī dažāda tipa dati, kas potenciāli izmantojami datu noliktavās. *Workflow Management Coalition* ir publicējusi 3 veidu datu definīcijas, kas saistītas ar darba plūsmu sistēmām [WFMC99]:

- lietojumprogrammu dati – dati kas tiek apstrādāti lietojumprogrammās,
- darba plūsmu būtiskie dati – nosaka izpildes ceļu izvēli, vaicājumi tiek izpildīti darba plūsmas izpildes laikā, dati netiek laboti, parasti arī netiek uzkrāti audita failos,
- darba plūsmu iekšējie dati – tiek uzkrāti audita failos, dati par darba plūsmas izpildi.

Workflow Management Coalition ir publicējusi arī savu audita failu formātu, kuru parasti ražotāji modificē. Tomēr parasti netiek uzglabāti dati, kas tiek apstrādāti darba plūsmas instancē.

Ja par datu avotu tiek izmantoti audita failu dati, tie nodrošina iespēju analizēt procesu izpildi attiecībā pret laiku, pret resursu izmantošanu, bet atbildēt uz jautājumiem par biznesa stāvokli parasti nav iespējams, jo pārsvarā komerciālie darba plūsmu pārvaldības rīki neglabā datu izmaiņas, kas notiek lietojumprogrammās, kuras tiek izsauktas darba plūsmas izpildes laikā [Mue01]. Tādēļ viena no problēmām, kas jārisina, ir kā veikt audita failu integrāciju ar citiem datu avotiem, kurā glabājas dati, kas radīti vai mainīti procesu izpildes laikā. Datu noliktava ir viens no veidiem, kā integrēt procesu izpildes datus ar biznesa specifiskiem datiem.

Ir aprakstīti vairāki dažāda tipa datu noliktavu projekti, kurus varētu saukt par procesu datu noliktavām, kaut arī autori dažādi interpretē šo jēdzienu. Tālāk tiks apskatīti daži no tiem, lai raksturotu dažādo „procesu datu noliktavas” jēdziena izpratni.

4.5.1. Ražošanas procesu datu noliktava

Datu noliktavu var izmantot arī ražošanas procesu analīzei. [JLK00] definē procesu datu noliktavu (*Process Data Warehouse*) „kā datu noliktavu, kas glabā datus par ražošanas procesu izpildi un rezultātiem pieredzes iegūšanai un atkārtotai izmantošanai”.

[JLK00] piedāvāto risinājumu raksturo:

- procesu datu noliktava integrē datus no dokumentu pārvaldības sistēmām, ražošanai specifiskām datu bāzēm, ražošanas rīkiem un iekārtām, trasēšanas rīkiem u.c. savā starpā vāji saistītiem dažāda tipa heterogēniem datu avotiem, izmantojot aktīvu repozitoriju tehnoloģiju,
- konceptuālā līmenī „uzņēmuma modelis” jāsadala vāji saistītos daļējos modeļos, kas katrs raksturo kādu atšķirīgu ražošanas procesa aspektu, loģiskā un fiziskā līmenī – ražošanā izmantoto rīku heterogenitāte ir daudz lielāka nekā tradicionāliem OLAP datu avotiem, tāpēc nepieciešams papildus standartizācijas solis, ne tikai datu, bet arī servisu līmenī, lai dati vispār būtu pieejami.

4.5.2. Darbu plūsmu datu noliktava

[BCDS01] autori argumentē, kāpēc darba plūsmu rīku procesu izpildes žurnālu datu analīzei var izmantot datu noliktavu. Katru ierakstu žurnāla failā raksturo, piemēram, laiks, notikuma tips, procesa identifikators, aktivitātes tips, aktivitātes stāvoklis, izpildītā programma, u.c. atribūti. Esošie analīzes rīki piedāvā ierobežotas analīzes iespējas – izpildīto procesu instanču skaits laika periodā, vidējais izpildes laiks, nav integrācijas iespēju ar citiem datu avotiem. Autori piedāvā darba plūsmas datu noliktavu (*Workflow Data Warehouse*). Darba plūsmas rīks, kas kalpo par datu avotu realizētajam prototipam – *HP Process Manager*.

Autori izstrādājuši universālu datu modeli, kas paredzēts daudzu savstarpēji saistītu faktu tipu glabāšanai - darba plūsmu izpilde var ģenerēt dažāda tipa faktus – piemēram, aktivitāte tiek izpildīta konkrētas plūsmas definīcijas kontekstā, pa noteiktu zaru. Risinājumu raksturo universāls modelis, no darba plūsmas rīka ražotāja neatkarīgs, dimensijas ir vienkāršas, nemainīgas, jaunums- *Behavior* dimensija, kas paredzēta analīzes vajadzībām, definē plūsmu izpildes tipiskus modeļus (*Patterns*). Fakti tiek dalīti atsevišķās tabulās – kopējā tabulā daudziem rīkiem kopējie atribūti, atsevišķās tabulās – modeļu specifiskie atribūti. Ir papildus faktu tabula

Process – Behavior, kur datu noliktavas fakti pēc ielādes tiek analizēti pēc atbilstības vienam vai vairākiem iepriekš definētiem uzvedības modeļu veidiem. Pēc tam iespējams analizēt „labu” vai „sliktu” izpildes scenāriju cēloņus un papildus faktu tabulas – *Process Data Fact* – kas apraksta, kādus datus katrs process ir mainījis. Dažādiem procesu tiptiem iespējams specificēt, kādu datu atribūtu izmaiņas vajag uzkrāt.

4.5.3. Izpildes pārvaldības sistēma

Izpildes pārvaldības sistēmas jēdzienu (*Performance Management System*) definē [KWL01], ar to saprotot sistēmu, kas centralizēti glabā un pārvalda visus procesu izpildes datus, gan finansiālos, gan nefinansiālos, kas vēl arī nodrošinātu sistemātisku procesu mērīšanu un datu laicīgu pieejamību.

Pamatā ir ideja, ka ilgtermiņā ir būtiski ne tikai pārbūvēt biznesa procesus, bet arī tos nepārtraukti turpināt uzlabot, atsaucoties uz Haringtonu [Har91] par mērījumu svarīgo lomu.

Rakstā [KWL01] piedāvātais risinājums ar datu noliktavu balstās uz metodi, kas sākas ar organizācijas un tās mērķu analīzi. Tiek noteikti uzņēmuma līmeņa mērķi, nosaka ieinteresētos, biznesa procesus un biznesa procesu mērķus. Šo informāciju izmanto, lai noskaidrotu atbilstošus izpildes rādītājus. Iesaistīti ir trīs kategoriju lietotāji - vadītāji, procesu īpašnieki, procesu dalībnieki. Konkrētajam procesam katrai lietotāju kategorijai tiek noskaidrotas informācijas vajadzības.

Arhitektūra ietver divas datu bāzes – datu noliktavu par mērījumu vērtībām un palīginformācijas datu bāzi par: 1) organizācijas struktūras elementiem- organizācijas struktūrvienībām, struktūrvienību vadītājiem, darbiniekiem, biznesa procesiem; 2) mērķu koku, mērķu atbilstību organizācijas struktūras elementiem, par mērķiem atbilstošiem mērījumiem un mērījumu sasniedzamajām vērtībām; 3) pieejas tiesībām gan izpildes rezultātiem, gan palīginformācijas datiem. Būtiska iezīme – jautājumi gan uzņēmuma, gan procesu līmenī tika formulēti, lai ietvertu sekojošus aspektus (skatījumus) – finansu, klientu, darbinieku, attīstības un vides aspektus. Procesu perspektīva tiek analizēta ierobežoti, procesu hierarhija tiek izmantota tikai lai analizētu izpildes laiku, bet ne darba plūsmas izpildes secību un nosacījumus, kas ir tipiski darba plūsmu datu noliktavām.

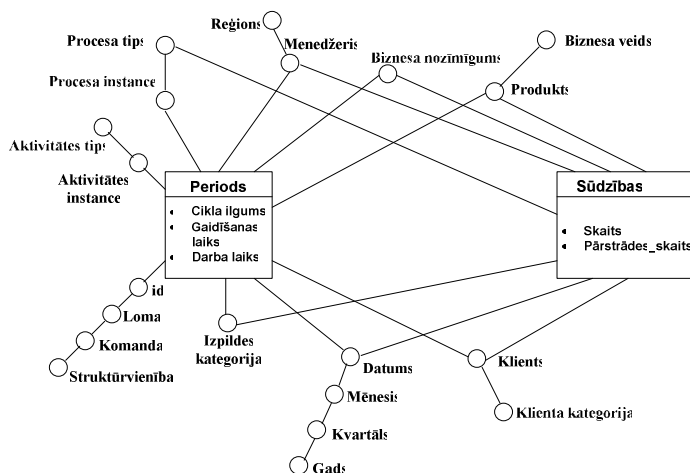
4.5.4. Korporatīvā izpildes mērīšanas sistēma

[LM04] piedāvā korporatīvo izpildes mērīšanas sistēmu (*Corporate Performance Measurement System - CPMS*), kur procesu izpildes dati tiek integrēti uzņēmuma datu noliktavā. Kā datu avotu iekšējo biznesa procesu raksturošanai [LM04] izmanto darba plūsmas sistēmas žurnāla failu. CPMS datu noliktavas projektējuma izveide notiek pēc sekojošas metodes:

- no kompānijas mērķiem tiek noteikti analizējamajiem biznesa procesiem specifiski mērķi pēc *Wal-Mart* metodes [Wes01];
- tiek paredzētas divas lietotāju kategorijas – procesu īpašnieki un struktūrvienību vadītāji, kuriem tiek definēti interesējošie biznesa jautājumi (noskaidro intervijās), kam savukārt piesaistīti atbilstoši mērījumi un norādīti mērījumu ieguves datu avoti (darba plūsmu sistēmu žurnāla faili

vai informācijas sistēmas). Rezultātā tiek izveidota tabula (biznesa jautājums, mērījums, datu avots). Netiek detalizēti aprakstīts, kā tiks integrēti procesu izpildes dati no žurnāla failiem, kas parasti rada problēmas;

- izmantojot *Dimensional Fact Model* [GMR98a], [GMR98b] notāciju, bet ne DFM metodi, izveidots datu noliktavas modelis, saskaņots ar jau eksistējošu korporācijas datu noliktavas modeli (skat. 24.att.).



24.att. Datu modelis “Korporatīvai izpildes mērīšanas sistēmai”, pēc [LM04]

Secinājumi par procesu mērīšanas sistēmām

Salīdzinot dažādās procesu datu noliktavu definīcijas, modeļus un realizācijas, var secināt:

- procesu izpildes mērīšana jāveic no dažādiem aspektiem. Nepietiek tikai ar darba plūsmu rīku žurnālfailu analīzi, jo tas ļauj analizēt procesa izpildi attiecībā pret laiku, bet nav iespējams izdarīt secinājumus par biznesa kopīgo stāvokli, jo datu izmaiņas notiek lietojumprogrammās. Tātad nepieciešama procesu izpildes datu integrēšana ar citiem datu avotiem. Tajā pašā laikā neinteresējoties par procesu izpildes gaitu, bet tikai par rezultātu, ko raksturo dati lietojumprogrammās, pietrūkst būtisks aspekts, lai analizētu veidus, kā procesa rezultātus uzlabot;

- izpildes rādītāju definēšanā būtu jābalstās uz kompānijas stratēģiju. Rādītāju noskaidrošanai jāizvēlas šim nolūkam piemērota metode. Realitātē parasti izmanto viena tipa datu avotu, piemēram, darba plūsmas žurnālu, līdz ar to rādītājus nevar brīvi definēt, to definēšanu ierobežo pieejamie dati;

- datu noliktava kā risinājums procesu mērīšanas sistēmai ir piemērota, ja galvenokārt jāveic ilgtermiņa procesu izpildes analīze, bet ne monitoringa gadījumā, tāpat datu noliktava būs piemērots risinājums organizācijas vajadzību analīzei.

5. PIEDĀVĀTĀS METODES KONCEPTUĀLO MODEĻU IZVEIDEI

5.1. Nodaļas nolūks

Iepriekšējās nodaļās aprakstītas eksistējošas metodes datu noliktavu konceptuālo modeļu izstrādei. Īpaša uzmanība veltīta jaunākajām tendencēm - mērķa vadītai pieejai un koncepcijai par procesu datu noliktavu. Analīzes laikā konstatēta dažādu pieeju un daudzu metožu eksistence, tajā pašā laikā neviena no metodēm nav pieņemta par standartu. Katrai no pieejām konstatēti gan trūkumi, gan pozitīvi aspekti. Katra no apskatītajām citu autoru metodēm izmantojama noteiktā situācijā.

Šīs nodaļas mērķis ir piedāvāt trīs jaunas metodes, kuras tiek veidotas atbilstoši metožu inženierijas nostādnēm, kas aprakstītas 5.2. apakšnodaļā. Novērtējot esošo situāciju datu noliktavu projektos, situāciju iespējams aprakstīt ar to raksturojošiem faktoriem, kas kalpo par pamatu esošu metožu komponentu izmantošanai vai adaptēšanai, vai nosaka vajadzību izstrādāt jaunu situācijai specifisku metodes komponentu. Apakšnodaļā 5.3. aprakstīta jauna lietotāju vadīta metode, apakšnodaļā 5.4. - jauna datu vadīta metode, bet apakšnodaļā 5.5. - jauna mērķu vadīta metode. Katrai metodei novērtēti situāciju raksturojošie faktori.

Metodes aprobētas Latvijas Universitātes datu noliktavas izstrādes projektos, katra no tām izmantota atsevišķas datuves izstrādē.

Par zinātnisko jauninājumu uzskatāmas katra no aprakstītajām metodēm kopumā, kā arī atsevišķi var runāt par šo metožu komponentu nozīmi. Šo metožu konstruēšanā izmantoti eksistējošu metožu komponenti adaptētā veidā un izstrādāti jauni komponenti. Katrai jaunajai metodei atbilstošās apakšnodaļas beigās dotas kopsavilkumu tabulas par komponentu lietošanas mērķiem un situācijas faktoriem, norādot komponenta tipu (jauns vai adaptēts) un raksturojot adaptētiem komponentiem adaptācijas būtību.

Situācijas faktori ir radušies pētījumā, kas veikti šī promocijas darba ietvaros, rezultātā un arī ir uzskatāmi par zinātnisko jauninājumu.

Metodes publicētas sekojošos rakstos: [BN04] - lietotāju vadīta metode, [SN05] - datu vadīta metode, [NSTN07a] un [NSTN07b] - mērķa vadīta metode.

5.2. Metodes definēšana

Informācijas sistēmu izstrādes laikā atkarībā no konkrētās biznesa situācijas var būt nepieciešams eksistējošas metodes adaptēt vai izstrādāt jaunu. Iegūto pieredzi nepieciešams aprakstīt, lai tā būtu atkārtoti izmantojama. Metodes jādziens tika dots 3. nodaļā. Tālāk tiks apskatīti jēdzieni, kas ir būtiski informācijas sistēmas izstrādes metodes aprakstīšanai.

"Metode ir pieeja sistēmas izstrādes projekta veikšanai, kas balstīta uz domāšanas veidu un sastāvoša no norādījumiem un likumiem, strukturētiem sistemātiskā veidā izstrādes aktivitātēs, kam atbilst noteikti izstrādes produkti" [Bri96].

"Tehnika ir procedūra, iepējams ar iepriekš definētu notāciju, lai veiktu izstrādes aktivitāti" [Bri96]. Saskaņā ar šo definīciju tehnikas jēdziens ietver gan notāciju, gan procedurālus aspektus, kā notācija lietojama.

"Rīks ir izstrādes aktivitātes atbalsta līdzeklis, iespējams automatizēts" [Bri96]. Kā piemērus var minēt CASE rīkus, veidnes (*template*).

Promocijas darbā izmantotas definīcijas no [Bri96], jo tās ir saskaņotas ar metožu inženierijas idejām, kas tiek apskatītas un izmantotas promocijas darbā.

5.2.1. *Metožu inženierija un jaunu metožu definēšana*

Runājot par informācijas sistēmas izstrādes metodi, ir būtiski runāt par:

- metodi veidojošām sastāvdaļām, teknikām, rīkiem;
- domāšanas veidu [Bri96], filozofiju [AF03], kas slēpjas aiz metodes, pieņēmumiem, skaidrojumiem, kāpēc sistēma strādā norādītajā veidā. Kā piemērs šādam pieņēmumam var būt lietotāju cieša iesaistīšana sistēmas izstrādē visa dzīves cikla laikā;
- metodes procesa un produkta aspektiem. Metožu aprakstos parasti atrodamas instrukcijas, kādā secībā veicamas izstrādes aktivitātes un strukturālās prasības produktiem, kas tiek saukti arī par nodevumiem.

[Bri96] piedāvā pētījumu ietvaru informācijas sistēmu izstrādes metodēm un rīkiem, nosaucot šo ietvaru par metožu inženieriju. "Metožu inženierija ir nozare informācijas sistēmu izstrādes metožu, tehniku un rīku projektēšanai, konstruēšanai un adaptēšanai" [Bri96]. Informācijas sistēmu izstrādes jomā pēc nozīmes līdzīgu terminu *metodoloģiju inženierija* ieviesa Kumars un Velke [KW92], apzīmējot ar to metožu konstruēšanu, adaptējot tās konkrētu izstrādes projektu vajadzībām.

Tiek izdalīta arī īpaša metožu kategorija - situāciju specifiskās metodes, ko [Bri96] definē kā "metodes, kas ir konfigurētas konkrētā projekta vajadzībām".

Metožu inženierija apskata trīs galvenās stratēģijas:

- Jauna metode tiek veidota, ja nav iespējams pielietot nevienu no esošām metodēm. Tās tiek sauktas par eksromptmetodēm [RDR03];
- Konstruēšana nozīmē, ka jaunā metode tiek uzbūvēta no metodes komponentiem jeb fragmentiem [RDR03]. Šo pieeju sauc arī par integrāciju [LVP07];
- Adaptācija [LVP07] nozīmē, ka kādai esošai metodei tiek izlaistas vai modificētas kādas no tās daļām.

Klasifikāciju papildina arī paplašinājumu bāzētas metodes [RDR03], kas nozīmē eksistējošas metodes papildināšanu ar jauniem jēdzieniem un īpašībām, bet to var uzskatīt par adaptācijas veidu metodes modificēšanas nozīmē. Biežāk lietotā stratēģija, ko aplūko metožu inženierijas kontekstā, ir metodes konstruēšana. Tā tiek izmantota arī promocijas darbā piedāvāto metožu izstrādei. Šajā stratēģijā ir būtisks metodes komponenta jēdziens, tādēļ tas tiks tuvāk apskatīts apakšpunktā 5.2.1.1.

5.2.1.1. Metodes komponenti un to lietošanas konteksts

Metodi var uzskatīt par metodes komponentu jeb fragmentu kopu [Har97]. Komponenti var būt vai nu produkta komponents, vai procesa komponents. Produkta komponenti var būt nodevumi - dokumenti, modeļi, diagrammas vai jēdzieni. Procesu komponenti var būt fāzes, aktivitātes, uzdevumi. Komponenti var tikt aizgūti no citām metodēm.

[Rol97] savukārt izmanto metodes "gabala" jēdzienu, lai apzīmētu metodes komponentu, kā arī izmanto konteksta jēdzienu, lai aprakstītu metodes komponenta lietošanu. Konteksts tiek definēts kā pāris $\langle \text{situācija}, \text{lēmums} \rangle$.

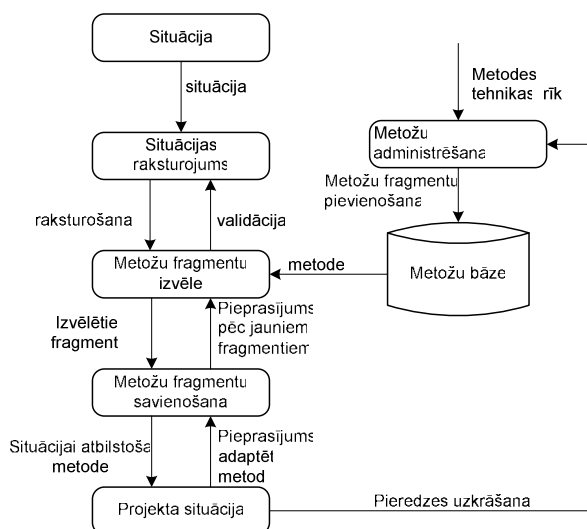
Lēmumu par metodes fragmenta piemērotību noteiktai situācijai nosaka: 1) metodes fragmenta lietošanas nolūks, 2) mērķa sasniegšanas veids (tehnika), 3) sasniedzamais mērķis (vērtējums).

Situācijas raksturošanai [Rol97] iesaka vērtēt: problēmas apgabalu (*problem domain*) un projekta sfēru (*project scope*). Vērtēšanas kritēriji, kas raksturo metodes situāciju metožu inženierijas kontekstā ir maz izpētīti [Rol97], [LVP07]. Apakšpunktā 5.2.1.3. tiks apskatīti situācijas faktori (*contingency factor*) informācijas sistēmu izstrādes jomā, kurus var izmantot par vienu no avotiem, lai iegūtu kritērijus, kā vērtēt metodes un metodes komponenta situāciju metožu inženierijā.

5.2.1.2. Metodes konstruēšanas process un reprezentācija

Metožu inženieriju var aplūkot no vairākiem aspektiem [Rol97]: metodes reprezentācija, process, kā šī reprezentācija tiek izveidota un metodes lietošanas raksturojums.

Jaunas, situācijai pielāgotas metodes konstruēšanas process aprakstīts vairākos avotos [Bri96], [Har97], [RDR03]. [Har97] piedāvātais procesa modelis redzams attēlā (skat.25.att.).

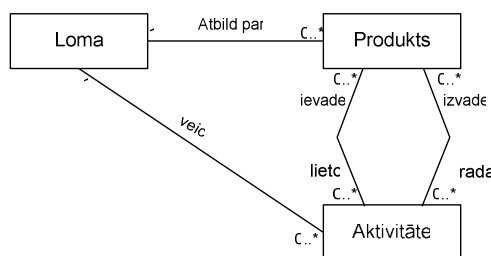


25. att. Situācijai atbilstošu metožu izveides process, pēc [Har97]

Galvenie procesa soļi ir: 1) situāciju raksturojošo faktoru noteikšana izstrādājamajai sistēmai, 2) metožu bāzes aizpildīšana ar metodēm, rīkiem, tehnikām, sadalot katru iepriekš minēto komponentos un novērtējot tiem raksturojošos faktorus un 3) jaunas metodes konstruēšana no metožu komponentiem, izvēloties tos atbilstoši 1. solī noteiktajiem faktoriem no metožu bāzes un tos savstarpēji integrējot.

Metožu reprezentēšanai ir liela nozīme, jo metožu inženierijai jāpalīdz izvēlēties no esošu metožu klāsta noderīgu gatavu vai adaptējamu metodi, vai arī jāpalīdz konstruēt jaunu metodi no citu metožu komponentiem. Būtisks ir jautājums, kā metodes reprezentāciju pierakstīt. Metožu inženierijā metožu reprezentācijas formālisti parasti balstās uz eksistējošiem semantiskiem modeļiem [RR01], piemēram, E/R modeli, UML, bet ir izstrādātas arī jaunas valodas metožu aprakstam, piemēram, MEL (*Method Engineering Language*) [Har97].

Promocijas darbā metožu apraksti tiks veidoti, bastoties uz programmatūras izstrādes procesa modeli, kā tas definēts *Software Process Engineering Metamodel* (SPEM) metamodeli [OMGd]. Attēlā (skat. 26.att.) redzama pamatstruktūra SPEM metamodelim.

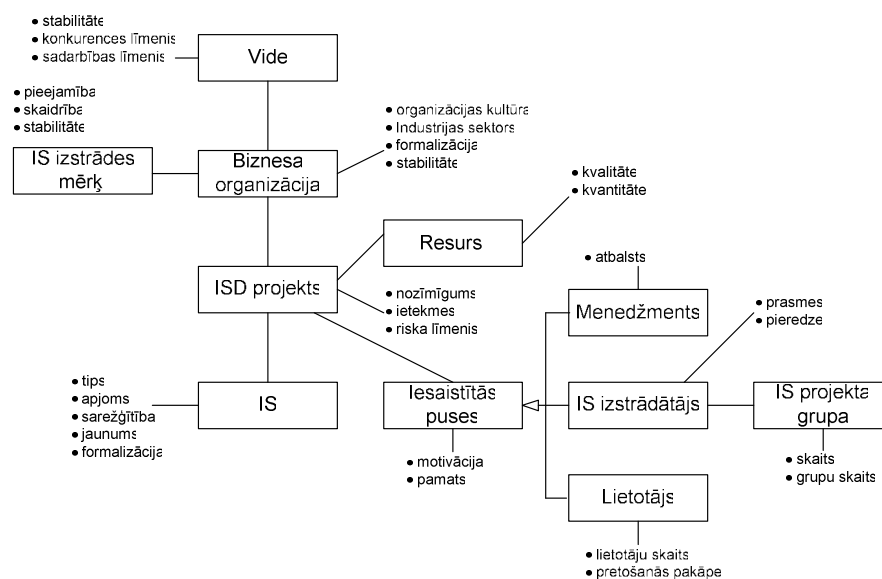


26.att. Programmatūras izstrādes procesa modelis, pēc [OMGd]

Pilns procesa metamodelis ir sarežģītāks, tajā 'Produkts' var sastāvēt no citiem 'Produktiem', katram no tiem var apskatīt stāvokļus un pieļaujamās pārejas atkarībā no stāvokļiem. Katra aktivitāte var sastāvēt no vēl sīkākām aktivitātēm, sauktām par soļiem. Aktivitātei var būt priekšnosacījumi un mērķis, aktivitāte ir saistīta ar aktivitātes grafu, kas apraksta soļu secību aktivitātē. Promocijas darbā lomas aspekti netiek analizēti.

5.2.1.3. Situācijas raksturojums

Metožu izstrāde, integrācija vai adaptācija metožu inženierijas ietvaros ir atkarīga no apkārtējā vides konteksta, ko var raksturot ar situācijas faktoriem. Viens no variantiem, kā aprakstīt situāciju, ir novērtēt sekojošus faktorus kā IS nozīmīgums, apjoms, atkarība no ārējiem faktoriem, cilvēkresursu pietiekamība, procedūru formalizācijas pakāpe u.c. Tomēr situācijas faktori tieši metožu inženierijai ir maz pētīti [LVP07]. Informācijas sistēmu izstrādes jomā turpretī ir bijuši daudzi priekšlikumi par situācijas faktoriem [SS96], [FF99], [MR05], [LVP07]. Šo pētījumu rezultāti var palīdzēt arī metožu inženierijā modeļu, tehniku un metožu konstruēšanai. Piemēram, informāciju sistēmu izstrādes jomā no literatūras apkopotie dažādie faktori ir redzami attēlā [LVP07] (skat.27.att). Situācijas faktori informācijas sistēmu izstrādes jomā atkarīgi no pieejas izvēles, prasībām, kādiem jābūt modeļiem, kā strukturēts izstrādes process, kā lietotāji iesaistās procesā. Piemēram, var minēt dažus faktorus - informācijas sistēmas izstrādes mērķu pieejamība, stabilitāte un skaidrība, ieinteresēto pušu motivācija, ieinteresētība.



27.att. Situāciju raksturojošie faktori IS izstrādei, pēc [LVP07]

Situācijas faktori var tikt noskaidroti divējādi [LVP07]: induktīvi un deduktīvi. Pirmajā gadījumā izpēta, kādi situācijas faktori ietekmējuši lēmumu pieņemšanu metožu inženierijas procesā, lai izvēlētos konkrētas procedūras, tehnikas utt. Otrajā gadījumā situācijas faktoros noskaidro no piejamās literatūras, kas ietver gan konceptuālus pamatojumus, gan konkrētu projektu aprakstus. Promocijas darbā tiek lietota induktīva pieeja situācijas faktoru noskaidrošanai.

5.2.2. Promocijas darbā piedāvāto metožu definēšana

Metodes apraksts tiks veidots, balstoties uz programmatūras izstrādes procesa modeli, kā tas definēts *Software Process Engineering Metamodel* (SPEM) metamodelī [OMGd].

Lai metodes apraksts būtu detalizēts, sistemātisks, saprotams un pārklātu visus būtiskos aspektus, ieskaitot aktivitātes un to detalizāciju soļos, produktus un tehnikas, metode tiks aplūkota divās detalizācijas pakāpēs un tiks aplūkoti gan metodes procesa, gan produkta aspekti.

Par pamatu aprakstam tiks ņemts metodes procesa aspekts, kur galvenā uzmanība vērsta uz aktivitāšu, procesa soļu secības aprakstīšanu. Vienlaicīgi katrai aktivitātei, atbilstoši SPEM metamodelim, tiks aprakstīti ar to saistītie produkti - gan aktivitātes ievadē, gan aktivitātes izvadē (nodevumi).

Ir izvēlēti divi detalizācijas līmeņi. Nedatalizētais apraksts ietver galvenās metodes aktivitātes un to secību. Detalizētajā metodes aprakstā tiek atspoguļota gan katra metodes aktivitāte, gan produkti. Katrai aktivitātei tiek definēti izpildes soļi un to secība, aprakstot arī pielietojamās tehnikas un paņēmienus.

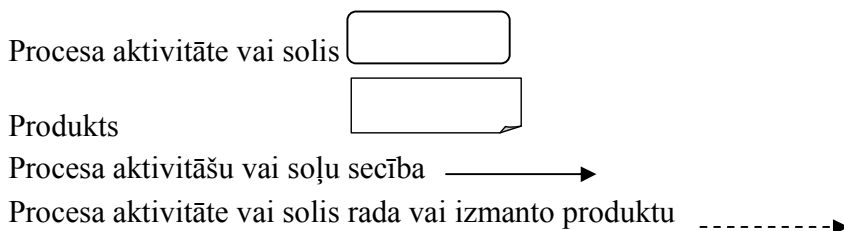
Metodes apraksta detalizētajā līmenī ir aprakstīti arī katra metodes komponenta (aktivitātes, ja netiek detalizēts soļos, vai soļa) lietošanas konteksts - lietošanas mērķis un izvēli ietekmējošie faktori. Atbilstoši apakšpunktā 5.2.1.1. rakstītajam, metodes komponents var būt gan procesa tipa komponents, gan produkta tipa komponents. Produkta tipa komponentu

lietošanas konteksti nav aprakstīti, jo promocijas darbā piedāvātajās metodēs tie sakrīt ar saistīto procesa komponentu kontekstiem.

Tā kā komponenta lietošanas kontekstu raksturo pāris <situācija, lēmums>, tad tiek aprakstīta gan lietošanas situācija, gan lēmums.

Ilustrācijai metodes aprakstā ir izmantoti piemēri no LU datu noliktavas, kuras konceptuālā modeļa izstrādē metode pielietota.

Metodes procesa modelī lietotie apzīmējumi :



5.3. Lietotāju vadītā metode

Šajā apakšnodaļā tiks piedāvāts jaunas lietotāju vadītās metodes (turpmāk tekstā - LVM) apraksts. Tiek izmantoti metožu inženierijas metožu konstruēšanas un metožu adaptācijas paņēmieni. Tiek izmantoti vai adaptēti eksistējošu metožu komponenti, kā arī izstrādāti jauni metožu komponenti, novērtējot metožu komponentu izvēles un lietošanas kontekstus.

5.3.1. Lietotāju vadītās metodes procesa modelis

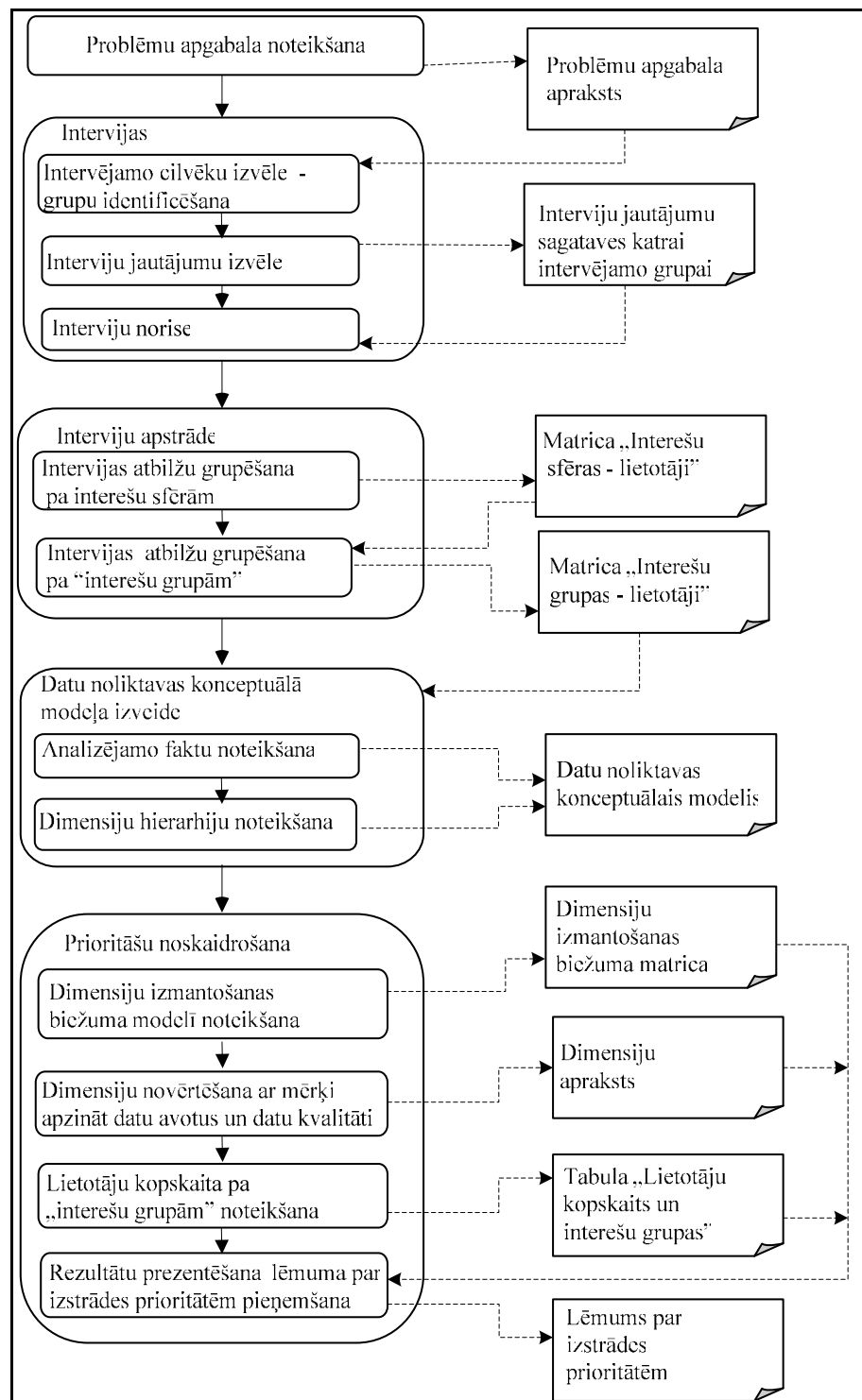
Atbilstoši nosaukumam - LVM, tā balstās uz lietotāju vajadzību noskaidrošanu intervijās, šīs informācijas apkopošanu, sistematizēšanu un izmantošanu datu noliktavas konceptuālā modeļa izstrādē. Izmantotas atsevišķas idejas no [KRRT98] - kā intervēt lietotājus, precīzāk, divu atsevišķu tipu jautājumu sagataves – „darbības mērķi un ietekmējošie faktori” un „analīzes prasības”. Šīs idejas papildinātas ar interviju rezultātu apstrādes veidu un izmantošanu.

Tālāk tiks dots LVM procesa modelis (skat.28.att.), bet sīkāks metodes apraksts atrodams punktā 5.3.2.

LVM procesa modeli raksturo sekojošas aktivitātes:

- problēmu apgabala noteikšana;
- intervijas, kur sākumā notiek intervējamo cilvēku izvēle. Tiek identificētas potenciālās lietotāju grupas. Notiek interviju jautājumu izvēle, lai varētu noskaidrot darba mērķus, kvalitātes rādītājus, kā arī ikdienas darbā analizējamus datus. Iepriekš tiek sagatavots jautājumu saraksts, pielāgoti varianti katrai potenciālai lietotāju grupai. Intervējamo cilvēku atbildes tiek reģistrētas tabulā;
- interviju apstrāde, kas notiek, izmantojot divu tipu matricas veida tabulas - „Interesu sfēras→lietotāji”, kurā katra tabulas šūniņa ir atbildes uz interviju jautājumiem, un “Interesu grupas→ lietotāji”, kas tiek lietota grupēšanas mērķiem;
- konceptuālā modeļa izstrāde, par pamatu ņemot noskaidrotās analīzes vajadzības un datu avotu modeļus;

- prioritāšu noskaidrošana: dimensiju apraksts ar mērķi apzināt datu avotus un datu kvalitāti, dimensiju izmantošanas biežuma novērtēšana, potenciālo lietotāju skaita noteikšana, rezultātu apspriešana un lēmuma par prioritātēm pieņemšana.



28.att. LVM procesa modelis

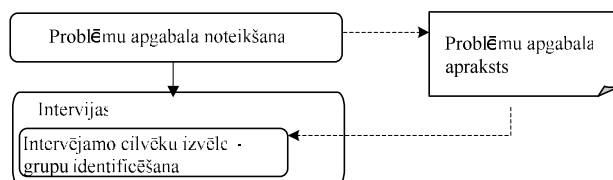
5.3.2. *Lietotāju vadītās metodes detalizēts apraksts*

1. aktivitāte "Problēmu apgabala noteikšana"

1. aktivitātes procesa aspekti

Lietotāju vadītās pieejas izvēle par galveno un primāro pieeju bieži ir kā sekas tam, ka sākumā nav iespējams precīzi izdalīt problēmas apgabalu. Tas šajā gadījumā pārklāj visas organizācijas funkcijas. Tādēļ, lai vēlāk ierobežotu problēmas apgabalu, būtiska loma ir prioritāšu noteikšanai, kas tiks veikta tālākās metodes aktivitātēs.

Aktivitātei "Problēmu apgabala noteikšana" nav detalizētu soļu. Attēlā (skat.29.att.) redzams fragments no metodes kopējā procesa modeļa, kas atspoguļo šīs aktivitātes saistību ar citām aktivitātēm un produktiem.



29.att. Aktivitāte "Problēmu apgabala apraksts" LVM metodei

Metodes komponentam, kas atbilst 1. aktivitātei, konteksts ir sekojošs:

- Situācijas raksturojums: no pasūtītāja puses nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic. Potenciāli iespējamie analizējamie biznesa procesi ir visas organizācijas darbības sfēras.
- Lēmums:
 - metodes komponenta lietošanas mērķis: metodes komponents nepieciešams problēmu apgabala definēšanai;
 - tehnika - pārrunas ar pasūtītāja pārstāvi un projekta sfēras apraksta sagatavošana;
 - vērtējums - tiek izmantots gatavs, datu noliktavām nespecifisks, metodes komponents, bet produktu (noteikto problēmu apgabalu un aprakstu) lielā mērā ietekmē situācija.

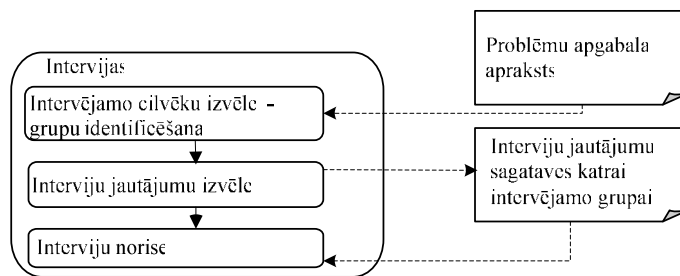
1. aktivitātes produkta aspekti

Šī soļa rezultātā tiek izveidots produkts "Problēmu apgabala apraksts" - dokuments, kurā tiek aprakstīts problēmu apgabals. Apraksta formai nav noteiktas sagataves.

2. aktivitāte "Intervijas"

2. aktivitātes procesa aspekti

Zīmējumā (skat.30.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Intervijas". Tālāk seko katra aktivitātes soļa apraksts.



30.att. Aktivitāte "Intervijas"

Šajā aktivitātē to veikšanas secībā var apskatīt trīs soļus :

2.1. solis. "Intervējamo cilvēku izvēle – grupu identificēšana"

Pielietojot lietotāju vadītu pieeju, būtiska ir intervējamo potenciālo lietotāju izvēle. Vienošanās par intervējamo kategorijām notiek ar pasūtītāja pārstāvi. Izvēle notiek pēc Kimbala metodē (skat. apakšpunktu 3.5.4.1) lietotiem 2 principiem. Organizāciju pārklāj horizontāli, lai būtu pārstāvētas dažādas lietotāju kategorijas un līdz ar to dažādas analīzes intereses. Organizāciju pārklāj arī vertikāli - sākot ar lietotājiem, kas veic datu analīzi, un beidzot ar dažādu līmeņu struktūrvienību vadītājiem, kuru pienākumos ietilpst lēmumu pieņemšana.

Piemēram, Latvijas Universitātes datu noliktavas gadījumā tika izvēlētas sekojošas lietotāju kategorijas – augstākā līmeņa vadība (LU vadība), vidējā līmeņa struktūrvienību vadītāji (fakultāšu dekāni, departamentu direktori), lietotāji - darbinieki, kas nodarbojas ar datu analīzi, LU gadījumā - departamentu darbinieki un fakultāšu izpilddirektori. Katrai lietotāju grupai nākošā solī tika izstrādātas pielāgotas intervijas.

Metodes komponentam, kas atbilst 2.1. solim, konteksts ir sekojošs:

- Situācijas raksturojums:
 - no pasūtītāja puses nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic;
 - nepieciešams noskaidrot pēc iespējas plašāka potenciālo lietotāju loka datu analīzes vajadzības.
- Lēmums:
 - metodes komponenta lietošanas mērķis: metodes komponents nepieciešams intervējamo personu loka noteikšanai;
 - tehnika - Kimbala metodes [KRRT98] fragments par diviem principiem lietotāju grupu izvēlē;
 - vērtējums: tiek izmantots gatavs metodes fragments.

2.2. solis "Interviju jautājumu izvēle"

Būtiska nozīme ir arī jautājumu saturam. Interviju jautājumiem dažas idejas, kā izvēlēties jautājumus, izmantotas pēc [KRRT98]. Jautājumu izvēle notiek pēc principa – galvenā prioritāte ir tiem jautājumiem, kas vērsti uz biznesam kritisku mērījumu noskaidrošanu, jo esošās biznesa

procesu un cilvēku datu analīzes vajadzības lielā mērā papildus uzziņamas arī no avotu datu modeļiem un datu avotu sistēmās jau lietotām atskaitēm.

Intervijās iekļaujami jautājumi no divām jautājumu grupām: „Organizācijas un konkrētās struktūrvienības darbības mērķi un ietekmējošie faktori” un „Analīzes prasības”. Daži jautājumu piemēri:

- Kādi ir kritēriji, kā vērtēt sasniegto? Kā novērtēt, ka darbs tiek darīts veiksmīgi?
- Kādas funkcijas ir viskritiskākās darbā?
- Kāda ir galvenā informācija, kas nepieciešama darbā, kā šī informācija tiek iegūta pašreiz?

Jautājumi no iepriekš sagatavota jautājumu saraksta tiek novērtēti, vai tie ir piemēroti konkrētajam biznesa procesam un tā veicējam, kas tiek intervēts.

Metodes komponentam, kas atbilst 2.2. solim, konteksts ir sekojošs:

- Situācijas raksturojums: gatavā veidā izmantot jautājumu sagataves no [KRRT98] nav iespējams, to saturs jāmodificē atbilstoši intervējamai personai, jo tiek intervēti daudzi lietotāji.
- Lēmums:
 - metodes komponenta lietošanas mērķis- sagatavoties intervijai;
 - tehnika - Kimbala metodes [KRRT98] komponents par interviju jautājumu sagatavju izveidošanu, jautājumus grupējot divās daļās, kas tika aprakstīts iepriekš šajā nodaļā;
 - vērtējums: tiek izmantots gatavs metodes [KRRT98] komponents, adaptējot tajā izmantoto produkta komponentu (jautājumu sagatavi), pielāgojot situācijai.

2.3. solis "Interviju norise"

Intervijas organizējamās atsevišķi organizācijas augstākā līmeņa vadības pārstāvjiem, lai noskaidrotu organizācijas stratēģijai būtiskus mērījumus. Vidējā līmeņa vadītājus intervijās apvieno ar viņiem pakļautajiem darbiniekiem. No vadītājiem tiek noskaidroti lēmumu pieņemšanai būtiski rādītāji. No darbiniekiem, kas tieši veic datu analīzi, tiek iegūts precīzāks skaidrojums par vadītājus interesējošiem rādītājiem no datu viedokļa. Bez tam katra datu analītiķa atsevišķa intervēšana dod tikai daļēju, nepilnīgu skatījumu uz organizācijas vajadzībām, kā to pamato arī [Goe05]. Intervijās tiek izmantoti jautājumi no iepriekš izstrādātām jautājumu sagatavēm.

Metodes komponentam, kas atbilst 2.3. solim, konteksts ir sekojošs:

- Situācijas raksturojums: potenciāli daudzi intervējamie, ko nosaka 2.1. solī identificētās lietotāju grupas;
- Lēmums:
 - metodes komponenta lietošanas mērķis- veikt intervijas;
 - tehnika - atsevišķu lietotāju grupu apvienošana pa vertikālo organizācijas pārklājumu, intervijās tiek izmantotas iepriekš 2.2. solī izveidotās sagataves, interviju atbildes tiek dokumentētas;
 - vērtējums: datu noliktavu kontekstā pieejamajā literatūrā, metodes komponents, kad zemākā līmeņa intervējamo grupas tiek apvienotas "vertikāli" noteiktā

situācijā, nav pieminēts. To var uzskatīt par metodes komponenta [KRRT98] adaptēšanu, pielietojot modificēšanu.

2. aktivitātes produkta aspekti

Aktivitāte izmanto produktu "Problēmu apgabala apraksts" kā vienu no avotiem potenciālo intervējamo grupu noteikšanai, pārrunājot šos jautājumus ar projekta pasūtītāju.

Šīs aktivitātes rezultātā tiek izveidots produkta komponents "Interviju jautājumu sagataves katrai intervējamo grupai". Jautājumu sagataves veido, grupējot jautājumus 2 daļās:

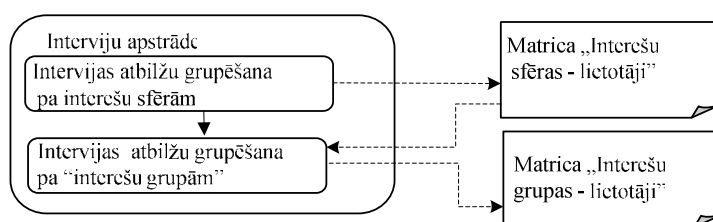
1. daļa. Organizācijas un konkrētās struktūrvienības darbības mērķi un ietekmējošie faktori. Jautājumi ir gan par organizāciju kopumā, gan par struktūrvienību, ko pārstāv intervējamie; jautājumu sagatave ietver jautājumus, ko grib sasniegt, kā vērtēt virzību uz mērķi, kas visbūtiskāk var ietekmēt mērķa sasniegšanu gan pozitīvi, gan negatīvi.

2. daļa. Analīzes prasības - kādu datu analīzi veic intervējamie, kādas ir vēl neapmierinātās analīzes vajadzības.

3. aktivitāte. "Interviju apstrāde"

3.aktivitātes procesa aspekti

Attēlā (skat.31.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Intervijas". Tālāk seko katra aktivitātes soļa apraksts.



31.att. Aktivitāte "Interviju apstrāde"

Šajā aktivitātē to veikšanas secībā var apskatīt divus soļus:

3.1. solis. "Intervijas atbilžu grupēšana pa interešu sfērām"

Interesu sfēras tiek definētas kā grupēšanas līdzeklis līdzīgām atbildēm. Interesu sfēru definēšanu veic intervētājs, balstoties uz interviju atbildēm, grupējot pēc lietotājiem nepieciešamās informācijas līdzības.

Piemēram, analīzes vajadzībās noskaidrotas prasības, ka nepieciešami sekojoši rādītāji: plānotais un esošais studentu skaits pa studiju programmu līmeņiem, fakultātēm; beidzēju skaits; budžeta un maksas studentu skaits. Prasības var grupēt kā interešu sfēru "Studentu statistikas dati".

Šādas grupēšanas rezultātā iespējams izveidot matricas veida tabulu "Matrica – Interesu sfēras →Lietotāji", kurai viena matricas dimensija ir „interesu sfēras”, otra dimensija ir „lietotāji” (intervētās lietotāju grupas vai atsevišķi lietotāji), matricas šūniņa – konkrētās atbildes uz

interviju jautājumiem, kas raksturo nepieciešamos datu analīzes rādītājus. Tabulas sagatave ar piemēriem apskatāma 3. aktivitātes produkta aspekta aprakstā.

Papildus priekšrocība intervētājam ir tad, ja ir labi zināmas datu avotu sistēmas un tās apakšsistēmas, jo tās norāda uz eksistējošiem biznesa procesiem ar uzkrātiem datiem. Tieši avotu sistēmu apakšsistēmu nosaukumi var kalpo par „interesu sfērām”. Šī pieeja izmantojama tikai kā papildus apsvērums, definējot interešu sfēras.

Metodes komponentam, kas atbilst 3.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: intervijās konstatēto prasību plašais spektrs, nepieciešamība grupēt atbildes pēc to līdzības, lai būtu iespējams vieglāk noteikt izstrādes prioritātes;
- lēmums:
 - metodes komponenta lietošanas mērķis- strukturēt interviju atbilžu dažādās datu analīzes prasības;
 - tehnika - intervijas atbilžu grupēšana, izmantojot sagataves matricu, kas aprakstīta metodes 3. aktivitātes "Interviju apstrāde" produkta aspektā (skat. produkts 3.1);
 - vērtējums: datu noliktavu kontekstā pieejamajā literatūrā metodes komponents, kad tiek veidotas šādas "Interesu sfēras", nav pieminēts. To var uzskatīt par jauna metodes komponenta konstruēšanu.

3.2. solis. Intervijas atbilžu grupēšana pa “Interesu grupām”

Solis 3.2. izmanto solī 3.1. izveidoto tabulu "Matrica – Interesu sfēras →Lietotāji" (skat. produkts 3.1.) un tā tiek pārveidota par līdzīga rakstura tabulu " Matrica - Interesu grupas → Lietotāji" (skat. produkts 3.2). Notiek interešu sfēru apkopošana lielākās grupās - "Interesu grupās", lai atvieglotu lēmuma pieņemšanu, ko izvēlēties par prioritāti izstrādei. Tiek meklēta līdzība starp jau nodefinētajām interešu sfērām un līdzības gadījumā tās tiek apvienotas vienā grupā.

Piemēram, apvienotas var tikt interešu sfēras „Studentu statistikas dati” un „Doktorantu statistikas dati”.

Rezultātā tiek izveidota jauna matrica, kurai viena matricas dimensija ir „interesu grupas”, otra dimensija ir „lietotāji” (intervētās lietotāju grupas vai atsevišķi lietotāji), matricas šūniņa tiek aizpildīta ar koeficienta k vērtību:

- $k=1$, ja lietotājs ir izteicis prasības par atbilstošo interešu grupu;
- $k=1,5$, ja lietotājs uzsvēris intervijās, ka atbilstošā interešu grupa ir "galvenā prioritāte".

Matricas izveides mērķis ir analizēt, cik ir potenciālo lietotāju katrai interešu grupai. Lai uzsvērtu kādas lietotāju kategorijas prasību objektīvu svarīgumu datu analīzē, var apspriest un izvēlēties papildus koeficientu p un to pielietot. Šai lietotāju kategorijai matricas šūnu vērtības būs $k \cdot p$. Tabulas sagatave ar piemēriem apskatāma 3. aktivitātes produkta aspekta aprakstā.

Metodes aprobācijas laikā LU datu noliktavas projektā no 19 interešu sfērām rezultātā tika izveidotas 11 interešu grupas. LU datu noliktavas gadījumā papildus koeficientu p izvēles pamatā bija pieņēmums, ka datu noliktava orientēta uz lēmumu pieņemšanu, nevis operatīvās

informācijas izmantošanu. Tika izmantoti papildus koeficienti $p = 1$ fakultāšu lietotāju grupām, $p = 2$ vadības un departamentu lietotājiem vai to grupām.

Metodes komponentam, kas atbilst 3.2. solim, konteksts ir sekojošs:

- situācijas raksturojums:
 - intervijās konstatēto prasību plašais spektrs, nepieciešamība grupēt atbildes pēc to līdzības, lai būtu vieglāk noteikt izstrādes prioritātes;
- lēmums:
 - metodes komponenta lietošanas mērķis - samazināt interešu sfēru skaitu, ja tas ir nepieciešams. Piemēram, interešu sfēru skaits iepriekšējā solī ir pietiekoši liels, lai būtu apgrūtināta izstrādes prioritāšu noteikšana. Otrs mērķis - noskaidrot potenciālo lietotāju skaitu katrai interešu grupai, ņemot vērā izteiktās prasības;
 - tehnika - interešu sfēru grupēšana un koeficientu pielietošana, izmantojot sagataves matricu, kas aprakstīta metodes aktivitātes "Interviju apstrāde" produkta aspektā (skat. produkts 3.2.),
 - vērtējums - datu noliktavu kontekstā pieejamajā literatūrā metodes komponents, kad tiek veidotas šādas "Interešu grupas", nav pieminēts. To var uzskatīt par jauna metodes komponenta konstruēšanu.

3. aktivitātes produkta aspekti

Aktivitāte "Interviju apstrāde" lieto un izveido diva tipa produktus - matricas veida tabulas. Sagataves abām tabulām promocijas darbā tiek dotas ar datiem no metodes aprobācijas projekta LU datu noliktavā.

3.1 produkts. Matrica – „Interešu sfēras” → „Lietotāji”

Viena matricas dimensija „Interešu sfēras”, otra dimensija „Lietotāji” (intervētās lietotāju grupas vai lietotāji), matricas šūniņa – konkrētās atbildes uz interviju jautājumiem - nepieciešamie datu analīzes rādītāji. Jēdzieni un matricas pielietošana aprakstīta pie metodes procesa aspektiem.

Fragmenti no izveidotās matricas redzams tabulā (skat. 1.tabula).

1.tabula

Matrica „Interešu sfēras” → „Lietotāji”

	Studentu dati	Studiju programmas un kursi	Darbinieki	Projekti
Rektora palīgs	Plānotais un esošais studentu skaits pa līmeņiem, fakultātēm. Beidzēju skaits.	Studiju programmas, to skaits un attīstība	Profesoru skaits. Personāla informācija ar algām, saraksti pa departamentiem, vakances, konkursi.	Projektu skaiti, finansējuma apjoms, struktūra
Kancelers	Budžeta un maksas studentu skaits	Studiju programmu virziena un līmeņa koeficienti, studiju maksa	Personāls, slodzes, algas, komandējumi, vieslektori	–

3.2. produkts. Matrica “Interešu grupas” → “Lietotāji”

Viena matricas dimensija ir „Interesu grupas”, otra dimensija „Lietotāji” (intervētās lietotāju grupas). Jēdzieni un matricas pielietošana aprakstīta pie metodes procesa aspektiem.

Matricas fragments redzams tabulā (skat. 2.tabula), tajā atspoguļoti piemēri no LU datu noliktavas projekta. Piecas interešu grupas tika vēlāk projektētas, izmantojot dimensionālo modelēšanu, dažas tika realizētas kā dimensijas (tabulā gaiši pelēkas), dažām interešu grupām trūka izejas datu – parādījās nepieciešamība pēc jaunām OLTP sistēmām (tabulā tumši pelēkas).

2.tabula

Matrica “Interesu grupas” → “Lietotāji”

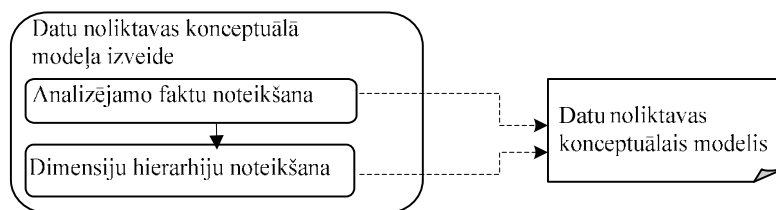
	Studenti un doktoranti	Darbinieki	Budžeta plānošana	Kontu apgrozījums	Telpas un saimniecība	Studiju programmas un kursi	Projekti	Rīkojumi	Ārējie sakari	Audits	Pārējais
Rektors	2	2	2	2	2	2	2	2	2	2	–
Rektora palīdzība	2	2	2		2	2	2	2	2	–	–
Kanclers	2	2		3	2	2	–	–	–	–	–
...											
Attīstības un plānošanas departaments	2	2	2	–	–	–	3		2	–	–
Ārējo sakaru departaments	2	2	–	–	–	–	–	–	2	–	–
...											
FMF	1	1	–	1	1	–	1	–	–	–	1
Pedagoģijas un psiholoģijas fakultāte	1	1	1	1.5	–	–	–	–	–	–	1
...											
Kopā:	38	34	9	29.5	24	16	14	8	8	4	13

4. aktivitāte. Datu noliktavas konceptuālā modeļa izveide

Balstoties uz apkopoto interviju informāciju, tiek veikta datuvju konceptuālā modelēšana. Modeļi veidoti, ņemot par pamatu interviju atbildes no matricas "Interesu sfēras"- "Lietotāju grupas", bet apvienojot analīzes prasības, atbilstoši tālāk noteiktajām interešu grupām. Modeļos iekļauti nepieciešamie atribūti un fakti, lai atbildētu uz visām lietotāju izvirzītajām problēmām un biznesa jautājumiem konkrētajai interešu grupai. Parasti vienai interešu grupai tiek izveidoti vairāki daudzdimensiju konceptuālie modeļi, jo vienā nav iespējams ietvert visas lietotāju prasības par analizējamajiem faktiem.

4.aktivitātes procesa aspekti

Zīmējumā (skat.32.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Datū noliktavas konceptuālā modeļa izveide". Tālāk seko katra aktivitātes soļa apraksts.



32.att. Aktivitāte "Datu noliktavas konceptuālā modeļa izveide"

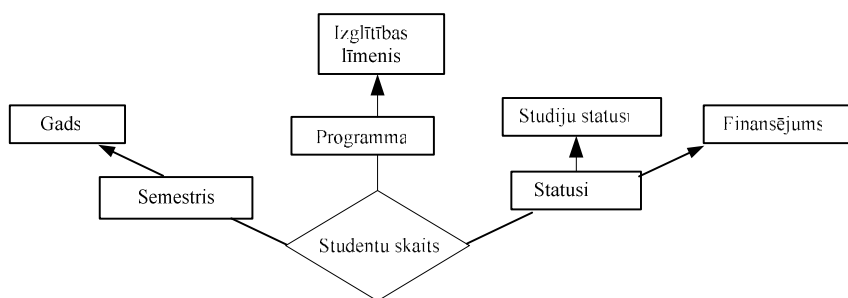
Šajā aktivitātē to veikšanas secībā var apskatīt divus soļus.

4.1. solis. Analizējamo faktu noteikšana

Konceptuālai modelēšanai izmanto modificētu M/ER notācību (skat. 4. aktivitātes produkta aspektu aprakstu). Šajā solī var lietot arī citu datu noliktavu konceptuālai modelēšanai paredzētu notācību. Modelējot no katras prasības izdala skaitlisku lielumu, kuru vajag analizēt. Ja prasībā ietverta arī informācija par analīzes dimensijām, tad to attēlo modelī kā dimensiju hierarhijas zemāko līmeni.

Piemēram, analīzes vajadzība "Maksas studentu skaits pa studiju programmu līmeņiem un pa mācību gada semestriem" ir sadalāma sekojoši - fakts: Studentu skaits; interesējošās analīzes dimensijas: Maksa/budžets; Studiju programmu līmeņi; Semestris. Cita šīs pašas interešu grupas "Studentu un doktorantu statistika" prasība "Analizēt pirmā semestra doktorantu skaitu pa gadiem un programmām" sadalāma sekojoši - fakts: Doktorantu skaits; interesējošās analīzes dimensijas: Satuss (studenta semestris); Studiju programma; Gads.

Izveido dimensiju modeļa sākotnējo variantu, par izejas punktu ņemot no prasībām noskaidrotos faktu atribūtus. Pēc nozīmes vienādus faktus apvieno vienā modelī. Piemēram - studentu skaits un doktorantu skaits tiek apvienoti vienā modelī. Abiem iepriekš minētajiem piemēriem, ņemot vērā arī prasīto analīzes dimensiju līmeņu savstarpējo hierarhisko pakļautību, var izveidot attēlā (skat.33. att.) redzamo modeli.



33. att. Dimensiju modeļa sākotnējais variants

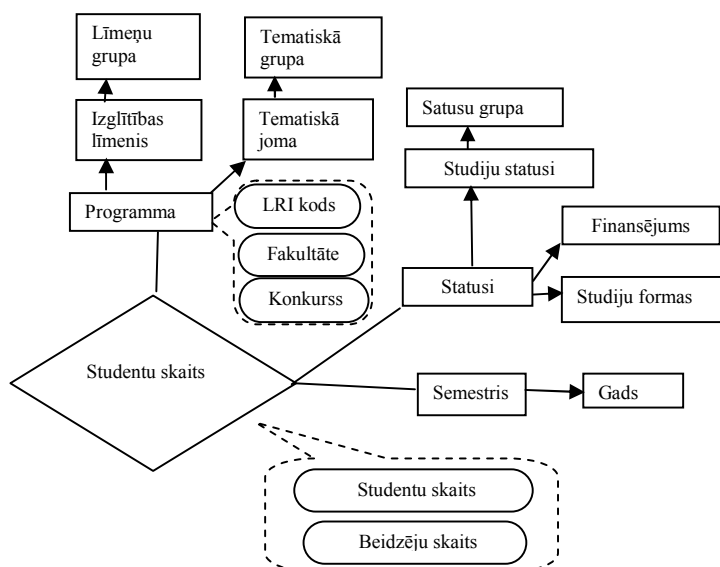
Metodes komponentam, kas atbilst 4.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: datu analīzes prasības, kas sagrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli;

- lēmums:
 - metodes komponenta lietošanas mērķis - no datu analīzes prasībām izdalīt, un ja iespējams, apvienot faktu atribūtus, noteikt dimensijas;
 - tehnika - katras dokumentētās prasības apstrāde, izdalot iespējamus faktus un dimensiju atribūtus, konceptuālā modelēšana;
 - vērtējums: pēc prasībām būvēta daudzdimensiju modeļa izstrāde tiek pielietota visās prasību vadītās pieejās, ieskaitot lietotāju vadītā pieejā, šāds sākotnējs modelis būtībā atbilst fakta definīcijai, piemēram, pēc ME/R konceptuālā modeļa. Tā kā ME/R konceptuālam modelim nav tiešā veidā aprakstīta metode, kā noteikt faktus no prasībām, metodes komponents adaptēts, pievienojot ideju, kā analizēt dokumentētās analīzes prasības.

4.2. solis. Dimensiju hierarhiju noteikšana

Šajā solī turpinās konceptuālā modeļa izstrāde, papildinot sākotnēji uzbūvētos vienkāršotos faktus ar dimensiju hierarhijām, hierarhiju līmeņu atribūtiem. Kā informācijas avots tiek izmantoti datu avotu datu modeļi un to apraksti. No tiem, balstoties uz eksistējošām datu savstarpējām atkarībām, tiek noskaidrotas papildus iespējamās hierarhijas un atribūti. Būtībā pielietojama jebkura no datu vadītām metodēm dimensiju hierarhiju noteikšanai, piemēram, DFM. Rezultātā, ņemot vērā, ka vienā modelī var apvienot faktus, kam sakrīt to definējošie dimensiju hierarhiju zemākie līmeņi, iegūstams konceptuālais modelis, kas redzams attēlā (skat.34.att.). Tajā apvienoti fakti - studentu skaits un beidzēju skaits. Notācijās apraksts dots tālāk šīs aktivitātes produkta aspektu aprakstā.



34.att. Konceptuālais modelis, papildināts ar hierarhijām

Metodes komponentam, kas atbilst 4.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: datu analīzes prasības, kas sagrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli;
- lēmums:
 - metodes fragmenta lietošanas mērķis - sakotnējo konceptuālo modeli, kas iegūts tikai balstoties uz prasībām, papildināt ar dimensiju hierarhijām un aprakstošiem atribūtiem;
 - tehnika - datu avotu modeļu analīze;
 - adaptēts DFM, var lietot arī citus, datu vadītus metožu komponentus. Par izejas punktiem DFM lietošanai izmanto tos faktus, kas konstatēti no prasībām iepriekšējā solī, atrodot atbilstošus atribūtus datu avotu modeļos, tālāk lieto DFM.

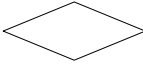
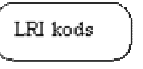
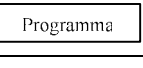
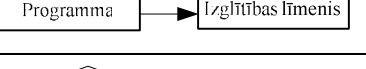
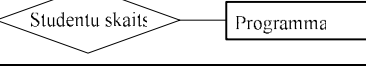

4. aktivitātes produkta aspekti

4.1. produkts - Konceptuālais modelis

Konceptuālai modelēšanai tika izmantota modificēta M/ER notācija. Konceptuālā modeļa piemērs no metodes aprobācijas projekta LU datu noliktavā, dots metodes atbilstošā soļa aprakstā. Diagrammas lasāmībai atribūti tika grupēti vizuāli, kas svarīgi, ja izmanto modeli arī prezentācijai. Apzīmējumi doti tabulā (skat.3.tabula.).

3. tabula

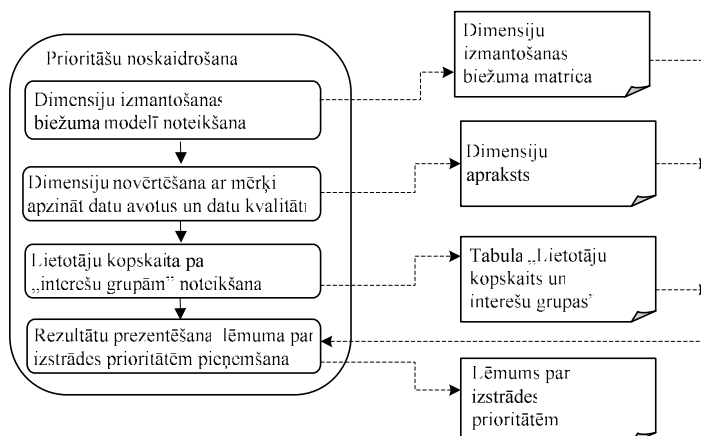
Apzīmējumi modificētai M/ER notācijai

Apzīmējums	Skaidrojums
	Fakts, piemēram, studentu skaits
	Fakta vai dimensijas atribūts, fakta gadījumā tiek saukts arī par mērījumu
	Dimensijas hierarhijas līmenis
	Dimensiju hierarhiju līmeņu saistība ar „roll-up” attiecību
	Attiecība starp faktu un dimensijas zemāko hierarhijas līmeni
	Papildus ieviests atribūtu grupēšanas līdzeklis diagrammas lasāmības dēļ. Viena atribūta gadījumā netiek lietots.

5. aktivitāte. Prioritāšu noskaidrošana.

5.aktivitātes procesa aspekti

Attēlā (skat. att.35.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Prioritāšu noteikšana". Tālāk dots katra aktivitātes soļa apraksts.



35. att. Aktivitāte "Prioritāšu noteikšana"

Šajā aktivitātē to veikšanas secībā var apskatīt četrus soļus.

5.1. solis. Dimensiju izmantošanas biežuma modelī noteikšana

Datu noliktava tiek plānota atbilstoši Kimbala koncepcijai [KRRT98] par „*DW bus architecture*”, kas balstās uz ideju par kopīgām dimensijām vairākiem daudzdimensiju modeļiem. Tiek izveidota matricas veida tabula, kas atspoguļo, kuras dimensijas konceptuālajos modeļos, kas tika izveidotas iepriekšējā aktivitātē, ir kopīgas visām datuvēm - daudzdimensiju modeļiem. Matricas veida tabulas rindas atspoguļo daudzdimensiju konceptuālā modeļa dimensijas, kolonnas atspoguļo daudzdimensiju modeļus. Tabula ar piemēru no metodes aprobācijas projekta LU datu noliktavā redzams aktivitātes produkta aspektu aprakstā (skat.4.tabula).

Metodes komponents var palīdzēt novērtēt potenciālo darba apjomu, kas svarīgs, lai pieņemtu lēmumus par realizācijas prioritātēm.

Metodes komponentam, kas atbilst 5.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: ir daudzas prasības un nepieciešamība noteikt prioritātes izstrādei;
- lēmums:
 - metodes komponenta lietošanas mērķis: novērtēt potenciālo darba apjomu dimensiju izstrādei;
 - tehnika - Kimbala metodes [KRRT98] „*DW bus architecture*” pielietošana;
 - vērtējums - tiek izmantots gatavs [KRRT98] metodes komponents.

5.2. solis. Dimensiju novērtēšana ar mērķi apzināt datu avotu datu kvalitāti

Tiek izveidota tabula, kas satur dimensiju un to atribūtu aprakstus. Tabulas struktūra dota aktivitātes produkta aspektu aprakstā (skat.5.tabula). Tiek dots datu kvalitātes novērtējumu un nepieciešamo transformāciju (*mapping*) apraksts. Ja iespējams, apraksta formāli, ja nē, tad aprakstoši, lai varētu novērtēt nepieciešamos resursus, kas saistīti ar datu kvalitātes problēmu novēršanu, kā arī lai novērtētu, vai nepieciešamie dati eksistē datu avotos.

Lai pieņemtu lēmumu par izstrādes prioritātēm, šī kvalitātes novērtējuma tabula un matrica par dimensiju izmantošanas biežumu ir viens no papildus informācijas avotiem, lai novērtētu darbu apjomu, bet ne interešu prioritātes.

Metodes komponentam, kas atbilst 5.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: daudzas prasības un nepieciešamība noteikt prioritātes izstrādei;
- lēmums:
 - metodes komponenta lietošanas mērķis: apzināt datu avotu datu kvalitāti;
 - tehnika - datu kvalitātes novērtēšanai raksturo: datu eksistenci, datu aizpildījumu, datu atbilstību pieļaujamai vērtību kopai. Tiek identificēts nepieciešamo transformāciju apjoms, lai iegūtu dimensijai vajadzīgos datus;
 - vērtējums - tiek konstruēts metodes komponents atbilstoši situācijai.

5.3. solis. Lietotāju kopskaita pa „interesu grupām” noteikšana

Tika izveidota tabula, kas dota aktivitātes produkta aspektu aprakstā (skat. 6.tabula), kur datuves grupētas pa „interesu grupām”, kādas tika identificētas iepriekšējos soļos, un norādīts arī potenciālais lietotāju skaits pa grupām. Izmantota informācija no matricas “Interesu grupas” → “Lietotāji”, bet netiek ņemti vērā koeficienti.

Metodes komponentam, kas atbilst 5.3. solim, konteksts ir sekojošs:

- situācijas raksturojums: daudzas prasības un nepieciešamība noteikt prioritātes izstrādei;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot potenciālo lietotāju skaitu dažādām datuvēm;
 - tehnika - izmanto dokumentētās prasības, kas grupētas, atbilstoši informācijai no 3. aktivitātē izveidotajām matricas veida tabulām;
 - vērtējums - metodes komponents izstrādāts atbilstoši situācijai.

5.4. solis. Rezultātu prezentēšana un lēmuma par izstrādes prioritātēm pieņemšana

Rezultātu prezentēšana pasūtītājam notiek jau konkrētu datuvju projektējumu veidā, kas jau iepriekš modelēti un novērtēti. Ja zināmi projektam piešķirtie resursi, to ietvaros iespējams piedāvāt atšķirīgas alternatīvas.

Priekšlikumu izvirzīšanai var izmantot sekojošus kritērijus:

- potenciālais lietotāju skaits katrai interešu grupai (skat. 6.tabula), ne tikai konkrētai datuvei,
- potenciālais lietotāju skaits, ņemot vērā koeficientus no matricas “Interesu grupas” → “Lietotāji”, kas atspoguļo vairāk tieši datu analīzes, ne operatīvās informācijas vajadzības,
- atbilstošo datu esamība un kvalitāte datu avotos,
- izstrādājamo datuvju sarežģītība, piemēram, dimensiju skaits, tajā skaitā kopējas dimensijas interešu sfērā,
- izstrādātāju zināšanas par datu avotiem.

Pēc diskusijām notiek vienošanās par kādu no ieteiktajiem priekšlikumiem, kādas datuves nepieciešams realizēt ar vislielāko prioritāti.

Metodes komponentam, kas atbilst 5.4. solim, konteksts ir sekojošs:

- situācijas raksturojums: daudzas prasības un nepieciešamība noteikt prioritātes izstrādei;
- lēmums:
 - metodes komponenta lietošanas mērķis: pieņemt lēmumu par izstrādes prioritātēm;
 - tehnika - dažādu faktoru (minēti iepriekš komponenta aprakstā) izvērtēšana un salīdzināšana ar pieejamiem resursiem;
 - vērtējums - tiek konstruēts metodes komponents atbilstoši situācijai.

5.aktivitātes produkta aspekti

5.1. produkts. Dimensiju izmantošanas biežuma matrica

Piemērs dimensiju izmantošanas biežuma tabulai dots ar datiem no metodes aprobācijas projekta LU datu noliktavā. Piemērs redzams tabulā (skat. 4.tabula). Tabula atspoguļo matricas fragmentu.

4 Tabula

Dimensiju izmantošanas biežums

Dimensijas/Datuves	Studentu statusa maiņa	Uzņemšanas rezultātu statistika	Studentu maksājumi	Studenta reģistrācija uz kursu
Laiks	+	+	+	+
Cilvēks	+	+	+	+
Adrese	+	+		
Programma	+	+	+	+
Statusi	+		+	+
Rikojums	+		+	
Finansējums		+		

Iepriekš veiktās konceptuālās modelēšanas laikā izveidotiem 18 zvaigznes shēmas modeļiem kopā bija 29 dimensijas.

5.2. produkts. Dimensiju apraksts

Dimensiju apraksta tabula (skat. 5.tabula) satur dimensiju atribūtu datu kvalitātes novērtējumu un nepieciešamo transformāciju (*mapping*) aprakstu, lai datus iegūtu no datu avotiem. Transformācijas tiek aprakstītas tikai tik daudz, lai novērtētu darbu apjomu. Apraksts šajā solī vēl nav uzskatāms par pabeigtu datu izguves, transformācijas un ielādes procesu specifikāciju.

5. tabula

Dimensiju novērtēšana

Atribūta nosaukums	Nepieciešamās transformācijas	Datu kvalitātes novērtējums	Vērtību piemēri

5.3. Produkts. Tabula „Lietotāju kopskaits un interešu grupas”

Tabulā (skat. 6.tabula) dots fragments no metodes aprobācijas projektā izveidotās tabulas un lietoti sekojoši apzīmējumi:

- **Studenti un doktoranti** - "Interešu grupas";
- Studentu statusa izmaiņas – datuves nosaukums, atbilst vienam modelim;
- kursīvā – jautājumu piemēri;
- (6+8+11) – lietotāju skaits: (1.– vadība; 2. - departamenti; 3. – fakultātes).

6. tabula

Lietotāju skaits pa „Interešu grupām”

Studenti un doktoranti (6+8+11)	Darbinieki (6+7+8)	Budžeta plānošana (3+2+1)	Kontu apgrozījums (4+4+11)	Telpas un saimniecība (6+4+6)
Studentu statusa izmaiņas <i>Studentu atbirums. Kāds ir studentu studēšanas ilgums?</i>	Darbinieku statusa maiņa <i>Darbinieku statistikas Darbinieku komandējumi, to ilgums, izmaksas</i>	Centralizētas dotācijas struktūrvienībām	Savstarpējie norēķini <i>Cik studenti no citām fakultātēm klausās kursu? Cik naudas par to saņem pasniedzēja struktūrvienība</i>	Telpas <i>LU telpu sadalījums pa fakultātēm un ēkām pa tipiem Firmas, kuras ir telpas LU</i>
Uzņemšanas rezultātu statistika <i>Reģioni, no kuriem nāk visvairāk reflektantu.</i>	Darbinieku publikācijas <i>Publikāciju saraksts pa nozarēm, to klasifikācija.</i>	Studiju maksas Plānotie pašu ieņēmumi	Transakcijas <i>Cik naudas ir iztērēts pa dažādām LU budžeta pozīcijām pa gadiem</i>	Telefonu rēķini <i>Telefonu rēķinu sadalījums pa numuriem</i>

5.4. Produkts. Lēmums par izstrādes prioritātēm

Tiek izveidots dokuments, kurā tiek aprakstīti pieņemtie lēmumi par izstrādes prioritātēm. Dokumentā ietverami apsvērumi, kas bija par pamatu šo lēmumu pieņemšanai.

5.3.3. Secinājumi un rezultāti par lietotāju vadīto metodi

Izstrādāta metode, balstoties uz lietotāju vadītu pieeju. Metodē tiek izmantoti gan 4 gatavi metodes komponenti, gan adaptēti 3 citu metožu komponenti atbilstoši situācijai, gan konstruēti 5 jauni komponenti, kas galvenokārt saistīti ar prioritāšu noteikšanu, lai noskaidrotu prasību un līdz ar to arī izstrādes prioritātes. Kopsavilkums par visiem metodes komponentiem dots tabulā (skat. 8. tabula), tajā raksturots komponenta lietošanas mērķis, situācijas faktori, norādīts, vai komponents jauns (J), adaptēts (A) vai paņemts gatavs (G) citas metodes komponents - G(S) apzīmē datu noliktavām specifisku, G(N) apzīmē komponentu, kas nav specifisks datu noliktavu jomai, adaptētam vai gatavam komponentam norādīti arī avoti.

Adaptētiem komponentiem atsevišķā tabulā (skat. 7.tabula) apkopots, kāda ir galvenā būtība komponenta adaptācijai, kādas izmaiņas vai papildinājumi veikti oriģinālajam komponentam. Par adaptējamiem komponentiem jāsecina, ka 2 adaptācijas gadījumos par pamatu kalpo citu lietotāju vadītu metožu komponenti. Tikai vienā adaptācijas gadījumā

izmantota cita tipa pieeja - datu vadītas metodes komponents. Datu vadītas metodes komponenta 4.2. izvēli adaptācijai nosaka apsvērumi, ka hierarhiskas attiecības datiem tikai ierobežotā apjomā nosakāmas no interviju materiāliem, no interviju materiāliem parasti nosakāmas tikai analīzes dimensijas, kā tas arī tika darīts metodes solī 4.1.

7. tabula

Adaptētie metodes komponenti

METODES KOMPONENTS	IZCELSMES AVOTS	J/A/G(N) VAI G(S)	ADAPTĀCIJA
LVM.2.3.	DN - [KRRT98] (skat. 2.3.4.1)	A	Intervējamo grupas tiek apvienotas "vertikāli" noteiktā situācijā
LVM.4.1.	DN - ME/R [SBHD98] (skat. 1.5.1.1.)	A	Izmanto ME/R notāciju, pievienojot ideju, kā analizēt dokumentētās analīzes prasības
LVM.4.2.	DFM [GMR98a] (skat. 2.2.1.)	A	DFM pievieno sākumpunktu pielietošanai, ko nosaka 4.1. solī

No kopsavilkuma par visiem metodes komponentiem, kas dots tabulā (skat. 8. tabula), vēl iegūstama lietotāju vadītai metodei raksturīga situācijas faktoru kopa (6 dažādi faktori):

LVM.F1 - nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic,

LVM.F2- potenciāli daudzi intervējamie, intervijās iesaistīti potenciāli visi darbinieki, kas veic datu analīzi,

LVM.F3- intervijās konstatēto prasību plašais spektrs,

LVM.F4- nepieciešamība grupēt atbildes pēc to līdzības, lai būtu iespējams vieglāk noteikt izstrādes prioritātes,

LVM.F5 -datu analīzes prasības, kas sagrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli,

LVM.F6-daudzas prasības un nepieciešamība noteikt prioritātes.

Šie faktori kopā ar katra konkrētā komponenta lietošanas mērķi kalpo par pamatu komponentu atkalizmantošanas iespējas noteikšanai. Promocijas darbā pēc katras no piedāvātajām metodēm uzskaitīti tai konstatētie faktori, un nodaļas beigās dots vērtējums visiem noteiktajiem faktoriem kopumā.

Metodes blakus efekts ir tāds, ka tiek noskaidrotas arī prasības pēc operatīvās informācijas. Ja šīs vajadzības var atrisināt ar datu noliktavas palīdzību un perspektīvā analizēt šo rādītāju tendences, tad var šīs prasības izmantot, prioritāte augstāka tādām operatīvām vajadzībām, kas ir saistītas ar citās intervijās izteiktām analīzes prasībām. Vislielākā prioritāte dodama tieši datu analīzes prasībām. Metodes trūkums – grūtības no prasībām izveidot daudzdimensiju modeli neprecīzi formulētu prasību dēļ, jo intervējamie cilvēki nezina OLAP rīku iespējas, bez tam pastāv tendence formulēt informācijas vajadzības, kas varētu noderēt konkrētajam cilvēkam, bet nav plašāka skata par uzņēmumu kopumā. Tāpēc, vērtējot un klasificējot interviju atbildes, vadības prasībām tika piemērots koeficients 2, jo vadības konkrētās informācijas vajadzības ir saistītas ar lēmumu pieņemšanu un uzņēmuma kopējiem mērķiem.

Situācijas faktori lietotāju vadītas metodes komponentiem

METODES KOMPONENTS	IZCELSMES AVOTS	J/A/G(N) VAI G(S)	KOMPONENTA MĒRĶIS	FAKTORI, KAS RAKSTURO SITUĀCIJU
LVM.1.	IS	G(N)	Problēmu apgabala definēšana	Nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic
LVM.2.1.	DN - (skat. 2.3.4.1) [KRRT98]	G(S)	Intervējamo personu loka noteikšanai	Nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic Iesaistīti potenciāli visi darbinieki, kas veic datu analīzi
LVM.2.2.	DN - [KRRT98]	G(S)	Sagatavoties intervijai	Potenciāli daudzi intervējamie
LVM.2.3.	DN - [KRRT98]	A	Interviju organizēšana	Potenciāli daudzi intervējamie
LVM.3.1.	DN projekta situācija	J	Strukturēt interviju atbilžu dažādās datu analīzes prasības	Intervijās konstatēto prasību plašais spektrs, Nepieciešamība grupēt atbildes pēc to līdzības, lai būtu iespējams vieglāk noteikt izstrādes prioritātes
LVM.3.2.	DN projekta situācija	J	1) Samazināt interešu sfēru skaitu no 3.1. ja apgrūtināta izstrādes prioritāšu noteikšana; 2) Noskaidrot potenciālo lietotāju skaitu katrai interešu grupai	Intervijās konstatēto prasību plašais spektrs; Nepieciešamība grupēt atbildes pēc to līdzības, lai būtu iespējams vieglāk noteikt izstrādes prioritātes
LVM.4.1.	DN - ME/R [SBHD98] (skat. 1.5.1.1.)	A	No datu analīzes prasībām izdalīt, un ja iespējams, apvienot faktu atribūtus, noteikt dimensijas	Datu analīzes prasības, kas sagrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli
LVM.4.2.	DFM [GMR98a] (skat. 2.2.1.)	A	Dimensiju hierarhiju noteikšana	Datu analīzes prasības, kas sagrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli
LVM.5.1.	DN- [KRRT98] „DW bus architecture”	G(S)	Novērtēt potenciālo darba apjomu dimensiju izstrādei	Daudzas prasības un nepieciešamība noteikt prioritātes izstrādei
LVM.5.2.	DN projekta situācija	J	Apzināt datu avotu datu kvalitāti	Daudzas prasības un nepieciešamība noteikt prioritātes izstrādei
LVM.5.3.	DN projekta situācija	J	Noskaidrot potenciālo lietotāju skaitu dažādām datuvēm	Daudzas prasības un nepieciešamība noteikt prioritātes izstrādei
LVM.5.4.	DN projekta situācija	J	Pieņemt lēmumu par izstrādes prioritātēm	Daudzas prasības un nepieciešamība noteikt prioritātes izstrādei

Lēmumu izmantot lietotāju virzītu metodi par pamatmetodi var ietekmēt vairāki iemesli:

- organizācija nav biznesa struktūra, tāpēc viens no pirmajiem analīzes mērķiem, kas tiek analizēts biznesa struktūrās - peļņa – varētu nebūt primārais intereses objekts, kaut arī jebkura organizācija, protams, nodarbojas ar savu finansu līdzekļu pārvaldību un analizē to izmantošanu;
- vadības ieinteresētība objektīvu kritēriju iegūšanai procesu novērtēšanai un jaunu lēmumu pieņemšanas pamatošanai;
- plaši darbības aspekti, kas vienlīdz svarīgi sekmīgai organizācijas funkcionēšanai, un katrā no sfērām ir aktuāli iegūt informāciju par attīstības gaitu;
- problemātiski ir izvēlēties datu avotu datu modeļus par izejas punktu dimensiju modeļa izveidei (skat. 3.4.apakšnodaļu par datu vadītu pieeju), ja ir daudz prasību. Nezinot prioritātes, ir grūti izvēlēties no potenciāli daudziem modeļiem, ko var iegūt, pielietojot datu vadītu metodi. Datu vadīta pieeja der, ja galvenajam pasūtītājam ir ļoti skaidri definēti mērķi un prioritātes, bet pastāvēs risks, ka projekts balstīsies tikai uz 1 cilvēka prasībām.

Jāsecina, ka metodes lietošanā lielākās grūtības ir noteikt interešu prioritātes un pēc tam izstrādes prioritātes, nav situācijai piemērotu gatavu risinājumu. Ar šo problēmu saistīti arī jaunie metodes komponenti 3.1., 3.2., 5.2., 5.3., 5.4.

Rezultāti:

- izstrādātajā metodē, balstoties uz lietotāju vadītu pieeju, tiek izmantotas gan gatavi metodes komponenti, gan adaptēti citu metožu komponenti atbilstoši situācijai, gan konstruēti jauni komponenti, kas galvenokārt saistīti ar prioritāšu noteikšanu, lai noskaidrotu datuvju izstrādes prioritātes. Ja sākotnēji nav skaidri definētas prasības no biznesa puses, metode piedāvā izmantot:

- vairākas “matricas”, grupējot “interesu sfēras”, ar mērķi noskaidrot potenciālo lietotāju skaitu datuvēm;
- koeficientus, lai raksturotu prasības pamatotību un atbilstību datu noliktavas mērķiem;
- koeficientus, lai raksturotu tiešā veidā izteiktas prasības par "galveno prioritāti";
- lēmumu pieņēmēju prasības un prasības, kas atbilst operatīvām vajadzībām;
- dimensiju skaitu, kas var kalpot kā papildus faktors prioritāšu noteikšanai.

- identificēti 7 situāciju raksturojoši faktori, kas ir būtiski datu noliktavu jomā, izvēloties un pielietojot metožu komponentus konceptuālo modeļu izveidei;

- metodes pielietojuma rezultātā sekmīgi izstrādātas un vairākus gadus tiek izmantotas datuves “Darbinieku komandējumi”, „Studentu statusu maiņa” Latvijas Universitātes datu noliktavā, ar kuras aprakstu iespējams iepazīties promocijas darba 3.pielikumā.

Rezultāti par Lietotāju vadītas metodes pielietošanas pieredzi Latvijas Universitātē kā arī par prioritāšu noteikšanu šīs metodes kontekstā publicēti rakstā:

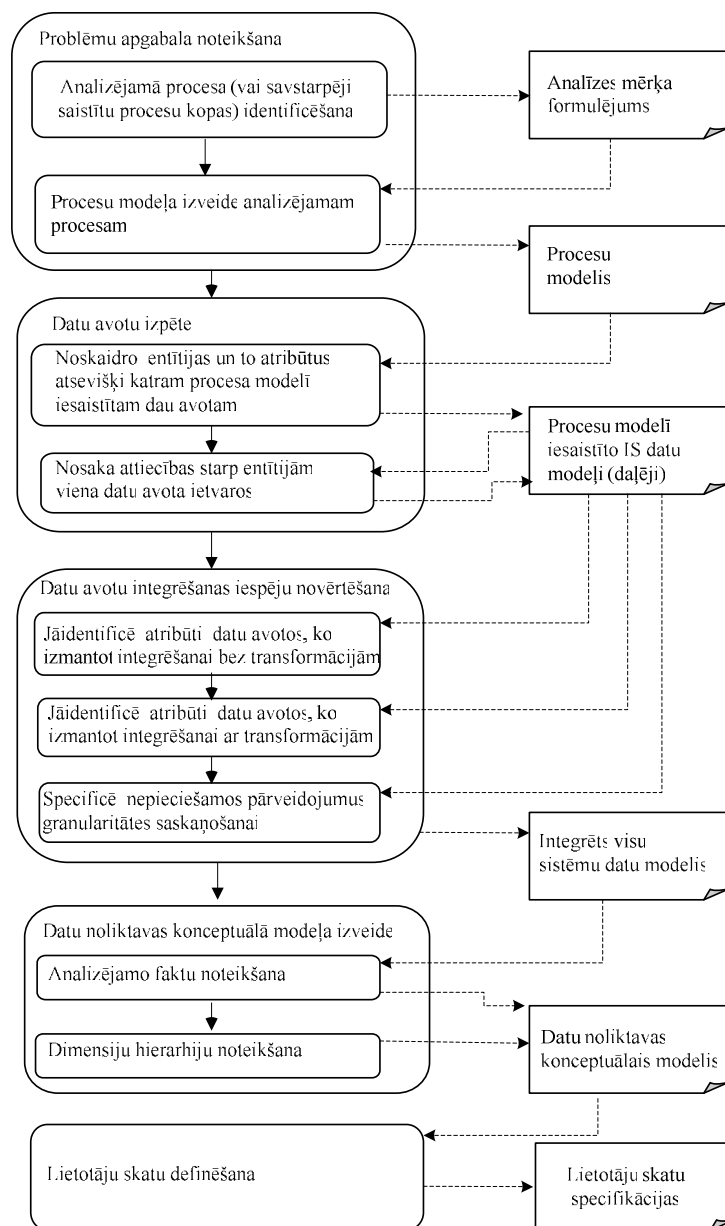
[BN04] J.Benefelds, L.Niedrite, „Comparison of Approaches in Data Warehouse Development in Financial Services and Higher Education”, ICEIS, Vol.1, 2004, p. 552-557.

5.4. Datu vadītā metode

Šajā apakšnodaļā ir dots apraksts jaunai datu vadītai metodei (tālāk tekstā DVM), kas paredzēta datu noliktavas konceptuālā modeļa izstrādei. Metodes izveidē tiek izmantoti metožu inženierijas metožu konstruēšanas un metožu adaptācijas paņēmieni. Tiek izmantoti vai adaptēti eksistējošu metožu komponenti, kā arī izstrādāti jauni metožu komponenti, novērtējot metožu komponentu izvēles un lietošanas kontekstus.

5.4.1. Datu vadītās metodes procesa modelis

Promocijas darbā tiek piedāvāta datu vadīta metode DVM, ko var definēt ar procesa modeli (skat. 36.att.), bet detalizēts metodes apraksts atrodams apakšpunktā 5.4.2.



36. att. Metodes DVM procesu modelis

DVM procesu modeli raksturo sekojošas aktivitātes:

- Problēmu apgabala noteikšana

Šīs aktivitātes nepieciešamību nosaka tas, ka pieeja, kurai atbilstoši tiek definēta šī metode, ir datu vadīta. Viens no rīcības scenārijiem būtu ņemt visu organizācijā pieejamo sistēmu datu modeļus un veidot globālu uzņēmuma datu modeli. Tomēr tas ir laikietilpīgi, līdz ar to dārgi, modelis saturēs daudz lieku datu, kas beigās var nebūt vajadzīgi analīzei. Lai ierobežotu šo globālo modeli un ļautu veidot uzņēmuma globālā datu modeļa apakškopu, ir vajadzīgi principi, kā šo ierobežošanu veikt.

Analizējamais process tiek uzzināts no pasūtītāja, bet netiek identificētas konkrētas analīzes prasības, analizējamie rādītāji, bet gan noskaidrots mērķis, kādam nolūkam, kāda procesa analīzei paredzēta datu noliktava. Analizējamajam augsta līmeņa procesam tiek veidots detalizētāks procesu modelis, kurā tiek atspoguļotas procesu soļu mijiedarbība ar uzņēmuma IS.

- Datu avotu izpēte

Tiek izveidoti datu modeļi atsevišķi katram no iepriekšējā aktivitātē izveidotā procesa modeļa datu avotiem, nosakot procesos izmantotos datu elementus (entītijas un to atribūtus). Attiecību starp entītijām noteikšanai viena modeļa ietvaros izmanto eksistējošus datu avotu modeļus (dokumentāciju) vai arī izmanto citus informācijas avotus, piemēram, DBPS metadatu vārdnīcas.

- Datu avotu integrēšanas iespēju novērtēšana

Identificē atbilstošus atribūtus datu avotos, kuru vērtības izmantojamas dažādu datu avotu integrēšanai tiešā veidā, bez transformācijām. Identificē eksistējošas integrācijas problēmas un nepieciešamās transformācijas, ja tādas iespējamas, lai šīs problēmas atrisinātu. Papildus specificējami arī nepieciešamie pārveidojumi ar visiem datiem kādā no datu avotiem, ne tikai ar „atslēgas laukiem” (lauki, kas piedalās attēlojumā), ja tādi pārveidojumi nepieciešami. Aktivitātes izpildes rezultātā tiek iegūts vienots pētāmajam procesam atbilstošs integrēts datu modelis.

- Datu noliktavas konceptuālā modeļa izveide

Nosaka analizējamās faktus. Nosaka dimensiju hierarhijas pēc kādas no zināmām datu vadītām datu noliktavas konceptuālā modeļa veidošanas metodēm, piemēram, ar DFM metodi. Par izejas punktu atribūtu koka būvēšanai izvēloties iepriekš noteiktos faktus, katram faktam (vai vairākiem faktu atribūtiem, ja tie ir vienas entītijas vai m:n attiecības atribūti) būvē savu atribūtu koku atbilstoši DFM metodei, arī tālāk pielieto DFM metodi.

- Lietotāju skatu definēšana

Tā kā datu vadītās metodoloģijās lietotāju analīzes prasības netiek identificētas pirms datu noliktavas konceptuālā modeļa izstrādes, tad tomēr lietotāja kontekstā pastāv divi aspekti: 1) kādiem datiem no konceptuālā datu noliktavas modeļa lietotājiem būtu jānodrošina piekļuve – pamatojums šai nostājai par pieejas ierobežošanu būs dots apakšnodaļā par lietotāju pieejas tiesību realizāciju; 2) kādas darbības ar šiem datiem ir pieļaujamas (pēc datu semantikas).

Lai formalizētu šos aspektus un nodrošinātu izstrādātājiem specifikāciju gan tiesību mehānisma, gan atskaišu realizācijai, promocijas darbā tiek piedāvāts ieviest lietotāja skatījuma jēdzienu.

5.4.2. *Datu vadītās metodes detalizēts apraksts*

Izstrādātā datu vadītā metode DVM tika aprobēta LU datu noliktavas projekta ietvaros, izstrādājot datuvi e-studiju vides WebCT lietošanas novērtēšanai. Tā kā detalizēti aprakstot metodi, tiks izmantoti piemēri no šī projekta, sākumā īsi tiks raksturots projekts, tālāk sekos metodes detalizēts apraksts.

5.4.2.1. *Metodes aprobācijas projekta raksturojums*

Metodes aprobācijas projekts saistīts ar e-studiju procesa analīzi. E-studijas ir ar tīklu nodrošināta zināšanu un iemaņu piegāde. E-studijas izmanto elektroniskas lietotnes un procesus, lai mācītos [Web]. Eksistē daudzas kursu pārvaldības sistēmas - *Learning Management Systems* (LMS), kas nodrošina e-studijas, kuru salīdzinājumi un īpašību analīze atrodami daudzu autoru darbos [BM01], [PSK01]. Problēma ir šo rīku arhitektūra, kā minēts [Bru04], tā nenodrošina vienlaicīgi gan mācību, gan arī administrēšanas vajadzības.

E-studiju vides, kā WebCT, ieviešana universitātē ietekmē dažādus procesus – gan pārvaldes, piemēram, studentu reģistrācijas kārtību uz kursiem, pasniedzēju slodzes, gan mācību procesus. Lai nodrošinātu kvalitatīvu mācību procesu, jāzin, kā notiek e-studijas – vai tās aizstāj pilnībā kāda kursa apmācību, vai tiek kombinētas ar tradicionālo apmācību. Nav skaidrs, kā šajā gadījumā vērtēt kursā iegūtās zināšanas un mācību procesa kvalitāti.

Parasti ir grūtības atrast piemērotus rādītājus un metodes, kā vērtēt e-studijas un to ietekmi – var analizēt kvantitatīvus rādītājus, var analizēt kvalitātes aspektus [Smi99],[Ber03]. Piemēram, interesi un piedalīšanos var vērtēt ar pieslēgumu skaitu un sūtīto e-pasta vēstuļu skaitu [SV01]. E-studiju vides piedāvā lietotāju darbību reģistrāciju žurnālfailā, ko var izmantot informācijas vākšanai par lietotāja darbībām un to analīzei [RK99].

Būtiski ir noskaidrot, kādus no šiem rādītājiem iespējams izmantot studenta mijiedarbības ar e-studiju vidi vērtēšanā un kā tos interpretēt, un vai šie gatavie statistikas rādītāji ir pietiekoši, vai jāmeklē citas rādītāju ieguves iespējas. LMS uzkrātos datus nevar izmantot analīzē kopā ar studenta datiem no citiem avotiem, parasti tie ir orientēti uz pasniedzēja skatījumu – statistika tiek atspoguļota par vienu atsevišķu kursu. Izpaliek organizācijas vadības skatījums par LMS izmantošanu kopumā.

Lai vērtētu e-studijas, ir bijuši vairāki pētījumi [MD04], [RK99], [SCHT03], [LPD03], [ZL01]. Datu noliktavas tradicionāli tiek izmantotas vadības informācijas analīzē [Inm02], [KRR98], [IGG03], tās izmanto arī tīmekļa datu analīzē [KM00]. Tikai vienā no pētījumiem par e-studiju analīzi tiek piedāvāts izmantot datu noliktavas iespējas [SV01], [SV02]. Datu glabāšanai tiek piedāvāts datu noliktavas modelis, bet netiek novērtēta realizācijas iespējamība, vai eksistē nepieciešamie dati.

LU projekta mērķis bija piedāvāt veidu, kā analizēt e-studiju vides izmantošanas ietekmi uz dažādiem universitātes procesiem. Piedāvāto risinājumu raksturo faktori:

- izvirzītie mērķi e-studiju analīzei:
 - e-studijās lietotās LMS sistēmas lietošanas vērtēšana – vai studenti un pasniedzēji pietiekami lieto e-studiju vidi, ar mērķi noskaidrot tālāku e-kursu izstrādes nepieciešamību, arī lai noskaidrotu problēmas neaktīvas izmantošanas gadījumā;
 - pasniedzēju aktivitātes vērtēšana, lai noskaidrotu pasniedzēju slodzes pieaugumu;
 - izstrādāto e-studiju vides e-kursu komponentu lietošanas vērtēšana – kurus komponentus izmanto vairāk, vai izmanto e-studiju vides specifiskos komponentus vai tikai mācību satura statistikas publicēšanas iespējas;
- orientēšanās uz plaši lietotu standarta LMS, kāds ir WebCT;
- datu noliktava kā vide – izvēli pamatoja lielais datu apjoms un nepieciešamība integrēt vairākus WebCT datu avotus ar studentu IS datiem analīzes mērķu noskaidrošanai, kā arī nepieciešamība analizēt datus visas universitātes mērogā;
- pieeja datu analīzei – skatu definēšana ar rādītājiem dažādās dimensiju hierarhiju detalizācijas pakāpēs, kas atspoguļo dažādas informācijas vajadzības atbilstoši šo datu izmantotāju biznesa funkcijām.

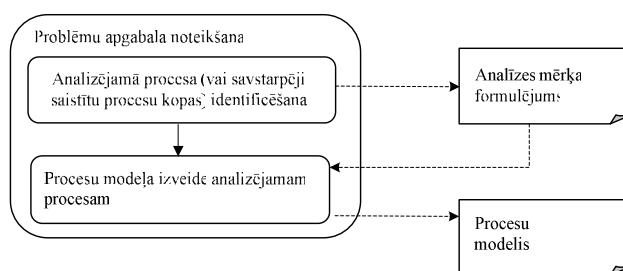
Detalizētāks pētījuma apraksts, kurā detalizētāk aprakstīti citu autoru pētījumi, kā arī ietverts metodes aprobācijas projekta realizācijas apraksts, dots promocijas darba 5. pielikumā.

5.4.2.2. *DVM detalizēts apraksts*

1. aktivitāte. "Problēmu apgabala noteikšana"

1. aktivitātes procesa aspekti

Zīmējumā (skat.37.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Problēmu apgabala noteikšana". Tālāk seko katra aktivitātes soļa apraksts.



37. att. Aktivitāte "Projekta sfēras noteikšana" DVM metodei

Šajā aktivitātē to veikšanas secībā var apskatīt divus soļus.

1.1. solis. Analizējamā procesa (vai savstarpēji saistītu procesu kopas) identificēšana

Intervējot pasūtītāju, tiek noskaidrots analizējamais process vai savstarpēji saistītu procesu kopa. Tiek noskaidrots mērķis, kādam nolūkam paredzēta datu noliktava, bet netiek identificētas konkrētas analīzes prasības, analizējamie rādītāji.

Mērķi M formulē pēc šablona M: „<viens no uzņēmuma darbības virzieniem> analīze”.

Būtībā <darbības virziens> izsaka kādu augšējā līmeņa procesu uzņēmumā.

Analīzes mērķa apraksts metodes aprobācijas projektam dots apakšpunktā 5.4.2.1.

Metodes komponentam, kas atbilst 1.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: nepieciešams mērīt un analizēt kādu noteiktu procesu organizācijā. Process organizācijas darbībai jauns, nav līdz ar to zināms, kādi varētu būt potenciālie rādītāji, ko mērīt;

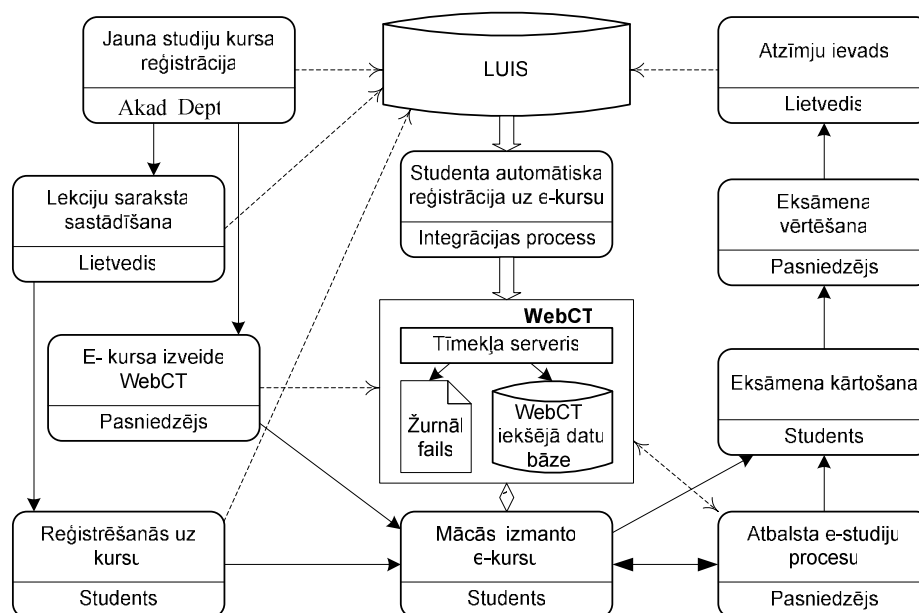
- lēmums:

- metodes komponenta lietošanas mērķis: noskaidrot analizējamo procesu;
- tehnika - intervija;
- vērtējums: izmantots gatavs metodes komponents.

1.2. solis Procesu modeļa izveide analizējamam procesam

Analizējamam augsta līmeņa procesam tiek veidots procesa modelis, kurā tiek atspoguļota procesa soļu mijiedarbība ar uzņēmuma IS. Procesu modeļa izstrādei izmantojami gan uzņēmuma darbību un procesus reglamentējoši dokumenti vai jau gatavi procesu modeļi, ja tādi ir pieejami, vai arī intervijas ar analizējamā procesa turētāju (atbildīgo personu). Intervijas ir tikai par uzņēmuma procesiem, jautājumi par analīzes vajadzībām netiek apskatīti.

Piemēram, atbilstoši analīzes mērķim "e-studiju analīze" aplūko sekojošu e-studiju procesa modeli (skat. 38.att.), kas izveidots, lai noskaidrotu procesa soļu mijiedarbību ar procesā izmantotajām sistēmām. Izmantotas intervijas ar LU informācijas sistēmas (LUIS) izstrādātāju pārstāvi, interfeisa izstrādātājiem starp LUIS un WebCT, sistēmu dokumentācija, kā arī promocijas darba autores personīgā pieredze e-kursu veidošanā un izmantošanā mācību procesā.



38. att. E-studiju procesa modelis

Metodes komponentam, kas atbilst 1.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: ir zināms analizējamais process vai dažu savstarpēji saistītu procesu kopa, nav zināmi rādītāji;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot datu avotus, ar kuriem mijiedarbojas analizējamais process;
 - tehnika - jebkura procesu modelēšanas tehnika;
 - vērtējums: metodes komponents adaptēts atbilstoši situācijai.

Produkta aspekti

1.1. produkts. Analīzes mērķa formulējums

Mērķa formulējums jāveic tādā formā, kā tika aprakstīts aktivitātes procesu aspektos, bet apraksts ir dokuments, kur sīkāk aprakstīts analizējamais process un analīzes mērķis.

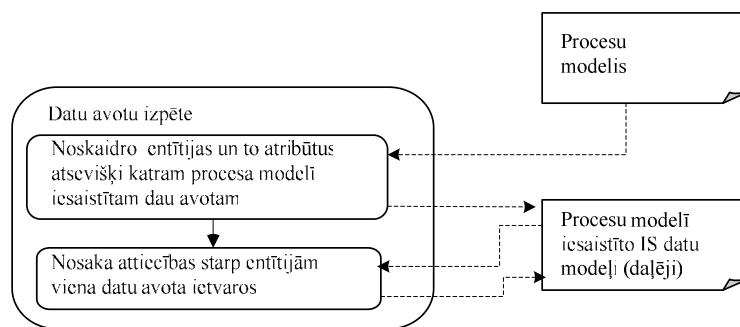
1.2. produkts. Procesu modelis

Procesu modelis tiek izstrādāts analizējamam procesam. Tas ietver procesa soļus un datu avotus. Piemērs dots iepriekš pie aktivitātes procesa aspektiem.

2. aktivitāte. "Datu avotu izpēte"

Procesa aspekti

Attēlā (skat.39.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Datu avotu izpēte". Tālāk seko katra aktivitātes soļa apraksts.



39.att. Aktivitāte "Datu avotu izpēte"

2.1. solis. Noskaidro entītijas un to atribūtus atsevišķi katram procesa modelī iesaistītam datu avotam

Tiek izveidoti datu modeļi katram no iepriekšējā aktivitātē izveidotā procesa modeļa datu avotiem. Katram no "Problēmu apgabala noteikšanas" aktivitātē identificētajiem datu avotiem nosaka procesu izmantotos datu elementus (entītijas un to atribūtus). Šie datu elementi veido apakškopu no visa datu avota modeļa, un tos var identificēt vairākos veidos- intervējot datu avotu

izstrādātājus, izpētot sistēmas projektējuma dokumentāciju, ja tajā šāda informācija ietverta, vai eksperimentu rezultātā, veicot darbības ar sistēmu atbilstoši procesa soļa prasībām un nosakot izmaiņas datus.

Tālāk piemērā aprakstīts, kā tiek analizēti dažādie datu avoti un iegūti datu modeļi metodes aprobācijas projektā. „E-studiju” datuvei kā potenciālie datu avoti, kas iesaistīti e-studiju procesos, tika identificēti sekojošie: 1) LUIS, 2) WebCT tīmekļa servera žurnālfaili un 3) WebCT iekšējā datu bāze, kuru datiem iespējams piekļūt, izmantojot API, saņemot rezultātu XML formātā. No minētajiem datu avotiem LUIS ir relāciju datu bāze un datu ieguvei tiek izmantotas datu noliktavu jomā zināmas metodes, tāpēc sīkāk tās netiks analizētas. Tomēr LUIS dati, kas iesaistīti procesos, tiek atspoguļoti modeļī, kas attēlo visu avotu datus (skat. 41.att., 2. aktivitātes produkta aspekts).

Apskatīsim datu avotu struktūru un izmantošanas iespējas.

WebCT iekšējā datu bāze

WebCT iekšējās datu bāzes datiem var piekļūt standarta veidā ar API, rezultāts būs XML fails. Bet šādā veidā iegūstami ne visi nepieciešamie dati. Otrs veids- analizēt iekšējās datu bāzes uzbūvi, kas veidota atbilstoši e-kursu uzbūvei, datu izguves nolūkam izstrādāts īpašs skripts.

No šī datu avota var iegūt sekojošu veidu datus:

- E-kursu uzbūves dati

Kurss sastāv no kursa komponentiem- atsevišķiem WebCT rīkiem. Skripta rezultātu atspoguļo tabula (skat.9. tabula) ar sekojošām kolonnām: datu savākšanas datums, kursa kods, rādītāji.

9. Tabula

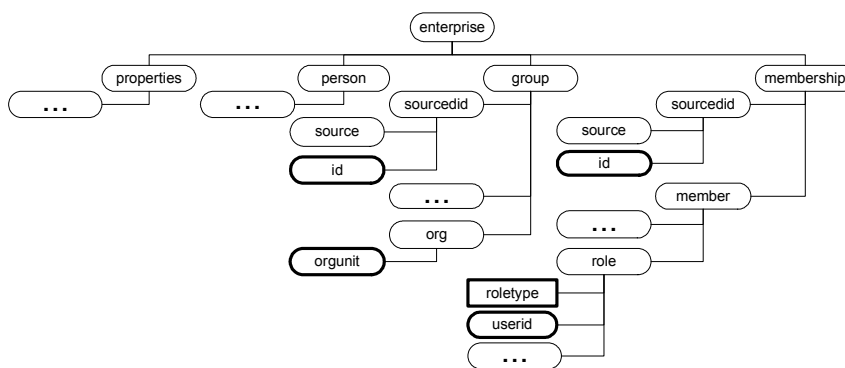
Dati par kursa uzbūvi un reģistrētiem un aktīviem studentiem

Datums	Kursa kods	Rādītāji
20041107	2DAT5080	SR=2;HTM-P=6;DOC-P=9;FIZ=218;HTM-F=3; DOC-F=8;DRB=89;; S=95;S_A=79;+S_B+M_B+B_B+MR_B+I_B+DB_B

Rādītāji ir simbolu virkne, kas apraksta dažādu komponentu skaitu vai apjomu, reģistrēto un aktīvo studentu skaitu.

- Lietotāju lomas un kursu kategorijas

Lai varētu analizēt studentu un pasniedzēju aktivitātesursos, datus par lietotāju lomām (students vai pasniedzējs) var iegūt no WebCT iekšējās datu bāzes XML faila veidā (skat. 40.att.), izmantojot WebCT API. Elementi un atribūti, kas tiek izmantoti datu noliktavai, ir attēloti ar biezāku līniju (skat. 40.att.).



40.att. XML fails – WebCT lietotāji un lomas

XML failā (skat. 40.att.) kokā ar sakni „group” ir vienu kursu raksturojošie dati, kas satur kursa kodu (elements *id*), WebCT kategoriju (elements *orgunit*), kas atbilst fakultātei un citus datus. Katram WebCT kursam atbilst viens koks ar sakni „group”.

Kokā ar sakni „membership” ir dati par viena kursa reģistrētiem lietotājiem. Elements *id* satur kursa kodu. Katram kursa lietotājam atbilst viens apakškoks ar sakni „member”, kas satur sekojošus datus: lietotāja *login* (elements *userid*), loma (atribūts *roletype*) un citus datus.

XML faila dati tiek ielādēti divās datu bāzes tabulās. Viena tabula satur kursa kodus un atbilstošas WebCT kategorijas. Otrā tabula satur datus par lietotājiem un viņu lomāmursos. Otrās tabulas fragments redzams tabulā (skat. 10.tabula).

10. tabula

Apstrādāti dati no XML faila (fragments)

Kursa kods	Login	Loma
2DAT5080	sd00028	Students

Ārēji datu avoti

Ārēji datu avoti tika izmantoti komponentu klasifikācijai (WebCT dokumentācija). Komponenti tiek saukti par statisku, ja tas ir izveidots un netiek mainīts kursa izmantošanas laikā. Dinamisks komponents [SCHT03] var tikt mainīts vai izveidots kursa norises laikā.

WebCT tīmekļa servera žurnālfaili

WebCT tīmekļa servera žurnālfaili ir lietošanas datu avots. WebCT žurnālfaili atbilst *Common Log Formātam* (CLF) [CLF]. No žurnālfaila iegūstami dati par WebCT lietošanu: datums un laiks, URL, lietotāja vārds, IP adrese, pārsūtīto baitu skaits. WebCT žurnālfaila ierakstu piemēri ir attēloti tabulā (skat. 11.tabula).

11. tabula

WebCT žurnālfaila fragments

213.175.115.2 - sd00028 [05/Nov/2004:12:22:38 +0200] "GET /SCRIPT/2DAT5080/scripts/student/serve_home?_homepage+START HTTP/1.1" 200 7134 0
213.175.115.2 - sd00028 [05/Nov/2004:12:22:43 +0200] "GET /SCRIPT/2DAT5080/scripts/student/serve_marks.pl?START+_homepage HTTP/1.1" 200 15795 0
213.175.115.2 - sd00028 [05/Nov/2004:12:22:46 +0200] "GET /SCRIPT/2DAT5080/scripts/student/serve_home?_homepage+START HTTP/1.1" 200 7134 0

WebCT tīmekļa servera žurnālfailu apstrāde notiek sekojoši: dati no žurnālfaila tiek daļēji apstrādāti un ielādēti datu bāzes tabulā. Tabulā (skat. 12.tabula) redzami apstrādāti piemēra (skat.11.tabula) dati.

12. tabula

Apstrādāti žurnālfaila dati

Datums	URL	Lietotājs	IP adrese	Baitu skaits
11/05/2004 12:22:38	GET /SCRIPT/2DAT5080/scripts/student/serve_home	sd00028	213.175.115.2	7134
11/05/2004 12:22:43	GET /SCRIPT/2DAT5080/scripts/student/serve_marks.pl	sd00028	213.175.115.2	15795
11/05/2004 12:22:46	GET /SCRIPT/2DAT5080/scripts/student/serve_home	sd00028	213.175.115.2	7134

Apstrādāti tiek URL – pēc apstrādes tie nesatur parametrus aiz “?” zīmes, jo parametri nav nepieciešami, lai identificētu izmantoto kursa komponenti. Ieraksti, kas attiecas uz WebCT standarta ikonām un attēliem, arī netiek tālāk izmantoti, šo žurnālfaila ierakstu pazīme - tajos neparādās lietotāja vārds. Šie ieraksti neraksturo konkrēta lietotāja darbības WebCT vidē.

Metodes komponentam, kas atbilst 2.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: procesā iesaistīti vairāki datu avoti;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot analizējamā procesa katra soļa izmantotos datus - entītijas un atribūtus;
 - tehnika - intervijas, datu avotu dokumentācijas izpēte;
 - vērtējums: metodes komponents adaptēts atbilstoši situācijai.

2.2. solis. Nosaka attiecības starp entītijām viena datu avota ietvaros

Attiecību starp entītijām noteikšanai viena modeļa ietvaros izmanto eksistējošus datu avotu modeļus (dokumentāciju) vai arī izmanto citus informācijas avotus, piemēram, DBPS metadatu vārdnīcas.

Metodes komponentam, kas atbilst 2.2. solim, konteksts ir sekojošs:

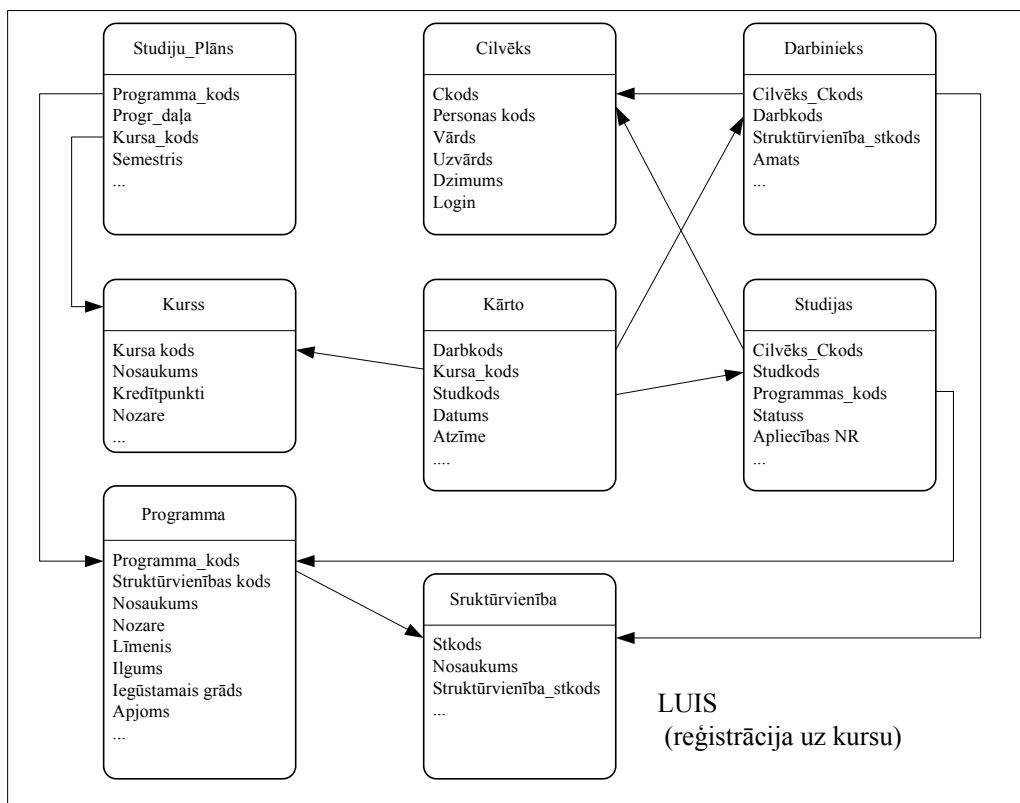
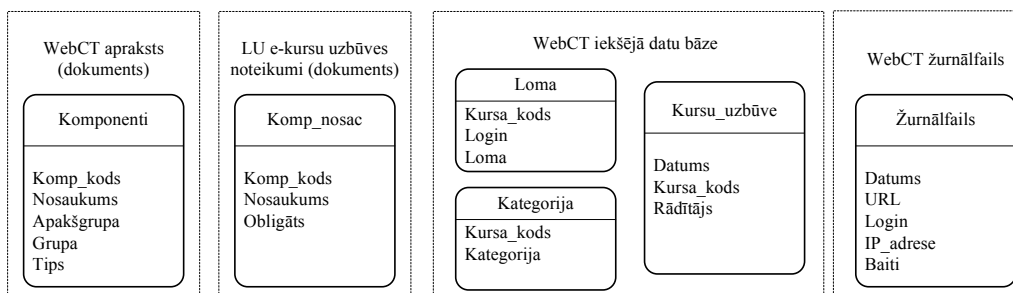
- situācijas raksturojums: procesā iesaistīti daudzi datu avoti;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot attiecības starp iepriekš noskaidrotajām entītijām, kādas eksistē pieejamajos datos datu avotos;
 - tehnika - dokumentācijas vai DBPS metadatu vārdnīcu izpēte;
 - vērtējums: izmanto gatavu metodes komponentu.

Produkta aspekti

2.1. produkts. Procesu modelī iesaistīto IS daļēji datu modeļi

Katrai informācijas sistēmai vai cita veida datu avotam, kas tiek izmantots procesā, tiek zīmēta datu modeļa apakškopa - tiek atspoguļoti tikai tie dati, kas tiek izmantoti procesā.

Iepriekš apskatītajā metodes aprobācijas projektā pēc visu procesā iesaistīto datu avotu izmantošanas analīzes kopējo situāciju atspoguļo modelis (skat. 41.att.).

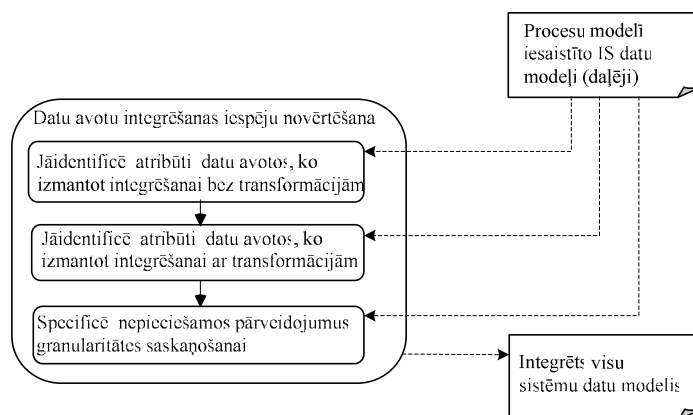


41.att. Datu avotos pieejamie dati

3. aktivitāte. "Datu avotu integrēšanas iespēju novērtēšana"

Procesa aspekti

Attēlā (skat.42.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Datu avotu integrēšanas iespēju novērtēšana". Tālāk seko katra aktivitātes soļa apraksts.



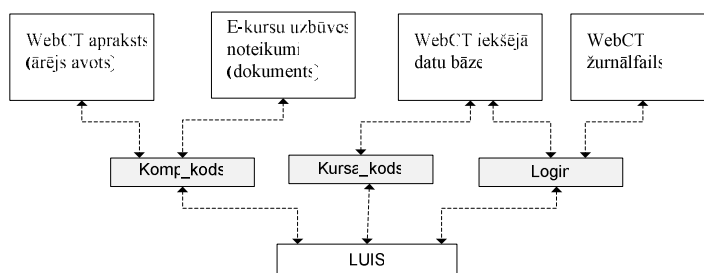
42.att. Aktivitāte "Datu avotu integrēšanas iespēju novērtēšana"

Šajā aktivitātē to veikšanas secībā var apskatīt trīs soļus.

3.1. solis. Jāidentificē atribūti datu avotos, ko izmantot integrēšanai bez transformācijām

Jāidentificē atbilstoši atribūti datu avotos, kuru vērtības izmantojamas dažādu datu avotu integrēšanai tiešā veidā, bez transformācijām – šim nolūkam izmantojami identiski vai semantiski līdzīgi lauku nosaukumi dažādos avotos, vienlaicīgi veicot arī datu izpēti atbilstošajos laukos. Izmantojami arī datu modeļu apraksti datu avotiem, ja tādi eksistē.

Piemēram, metodes aprobācijas projektā, datu avotu integrēšana veikta, izmantojot šim nolūkam piemērotus atribūtus *Login*, *Kursa_kods* un *Komp_kods* (skat. 43.att.). Lai pilnībā izmantotu žurnālfailu, bija nepieciešams atrisināt dažas problēmas, piemēram, kā identificēt komponentus, sesijas, aktivitātes no žurnālfaila datiem. Šīs problēmas tiks apskatītas kā piemēri nākošā soļa aprakstā.



43.att. Datu avotu integrēšanai izmantojamo atribūtu piemēri

Metodes komponentam, kas atbilst 3.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot atribūtus, kas kopīgi vairākiem datu avotiem;
 - tehnika - identisku vai semantiski līdzīgu lauku nosaukumu atrašana dažādos avotos, datu izpēte atbilstošajos laukos, datu modeļu aprakstu izpēte datu avotiem;
 - vērtējums: izmantots gatavs metodes komponents.

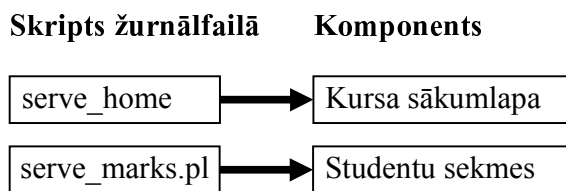
3.2. solis. Jāidentificē atribūti datu avotos, ko izmantot integrēšanai ar transformācijām

Jāidentificē eksistējošas integrācijas problēmas un nepieciešamās transformācijas, ja tādas iespējamas, lai šīs problēmas atrisinātu. Integrācijas problēmu piemērs: semantiski viens un tas pats jēdziens tiek glabāts vienā datu avotā vienā laukā, otrā – divos laukos. Jādefinē attēlojums, kas ļauj izveidot attiecību starp abos modeļos ietvertā identiskā jēdziena dažādajām fiziskajām reprezentācijām. Attēlojums pierakstāms vai nu izmantojot ETL attēlojumus lietojamās metodes (datu izguves, datu ielādes un transformācijas process datu noliktavās), piemēram, [LVT04], vai vienkāršākajā gadījumā, ja transformācijas nebija nepieciešamas, globālajā datu modelī grafiski zīmējot attiecības starp abu modeļu atbilstošām entītijām.

Piemēram, metodes aprobācijas projektā bija nepieciešams veikt sekojošas transformācijas:

- Komponentu un kursu kodu identificēšana no žurnālfaila URL lauka

WebCT žurnālfailā URL ietver kursa kodu, darbības (skripta) un/vai faila nosaukumu, ar kuru notika darbība. Visām iespējamām WebCT darbībām ar kursiem, kas parādās žurnāla failos, nepieciešams piesaistīt noteiktu komponentu. Tabulā (skat. 11.tabula) piemēros redzamajiem skriptiem atšifrējums par komponentiem redzams zīmējumā (skat.44.att.). Skriptu nosaukumu atšifrēšana ir manuālas izpētes rezultāts.



44.att. Skriptu nosaukumu pārtulkošana par komponentiem

Metodes komponentam, kas atbilst 3.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot atribūtus, kas izmantojami sistēmu integrēšanai, bet pielietojot transformācijas;
 - tehnika - identisku vai semantiski līdzīgu lauku nosaukumu atrašana dažādos avotos, datu izpēte atbilstošajos laukos, datu modeļu aprakstu izpēte datu avotiem;
 - vērtējums: izmantots gatavs metodes komponents.

3.3. solis. Specificē nepieciešamos pārveidojumus granularitātes saskaņošanai

Papildus specificējami arī visi nepieciešamie pārveidojumi ar pārējiem datiem kādā no datu avotiem, ne tikai ar laukiem, kas piedalās attēlojumā (skat. iepriekšējos soļus 3.1. un 3.2.). Piemēram, varētu būt atšķirīgas datu detalizācijas pakāpes dažādos datu avotos, tad jāveic darbības, lai tās saskaņotu, datalizētākā avotā veicot datu agregāciju.

Piemēram, metodes aprobācijas projektā, lai WebCT žurnālfails kā datu avots detalizācijas ziņā atbilstu izmantošanas mērķim - noskaidrot lietotāju aktivitātes WebCT, bija nepieciešams veikt sekojošas darbības:

- Sesijas identificēšana no žurnālfaila datiem;

Visiem lietotājiem tiek identificētas sesijas. *Sesija* ir lietotāja aktivitāšu secība no pieslēgšanās (*login*) līdz izešanai no sistēmas (*logout*) vai pārejai uz citu tīmekļa vietni. WebCT vidē maksimālais laiks, kad lietotājs, neveicot nekādas aktivitātes, paliek pieslēdzies sistēmai, ir 60 minūtes. Tāpēc, ja žurnālfailā lietotājam parādās ieraksti ar laika starpību starp pēc kārtas esošiem ierakstiem mazāku par 60 minūtēm, tad visas šīs aktivitātes tiek uzskaitītas par vienu sesiju. Par katru sesiju var iegūt sekojošu informāciju:

- lietotājs – WebCT visiem lietotājiem ir nepieciešama autorizācija. Arī LUIS katram studentam un pasniedzējam glabājas piešķirtais lietotāja vārds. Integrējot informāciju no šiem diviem datu avotiem, ir iespējams iegūt informāciju par lietotāju;
- garums – laiks sekundēs no sesijas pirmās aktivitātes līdz pēdējai aktivitātei;
- IP adrese, kas tiek glabāta, lai pēc tam varētu noskaidrot, no kuras vietas notika aktivitāte, piemēram, no universitātes fakultāšu datorklasēm, kopmītnēm vai ārpus universitātes;
- sesijas veids – tiek noteikts pēc sesijas garuma.

- Aktivitātes identificēšana no vairākiem žurnālfaila ierakstiem.

Aktivitāte tiek definēta kā viens lietotāja klikšķis uz WebCT komponentu. Vispārīgā gadījumā vienam klikšķim WebCT žurnālfailos var atbilst vairāk nekā viens ieraksts. Piemēram, ar vienu klikšķi uz kādu komponentu var tikt izpildīti vienlaicīgi vairāki skripti, kas parādās kā atsevišķi ieraksti žurnālfailā. Tāpēc vienkārši saskaitot ierakstu skaitu žurnālfailā, iegūtais skaitlis nekorekti attēlos reālo aktivitāšu skaitu. Par aktivitāti tika uzskatīts viens vai vairāki ieraksti žurnālfailā, kam ir vienāda sesija, komponents un laiks. Tādā veidā vairāki ieraksti, starp kuriem starpība mazāka nekā sekunde, tiek apvienoti vienā aktivitātē. Katrai aktivitātei var aprēķināt laiku kā starpību starp nākošā ieraksta laiku un dotās aktivitātes ieraksta laiku, kā arī pārsūtīto datu apjomu. Ja aktivitātei atbilst vairāki ieraksti žurnālfailā, tad to datu apjomi tiek summēti.

Apstrādāta žurnālfaila piemērs redzams tabulā (skat. 13.tabula). Tā apstrādē izmantota aktivitātes definīcija. Aktivitātes grupētas sesijās atbilstoši sesijas definīcijai, piešķirot katrai sesijai unikālu kodu. Pēc atribūta *Login* noteikts Lietotājs un Lietotāja loma, bet no URL - kursa kods un aktivitātes laikā izpildītā darbība (skripti), kas vēl papildus identificēts kā noteikts komponents.

13.tabula

Rezultāts, summējot 12. tabulas ierakstus

Sesijas sākuma laiks	Kursa kods	Komponents	Cilvēks	Loma	Sesijas ID	Laiks	Aktivitāšu skaits	Apjoms
11/05/2004 12:22:36	2DAT5080	Kursa sākumlapa	VārdsUzvārds	Students	327878	8	3	57
11/05/2004 12:22:43	2DAT5080	Studentu sekmes	VārdsUzvārds	Students	327878	3	1	15

Metodes komponentam, kas atbilst 3.3. solim, konteksts ir sekojošs:

- situācijas raksturojums: procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt;
- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot, kādas darbības jāveic, lai datu avoti ar dažādu datalizācijas pakāpi būtu integrējami, ja integrācija ir iespējama;
 - tehnika - datu modeļu aprakstu izpēte datu avotiem;
 - vērtējums: metodes komponents adaptēts atbilstoši situācijai.

Produkta aspekti

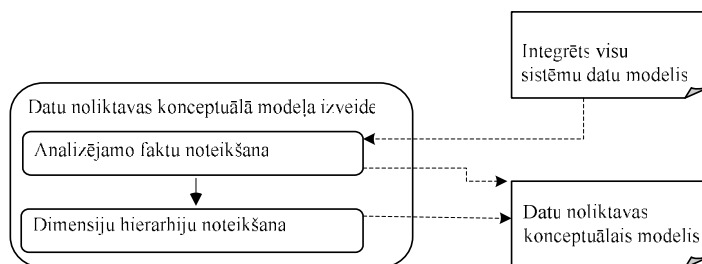
3.1. produkts. Integrēts visu sistēmu datu modelis

Soļa izpildes rezultātā tiek iegūts vienots pētāmajam procesam atbilstošs integrēts datu modelis, kas tiks izmantots kā pamats datu noliktavas konceptuālā modeļa izveidei.

4. aktivitāte. Datu noliktavas konceptuālā modeļa izveide

Procesa aspekti

Attēlā (skat.45.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Datu noliktavas konceptuālā modeļa izveide". Tālāk seko katra aktivitātes soļa apraksts.



45. att. Aktivitāte "Datu noliktavas konceptuālā modeļa izveide"

Šajā aktivitātē to veikšanas secībā var apskatīt divus soļus.

4.1. solis. Analizējamo faktu noteikšana

Par pamatu ņemot principu – meklēt $m:n$ attiecības datu modeļos, sāk zīmēt atribūtu koku pēc DFM metodes.

Metodes komponentam, kas atbilst 4.1. solim, konteksts ir sekojošs:

• situācijas raksturojums: dots analizējamā procesa izmantojamo datu integrēts modelis. Nav zināms, kādi ir nepieciešamie rādītāji datu analīzei;

- lēmums:
 - metodes komponenta lietošanas mērķis: noskaidrot, kādus atribūtus var izmantot par faktiem;
 - tehnika - integrētā datu modeļa izpēte, tiek meklētas $m:n$ attiecības starp entītijām;

- vērtējums: adaptēts DFM metodes komponents, pielietots integrētajam datu modelim.

4.2. solis. Dimensiju hierarhiju noteikšana

Izmantojot globālo datu modeli un izmantojot eksistējošās attiecības starp dažādo datu avotu modeļiem, veido atribūtu koku saskaņā ar DFM metodi, turpinot iepriekšējā solī iesāktos konceptuālos modeļus. Tā kā abi aktivitātes soļi ir cieši saistīti, tālāk tiks apskatīti vairāki piemēri, kur katrā no tiem ietverti abi iepriekš minētie soļi.

Metodes komponentam, kas atbilst 4.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: dots analizējamā procesa izmantojamo datu integrēts modelis. Nav zināms, kādi ir nepieciešamie rādītāji datu analīzei un attiecībā pret kādām dimensijām nepieciešams analizēt;

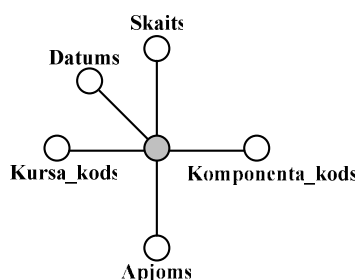
- lēmums:

- metodes komponenta lietošanas mērķis: identificēt dimensijas un dimensiju līmeņu hierarhijas;
- tehnika - integrētā datu modeļa izpēte, tiek pielietota DFM metode;
- vērtējums: adaptēts DFM metodes komponents, pielietojot integrētajam datu modelim.

Piemēri 4.1. un 4.2. soļu pielietošanai

1. piemērs. E-kursu uzbūves shēmas izveide

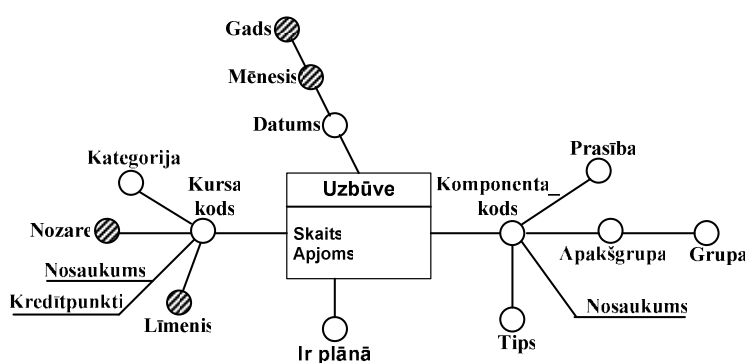
Analizējamā fakta noteikšanai par pamatu ņemot principu – meklēt $m:n$ attiecības datu modeļos - viens no variantiem, ko izmantot par pamatu fakta noteikšanai, ir Kursu_Uzbūve (skat. 41. att.). Būtībā šī tabula reprezentē $m:n$ attiecību starp Kurss un Komponenta ar attiecības atribūtiem Datums, Skaits un Apjoms. Komponenta_kods, Skaits un Apjoms ir iegūstami no Kursu_Uzbūves tabulas lauka "Rādītājs" datiem (skat. 9. tabula).



46.att. Kursu_uzbūves atribūtu koks

Tālāk, izmantojot globālo datu modeli un eksistējošās attiecības starp dažādo datu avotu modeļiem, saskaņā ar DFM metodi veido atribūtu koku. Tiek turpināts iepriekš iesāktais atribūtu koks (skat. 46.att.) ar faktu Uzbūve kā izejas punktu (faktu definē atribūti Kurss_kods, Komponenta_kods un Datums un to raksturojošie faktu atribūti *apjoms* un *skaits*). Rezultāts DFM metodes tālākai pielietošanai globālajā integrētajā modelī redzams zīmējumā (skat. 47.att.).

Zvaigznes shēma papildināta ar aprēķinātiem atribūtiem (turpmāk visos attēlos atribūti ar svītrojumu), kas izrēķināmi no atribūtiem, kas ņemti no globālā modeļa, piemēram, no datuma izrēķināti mēnesis un gads. Iespējams veidot sarežģītākas hierarhijas aprēķinu rezultātā, bet jāņem vērā, ka šie aprēķinātie atribūti tomēr atspoguļo izstrādātāju priekšstatu par datu analīzes vajadzību, tādēļ šādu aprēķinātu lauku iekļaušana modelī ir rūpīgi apsverama. Pieņemot lēmumu par aprēķināmo atribūtu nepieciešamību, tiek izmantotas gan izstrādātāju zināšanas par līdzīgu datu noliktavu modeļiem, piemēram, Laika dimensiju citos modeļos, gan zināšanas par pašiem atribūtiem, piemēram, *kursa_kods* ir salikts teksta lauks, kurā ietverti zinātņu nozares kods un kursa sarežģītības līmeņa kods, tādēļ acīm redzama ir vajadzība atšifrēt šos kodus lietotājam saprotamā formā.



47.att. E-kursu uzbūves zvaigznes shēma DFM notācijā

E-kursa 'Uzbūves' shēmā tiek uzkrāti dati par e-kursu uzbūvi, t.i. e-kursa komponentu *skaitis* un *datu apjoms* komponentam e-kursā uz noteikto laika brīdi. Tiek izmantotas sekojošas dimensijas:

Dimensijai 'Laiks' ir atribūti *Datums*, *Mēnesis*, *Gads*.

'Komponents' satur informāciju par visiem WebCT komponentiem. Šai dimensijai ir sekojoši atribūti: *Nosaukums*, *Apakšgrupa*, *Grupa*, *Prasība* – vai komponents ir obligāts e-kursu uzbūvei, *Tips* – statisks vai dinamisks.

'Kurss' satur informāciju par WebCT e-kursiem, un tos apraksta atribūti *WebCT kursa kods* un *kategorija*, *Līmenis* – bakalaura, maģistra, doktorantūras un profesionālām programmām, *Nosaukums*, *Kreditpunkti*– kredītpunktu skaits par kursu, *Nozare* – zinātnes nozare.

Vēlāk modelis papildināts ar atribūtiem, kas iegūstami no citām datu noliktavas zvaigznes shēmām, lai raksturotu e-kursa stāvokli dotajā laika momentā. Sākotnējā modelī, kas tika iegūts tikai no datiem datu avotā – tieši vai aprēķinu ceļā, šie atribūti neietilpa. Tie ir: pēdējais studentu pieslēguma datums, pēdējais dizainera pieslēguma datums, aktivitātes līmenis un uzbūves līmenis.

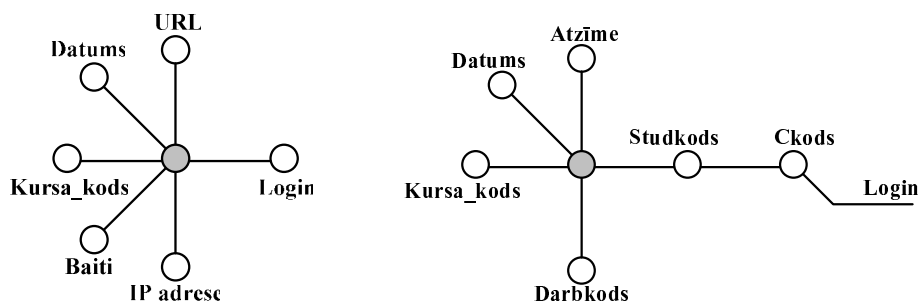
Dimensija 'Plāns' satur tikai vienu atribūtu '*Ir plānā*', kas norāda, ka e-kurss tiek pasniegts noteiktā semestrī. Šim atribūtam ir divas vērtības: *pasniegts/netiek pasniegts*.

Lai datus par e-kursu uzbūvi un izmantošanu varētu analizēt dažādos detalizācijas līmeņos, dimensijā 'Komponents' un dimensijā 'Kurss', kā redzams zīmējumā (skat.47.att.), atribūti veido

vairākas hierarhijas. Dimensijā 'Komponents' viena hierarhija sastāv no komponentu grupas un apakšgrupas. Otra hierarhija ir komponenta tips (dinamisks vai statisks), trešā - komponentu obligātuma īpašība no e-kursu izstrādes prasībām. Dimensijā 'Kurss' ir kategorijas, līmeņa un nozares hierarhijas.

2. piemērs. E-kursu lietošanas un aktivitātes shēmas izveide

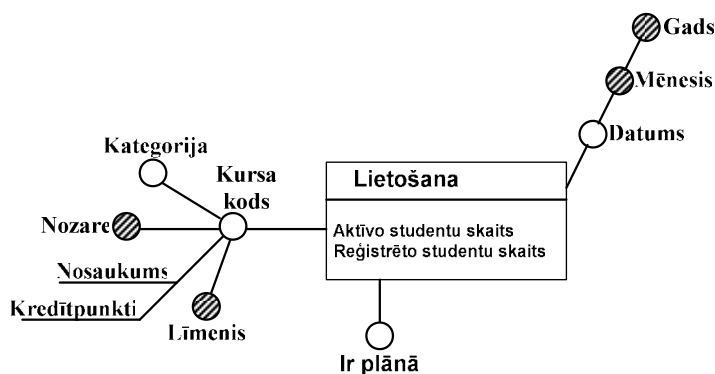
Izpētot globālo modeli un salīdzinot dažādās $m:n$ attiecības, kas eksistē modelī, tika konstatēts, ka divos gadījumos iegūstamas līdzīgas pēc uzbūves, bet dažādas pēc satura atribūtu koku sākuma daļas - $m:n$ attiecību atribūti (skat. 48.att.).



48.att. $m:n$ attiecību atribūti žurnālfailā un LUIS studentu reģistrācijas tabulā

Būtībā tās ir divas dažādas $m:n$ attiecības starp atribūtiem *Kursa_kods*, *Login* un *Datums*. Pirmā ir iegūta no žurnālfaila un raksturo aktīvos studentus, kas kaut reizi ir pieslēgušies WebCT, bet otra – uz atbilstošo kursu reģistrētos studentus (LUIS tabula *Kārto*), kam vajadzētu izmantot WebCT šim kursam paredzētos materiālus. Tātad varētu interesēties par faktu, ka studentam būtu jāizmanto WebCT, un reālo fakta izpildi. Šī fakta analīzi var veikt arī detalizētu ierakstu līmenī, bet šajā gadījumā datu apjomu apsvērumu dēļ tika nolemts vispirms analizēt summārus rādītājus - studentu kopsummā pa kursiem uz noteiktu datumu, tādēļ dimensija *Students* netiek veidota. Atbilstošie faktu atribūti ir 'Reģistrēto studentu kopskaits uz kursu' un 'Aktīvo studentu kopskaits kursā'.

Kopējā "Lietošanas zveigznes shēma" redzama attēlā (skat. 49.att.).

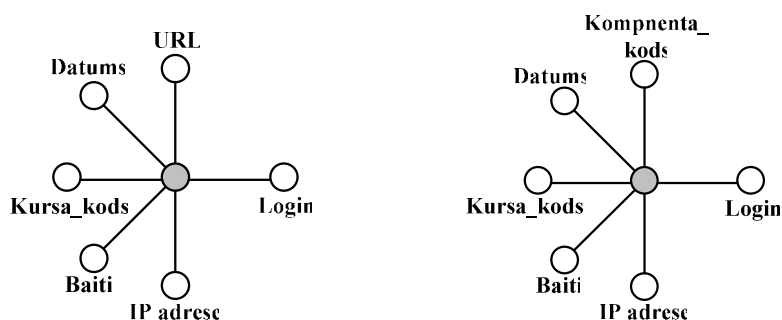


49.att. Lietošanas zveigznes shēma

Faktam *Lietošana* ir mērījumi *reģistrēto un aktīvo studentu skaits* uz doto laika brīdi. Šai shēmai tiek izmantotas dimensijas: *Kurss, Laiks un Plāns*, kas jau iepriekš tika aprakstītas.

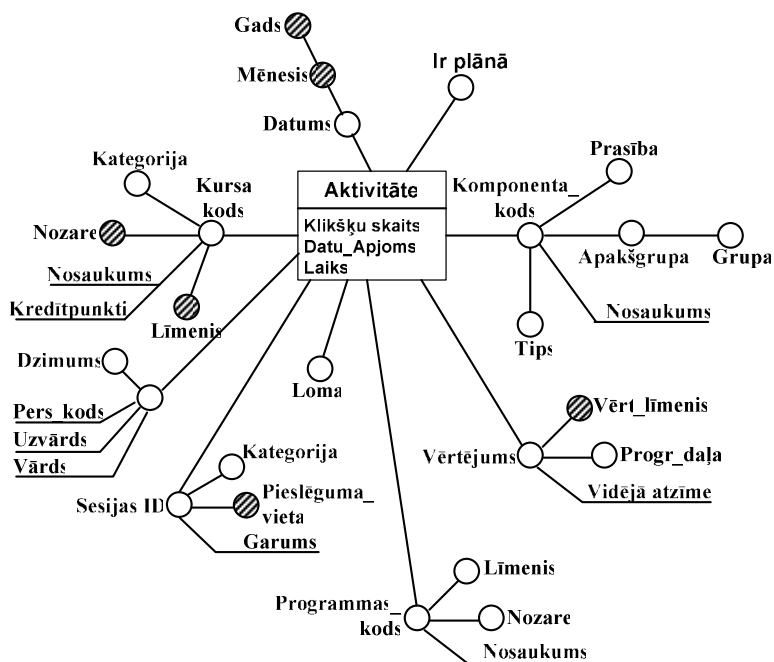
3. piemērs. Aktivitātes shēmas izveide

Aktivitātes zvaigznes shēma tiek veidota ar mērķi analizēt informāciju par studentu un dizaineru aktivitātēm WebCT e-kursu izmantošanas laikā. Par izejas punktu tika izvēlēta m:n attiecība, kas iegūstama, apstrādājot WebCT žurnāltabulas datus, kā tika aprakstīts iepriekš pie nepieciešamajām transformācijām, un URL skriptu atšifrējot ar atbilstošo komponenta kodu. Sākotnējie atribūtu koki atspoguļoti attēlā (skat.50.att.).



50.att. m:n attiecību atribūti žurnālfailā un no URL, skriptu atšifrējot ar komponenti

Papildinot sākotnējo m:n attiecības atribūtu koku saskaņā ar DFM ar atribūtiem no pārējā globālā modeļa, iegūst Aktivitātes shēmu (skat. 51.att.).



51.att. Aktivitātes zvaigznes shēma

Faktam 'Aktivitāte' ir sekojoši mērījumi: aktivitāšu skaits (klikšķu skaits), datu apjoms un laika perioda garums, kurā lietotājs ir strādājis ar kursa komponentu. Dimensijas 'Laiks', 'Kurss' un 'Plāns' jau tika aprakstītas iepriekš. Šajā shēmā parādās jaunas dimensijas:

Dimensijā 'Persona' ir dati par visiem e-kursu lietotājiem (atribūti Vārds, Uzvārds, Personas kods, Dzimums).

Dimensija 'Vērtējums' apraksta studentu sekmes e-kursos (atribūti Vērtējums, Vērt_līmenis – vai atzīme ir apmierinoša, neapmierinoša, Vidējā atzīme – pa visiem studenta kursiem, Progr_daļa – raksturo vairāk kursu, kurā iegūts vērtējums - obligāta, ierobežota izvēle, brīva izvēle).

Dimensijas 'Loma' atribūts Loma norāda lietotāja lomu e-kursā. Sistēmā ir četras lietotāju lomas: students, dizaineris, asistents, viesis.

Dimensijai 'Studiju Programma' ir atribūti Nosaukums, Līmenis (bakalaura, maģistra, profesionālā), Nozare.

Vienu pieslēgšanos WebCT raksturo dimensija 'Sesija' (atribūti Kategorija – īsa vai gara sesija, Garums - sekundēs, Pieslēguma vieta, kas ir noteikta pēc IP adreses). Garumu kategoriju klasificēšana tika izveidota pēc sekojoša principa: īsa (0-1 minūte); vidēja (1-10 minūtes); gara (10-60 minūtes); ļoti gara (>60 minūtes).

Produkta aspekti

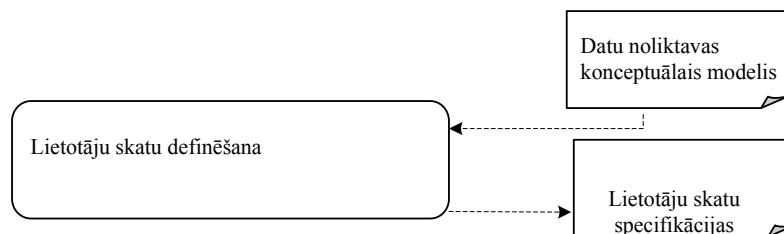
4.1. produkts. Datu noliktavas konceptuālais modelis

Aktivitātes rezultātā tiek iegūts datu noliktavas konceptuālais modelis DFM notācijā, ko nosaka DFM metodes pielietošana aktivitātes procesa aspektiem atbilstošajos soļos. Piemēri konceptuālajiem modeļiem DFM notācijā apskatāmi iepriekš pie soļu 4.1. un 4.2. apraksta.

5. aktivitāte. Lietotāju skatu definēšana

Procesa aspekti

Attēlā (skat.52.att.) redzams fragments no kopējā metodes procesa modeļa, kas atspoguļo aktivitāti "Lietotāju skatu definēšana". Aktivitātei nav detalizācijas pa soļiem.



52.att. Aktivitāte "Lietotāju skatu definēšana"

Lai formalizētu analīzes prasības un nodrošinātu izstrādātājiem specifikāciju gan tiesību mehānisma, gan atskaišu realizācijai, promocijas darbā tiek piedāvāts ieviest lietotāju skatījumu jēdzienu. Sākotnējie lietotāju skatījumi tiek definēti, izejot no izstrādātā dimensiju modeļa, ņemot

vērā, kādi ir iespējamie fakti un dimensiju agregācijas iespējas, kā arī potenciālo lietotāju atbildības jomas. Šī pieeja balstās uz Inmona datu vadītajā metodē izteikto pieņēmumu, ka datu noliktavā OLAP lietotnes tiek izstrādātas iteratīvi, izstrādes dzīves cikla beigās. Skatu definīcijas pēc to formulēšanas tiek precizētas, apspriežoties ar potenciālo lietotāju pārstāvi.

Katra skatījuma definīciju veido $m+2$ elementu kopa $(R, G_m(L_{mj}), F)$, kur :

$0 \leq m \leq n$; n – dimensiju skaits, bet pārējie elementi ir ar sekojošu nozīmi:

R - Rādītājs, ko analizē – Rādītāja nosaukums, kas biznesa terminos definē analizējamo faktu, agregācijas funkciju un dimensijas hierarhiju līmeņu detalizācijas pakāpi;

$G_m(L_{mj})$ – ierobežojums uz dimensijas D_m hierarhijas līmeni L_{mj} ; kur

$1 \leq j \leq k$; k – hierarhijas līmeņu skaits dimensijai D_m ; $k \leq D_m$ atribūtu skaits;

L_{m1} – hierarhijas līmenis, kas tiek izmantots fakta definīcijā, bet L_{mk} – augšējais hierarhijas līmenis.

$G_m(L_{mj})$ var tikt pierakstīts trīs veidos:

- `Dimensijas_nosaukumsm(Lmj)` // tas ir analīzes tipa ierobežojums
`Dimensijas_nosaukumsm` - dimensijas D_m nosaukums; detalizācija tiek paredzēta līdz dimensijas `Dimensijas_nosaukumsm` dimensijas hierarhijas līmenim L_{mj} ,

- `Dimensijas_nosaukumsm(Lmj = „līmeņa_vērtība“)` // tas ir analīzes tipa ierobežojums

Šajā gadījumā tiek ņemti tikai dimensijas `Dimensijas_nosaukums` ieraksti ar atbilstošu atribūta L_{mj} vērtību, šajā gadījumā “līmeņa_vērtība”.

- `Dimensijas_nosaukumsm(Lmj = Value)` // tas ir datu piederības (tiesību) tipa ierobežojums

Šajā gadījumā rādītājs katrai individuālai “Value” vērtībai ir aprēķināms atsevišķi katram konkrētajam lietotājam, katram no lietotājiem var būt pieejama sava datu kopa (atskaite) atkarībā no Value vērtības.

$F (f_x)$ - Funkcija, kas veic pamatfaktu agregāciju

f_x – faktu atribūts, $0 \leq x \leq z$, z – faktu atribūtu skaits.

Definējot ierobežojumus, ieviests iedalījums - datu analīzes tipa ierobežojumi un datu piederības (tiesību) tipa ierobežojums:

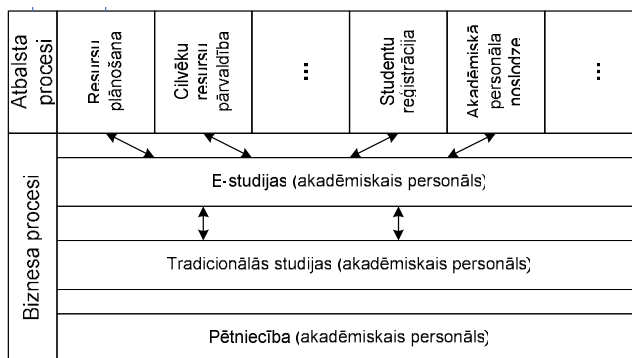
- Datu analīzes tipa ierobežojums ir ierobežojums, ko definē datu noliktavas izstrādātājs, balstoties uz analīzes mērķi un analīzes mērķa neformālo aprakstu. Parasti šis ierobežojums attiecas uz visiem lietotājiem un ir visiem vienāds.

Piemēram, `Kurss(Fakultāte)` nozīmē, ka lietotājam fakti, kam piesaistīta dimensija `Kurss`, ir jāskata `Fakultātes` agregācijas līmenī, nevis detalizēti pa individuāliem kursiem. Vai otra veida piemērs – `Loma(loma=Students)` nozīmē atlasīt datus no faktu tabulas, kurai ir piesaistīta dimensija `Loma`, bet tikai par studentiem.

- Datu piederības tipa ierobežojums ir ierobežojums, kas nozīmē, ka lietotājam atkarībā no viņa ieņemamā amata, struktūrvienības, tiek definēts, kādiem datiem šis lietotājs drīkst piekļūt.

Piemēram, Kurss(Fakultāte=*Value*) nozīmē, ka lietotājs piekļūs savas fakultātes datiem Dimensijā un tiem piesaistītiem datiem faktā tabulā.

Tālāk tiks apskatīti piemēri skatu definīcijām no metodes aprobācijas projekta. Tā kā viens no datu noliktavas izveides mērķiem bija apzināt e-studiju vides ietekmi uz universitātes procesiem, tad tika izdalīta vadību interesējošo funkciju apakškopa, kuru savstarpējā mijiedarbība aplūkojama zīmējumā (skat. 53.att.).



53.att. Universitātes procesi un to mijiedarbība

Šo procesu izpildītājiem datu analīzei atbilstoši tika definēti četri skatījumi – universitātes vadības, fakultāšu dekānu, departamentu vadītāju un pasniedzēju skatījumi.

Katra skatījuma definīciju veido (precīzs definīcijas apraksts dots iepriekš):

- rādītāji, ko analizē;
- analīzes dimensiju hierarhiju līmenis, līdz kuram interesē detalizācija konkrētajam rādītājam;
- funkcija, kas veic pamatfaktu agregāciju.

Vairākiem lietotāja skatījumiem var būt kopīgi rādītāji, kas katrā no tiem atšķiras ar hierarhijas līmeni detalizācijai.

Tālāk aprakstā - tabulās, kur definēti skatījumi, izmantoti apzīmējumi, ko ilustrēšu ar piemēru viena rādītāja aprakstam:

- Rādītājs – „Aktīvo studentu vidējā aktivitāte pēc klikšķu skaita”,
- Laiks(mēnesis) - dimensijas Laiks detalizācija tiek paredzēta līdz līmenim “mēnesis”,
- Loma(loma=students) - tiek ņemti tikai dimensijas Loma ieraksti ar atbilstošo atribūta vērtību, šajā gadījumā “students” ,
- Kurss(nosaukums=*Value*) - rādītājs katrai individuālai “*Value*” vērtībai ir aprēķināms atsevišķi,
- Funkcija: $SUM(klikšķu_skaits)/SUM(aktīvo_studentu_skaits)$.

Universitātes vadības skatījums

Universitātes vadībai interesē e-studijas vērtēt no kursu izmantojamības aspekta – vai studenti vispār izmanto. Rādītāji, kas raksturo e-kursu izmantošanu, vadības skatījumā doti tabulā (skat. 14.tabula). Šos rādītājus iespējams salīdzināt ar finansiāliem rādītājiem WebCT iegādei un uzturēšanai, kā arī ar e-kursu izstrādei ieguldīto naudu. Finanšu rādītāji e-kursu novērtēšanas datu noliktavā nav ietverti. Analīze ietver visas universitātes datus, detalizācija ir tikai līdz fakultāšu līmenim, laika dimensijā – viss atskaites periods vai pa mēnešiem. Vadību interesē arī dati par pasniedzēju aktivitāti – pieslēgumu kopskaits. Vērtēšana paredzēta atskaites perioda beigās.

14.tabula

Vadības skatījuma definīcija

Kods	Rādītājs	Shēma	Analīzes dimensijas un hierarhiju detalizācijas pakāpe	Funkcijas
MV1	Reģistrēto un aktīvo studentu vidējā aktivitāte pēc klikšķu skaita	Lietošana un Aktivitāte	Kurss(kategorija) Laiks(mēnesis) Loma(loma=Students)	SUM(klikšķu_skaits)/SUM(akt_st.skaits) SUM(klikšķu_skaits)/SUM(reg_st.skaits)
MV2	Reģistrēto un aktīvo studentu vidējā aktivitāte pēc laika minūtēs	Lietošana un Aktivitāte	Kurss(kategorija) Laiks(mēnesis) Loma(loma=Students)	SUM(laiks)/SUM(akt_st.skaits); SUM(laiks)/SUM(reg_st.skaits)
MV3	Sesiju skaits	Aktivitāte	Laiks(mēnesis) Sesija(kategorija)	COUNT_DISTINCT (Sesija_id)
MV4	Vidējais sesijas garums	Aktivitāte	Laiks(mēnesis)	
MV5	e-Kursi mācību plānā semestrī skaits	Lietošana	Plānā (ir_plānā=Jā)	COUNT_DISTINCT(Kurss_id)
MV6	Reģistrēto studentu skaits semestrī	Lietošana	Plānā (ir_plānā=Jā)	SUM(Reģistrēto_studentu_skaits)
MV7	Aktīvo studentu skaits semestrī	Lietošana	Plānā (ir_plānā=Jā)	SUM(Aktīvo_studentu_skaits)
MV8	Aktīvo pasniedzēju skaits	Aktivitāte	Loma(loma=dizaineris vai loma=asistents)	COUNT_DISTINCT(persona_id)

Analīzes rezultātu piemēri ir noformēti tabulu veidā. Datu analīzes rezultātus vadības skatījuma rādītājiem MV6, MV7, MV5 atspoguļo dati tabulā (skat. 15.tabula). Aktivitātes koeficienti abos gadījumos tiek aprēķināti no iegūtajām rādītāju vērtībām. Tabula sakārtota pēc kursu aktivitātes koeficienta pa fakultātēm.

15. tabula

Vadības skata rādītāju rezultāti

WebCT kategorija	Studentu reģistrāciju uz kursu skaits	Aktīvo studentu skaits	Studentu aktivitātes koeficients	Kursu skaits
LU-PPF	1057	337	31.88%	21
LU-FMF	5343	3952	73.97%	128
...				
LU-TF	101	72	71.29%	13

Fakultāšu dekānu skatījums

Fakultātes vadības datu analīzes mērķis ir novērtēt gan e-kursu izmantošanas rādītājus (līdzīgus kā universitātes vadībai) savas fakultātes ietvaros – cik “daudz” māca, kā arī savas fakultātes ietvaros novērtēt mācību procesu raksturojošus rādītājus – “kāda veidā māca”, piemēram, kādus e-kursu komponentus, dinamiskus vai statiskus izmanto, arī detalizēti līdz e-kursa komponentiem. Interesē arī aktīvākie pasniedzēji e-kursu izmantošanā fakultātē, pēc izmantošanas laika un pieslēgumu skaita, interesē, kā WebCT izmantošana ietekmē pasniedzēju slodzi.

No vadības skata izmantojamiem rādītājiem mainās analīzes dimensiju hierarhiju izmantošana, piemēram, rādītājam MV3 (sesiju skaits) hierarhiju detalizācija notiek sekojoši:

Vadības skatījumā - Laiks(mēnesis); Sesija(kategorija)

Fakultāšu skatījumā : Laiks(mēnesis), Kurss(Kurss), Sesija(kategorija), bez tam analizē tikai atbilstošās fakultātes ietvaros (Kurss (Kategorija = Value)).

Līdzīgi, mainot hierarhijas, var izmantot MV1 un MV2 rādītājus. Fakultātes dekānu skatījumu definē tabulā (skat.16.tabula) redzami jauni rādītāji:

16. tabula

Fakultātes dekāna skatījums

Kods	Rādītājs	Shēma	Analīzes dimensijas un hierarhiju detalizācijas pakāpe	Funkcijas
FV1	Fakultātes konkrētu dizaineru aktivitāte pēc klikšķu skaita un laika minūtēs (kopā pa kursu)	Aktivitāte	Kurss(Kurss) Loma(Loma=Dizaineris) Kurss(kategorija=Value) Persona(Personas_kods)	SUM(klikšķu_skaits) ; SUM(laiks);
FV2	Fakultātes konkrētu dizaineru aktivitāte pēc klikšķu skaita un laika minūtēs – pa komponentiem	Aktivitāte	Kurss(Kurss) Loma(Loma=Dizaineris) Kurss(kategorija=Value) Komponents (Nosaukums) Persona (Pers_kods)	SUM(klikšķu_skaits) ; SUM(laiks)
FV3	Fakultātes aktīvākie dizaineri - kopējais klikšķu skaits pa komponentiem un laiks	Aktivitāte	Kurss(Kurss) Loma(Loma=Dizaineris) Kurss(kategorija=Value) Komponents (Nosaukums) Persona(Pers_kods)	TOP_TEN (klikšķu_skaits); TOP_TEN(laiks)

Analīzes rezultātus atspoguļo Tabulā (skat.17.tabula) apskatāmie dati rādītājam FV3 (fakultātes 10 aktīvākie dizaineri) laikā no 01.09.2004 līdz 30.01.2005, Fakultāte 'LU-FMF'.

17.tabula

Aktīvākie 10 dizaineri pēc aktivitāšu skaita (fragments)

Komponents		Aktivitāšu skaits		
		Dinamisks	Statisks	Kopā
Uzvārds	Vārds			
U1	V1	54633	5525	60158
U2	V2	5673	3339	9012
...				
U10	V10	2925	2668	5593

Departamentu vadītāju skatījums

Šis skatījums paredzēts departamentu vadītājiem, kuru funkcijās ietilpst mācību procesa organizēšana un mācību procesa kvalitātes nodrošināšana, līdzīgi kā fakultāšu skatījumā interesē pasniedzēju aktivitātes, bet visas universitātes līmenī, kā arī interesē veids, kā WebCT tiek izmantots, kādi komponenti - pārsvarā dinamiskie vai statiskie, kādas ir gala atzīmes abos gadījumos. Šis skatījums nodrošina tādu vērtēšanas iespēju, kur jau laicīgi iespējams iepazīties ar informāciju par izmantošanas tendencēm.

No iepriekšējos skatos definētajiem rādītājiem, mainot analīzes dimensiju hierarhiju, izmantojam, piemēram FV3 (10 aktīvākie dizaineri) ar jaunām hierarhijas detalizācijām: Kurss (Kurss), Loma (Loma=Dizaineris), Komponenti (Kategorija), izmanto arī citus iepriekš definētus rādītājus.

Departamentu vadītāju skatījumu definē jauni rādītāji tabulā (skat.18.tabula).

18.tabula

Departamenta skatījums

Kods	Rādītājs	Shēma	Analīzes dimensijas un hierarhiju detalizācijas pakāpe	Funkcijas
DV1	Dizaineru aktivitāte (kopējais laiks minūtēs, klikšķu skaits) pa komponentiem	Aktivitāte	Loma (Loma=Dizaineris) Komponents(Nosaukums)	SUM (klikšķu_skaits, laiks)
DV2	Dizaineru aktivitāte (kopējais laiks minūtēs, klikšķu skaits) pa komponentu tipiem	Aktivitāte	Loma (Loma=Dizaineris) Komponents(Tips)	SUM (klikšķu_skaits, laiks)

Pasniedzēja skatījums

Pasniedzēja skatījums uz e-kursa vērtēšanu ir saistīts ar sava pasniegtā kursa izmantošanas analīzi – piemēram, konkrēto studentu aktivitāte (klikšķu skaits un laiks minūtēs) pa komponentiem; kādas kursa komponentes un cik daudz izmanto - kopskaits pa visiem studentiem (laiks un klikšķu skaits); sesiju skaits pa mēnešiem kursam, sesiju garumu analīze. Pasniedzēja skatījumu definē jauni rādītāji tabulā (skat.19.tabula).

19.tabula

Pasniedzēja skatījums

Kods	Rādītājs	Shēma	Analīzes dimensijas un hierarhiju detalizācijas pakāpe	Funkcijas
TV1	Kopējais laiks minūtēs konkrētiem studentiem	Aktivitāte	Loma(Loma=Students) Komponents(Nosaukums) Kurss(Nosaukums=Value) Persona(Pers_kods)	SUM(laiks)
TV2	Kopējais klikšķu skaits konkrētiem studentiem	Aktivitāte	Loma(Loma=Students) Komponents(Nosaukums) Kurss(Nosaukums=Value) Persona(Pers_kods)	SUM(klikšķu_skaits)
TV3	Kursa uzbūve – komponentu skaits un apjoms	Uzbūve	Komponents(Nosaukums) Kurss(Nosaukums=Value)	Skaits Apjoms

Tabula (skat.20.tabula) attēlo rādītāja TV3 rezultātus, kā laika gaitā mainās kursu uzbūve pa komponentiem.

20.tabula

Kursu komponentu izmaiņas

Kurss	Akadēmiskā angļu rakstu valoda II		Akadēmiskā angļu rakstu valoda IV	
	Datums	30-Jan-05	Datums	30-Jan-05
Komponents	Skaitis	Izmaiņas	Skaitis	Izmaiņas
Dokumenti satura rādītājos	17	-	23	-4
Dokumenti izkārtojuma lapās	23	-2	16	6
Jautājumi paštestos	30	-	60	2
...				
Jautājumi testos	37	-	25	-

Metodes komponentam, kas atbilst 5. aktivitātei, konteksts ir sekojošs:

- situācijas raksturojums: dots konceptuālais daudzdimensiju modelis. Nav zināms, kā notiks datu analīze, jo sākumā tika noteikts tikai datu analīzes mērķis, bet precīzas datu analīzes vajadzības nav noskaidrotas;
- lēmums:
 - metodes komponenta lietošanas mērķis: aprakstīt dažādām lietotāju grupām atbilstoši amatam, struktūrvienībai, kurā strādā un veicamajām darba funkcijām lietotāju skatus;
 - tehnika - datu noliktavas konceptuālā modeļa analīze, dažādām lietotāju grupām atbilstošu lietotāju skatu definēšana. To veic izstrādātājs;
 - vērtējums: izstrādāts jauns metodes komponents.

Produkta aspekts

5. produkts. Lietotāju skatu specifikācijas.

Specifikācijas tiek definētas tabulā, kurai ir sekojošas kolonnas: Kods; Rādītājs; Shēma; Analīzes dimensijas un hierarhiju detalizācijas pakāpe; Funkcijas. Piemēri lietotāju skatu specifikācijām doti aktivitātes procesa aspekta aprakstā, piemēram, pasniedzēja skats dots tabulā (skat.19.tabula).

5.4.3. Secinājumi un rezultāti par datu vadīto metodi

Izstrādāta metode, balstoties uz datu vadītu pieeju. Metodē tiek izmantoti gan 4 gatavi metodes komponenti, gan adaptēti 5 citu metožu komponenti atbilstoši situācijai, gan konstruēti 1 jauns komponents. Kopsavilkums par visiem metodes komponentiem dots tabulā (skat. 22.tabula), tajā raksturots komponenta lietošanas mērķis, situācijas faktori, norādīts, vai komponents jauns (J), adaptēts (A) vai gatavs (G) citas metodes komponents. G(S) apzīmē datu noliktavām specifisku, G(N) apzīmē komponentu, kas nav specifisks datu noliktavu jomai, adaptētam vai gatavam komponentam norādīti arī avoti.

Adaptētiem komponentiem atsevišķā tabulā (skat. 21.tabula) apkopots, kāda ir galvenā būtība komponenta adaptācijai. Par adaptējamiem komponentiem datu vadītai metodei jāsecina, ka 3 adaptācijas gadījumos par pamatu kalpo metožu komponenti, kas nav specifiski datu noliktavām. Par metožu komponentiem izmantotas modelēšanas un integrācijas metodes no IS izstrādes jomas. Datu noliktavām specifiskas metodes komponenti tiek izmantoti, lai noteiktu datu noliktavu elementus - faktus un hierarhijas.

Jaunais metodes komponents izstrādāts kā līdzeklis, lai formalizētu analīzes iespējas un nodrošinātu izstrādātājiem specifiskā gan tiesību mehānisma, gan atskaišu realizācijai, jo viens no situāciju raksturojošiem faktoriem, kad tiek pielietotas datu vadītas metodes ir precīzu datu analīzes prasību trūkums. Lietotāju skati tiek definēti izejot no dimensiju modeļa, lietotāja amata, veicamajām funkcijām. Skati tiek precizēti pēc to formulēšanas un apspriešanas ar potenciālo lietotāju pārstāvjiem. Par pamatu ņemot lietotāju prasības nevis datu modeli, lietotāju skatus kā līdzekli OLAP lietotnes specificēšanai iespējams izmantot arī lietotāju vadītā metodē.

21.tabula

Adaptētie metodes komponenti

METODES KOMPONENTS	IZCELSMES AVOTS	J/A/G(N) VAI G(S)	ADAPTĀCIJA
DVM.1.2.	IS (procesu modelēšanas tehnikas)	A	Procesu modelis tiek veidots ierobežots, pārējie procesi tiek modelēti tikai saistībā ar analizējamo procesu
DVM.2.1.	IS (datu modelēšanas tehnikas);	A	Datu modeļi tiek veidoti ierobežoti - tikai tie dati ko izmanto 1.2. noteiktie procesi.
DVM.3.3.	DN modeļu integrācijas metodes	A	Datalizētākā avotā specificē datu agregāciju, lai izstrādes laikā varētu veikt datu integrāciju
DVM.4.1.	DFM [GMR98a] (skat. 2.2.1.)	A	DFM metode tiek pielietota integrētajam datu modelim
DVM.4.2.	DFM [GMR98a] (skat. 2.2.1.)	A	DFM metode tiek pielietota integrētajam datu modelim

No kopsavilkuma par visiem metodes komponentiem, kas dots tabulā (skat. 22. tabula), ir iegūstama datu vadītai metodei raksturīga situācijas faktoru kopa (7 dažādi faktori):

- DVM.F1- process organizācijas darbībai jauns;
- DVM.F2- procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt;
- DVM.F3- zināms analizējamais process vai dažu savstarpēji saistītu procesu kopa;
- DVM.F4- dots (iegūstams) analizējamā procesa izmantojamo datu integrētais modelis;
- DVM.F5- nav zināmi potenciālie rādītāji, ko mērīt;
- DVM.F6- nav zināms attiecībā pret kādām dimensijām nepieciešams analizēt rādītājus;
- DVM.F7- nav zināms datu analīzes veids, jo sākumā tika noteikts tikai datu analīzes mērķis, bet precīzas datu analīzes vajadzības nav noskaidrotas.

Šie faktori kopā ar katra konkrētā komponenta lietošanas mērķi kalpo par pamatu komponentu atkalizmantošanas iespējas noteikšanai. Promocijas darbā pēc katras no

piedāvātajām metodēm uzskaitīti tai konstatētie faktori, un nodaļas beigās dots kopsavilkums visiem noteiktajiem faktoriem.

Rezultāti:

Izstrādāta Datu vadīta metode, kā konstruēt datu noliktavas zvaigznes shēmu, balstoties uz analizējamās sfēras procesos iesaistīto sistēmu datu avotu modeļiem.

Metodē eksistējoša datu vadīta metode DFM paplašināta varāku datu avotu integrēta modeļa gadījumam.

Metode papildināta ar lietotāju skatiem, kas procesus raksturojošus rādītājus grupē atbilstoši lietotāju kategorijām, viņu datu pieejas tiesībām un datu analīzes mērķim, kuram datalizēti rādītāji iepriekš netiek specificēti.

Metode aprobēta LU datu noliktavā, izstrādājot datuves e-studiju analīzei.

Nodaļā aprakstītā datu vadītā metode un tās pielietojums LU datu noliktavas projektā aprakstīts un publicēts rakstā:

D. Solodovnikova, L. Niedrite, "Using Data Warehouse Resources for Assessment of E-Learning Influence on University Processes", *In Proceedings of ADBIS'2005*, Tallinn, 2005.

Situācijas faktori datu vadītas metodes komponentiem

METODES KOMPONENTS	IZCELSMES AVOTS	J/A/G(N) vai G(S)	MĒRĶIS	FAKTORI, KAS RAKSTURO SITUĀCIJU
DVM.1	IS	G(N)	Noskaidrot analizējamo procesu	Process organizācijas darbībai jauns, līdz ar to nav zināms, kādi varētu būt potenciālie rādītāji, ko mērīt
DVM.1.2.	IS (procesu modelēšanas tehnikas)	A	Noskaidrot datu avotus, ar kuriem mijiedarbojas analizējamais process	Zināms analizējamais process vai dažu savstarpēji saistītu procesu kopa, nav zināmi rādītāji
DVM.2.1.	IS (datu modelēšanas tehnikas)	A	Noskaidrot analizējamā procesa katra soļa izmantotos datus - entītijas un atribūtus	Procesā iesaistīti vairāki datu avoti
DVM.2.2.	IS (datu modeļu analīze; metadatu vārdnīcu analīze)	G(N)	Noskaidrot attiecības starp noskaidrotajām entītijām, kādas eksistē pieejamajos datos datu avotos (viena modeļa ietvaros)	Procesā iesaistīti daudzi datu avoti
DVM.3.1.	IS un DN modeļu integrācijas metodes	G(S)	Noskaidrot atribūtus, kas kopīgi vairākiem datu avotiem	Procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt
DVM.3.2.	IS un DN modeļu integrācijas metodes	G(S)	Noskaidrot atribūtus, kas izmantojami sistēmu integrēšanai, bet pielietojot transformācijas	Procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt
DVM.3.3.	DN modeļu integrācijas metodes	A	Noskaidrot, kādas darbības jāveic, lai datu avoti ar dažādu datalizācijas pakāpi būtu integrējami, ja iespējams	Procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt
DVM.4.1.	DFM [GMR98a] (skat. 2.2.1.)	A	Noskaidrot, kādus atribūtus var izmantot par faktiem	Dots analizējamā procesa izmantojamo datu integrēts modelis Nav zināms, kādi ir nepieciešamie rādītāji datu analīzei
DVM.4.2.	DFM [GMR98a] (skat. 2.2.1.)	A	Identificēt dimensijas un dimensiju līmeņu hierarhijas	Dots analizējamā procesa izmantojamo datu integrēts modelis. Nav zināms, kādi ir vajadzīgie rādītāji datu analīzei. Nav zināms attiecībā pret kādām dimensijām nepieciešams analizēt rādītājus
DVM.5.	DN projekta situācija	J	Aprakstīt dažādām lietotāju grupām atbilstoši ieņemamam amatam, struktūrvienībai, kurā strādā un veicamajām darba funkcijām lietotāju skatus	Dots konceptuālais daudzdimensiju modelis. Nav zināms, kā notiks datu analīze, jo sākumā tika noteikts tikai datu analīzes mērķis, bet precīzas datu analīzes vajadzības nav noskaidrotas

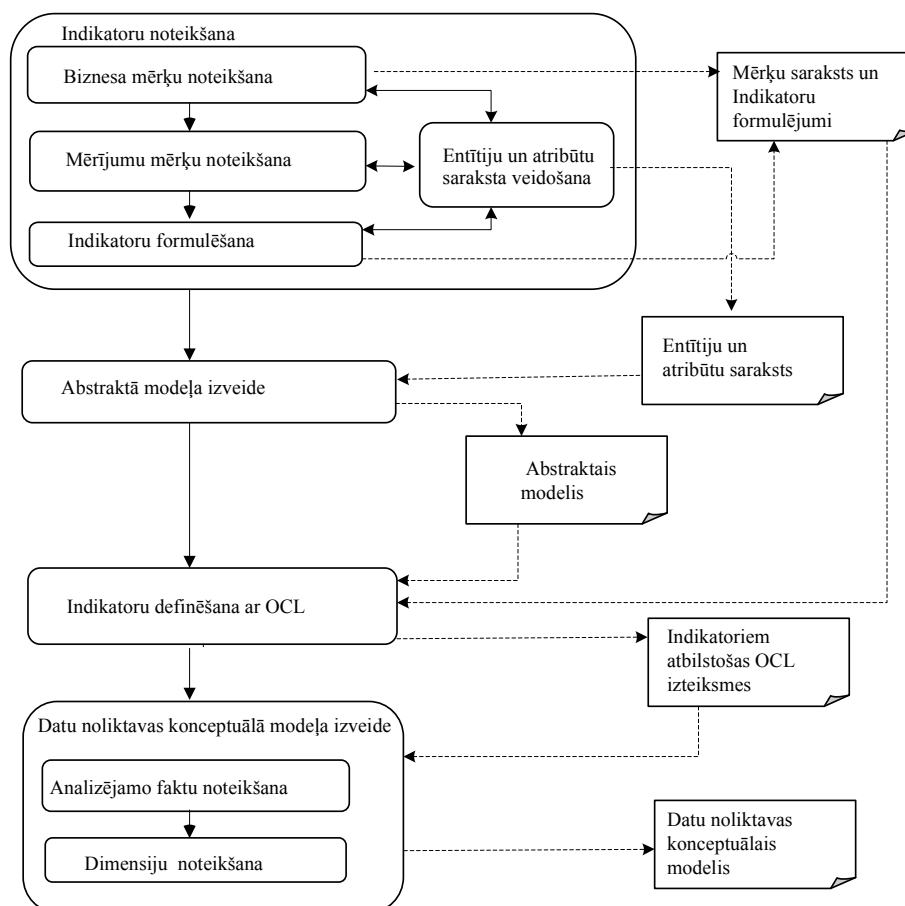
5.5. Mērķu vadītā metode

Promocijas darbā tiek piedāvāta mērķu vadītā metode, kas paredzēta datu noliktavas konceptuālā modeļa izstrādei - MVM (mērķu vadītā metode). Metode tiks detalizētāk aprakstīta, izmantojot tās pielietojumu "Procesu mērīšanas sistēmas" izstrādē, kura balstīta uz šim nolūkam izstrādātu datu noliktavu. Analizējamie biznesa procesi, kuri tiks izmantoti metodes ilustrācijai ir divi: „Studentu reģistrācija uz kursiem” un „E-studijas” - datuves otrās versijas izstrāde.

5.5.1. Mērķu vadītās metodes procesa modelis

Promocijas darbā tiek piedāvāta mērķu vadītā metode MVM, kas paredzēta datu noliktavas konceptuālā modeļa izstrādei un kas balstīta uz GQ(I)M metodi biznesa mērķu, mērījumu mērķu un indikatoru noteikšanai. Datu noliktava tiek veidota ar mērķi mērīt biznesa procesus.

Attēlā dots MVM procesu modelis (skat. 54.att.), bet sīkāks metodes apraksts dots punktā 5.5.2.



54.att. Mērķu vadītās metodes procesa modelis

MVM procesa modeli raksturo sekojošas aktivitātes:

- Indikatoru noteikšana. Sākumā nosaka biznesa mērķus, tad mērījumu mērķus, tad indikatorus, pielietojot GQ(I)M metodi.

- Abstraktā modeļa izveide. Izmantojot GQ(I)M metodi, vienlaicīgi ar mērķu, jautājumu un indikatoru noteikšanu tiek identificētas arī procesos iesaistītās entītijas un to atribūti. MVM metodes ietvaros tiek piedāvāts šo modeli, kas tiek nosaukts par abstrakto modeli un kas sastāv no identificētajām entītijām un atribūtiem, izstrādāt kā UML 2.0 *Structure* diagrammu, atbilstoši MVM metodes ietvaros piedāvātajam Indikatoru noteikšanas metamodelim (skat.55.att.).
- Indikatoru definēšana ar OCL. Indikatori, kas tika definēti saskaņā ar GQ(I)M, pēc tam tiek definēti ar OCL izteiksmju palīdzību, izmantojot uzbūvētā modeļa entītijas un atribūtus.
- Datu noliktavas konceptuālā modeļa izveide. Tiek analizēta OCL izteiksmju struktūra, lai noteiktu potenciālos faktus un dimensijas.

5.5.2. Mērķu vadītās metodes detalizēts apraksts

5.5.2.1. Metodes aprobācijas projekta raksturojums

Lai ilustrētu piedāvātās metodes pielietojumu, tiks aplūkots biznesa procesu piemērs - divas cieši saistītas darba plūsmas universitātē. Šo procesu mērīšanas vajadzībām tika nolemts izmantot datu noliktavu, integrējot jau eksistējošā datu noliktavā nepieciešamos rādītājus.

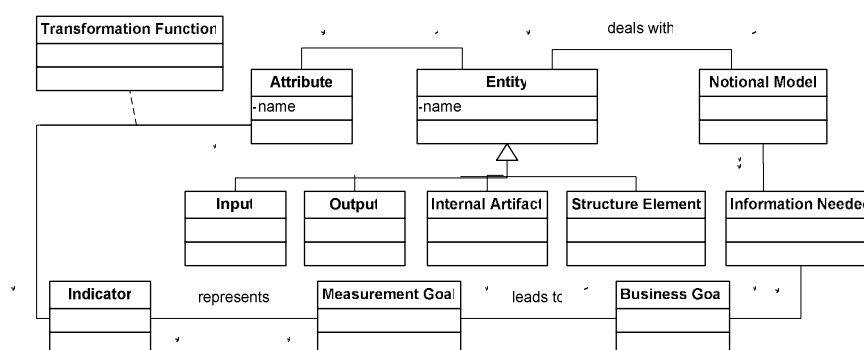
Pieteikšanās uz studiju kursiem ir darba plūsma, kurā studenti, atbilstoši apgūstamajai studiju programmai, piesakās uz izvēles kursiem, kurus tie vēlētos apgūt nākamajā semestrī [Med05]. Studenti var arī atteikties no jau izvēlēta studiju kursa. Izdarītā izvēle ir jāakceptē, bet līdz pieteikšanās perioda beigām iespējams atteikties arī no akceptēta kursa. Tā kā procesa organizācija neapmierināja tajā iesaistītos dalībniekus, tika nolemts veikt procesa mērīšanu, lai noskaidrotu nepieciešamās izmaiņas.

E-studijas ir darba plūsma, kurā katram e-kursam, ja tāds ir izstrādāts e-studiju vidē WebCT, tiek automātiski pierēģistrēti visi Studentu informācijas sistēmā (LUIS) uz atbilstošo kursu reģistrētie studenti. Studenti semestra laikā izmanto e-kursu materiālus, pasniedzējs savukārt novērtē studentu sasniegumus un atbalsta studentu mācību procesu. Universitātē jau vairākus gadus tiek realizēts e-universitātes projekts, kura tālākai attīstībai bija būtiski noskaidrot ar jau esošo kursu izmantošanu saistītos jautājumus.

5.5.2.2. Indikatoru noteikšanas metamodelis

Par pamatu procesu mērīšanai nepieciešamo indikatoru noteikšanai tika izmantota [PGF96] piedāvātais mērķu vadītais mērīšanas process GQ(I)M, kas savukārt lielā mērā ir balstīts uz GQM metodi [BR88]. GQ(I)M ir saīsinājums no *Goal- Question-Indicator-Measure* (Mērķis – Jautājums – Indikators - Mērījums) un indikatoru izmantošana atšķir šo metodi [PGF96] no GQM. GQ(I)M metode aprakstīta arī promocijas darba apakšnodaļā 4.4. un 4.pielikumā.

Indikatoru noteikšanai vajadzīgie GQ(I)M metodes pamatelementi un to savstarpējā saistība ir aprakstāma ar Indikatoru noteikšanas metamodeli (skat. 55.att.), papildus ieviešot asociāciju starp klasēm *Indicator* un *Attribute* un asociācijas klasi, kas apraksta transformācijas funkciju.



55.att. Indikatoru noteikšanas metamodelis

Atbilstoši GQ(I)M metodei sākumā tiek identificēti biznesa stratēģiskie mērķi (*Business goal*) un noskaidrotas informācijas vajadzības, lai sasniegtu mērķi (*Information needed*).

Tiek būvēts abstraktais modelis (*Notional Model*), kas satur entītijas (*Entity*), par kurām ir nepieciešama informācija. Entītijas var būt 4 veidu - Ievade (*Input*), piemēram, resursi, Izvade (*Output*), piemēram, produkti, efekti, Iekšējs artefakts (*Internal artifact*), līdzekļi (*things they hold*) – piemēram, dati, rīki, zināšanas vai Struktūras elements (*Structure element*), piemēram, apakšprocesi, aktivitātes, plūsmas. Katru entītijai raksturo īpašības (*Attributes*). Entītijas var būt savā starpā saistītas. Šāds modelis tiek būvēts mērķa vadītās mērīšanas (GQM) procesā, intervējot biznesa procesu dalībniekus (*stakeholders*), lai noskaidrotu biznesa mērķus, bet vienlaikus tiek vākta arī informācija par dalībnieku pārvaldītajām (*managed*), ietekmētajām (*affected*) entītijām un to raksturojošajām īpašībām.

Tālāk tiek noskaidroti apakšmērķi jeb mērījumu mērķi (*Measurement goals*). Tiek uzdoti jautājumi, lai noskaidrotu kvantitatīvu lielumu kopā ar tā attēlošanas veidu, kas tiek saukts par indikatoru (*Indicator*). Indikatori būtībā ir tā informācija par procesu, kas interesē lietotāju, lai noskaidrotu mērķa sasniegšanu. Indikatora apraksts satur datu elementus, kuri netika atsevišķi izdalīti metamodelī, jo nolūks bija aprakstīt tieši indikatoru saistību ar atribūtiem abstraktajā modelī. Saistību starp indikatoriem un atribūtiem var raksturot likumi jeb transformācijas funkcija (*Transformation function*).

5.5.2.3. Mērķu vadītās metodes procesa un produktu aspekti

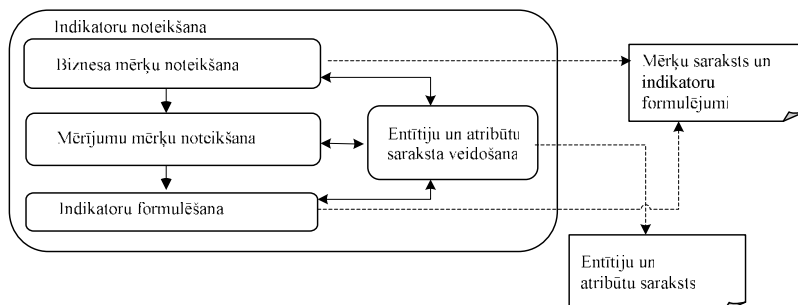
Tālāk dots mērķa vadītās metodes detalizēts apraksts, atbilstoši metodes procesa modelim, kas dots 5.5.1. punktā (skat. 54.att.).

1. aktivitāte. Indikatoru noteikšana

Ja par primāru tiek uzskatīts uzdevums mērīt procesus, tad būtiski ir noskaidrot, kādi būs indikatori, kas tiks mērīti. Šīs aktivitātes mērķis ir tieši indikatoru noskaidrošana. Metodes soļi tiek veikti, intervējot atbildīgo darbinieku (vai tā pilnvarotu darbinieku) par procesu, kuru paredzēts mērīt.

Procesa aspekti

Attēlā (skat. 56.att.) redzams fragments no metodes kopējā procesa modeļa, kas atspoguļo šīs aktivitātes saistību ar citām aktivitātēm un produktiem.



56.att. Aktivitāte "Indikatoru noteikšana"

Šajā aktivitātē to veikšanas secībā var apskatīt četrus soļus (izņemot 1.4. kas tiek atkārtoti veikts pēc katra no pārējiem soļiem).

1.1. solis. Biznesa mērķu noteikšana

Atbilstoši GQ(I)M metodei [PGF96] (skat. apakšnodaļu 4.4. un 4.pielikumu), tiek identificēti biznesa mērķi. Metodes aprobācijas projektā, tika identificēti divi biznesa mērķi atbilstoši diviem cieši saistītiem procesiem, kurus bija plānots mērīt:

- "Piedāvāt iedarbīgu un produktīvu studentu reģistrācijas procesu"
- "Piedāvāt iedarbīgu un produktīvu e-studiju procesu"

Metodes komponentam, kas atbilst 1.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: zināms viens vai daži sastīti procesi, kas jāmēra; nav zināmi indikatori, ko mērīt;
- lēmums:
 - metodes komponenta lietošanas mērķis: identificēt biznesa mērķus;
 - tehnika - GQ(I)M biznesa mērķu noteikšanas tehnika;
 - vērtējums: izmantots gatavs komponents.

1.2. solis. Mērījumu mērķu noteikšana

Noskaidrotie biznesa mērķi tiek analizēti un detalizēti līdz apakšmērķiem. Atbilstoši GQ(I)M metodei [PGF96] (skat. apakšnodaļu 4.4 un 4.pielikumu) biznesa mērķiem tiek identificēti atbilstoši mērījumu mērķi. Piemēram, metodes aprobācijas projektā, tika noskaidroti trīs mērīšanas apakšmērķi, kur mērķi G1 un G2 atbilst pirmajam biznesa mērķim un mērķis G3 – otrajam:

- G1. Uzlabot reģistrācijas procesa iedarbīgumu no studenta skatu punkta;
- G2. Uzlabot reģistrācijas procesa un ar to saistīto procesu efektivitāti no akadēmiskā departamenta skatu punkta;
- G3. Uzlabot ieguldījumu efektivitāti e-studijās no vadības skatu punkta.

Metodes komponentam, kas atbilst 1.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: zināms viens vai daži sastīti procesi, kas jāmēra. Nav zināmi indikatori, ko mērīt;

- lēmums:

- metodes komponenta lietošanas mērķis: identificēt mērījumu mērķus;
- tehnika - GQ(I)M mērījumu mērķu noteikšanas tehnika;
- vērtējums: izmantots gatavs komponents.

1.3. solis. Indikatoru formulēšana

Atbilstoši GQ(I)M metodei [PGF96] (skat. apakšnodaļu 4.4 un 4.pielikumu) tālāk tiek formulēti jautājumi, kas raksturo mērķa sasniegšanu. Atbilstoši jautājumiem tiek definēti indikatori, kas ir atbildes uz jautājumiem.

23.tabula

Mērķi, jautājumi un indikatori			
Mērķis G1			
Jautājums		Indikators	
Q2	Kādas ir reģistrācijas uz kursiem kritiskās vietas?	I3	Neapmierināto pieteikumu skaits
		I5	Izvēles kursu skaits ar lielu pieteikumu skaitu
		I6	Piereģistrēto studentu skaits attiecībā pret visu pieteikumu skaitu
Q3	Kāda ir kursu atcelšanas kārtība, ja piesakās mazs studentu skaits?	I8	Atcelto kursu skaits
Q4	Cik studentiem nebija pieejams reģistrācijas process izmantojot Internetu un kāpēc?	I10	Studentu skaits ar finansu parādiem
		I11	Studentu skaits ar akadēmiskajiem parādiem
Q5	Vai kursu piedāvājums bija pieejams laicīgi?	I12	Piedāvāto kursu skaits reģistrācijas procesa sākumā
Mērķis G2			
Jautājums		Indikators	
Q6	Cik studenti pierēģistrējās uz studiju kursiem Internetā?	I17	Nepilna laika studentu skaits, kas pierēģistrējies uz katru atsevišķu kursu
		I18	Kopējais studentu skaits, kas pierēģistrējušies uz kursiem
		I19	Kopējais nepilna laika studentu skaits, kas pierēģistrējušies uz kursiem
Q8	Kā mainījās studiju padomnieku slodze reģistrācijas laikā?	I22	Studiju padomnieka pierēģistrēto studentu skaits uz kursu
Q10	Kā mainījās darba plūsmu aktivitātes pa reģistrācijas dienām?	I26	Pieteikumu atcelšanas skaits
		I27	Kopējais reģistrācijas aktivitāšu skaits kursā
		I28	Reģistrācijas anulēšanu skaits katrā kursā

Mērķi, jautājumi un indikatori

Mērķis G3			
Jautājums		Indikators	
Q11	Kādi ir ieguldījumu rezultāti attiecībā pret pieejamību?	I30	Ieguldījumi pa fakultātēm
		I31	Reģistrēto studentu skaits e-kursos pa fakultātēm laika periodā
		I32	Studentu skaits, kas lieto e-kursus pa fakultātēm laika periodā
Q12	Kādi ir ieguldījumu rezultāti attiecībā pret lietošanu?	I30	Ieguldījumi pa fakultātēm
		I32	Studentu skaits, kas lieto e-kursus pa fakultātēm laika periodā
		I34	Studentu klikšķu skaits pa fakultātēm laika periodā
		I35	Studentu e-kursu lietošanas laiks pa fakultātēm laika periodā.
Q13	Kādi ir ieguldījumu rezultāti attiecībā pret studentu?	I30	Ieguldījumi pa fakultātēm
		I32	Studentu skaits, kas lieto e-kursus pa fakultātēm laika periodā
		I36	Studentu sesiju skaits e-kursos pa fakultātēm e-kursos

Piemēram, metodes aprobācijas projektā apskatāmo divu procesu gadījumā noteiktajiem mērķiem atbilstošie jautājumi un indikatori ir attēloti tabulā (skat. 23. tabula.). Kopā tika identificēti 13 jautājumi un tiem atbilstoši 36 indikatori. Promocijas darbā iekļautajā tabulā atstāti tikai tie indikatori, kas ir būtiski, lai izveidotu datu noliktavas modeli. Pārējie indikatori bija nepieciešami procesa mērīšanai un kontrolei (formulē, ko vajag mērīt), bet nemainīja iegūto datu noliktavas modeli.

Metodes komponentam, kas atbilst 1.3. solim, konteksts ir sekojošs:

- situācijas raksturojums: zināms viens vai daži sastīti procesi, kas jāmēra. Nav zināmi indikatori, ko mērīt;
- lēmums:
 - metodes komponenta lietošanas mērķis: identificēt mērāmos indikatorus;
 - tehnika - GQ(I)M jautājumu un indikatoru noteikšanas tehnika;
 - vērtējums: izmantots gatavs komponents.

1.4. solis. Entītiņu un atribūtu saraksta veidošana

Soļu 1.1 - 1.3. izpildes laikā, noskaidrojot biznesa mērķus, mērīšanas mērķus, jautājumus, indikatorus, atbilstoši GQ(I)M metodei [PGF96] (skat. apakšnodaļu 4.4. un 4.pielikumu) tiek veidots entītiņu un atribūtu saraksts par tiem jēdzieniem, kas ietverti šajos mērķos, jautājumos un indikatoros.

Metodes komponentam, kas atbilst 1.4. solim, konteksts ir sekojošs:

- situācijas raksturojums: zināms viens vai daži sastīti procesi, kas jāmēra. Nav zināmi indikatori, ko mērīt;
- lēmums:
 - metodes komponenta lietošanas mērķis: izveidot entītiju un atribūtu sarakstu, kas saistīti ar mērījumu indikatoru noteikšanas soļiem 1.1, 1.2., 1.3.;
 - tehnika - intervijas, kas veiktas, lai pielietotu GQ(I)M metodes komponentus, kas izmantoti 1.1. - 1.3. soļos;
 - vērtējums: izmantots gatavs komponents.

Produkta aspekti

1.1. produkts. Mērķu saraksts un indikatoru formulējumi

Tiek izveidota tabula, pēc struktūras atbilstoša tabulai, kas izmantota kā piemērs 1.3. solī. (skat. 23. tabula).

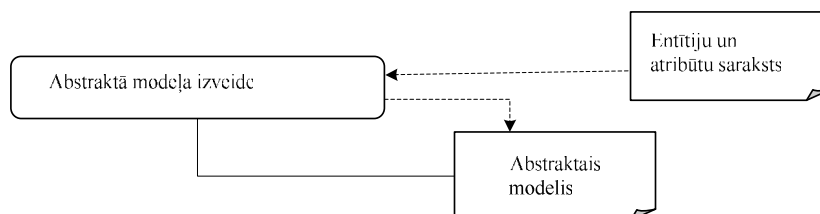
1.2. produkts. Entītiju un atribūtu saraksts

Netiek veidots modelis, neformālā veidā pieraksta entītijas un tās raksturojošos atribūtus.

2. aktivitāte. Abstraktā modeļa izveide

Procesa aspekti

Attēlā (skat. 57.att.) redzams fragments no metodes kopējā procesa modeļa, kas atspoguļo aktivitātes "Abstraktā modeļa izveide" saistību ar citām aktivitātēm un produktiem. Aktivitātei nav sīkākas detalizācijas pa soļiem.



57.att. Aktivitāte "Abstraktā modeļa izveide"

Paralēli mērķu definēšanai tiek identificētas uzņēmuma darbībai būtiskākās entītijas (procesu dalībnieki, objekti, procesi u.c) un tos raksturojoši atribūti, no kuriem tiek izveidots abstrakts modelis (*Notional Model*) ar paplašinātu UML 2.0 *Structure* diagrammas palīdzību atbilstoši iepriekš piedāvātajam Indikatoru noteikšanas metamodelim.

Paplašināta UML 2.0 *Structure* diagramma iepriekš izmantota [LK05] procesu biznesa konteksta aprakstam, apzināti neiedziļinoties detalizētu procesu plūsmu kontekstā. [LK05] autori balstās uz [OMG] klases definīciju, ka klase ir "objektu kopa, kam ir vienādas īpašību specifikācijas, ierobežojumi un semantika. Klases mērķis ir definēt objektu klasifikāciju un specificēt īpašības, kas raksturo objektu struktūru un uzvedību", tādēļ to var izmantot arī biznesa procesu modelēšanai.

Arī GQ(I)M metodē abstraktajā modelī interesējās par procesa kontekstu, bet nedaudz citā aspektā nekā [LK05], atbilstoši Indikatoru noteikšanas metamodelim.

Metodes komponentam, kas atbilst 2.aktivitātei, konteksts ir sekojošs:

- situācijas raksturojums: identificēti indikatori, kas jāmēra. Noskaidrota entītijū un atribūtu kopa, kas iegūta indikatoru identificēšanas gaitā,
- lēmums:
 - metodes komponenta lietošanas mērķis: izveidot datu modeli no noskaidrotajām entītijām un atribūtiem 1.4. solī, atbilstoši indikatoru noteikšanas metamodelim (skat. 55.att.) ;
 - tehnika - datu modelēšana;
 - vērtējums: izstrādāts jauns metodes komponents.

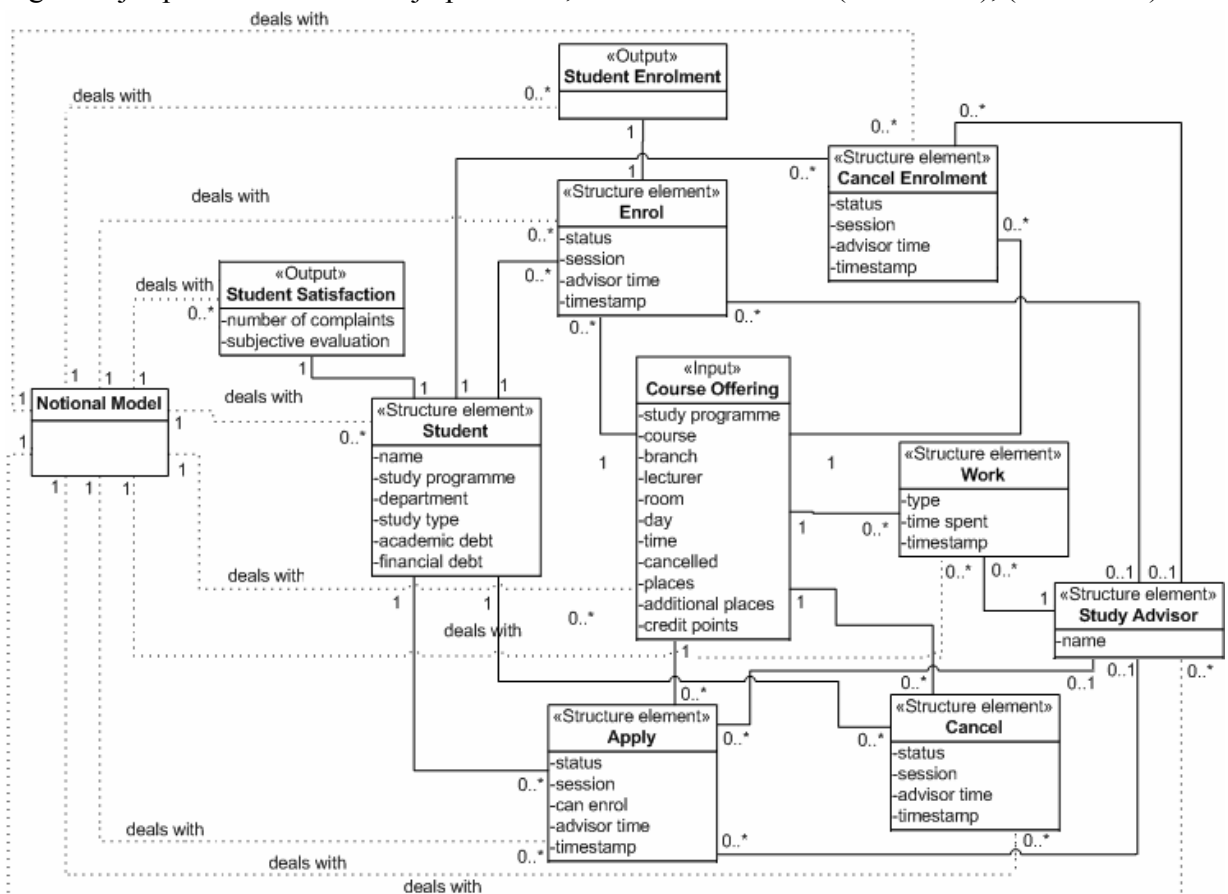
Produktu aspekti

2. produkts. Abstraktais modelis.

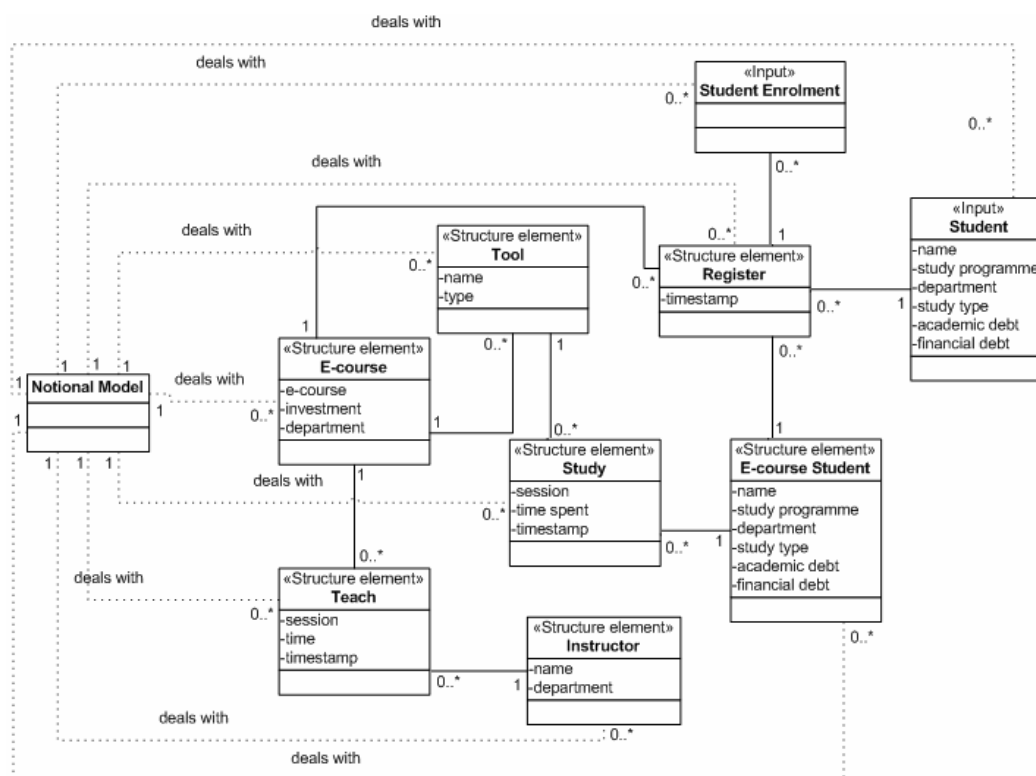
Aktivitātes rezultāts ir Abstraktais modelis, kas veidots ar UML 2.0 *Structure* diagrammas palīdzību. Tālāk doti divi piemēri.

Abstrakto modeļu piemēri

Metodes aprobācijas projekta ietvaros abiem procesiem tika izveidoti abstraktie modeļi reģistrācijas procesam un e-studiju procesam, kuri redzami attēlos (skat.58.att.), (skat.59.att.).



58.att. Abstraktais modelis studentu reģistrācijai uz kursiem

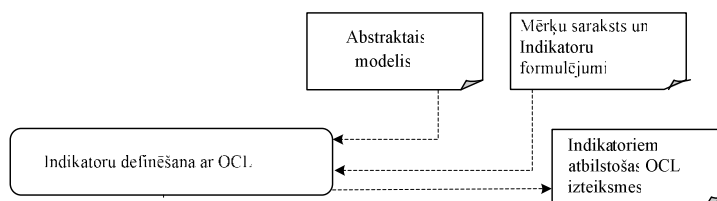


59.att. Abstraktais modelis e-studiju procesam

3. aktivitāte. Indikatoru definēšana ar OCL

Procesa aspekti

Attēlā (skat.60.att.) redzams fragments no metodes kopējā procesa modeļa, kas atspoguļo šīs aktivitātes saistību ar citām aktivitātēm un produktiem. Aktivitātei "Identifikatoru definēšana ar OCL" nav sīkākas detalizācijas pa soļiem.



60.att. Aktivitāte "Identifikatoru definēšana ar OCL"

OCL tiek izmantots kā līdzeklis datu ierobežojumu pierakstam, lai formāli pierakstītu mērījumu sastību ar atribūtiem. Kad abstrakts modelis ir uzbūvēts, indikatori, kas tika identificēti saskaņā ar GQ(I)M metodi, tiek formulēti ar OCL vaicājuma operācijām [OMGb], kas atgriež vērtību vai vērtību kopu, izmantojot objektus, atribūtus un asociācijas no abstraktā modeļa. Indikatoru definīciju piemēri, kas atbilst mērījumu mērķa G1 indikatoriem (skat. 23.tabula), doti

tabulā (skat. 24.tabulu), bet visiem promocijas darbā minētajiem indikatoriem, kas atbilst mērījumu mērķiem G1, G2 un G3, formulējumi ar OCL doti 6. pielikumā .

24. tabula

Indikatoru definēšana ar OCL

I3	context Notional Model::I3():Integer body: Apply -> select(status='Applied' and can enrol='No') -> size()
I5	context Notional Model::I5():Integer body: Course Offering -> select(Apply -> select(status='Applied') -> Student -> asSet() -> size() > places and branch='Language courses') -> asSet() -> size()
I6	context Course Offering::I6():Real body: (Enroll -> select(status='Enrolled') -> Student -> asSet() ->size()) / (Apply -> select(status='Applied') -> Student -> asSet() -> size())
I8	context Notional Model::I8():Integer body: Course Offering -> select(cancelled='Yes') -> size()
I10	context Notional Model::I10():Integer body: Student -> select(financial debt='Yes') -> size()
I11	context Notional Model::I11():Integer body: Student -> select(academic debt='Yes') -> size()
I12	context Notional Model::I12():Set(Course Offering) body: Course Offering

Metodes komponentam, kas atbilst 3. aktivitātei, konteksts ir sekojošs:

- situācijas raksturojums: identificēti indikatoru, kas jāmēra. Noskaidrota entītiju un atribūtu kopa, kas iegūta indikatoru identificēšanas gaitā;
- lēmums:
 - metodes komponenta lietošanas mērķis: formāli definēt indikatorus;
 - tehnika - OCL lietošana, tiek izmantots abstraktais modelis no 2. aktivitātes;
 - vērtējums: izstrādāts jauns metodes komponents.

Produktu aspekti

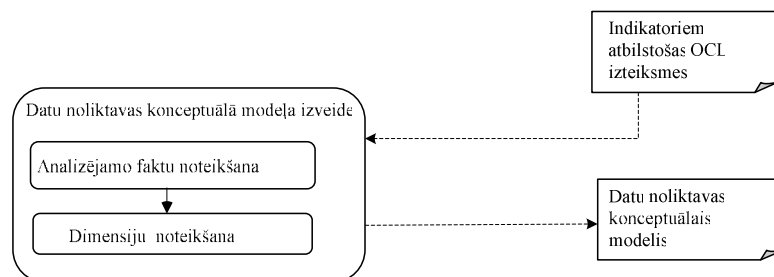
3.1. produkts. Indikatoriem atbilstošas OCL izteiksmes

Aktivitātes rezultāts ir OCL izteiksmes, kas apraksta katru indikatoru. Piemērs tabulai, kurā var dokumentēt gan indikatorus, gan OCL izteiksmes tika dots pie aktivitātes procesu aspektu apraksta (skat. 24.tabula).

4. aktivitāte. Datu noliktavas konceptuālā modeļa izveide

Procesa aspekti

Attēlā (skat.61.att.) redzams fragments no metodes kopējā procesa modeļa, kas atspoguļo aktivitātes "Datū noliktavas konceptuālā modeļa izveide" saistību ar citām aktivitātēm un produktiem.



61.att. Aktivitāte "Datu noliktavas konceptuālā modeļa izveide"

Šajā aktivitātē to veikšanas secībā var apskatīt divus soļus.

4.1. solis. Analizējamo faktu noteikšana

OCL vaicājuma operācijas, kas definē indikatorus, tālāk tiek analizētas, lai iegūtu datu noliktavas modeli. Sākumā identificē potenciālus faktus. Par potenciāliem faktiem uzskata rādītājus, kas tiek iegūtas ar operācijām, kuru rezultāts ir skaitliska vērtība, piemēram, `sum()`, `size()`, `round()`, `multiplication`, `division`, utml. Pētījuma ietvaros apskatītajiem gadījumiem tika identificēti sekojoši potenciāli fakti datu noliktavas zvaigznes shēmām (skat. 62.att, 63.att.), kas tiek iegūti no tabulā (skat. 24.tabula) un 6.pielikumā aprakstītām OCL vaicājuma operācijām:

`Apply->size()` (pieteikumu skaits - "*Number of applications*"), `Student->size()` (studentu skaits - "*Number of students*"), `Course Offering->size()` (piedāvāto kursu skaits), `Cancel->size()` (atteikumu skaits - "*Number of cancellations*"), `Enrol->size()` (reģistrāciju skaits - "*Number of enrolments*"), `Cancel Enrolment->size()` (atcelto reģistrāciju skaits - "*Number of enrolment cancellations*"), `E-course.investment->sum()` (Izmaksas - "*Investment*"), `Study.time spent->sum()` (Studijām patērētais laiks - "*Time spent*"), `Register->size()` (Reģistrāciju skaits - "*Number of registrations*"), `E-course Student->size()` (E-kursu studentu skaits - "*Number of e-course students*"), `Study->size()` (Studēšanas aktivitāšu skaits - "*Number of studying activities*"), `Study.session->size()` (Sesiju skaits - "*Number of sessions*").

Metodes komponentam, kas atbilst 4.1. solim, konteksts ir sekojošs:

- situācijas raksturojums: dots formāls indikatoru pieraksts ar OCL;
- lēmums:
 - metodes komponenta lietošanas mērķis: identificēt faktus;
 - tehnika - OCL vaicājumu analīze;
 - vērtējums: izstrādāts jauns metodes komponents.

4.2. solis. Dimensiju noteikšana.

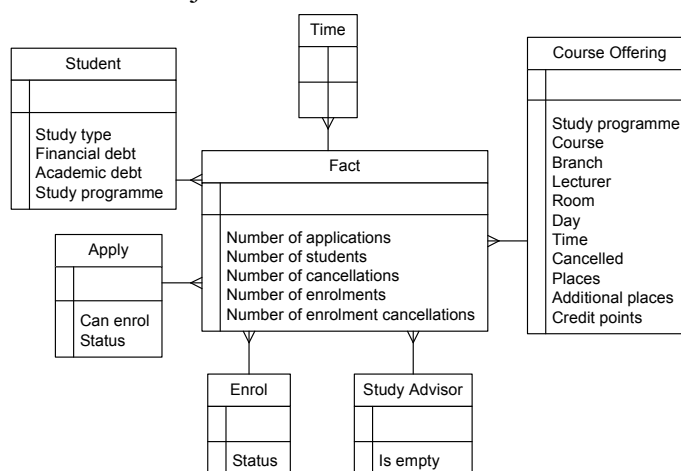
Piedāvātajā metodē tiek noskaidrotas potenciālās dimensijas un dimensiju atribūti. Pirmkārt par potenciālām dimensijām tiek uzskatītas klases, kas parādās OCL izteiksmes *context* sadaļā, izņemot klasi *Notional Model*. Par dimensiju atribūtiem uzskata klases atribūtus. Piemēram, šādā veidā tiek iegūtas dimensija *Course Offering* no tabulā (skat. 24.tabula un 6.pielikums)

aprakstītām OCL operācijām. Papildus dimensijām tiek identificēti pārējie dimensiju atribūti, kas tiek izmantoti OCL vaicājuma operācijas "select" izteiksmē. Piemēram, no tabulā (skat. 24.tabula un 6.pielikums) dotām OCL vaicājuma operācijām iegūst papildus nepieciešamos dimensiju atribūtus: klases *Apply* atribūti *status*, *can enrol*, klases *Student* atribūti *study type*, *financial debt*, *academic debt*, klases *Enrol* atribūts *status*, klases *E-course* atribūts *department*, klases *Register* atribūts *timestamp* un klases *Study* atribūts *timestamp*. Šos atribūtus grupē dimensijās pēc piederības noteiktai klasei. Pēc dimensiju noteikšanas var izrādīties, ka ne visi klases atribūti ir iekļauti datu noliktavas modelī. Šie atribūti var parādīt jaunus analīzes virzienus, kas iepriekš netika apspriesti. Šajā gadījumā papildus atribūti var tikt iekļauti modelī pēc intervijām ar lēmumu pieņēmējiem.

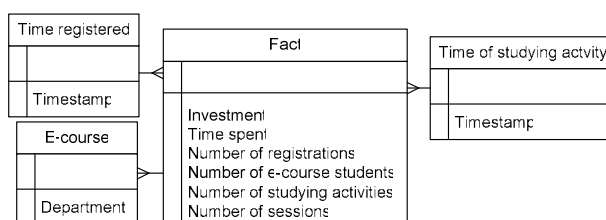
Visbeidzot modelim pievieno laika dimensiju, ja tā netika iegūta iepriekš aprakstītas analīzes laikā.

Pētījumā apskatītajiem indikatoriem tika izveidoti divi datu noliktavas modeļi, kas ir attēloti zīmējumos (skat. 62.att.) un (skat.63.att.). Potenciālais fakts *Course Offering->size()* netika iekļauts modelī, jo tā vērtība vienmēr būs 1.

Dimensiju hierarhiju noteikšanai papildus jāizmanto datu avotu modeļi, lai noskaidrotu identificēto dimensiju atribūtu eksistējošas atkarības datu avotos.



62.att. Zvaigznes shēma Reģistrācijas procesam



63.att. Zvaigznes shēma E-studiju procesam.

Metodes komponentam, kas atbilst 4.2. solim, konteksts ir sekojošs:

- situācijas raksturojums: dots formāls indikatoru pieraksts ar OCL;

- lēmums:
 - metodes komponenta lietošanas mērķis: dimensiju un dimensiju atribūtu noteikšana;
 - tehnika - OCL vaicājumu analīze;
 - vērtējums: izstrādāts jauns metodes komponents.

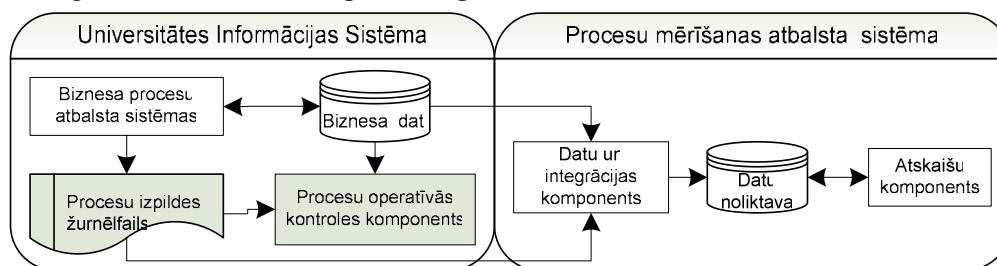
Produktu aspekti

3. produkts. Datu noliktavas konceptuālais modelis.

Aktivitātes rezultāts ir datu noliktavas shēmas. Piemērs tika dots pie aktivitātes procesu aspektu apraksta (skat. 62.att, 63.att).

5.5.3. Procesu mērīšana un procesu kontrole izmantojot datu noliktavu

Kopējā shēma procesu pārvaldības risinājumam ir redzama attēlā (skat.64.att.). Pelēkā krāsā iezīmētas procesu kontroles risinājuma sastāvdaļas - Procesu operatīvās kontroles komponents un procesu izpildes žurnālfails. Procesu operatīvās kontroles komponents daļu no atskaitēm veido no procesu izpildes žurnāla failu datiem, citas no biznesa datiem, neveicot šo datu integrāciju. Dati integrētā veidā tiek analizēti Procesu mērīšanas atbalsta sistēmā, kuras pamatā ir datu noliktavas zvaigznes shēmas, kas ir aplūkotas punktā 5.5.2.



64.att. **Procesu mērīšanas un procesu kontroles shēma**

Procesu operatīvās kontroles komponents nodrošina ar procesu darba plūsmu saistīto indikatoru analīzi tiešā veidā no žurnālfaila procesu izpildes laikā, kā arī šie indikatori tiek izmantoti datu noliktavā procesu mērīšanai. Ar procesu mērīšanas rezultātiem var iepazīties promocijas darba 7.pielikumā. Procesu operatīvās kontroles komponents realizēts kā Universitātes informācijas sistēmas sastāvdaļa. Tā kā būtiska loma gan procesu kontrolē, gan mērīšanā ir procesu izpildes žurnālfailam, ko izmanto datu uzkrāšanai par procesu izpildi, tad tālāk seko darba plūsmas izpildes žurnāla tabulas formāta apraksts.

Darba plūsmas izpildes žurnāla tabulas formāts

Darba plūsmu e-studiju gadījumā atspoguļo WebCT žurnālfaili, kas ir tīmekļa servera žurnālfaili, kas tiek ielādēti kopējā USERLOG tabulā (skat.25.tabula).

Žurnāltabulas formāts procesu izpildes uzkrāšanai

dt	usr	proc	ip	sk	tim
datums	login	WebCt darbība no URL (<i>script</i>)	ip adrese	cik reizes dienā	kopējais saņemto baitu skaits

Pielietojot MVM metodi, nav būtiski, vai vajadzīgie dati datu avotos eksistē. Trūkstošo vajadzīgo datu uzkrāšana tiek nodrošināta, sākot no datu noliktavas izstrādes momenta. Datu avotu sistēmās tiek iestrādātas datu noliktavai vajadzīgo datu uzkrāšanas funkcijas.

Kopējā žurnāla failā USERLOG tiek ierakstītas arī darbības, kas atbilst citu mērāmo procesu aktivitātēm, piemēram, pieteikšanās uz kursiem procesam, kam realizētas atbilstošas funkcijas Biznesa procesu atbalsta sistēmā (aprobācijas projekta gadījumā - LUIS). Kuriem procesiem krāt datus, lai vēlāk varētu ielādēt datu noliktavā, tiek noteikts, izmantojot abstrakto (*Notional*) modeli. Katrai klases "Entīcija" apakšklasei "*Structure element*", kas apraksta procesu aktivitātes, piemēram "*Apply*", "*Enrol*", tiek veidoti ieraksti žurnāla tabulā pie katras darba plūsmas izpildes. Aprobācijas projekta gadījumā LUIS procedūrās tika iestrādāta iespēja, ka katrs interesējošās procedūras izsaukums tiek fiksēts žurnālā laukā PROC sekojošā formātā:

<procedūras_nosaukums> (<kursa_kods>).

5.5.4. Secinājumi un rezultāti par mērķu vadīto metodi

Izstrādāta metode, balstoties uz mērķu vadītu pieeju, kurā izmantoti 4 gatavi metodes komponenti un konstruēti 4 jauni komponenti. Metodes komponenti doti tabulā (skat. 26. tabula), tajā raksturots komponenta lietošanas mērķis, situācijas faktori, norādīts, vai komponents jauns (J) vai gatavs G(N) citas metodes komponents, kas nav specifisks datu noliktavu jomai, gatavam komponentam norādīti arī avoti. Adaptēti komponenti no citām metodēm netiek izmantoti.

No kopsavilkuma par visiem metodes komponentiem, kas dots tabulā (skat. 26. tabula), vēl iegūstama mērķu vadītai metodei raksturīga situācijas faktoru kopa (3 dažādi faktori):

MVM.F1 - zināms viens vai daži sastīti procesi, kas jāmēra;

MVM.F2 - Nav zināmi indikatori, ko mērīt;

MVM.F3 - Indikatoru identificēšanas procesā noskaidrota entīciju un atribūtu kopa, no kuriem izveidots datu modelis;

Šie faktori kopā ar katra konkrētā komponenta lietošanas mērķi kalpo par pamatu komponentu atkalizmantošanas iespējas noteikšanai. Nodaļas beigās dots kopsavilkums visiem noteiktajiem faktoriem.

Nodaļā aprakstītā metode un tās pielietojums LU datu noliktavā aprakstīts un publicēts rakstos: [NSTN07a] Niedrite L., Solodovnikova D., Treimanis M., Niedritis A., „The Development Method for Process-Oriented Data Warehouse”, In: WSEAS Transactions on Computer Research, Issue 2, Vol. 2, February 2007, pp. 183 – 190.

[NSTN07b] Niedrite L., Solodovnikova D., Treimanis M., Niedritis A.: “Goal-Driven Design of a Data Warehouse-Based Business Process Analysis System”, In: Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering And Data Bases (AIKED '07), Corfu, February 2007.

Situācijas faktori mērķu vadītas metodes komponentiem

METODES KOMPONENTS	IZCELSMES AVOTS	J/A/G(N) vai G(S)	MĒRĶIS	FAKTORI, KAS RAKSTURO SITUĀCIJU
MVM.1.1.	GQ(I)M [PGF96] (skat. apakšnodaļu 4.4)	G(N)	Identificēt biznesa mērķus	Zināms viens vai daži sastīti procesi, kas jāmēra; Nav zināmi indikatori, ko mērīt
MVM.1.2.	GQ(I)M [PGF96] (skat. apakšnodaļu 4.4)	G(N)	Identificēt mērījumu mērķus	Zināms viens vai daži sastīti procesi, kas jāmēra; Nav zināmi indikatori, ko mērīt
MVM.1.3.	GQ(I)M [PGF96] (skat. apakšnodaļu 4.4)	G(N)	Identificēt mērāmos indikatorus	Zināms viens vai daži sastīti procesi, kas jāmēra; Nav zināmi indikatori, ko mērīt
MVM.1.4.	GQ(I)M [PGF96] (skat. apakšnodaļu 4.4)	G(N)	Izveidot entītiju un atribūtu sarakstu, kas saistīti ar mērījumu indikatoru noteikšanas soļiem	Zināms viens vai daži sastīti procesi, kas jāmēra; Nav zināmi indikatori, ko mērīt
MVM.2	DN projekta situācija	J	Izveidot datu modeli no noskaidrotajām entītijām un atribūtiem 1.4. solī, atbilstoši indikatoru noteikšanas metamodelim	Indikatoru identificēšanas procesā noskaidrota entītiju un atribūtu kopa
MVM.3.	DN projekta situācija	J	Formāli definēt indikatorus	Indikatoru identificēšanas procesā noskaidrota entītiju un atribūtu kopa; Izveidots datu modelis;
MVM.4.1.	DN projekta situācija	J	Identificēt faktus	Dots formāls indikatoru pieraksts ar OCL
MVM.4.2.	DN projekta situācija	J	Identificēt dimensijas	Dots formāls indikatoru pieraksts ar OCL

5.6. Nodaļas secinājumi

Tālāk katrai no piedāvātajām jaunajām metodēm atkārtoti doti situāciju raksturojošie faktori, kas tika konstatēti aprakstot metodes, lai novērtētu, vai konstatētie faktori ir līdzīgi, vai katrai metodei unikāli.

Lietotāju vadītai metodei:

- LVM.F1 - nav izdalīts viens vai daži konkrēti biznesa procesi, kuru mērīšana jāveic,
- LVM.F2- potenciāli daudzi intervējamie, intervijās iesaistīti potenciāli visi darbinieki, kas veic datu analīzi,
- LVM.F3- intervijās konstatēto prasību plašais spektrs,
- LVM.F4- nepieciešamība grupēt atbildes pēc to līdzības, lai būtu iespējams vieglāk noteikt izstrādes prioritātes,
- LVM.F5 -datu analīzes prasības, kas sargrupētas pēc nozīmes, jāpārveido par atbilstošu daudzdimensiju konceptuālo modeli,
- LVM.F6-daudzas prasības un nepieciešamība noteikt prioritātes.

Datu vadītai metodei:

- DVM.F1- process organizācijas darbībai jauns,
- DVM.F2- procesā iesaistīti daudzi datu avoti, kurus nepieciešams integrēt,
- DVM.F3- zināms analizējams process vai dažu savstarpēji saistītu procesu kopa,
- DVM.F4- dots (iegūstams) analizējamā procesa izmantojamo datu integrēts modelis,
- DVM.F5- nav zināmi potenciālie rādītāji, ko mērīt,
- DVM.F6- nav zināms attiecībā pret kādām dimensijām nepieciešams analizēt rādītājus,
- DVM.F7- nav zināms datu analīzes veids, jo sākumā tika noteikts tikai datu analīzes mērķis, bet precīzas datu analīzes vajadzības nav noskaidrotas.

Mērķa vadītai metodei:

- MVM.F1 - zināms viens vai daži sastīti procesi, kas jāmēra,
- MVM.F2 - Nav zināmi indikatori, ko mērīt,
- MVM.F3 - Indikatoru identificēšanas procesā noskaidrota entītiņu un atribūtu kopa, no kuriem izveidots datu modelis,

Salīdzinot šos faktorus, redzams, ka situāciju nepieciešams analizēt attiecībā pret sekojošiem kritērijiem (atmestas faktoru konkrētās vērtības):

- process pazīstamība konkrētajai organizācijai,
- analizējamo (mērāmo) procesu skaits,
- process ir vai nav identificēts (izvēlēts), kurš jāmēra,
- intervējamo skaits,
- analīzes prasību spektrs,
- prasību prioritātes,

- integrējamo datu avotu skaits,
- datu avotu modeļi un to integrēšanas iespējas,
- mērāmie indikatori,
- indikatoru analīzes dimensijas,
- analīzes veids .

Ne visām metodēm katrs no minētajiem kritērijiem ir būtisks. Kritēriju vērtības, kādas tās ir katrai no metodēm, saskaņā ar situāciju faktoriem, kas iegūti no metožu apraksta, dotas tabulā (skat. 27. Tabula). Ja kāda kritērija vērtība nav ietekmējusi lēmumu pieņemšanu, konstruējot jaunās metodes, tad tabulā tas apzīmēts ar "-".

27. Tabula

Situācijas faktoru kopsavilkums un salīdzinājums metodēm

	LVM	DVM	MVM
Process pazīstams vai jauns	-	Jauns DVM.F1	Zināms MVM.F1
Procesu skaits	Daudzi LVM.F1	1 vai daži DVM.F3	1 vai daži MVM.F1
Process ir vai nav identificēts, kurš jāmēra	nav identificēts LVM.F1	ir identificēti DVM.F3	ir identificēti MVM.F1
Intervējamo skaits	Daudzi LVM.F2	-	-
Analīzes prasību spektrs	Plašs LVM.F3, LVM.F6	-	-
Prasību prioritātes	Jānosaka LVM.F4, LVM.F6	-	-
Integrējamo datu avotu skaits	-	Daudzi DVM.F2	-
Datu avotu modeļi un to integrēšanas iespējas	-	Zināmi DVM.F4	Modelis tiek iegūts MVM.F3
Mērāmie indikatori	Tiek noskaidroti intervijās LVM.F5	nav zināmi DVM.F5	Nav zināmi MVM.F2, bet tiek iegūti MVM.F3
Indikatoru analīzes dimensijas	Tiek noskaidroti intervijās LVM.F5	nav zināmi DVM.F6	Nav zināmi MVM.F2, bet tiek iegūti MVM.F3
Analīzes veids	Tiek noskaidroti intervijās LVM.F5	nav zināms DVM.F7	Nav zināmi MVM.F2, bet tiek iegūti MVM.F3

Būtībā var runāt tāpat par 11 dažādiem situāciju faktoriem, kuru vērtības ir noteikušas lēmumu pieņemšanu, kādu pieeju izvēlēties, kā arī konkrētās metodes konstruēšanas laikā, kādu metodes komponentu izvēlēties, vai arī kādu komponentu izstrādāt no jauna.

Secinājumi par piedāvātajām metodēm un jauninājuma izvērtējums dots promocijas darba nobeigumā.

6. DATU NOLIKTAVAS REALIZĀCIJĀ IZMANTOJAMĀS METODES

6.1. Nodaļas nolūks

Datu noliktavu realizācijā, neatkarīgi no tā, kāda metode ir bijusi par pamatu konceptuālā modeļa izveidei, arī sastopama virkne problēmu. Promocijas darbā tiek piedāvāts risinājums divām problēmām - viena metode ir pieejas kontrolei datu noliktavās, otra metode ir saistīta ar lēni mainīgu faktu glabāšanas un izmantošanas problēmu.

Datu noliktavas pēc būtības rada draudu datu drošībai [KRRT98] – no vienas puses tās nodrošina pēc iespējas ērtāku, ātrāku, plašāku pieeju visiem integrētajiem datiem, no otras puses - šādi integrēti dati iegūst papildus vērtību un līdz ar to kļūst vēl sensitīvāki, tādēļ arī datu noliktavām nepieciešama pieejas kontrole. Promocijas darba ietvaros izstrādāta un aprakstīta metode pieejas kontrolei (skat. apakšnodaļu 6.2.). Izvērtēta virkne aspektu, kas ir būtiski pieejas kontrolei datu noliktavām (skat. punktu 6.2.1). Izstrādātajai metodei punktā 6.2.2. tiek piedāvāts pieejas kontroles modelis, bet punktā 6.2.3. - šim modelim atbilstošs pieejas kontroles mehānisms, izmantojot dinamiskos skatus. Piedāvātais pieejas kontroles modelis un tam atbilstošais pieejas kontroles mehānisms ir aprakstīts rakstā [NG05].

Datu noliktavās atsevišķos gadījumos sastopami tā sauktie "lēni mainīgie fakti", kuru glabāšana un izmantošana noteiktās situācijās ir problemātiska. Promocijas darba ietvaros izstrādāta un tiek apskatīta jauna metode (skat. apakšnodaļu 6.3.), kā glabāt šos "lēni mainīgos faktus" dalītu faktu veidā, pielietojot svarus, lai būtu iespējams realizēt vienlaikus divu veidu tipiskas datu noliktavu atskaites - iegūt fakta raksturojumu noteiktā laika momentā un iegūt tā sauktās fakta dinamikas atskaites.

Metodes aprobētas Latvijas Universitātes datu noliktavas projektā un metodes publicētas rakstos: [NG04] - pieejas kontroles metode, [GN06] - metode lēni mainīgu faktu glabāšanai dalītā un "svarotā" veidā.

6.2. Metode pieejas kontrolei datu noliktavām

[TCS03] definē "pieejas kontroli kā veidu subjektu mijiedarbības ar objektiem organizēšanai, kur pieeja ir informācijas pārvešana no objekta uz subjektu, bet subjekti ir aktīvās entītijas, kas pieejas rezultātā saņem vai maina informāciju par (vai datus no) pasīvajām entītijām jeb objektiem".

Pieejas kontrole ir viens no būtiskiem sistēmas drošības jautājumiem. Kādi stāvokļi sistēmai ir droši, tiek aprakstīts ar drošības politikas palīdzību [Bis02]. "No pieejas kontroles perspektīvas drošības politika ir likumu kopa, ko organizācija pieņem, lai regulētu, kurš var piekļūt kādiem resursiem"[Ben06]. Pieejas kontroles jēdziens [Ben06] tiek lietots kā sinonīms autorizācijai. Autorizācija nodrošina, ka pieprasītā aktivitāte vai piekļuve objektam ir iespējama, ja subjektam ir piešķirtas atbilstošas tiesības [TCS03].

Visbiežāk sistēmas pieejas kontrolei izmanto pieejas kontroles matricu, kas salīdzina subjektu, objektu un pieprasīto darbību. Pieejas kontroles matricas definīcija no [Ben06]: "Apzīmējot ar A pieejas matricu, S - subjektu kopu, O - objektu kopu, R - pieejas tiesību kopu, ko definē noteikta politika, tad matricas šūna A(s,o) reprezentē tiesību kopu $R_{so} \subset R$, kas ierobežo subjekta s pieeju objektam o".

Pieejas tiesība "ir sakārtots pāris <objekts, tiesību kopa>, kur tiesību kopa ir apakškopa no visām atļautajām operācijām, kas var tikt veikta ar objektu" [SGG05].

Drošības politiku specificēšanai tiek lietoti dažādi formālismi - pieejas kontroles modeļi. Savukārt "pieejas kontroles mehānisms ir konkrēta metode, rīks, procedūra vai to kopums noteiktas pieejas kontroles politikas implementēšanai" [Ben06]. Pieejas kontroles matrica ir viens no pieejas kontroles modeļiem, bet tā realizācijas dažādās sistēmās- pieejas kontroles mehānismi, kas atbilst šim modelim.

Datu bāzu jomā pieejas kontroles modeļiem būtu jāņem vērā sekojoši principi - 1) jābalstās uz loģiskā datu modeļa jēdzieniem (relāciju datu bāzei izmanto tabulas, atribūtus), 2) jāatbalsta arī satura- balstīta pieejas kontrole, kas atkarībā no datu vienības satura atļauj vai aizliedz piekļūt šai datu vienībai. [BS05]. Datu noliktavām būtiski ir gan datu bāzēm rekomendētie principi, gan specifiskas īpašības, kas tiks apskatītas punktā 6.2.1.

6.2.1. Prasības pieejas kontrolei datu noliktavās

Datu noliktavām sākotnēji pamatā bija ideja, ka tām jānodrošina uzņēmumam iespējami plašāks skats uz biznesu. Kopsakarības tiek meklētas visos datu noliktavā integrētajos datos, un datu analītiķiem tiek dota pieeja visai datu noliktavai. Datu noliktavu attīstības vēsture pierādīja, ka pastāv vairāki faktori, kas jāņem vērā:

- sensitīvu datu esamība - datu noliktava var ietvert gan personas datus, gan piemēram, ar veselības aprūpi saistītus datus, bez tam datu noliktavās dati iegūst papildus vērtību to integrācijas rezultātā [KRRT98];
- finansiāli apsvērumi - datu īpašnieks- kompānija vai struktūrvienība var gribēt datus pārdot, nevis vienkārši dalīties ar informāciju, kaut vai ar tās pašas kompānijas citu struktūrvienību [WME+01];
- lietotāju skaita pieaugums - lietotāju loks vairs nav tikai daži augstu amatu ieņemoši lietotāji [PP01].

Ieviešot datu noliktavā pieejas kontroli, jāņem vērā, ka tas var traucēt analīzes iespējām integrētajiem datiem, kā arī iespējams pat iegūt aplamus rezultātus [PP00], it īpaši gadījumos, kad lietotājs nav informēts par ierobežojumiem. Rezultāts var tikt nepareizi interpretēts, piemēram, rādītājs, kas iegūts tikai no dažu pieejai atļautu departamentu datiem, var nepareizi tikt interpretēts kā visu organizāciju raksturojošs rādītājs.

Ja salīdzinām datu noliktavas ar transakciju apstrādes sistēmām, mēs varam runāt par dažām datu noliktavu specifiskām īpašībām [KRRT98], [Inm02], kas nosaka prasības pieejas kontrolei:

- datu noliktavas integrē datus no daudziem datu avotiem;

Katrā no datu avotiem ir realizēti savi pieejas kontroles mehānismi, līdz ar to vienam atsevišķam lietotājam var būt iespējamas pretrunīgas tiesības uz vieniem un tiem pašiem datiem, kas glabājas vairāk kā vienā datu avotā. Pretrunīgs var būt gan pieejamo datu spektrs, gan datu detalizācijas līmenis, gan iespējamās veicamās darbības.

- Veicamās operācijas ar datiem.

Ja apskata tās operācijas, ko mēdz apskatīt autorizācijas kontekstā datu bāzu pārvaldības sistēmām, tad tipiskas operācijas, kas paredzētas datu noliktavas lietotājam, ir tikai lasīšanas operācijas. Iespējama arī OLAP operāciju ierobežošana, [WME+01] iesaka ierobežot *drill-down*, *roll-up*, *slice*, *dice*, *drill – through*. Tomēr jāņem vērā, ka ne visas OLAP operācijas var izpildīt uz visiem objektiem, piemēram, uz faktiem var izpildīt tikai lasīšanas operācijas.

- Datu noliktavu objekti.

Tā kā datu noliktavu daudzdimensiju modeļu specifika ir tā, ka tiek nodalītas divu tipu entītijas - dimensijas un fakti, kā arī dimensiju entītijām ir raksturīga hierarhiska uzbūve (dimensiju līmeņi), tad, izejot no datu noliktavu daudzdimensiju modeļu īpašībām, vairāki autori [WME+01], [PP00], [PP01] ierosina datu noliktavā ierobežot sekojošus objektus: dimensijas, dimensiju hierarhijas, faktus. [PP00] ierosina papildus diferencēt pieeju dažādiem faktu detalizācijas/agregācijas līmeņiem.

- Dažādi datu izmantošanas mērķi un dažādas lietotāju kategorijas.

Datu noliktavās var tikt veikta gan OLAP analīze un datizrāce, gan lietotas operatīvās atskaites. Katrai lietotāju kategorijai var būt atšķirīgas vajadzības, kuras būtu jāņem vērā, veidojot pieejas kontroles modeli konkrētai datu noliktavai. Operatīvo atskaišu gadījumā tiesības var sakrist ar lietotāja tiesībām datu avotu sistēmā, jo parasti tie būs tie paši lietotāji, bet nomainīta vide. OLAP un datizrāces gadījumā tie varētu būt pavisam citi lietotāji ar citām vajadzībām un funkcijām, līdz ar to viņiem tiesības jādefinē datu noliktavā.

- Iespēja vienai datu noliktavai izmantot vairākas dažādas lietojumprogrammas.

Datu noliktavu gadījumā ir tipiska situācija, ka ar vienu datu noliktavu iespējams strādāt ar dažādiem rīkiem - OLAP, datizrāces, atskaišu rīkiem. No katras kategorijas rīki var būt arī vairāki. Šajos rīkos bieži vien tiek piedāvāta iespēja pašā rīkā nodrošināt lietotājam pieejamo datu kontroli, bet daudzu rīku gadījumā vienā datu noliktavā ir grūti garantēt šo tiesību saskaņotu administrēšanu. Ņemot vērā iepriekšminētos apsvērumus par vairāku rīku lietošanu iespēju, datu noliktavām drošāku un saskaņotu risinājumu iespējams realizēt datu bāzes līmenī.

6.2.2. Pieejas kontroles modelis datu noliktavai

6.2.2.1. Datu noliktavu pieejas kontroles modeļu apskats

Pieejas kontroles modeļus datu noliktavām var klasificēt pēc datu noliktavas arhitektūras komponentes, kur tos paredzēts realizēt, piemēram, relāciju datu bāzē, OLAP serverī vai lietotnē. Lietotnes risinājums tiek uzskatīts par nedrošāko [PP00], jo var tikt apiets. Realizējot pieejas kontroles modeli relāciju datu bāzes līmenī, tiek izmantotas dažādu DBPS iespējas, kā, piemēram, skati, lomas, kā arī tāda iespēja kā VPD (*Oracle Virtual Private Database*).

Eksistē divas pretējas pieejas autorizācijas modeļiem „open world” – viss atļauts, izņemot to, kas tiešā veidā aizliegts, vai pretēja - „closed world” pieeja. [PP00] uzskata, ka OLAP gadījumā „augsta līmeņa” lietotāju dēļ „open world” pieeja ir piemērotāka. Vēl mēdz pielietot tā saukto „ownership” pieeju – datu noliktavas datu īpašniekiem jābūt tiem pašiem, kas datu avotā, bet tad ir problēma noteikt, kas būs īpašnieks agregātdatiem no daudziem avotiem.

Datu noliktavām dažādi autori piedāvā savus pieejas kontroles modeļus, piemēram, [WME+01] savu autorizācijas modeli definē subjektu kopai S, fakti kopai F, dimensiju kopai D, un četrām darbībām $T = \{read, drill-down, roll-up, slice\}$. [PP00],[PP01] definē prasības OLAP pieejas tiesībām, piemēram, "noslēpt kubu", "noslēpt noteiktu mērījumu", "noslēpt kuba apakškopu, ierobežojot kādas dimensijas vērtību", "noslēpt noteiktu datalizācijas līmeni". Pieejas ierobežojumi tiek definēti pēc „open world” principa ar negatīvu ierobežojumu palīdzību.

Konceptuālai pieejas tiesību modelēšanai [FTVP04] piedāvā UML paplašinājumu, lai varētu izstrādāt daudzdimensiju konceptuālos modeļus, tajos iekļaujot arī informāciju par lietotāju - kādām īpašībām jābūt spēkā, lai piekļūtu informācijai.

No praksē realizētu projektu aprakstiem pieejama informācija ir par MIT (*Masačūsetas Tehnoloģiskā Institūta*) datu noliktavas pieejas kontroles modeli [Tho00]. MIT datu noliktavā nodrošina pieejamo datu kontroli datu bāzes tabulu ierakstu līmenī. MIT datu noliktavas lietotāji datus var aplūkot tikai caur dažādiem skatiem. Ir izveidota pieejamo datu tabula - t.s. drošības tabula, kurā glabājas lietotāju vārdi un kodi. Arī katram ierakstam datu noliktavas fakti tabulās tiek pievienoti šādi kodi. Lietotāji caur skatiem redz tikai tos datus, kuru ierakstiem kods sakrīt ar lietotājam piešķirto kodu. Jebkuri ierobežojumi ir veidoti tikai fakti tabulām, ne dimensijām.

6.2.2.2. *Satura vadītais pieejas kontroles modelis datu noliktavām*

Piedāvātais pieejas kontroles modelis ir satura-vadīts un ņem vērā to, ka datu noliktavām pieļaujamā darbība ir lasīšanas operācija. Pieejas kontroles modeli definē:

- ierobežotā dimensija ir tā zvaigznes shēmas dimensija, kuras vienai kolonnai – ierobežojošai kolonnai eksistē nosacījums, kas pēc datu satura ļauj noteikt, kuri ieraksti no šīs dimensijas ir pieejami lietotājam.

- Horizontālais ierobežojums ir to tabulas ierakstu kopa, kas pieejama lietotājam. Ar I apzīmē tabulas ierakstu kopu, ar H_i apzīmē horizontālo ierobežojumu, $H_i \subset I$.

Horizontālo ierobežojumu definē atbilstoši ierobežojošās kolonnas saturam kādā no veidiem:

- pārskaitot pieļaujamās vērtības $\langle \text{vērtība}_1 \rangle, \dots, \langle \text{vērtība}_i \rangle$;
- norādot pilnas vērtības vietā tās sākuma simbolus un % zīmi.

- Vertikālais ierobežojums ir to kolonnu kopa, kas pieejama lietotājam. Ar K apzīmē visu tabulas kolonnu kopu, V_i - visu lietotājam pieejamo kolonnu kopu, $V_i \subset K$.

Vertikālo ierobežojumu V_i pieraksta: $\langle \text{Tabula} \rangle (\langle \text{kolonna}_1, \dots, \text{kolonna}_i \rangle)$, kur vertikāli ierobežotai tabulai pārskaita visas atļautās kolonnas. Ja lietotājam pieejama visa tabula, norāda tikai tabulas nosaukumu.

- Datu pieejas tiesības apzīmē lietotāja iespēju lasīt tos datus no tabulām, kas atbilst gan vertikālajam, gan horizontālajam ierobežojumam. Ja salīdzina ar pieejas kontroles matricas

modeli, vienīgā operācija, kas tiek aplūkota, ir lasīšanas operācija, bet objekts - tabulas šūnas, ko veido $V_i \cap H_i$.

Ierobežotās dimensijas horizontālais ierobežojums kalpo par pamatu datu pieejas tiesību definēšanai visā zvaigznes shēmā.

Faktu tabulas pieejamie ieraksti - fakti tabulas horizontālais ierobežojums - tiek atrasti, izmantojot ierobežotās dimensijas horizontālo ierobežojumu. Pieejami ir tie fakti tabulas ieraksti, kuru ārējās atslēgas vērtības atbilst ierobežotās dimensijas horizontālajam ierobežojumam, t.i. tajā iekļauto ierakstu primārajām atslēgām.

Pārējo fakti tabulai piesaistīto dimensiju horizontālie ierobežojumi nosakāmi, izmantojot iepriekš noskaidroto horizontālo ierobežojumu fakti tabulai. Pieejami ir tie pārējo dimensiju ieraksti, kuru primāro atslēgu vērtības ietilpst fakti tabulas horizontālajā ierobežojumā.

Vienkāršākā gadījumā ierobežotā dimensija ir viena (skat. mehānisma aprakstu apakšpunktā 6.2.3.2.) , bet iespējamās arī vairākas ierobežotās dimensijas vienai datuvei (skat. mehānisma aprakstu apakšpunktā 6.2.3.3.).

6.2.3. Pieejas kontroles mehānisms, izmantojot dinamiskos skatus

Pieejas kontroles mehānisms, kas atbilst iepriekš definētajam pieejas kontroles modelim, veidots, izmantojot dinamiskos skatus.

Ar dinamiskajiem skatiem apzīmēti skati, kas tiek aizpildīti tikai ar tādiem datiem, kas attiecīgajam lietotājam ir pieejami bāzes tabulās (zvaigznes shēmas fakti un dimensiju tabulas).

Pieejas kontroles mehānisms veidots tā, lai nebūtu nepieciešamības veidot sarežģītas, manuāli administrējamās datu struktūras. Piedāvātais mehānisms izmanto pa vienam dinamiskam skatam katrai datu noliktavas dimensijai un fakti tabulai, kā arī virkni funkciju, kas realizē datu pieejas tiesību kontroli konkrētajam lietotājam. Šīs funkcijas tiek izmantotas dinamisko skatu definīcijās. Šie dinamiskie skati pēc kontroles funkciju izpildes tiek aizpildīti tikai ar tiem datiem, uz kuriem lietotājam ir datu pieejas tiesības.

Pieejas kontroles mehānisms apmierina sekojošās kvalitātes prasības:

- ja lietotājam nav pieejama neviena rindiņa datu noliktavas tabulā – fakti tabulā vai dimensiju tabulā, tad attiecīgā tabula lietotājam nav jāredz,
- ja lietotājam nav pieejama neviena kolonna datu noliktavas tabulā – fakti tabulā vai dimensiju tabulā, tad attiecīgā tabula lietotājam nav jāredz.

Risinājums apmierina arī izvirzītās kvalitātes prasības. Tukšie dinamiskie skati lietotājiem tiek padarīti neredzami, lietotājiem atņemot tiesības lasīt tukšos dinamiskos skatus.

Problēmas, kas saistītas ar šiem kvalitātes kritērijiem, var rasties ar datu pieejas rīkiem, kuros iepriekš jādefinē lietotājam pieejamo datu metadati, kā to prasa rīka uzbūves principi. Šajā gadījumā tukšajiem dinamiskajiem skatiem atbilstošie elementi (piemēram, folderi *Oracle Discoverer* gadījumā) teorētiski var būt redzami, jo viens un tas pats skats tiek aizpildīts dinamiski un tiek izmantots visiem lietotājiem vienas un tās pašas atskaites veidošanai. *Oracle Discoverer* rīkā šī problēma ir atrisināta - *Oracle Discoverer* padara folderus neredzamus tiem

lietotājam, kuriem atbilstošās tabulas vai skati datu bāzē nav pieejami, pārējiem lietotājiem tas pats folderis ir redzams. Citu rīku gadījumā būtu nepieciešama šī aspekta pārbaude.

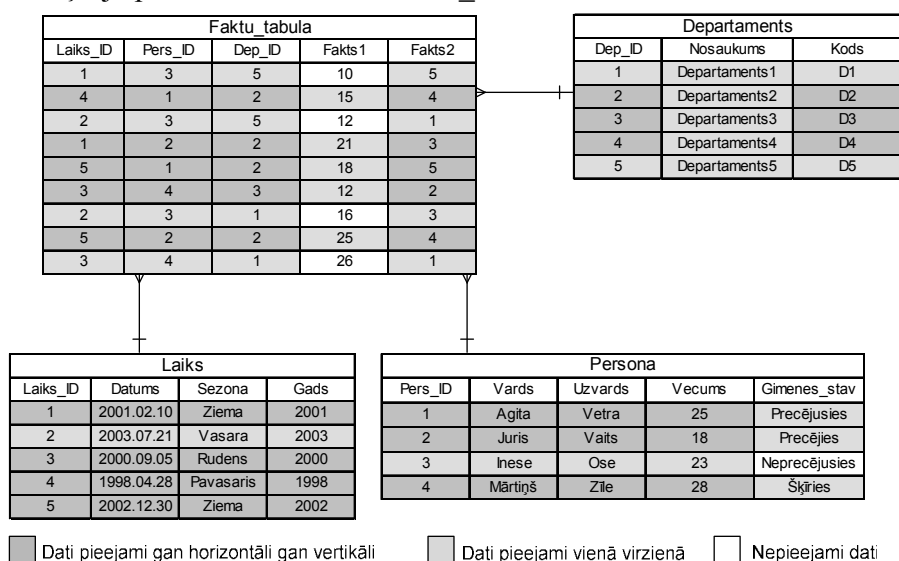
Definējot datu pieejas tiesības dinamisko skatu gadījumā, ir jādomā par ierobežojumu nepieciešamību. Katrs no tiem ievērojami palielina vaicājumu atbildes laiku. It īpaši tas attiecas uz vertikālo ierobežojumu.

Lietotājiem pieejamo datu kontrole, izmantojot dinamiskos skatus, tiks tālāk apskatīta vairākiem atsevišķiem gadījumiem - realizācija ar vienu ierobežoto dimensiju, kas ir vienkāršākais gadījums, kas pēc tam tiek papildināts vairāku ierobežoto dimensiju gadījumam. Pēdējais tiks apskatīts gadījums, ja lietotāju tiesības tiek integrētas no vairākiem datu avotiem.

6.2.3.1. Piemēra apraksts datuvei ar ierobežojošu dimensiju

Tālāk promocijas darbā tiks apskatīts datuves piemērs ar ierobežoto dimensiju 'Departaments' ar ierobežojošo kolonnu 'Kods', kuras dati definē horizontālo ierobežojumu.

Aplūkotā piemēra zvaigznes shēmas modelis ar vizuāli attēlotu vertikālo un horizontālo ierobežojumu vienam noteiktam lietotājam redzams attēlā (skat.65.att.). Šim lietotājam horizontālais ierobežojums ļauj aplūkot datus, kas atbilst 'D2', 'D3' un 'D4', kas ir ierobežojošās kolonnas 'Kods' konkrētajam lietotājam atļautās vērtības. Konkrētajam lietotājam vertikālais ierobežojums neļauj aplūkot kolonnas 'Gimenes_stav' un 'Fakts1'.



65.att. Piemērs zvaigznes shēmas modelim ar tiesībām.

Attēlā (skat.65.att.) tumši pelēkas ir iekrāsotas lietotājam pieejamās tabulu ierakstu vērtības. Gaiši pelēkās šūnas atbilst tikai horizontālajam vai tikai vertikālajam ierobežojumam, neiekrāsotās šūnas neatbilst nevienam datu pieejas nosacījumam.

6.2.3.2. Pieejas kontroles mehānisma realizācija ar vienu ierobežoto dimensiju

Lai ilustrētu pieejas kontroles mehānisma darbības principus, tiks izmantots attēlā (skat.65.att.) redzamais zvaigznes shēmas piemērs ar tajā dotajām datu pieejas tiesībām.

Piedāvātā pieejas kontroles mehānisma datu struktūra datu pieejas tiesību glabāšanai vienkāršākajam gadījumam - viens datu avots un viena ierobežotā dimensija - ir redzama attēlā (skat. 66.att.). Horizontālais ierobežojums, kas ir biežāk lietotais ierobežojuma veids un var būt atšķirīgs katram lietotājam, tiek aprakstīts tabulā 'Lietotaji' kolonnā 'Rindas'. Vertikālais ierobežojums tiek aprakstīts tabulas 'Lomas' kolonnā 'Kolonnas'. Lomas tiek definētas tikai tad, ja ir nepieciešams ieviest jaunu atšķirīgu vertikālo ierobežojumu. Salīdzinoši ar horizontālo ierobežojumu, kas tiek definēts katram lietotājam, lomu skaits parasti ir neliels.

Tabulu 'Lomas' un 'Lietotaji' aizpildījumā ar datiem (skat. 66.att.) iepriekš apskatītajam piemēram (skat.65.att.) vertikālam ierobežojumam atbilst 'Loma3', bet horizontālajam ierobežojumam lietotājiem 'Lietot2' un 'Lietot4' atbilstošais ierobežojums.

Lomas		
ID	Nosaukums	Kolonnas
1	Loma1	Laiks,Departaments,Persona,Faktu_Tabula
2	Loma2	Laiks,Departaments,Faktu_Tabula
3	Loma3	Laiks,Departaments,Faktu_Tabula(Fakts2), Persona(Vards,Uzvards,Vecums)

Lietotaji			
ID	Lietot_vards	Rindas	Loma_ID
1	Lietot1	D2	2
2	Lietot2	D2,D3,D4	3
3	Lietot3	D%	1
4	Lietot4	D2,D3,D4	3

66.att. Datu struktūras datu pieejas tiesību glabāšanai ar vienu ierobežoto dimensiju

Ne visos gadījumos ir nepieciešams lietot vertikālo ierobežojumu. Ir tādas situācijas, kad visai datuvei nav nepieciešams vertikālais ierobežojums, piemēram, statistikas datuve varētu būt publiska. Tādā gadījumā vertikālais ierobežojums netiek definēts nevienai datuves tabulai.

Vertikālā ierobežojuma lietošanu nosaka prasības datu noliktavai un lietotāju vajadzības datu noliktavā, savukārt tieši horizontālais ierobežojums ir tā pieejas tiesību daļa, ko var mantot no datu avotiem. Nepieciešamības gadījumā mantotās tiesības var mainīt, var pievienot arī papildus ierobežojumus, ko nosaka citi datu avoti vai arī prasības datu noliktavai. Realizācija vairāku avotu gadījumā tiks apskatīta tālāk apakšpunktā 6.2.3.4.

Piedāvātajam autorizācijas modelim speciālgadījumā ar vienu ierobežoto dimensiju nepieciešamās darbības dinamisko skatu izveidošanai un uzturēšanai ir sekojošas:

- horizontālā ierobežojuma definēšana,
- vertikālā ierobežojuma definēšana,
- dinamisko skatu lasīšanas tiesību piešķiršana lietotājiem.

Horizontālā ierobežojuma definēšana

Lai definētu horizontālo ierobežojumu, tiek veidota funkcija, kuras izsaukums tiek iekļauts ierobežotās dimensijas dinamiskā skata definīcijā tādā veidā, lai šis skats atgrieztu to tabulas datu apakškopu, kura attiecīgajam lietotājam ir pieejama. Ja ierobežotajai dimensijai nav nepieciešams nekāds vertikālais ierobežojums, tad šai dimensijai atbilstošā skata vispārīgā definīcija sakrīt ar (1) kodā doto. Ja vertikālais ierobežojums ir nepieciešams, tad jāpārskaita visas kolonnas, tām pielietojot vēl vertikālā ierobežojuma funkciju, kas aprakstīta nākamajā solī.

Iepriekš aplūkotajam piemēram (skat.65.att.) ierobežotā dimensija ir 'Departaments'. Līdz ar to šai dimensijai atbilstošā dinamiskā skata definīcija, kurā realizēts tikai horizontālais ierobežojums, dots kodā (2).

```
Create view <ierobežotā_dim>_skats as select * from <ierobežotā_dim> where  
hor_func(<ierobežotās_kol_vert>,<lietot_vards>,<hor_ties_kol_nosaukums>)=1
```

(1)

```
Create view departaments_skats as select * from departaments where  
hor_func(kods,lietotajs,rindas)=1
```

(2)

Horizontālo ierobežojumu veidojošai funkcijai *hor_func* ir trīs parametri:

- ierobežotās kolonnas vērtība katram konkrētajam ierakstam. Iepriekš aplūkotajam piemēram (skat. 65.att.) - kolonnas 'kods' vērtības.

- Lietotāja vārds - tā lietotāja vārds, kurš izpilda skata definīcijā ietverto vaicājumu. Izmantojot Oracle DBPS, lietotāja vārdu var iegūt ar vēl citas funkcijas - funkcijas 'user' palīdzību.

- Kolonnas nosaukums, kurā datu struktūrās, kas apraksta lietotāja pieejas tiesības, glabājas horizontālā ierobežojuma informācija. Apskatāmā piemēra gadījumā tā ir tabulas 'Lietotaji' kolonna 'Rindas'.

```
Select <hor_ties_kol_nosaukums> into ir_pieejams from Lietotaji where  
Lietot_vards=lietotajs;  
If ierobežotās_kol_vert=substr(ir_pieejams) then return 1 else return 0;
```

(3)

Vertikālā ierobežojuma definēšana

Vertikālā ierobežojuma realizācijai nepieciešamas vairākas līdzīgas vertikālā ierobežojuma funkcijas, kas atgriež lietotājam atļautos datus. Tās var būt divu veidu:

- dimensiju tabulu funkcijas.

Šīm funkcijām ir pieci ieejas parametri - 1) lietotāja vārds (lietotājs), 2) dimensijas tabulas primārā atslēga (primārā_atslēga), 3) kolonnas vārds, kurai šī funkcija tiek pielietota (kolonna), 4) dimensijas vārds, kurā kolonna atrodas (bāzes_tabula), un 5) tās kolonnas vārds, kurā tabulā 'Lomas' glabājas vertikālo ierobežojumu aprakstošie dati (kolonnas).

Tā kā kolonnām dimensiju tabulā var būt dažādi datu tipi, piemēram, NUMBER, DATE vai VARCHAR, tad vertikālo ierobežojumu nevar realizēt ar vienu funkciju. Funkciju skaits sakrītīs ar dažādo lietoto datu tipu skaitu dimensijā.

- faktu tabulu funkcijas.

Ja vertikālais ierobežojums jāveido arī faktu tabulām, tad katrai faktu tabulai ir nepieciešams savs komplekts ar funkcijām, kuras atgriež dažādus datu tipus. Savukārt katrai faktu tabulai šis komplekts ir individuāls tādēļ, ka katrai faktu tabulai var būt atšķirīgas uzbūves primārā atslēga, kas sastāv no visām dimensiju tabulu ārējām atslēgām. Faktu tabulas vertikālo ierobežojumu veidojošās funkcijas kā ieejošos parametrus saņem 1) lietotāja vārdu (lietotājs), 2) kolonnas nosaukumu (kolonna) un 3) visu salikto primāro atslēgu (primārās_atslēgas_kolonnas).

Vertikālā ierobežojuma veidojošo funkciju izpildāmās operācijas var sadalīt trīs daļās, kā tas dots (4) kodā.

```
1. select kolonna into šūnas_vērtība from bāzes_tabula where
   primārā_atslēga=primārās_atslēgas_kolonnas;
2. select kolonnas into pieejamās_kolonnas from lomas, lietotaji where
   lietot_vards=lietotājs and lietotaji.loma_id=lomas.id;
3. If kolonna=substr(pieejamās_kolonnas) then return šūnas_vērtība else
   return null;
```

(4)

Kad ir izveidotas funkcijas, kuras realizē vertikālo ierobežojumu, var definēt skatus. Ir jāveido divu veidu skati. Dimensijām atbilstošo dinamisko skatu definīciju vispārīgā forma ir dota kodā (6), bet faktu tabulām atbilstošo skatu vispārīgā forma – kodā (5). Vērā tiek ņemts arī jau iepriekš definētais horizontālais ierobežojums un tiek izmantotas vertikālā ierobežojuma funkcijas.

```
Create view <faktu_tabula>_skats as select <primārās_atslēgas_kolonnas>,
<funkc(<lietotājs>,<kolonna1>,<primārās_atslēgas_kolonnas>)>,...,
<funkc(<lietotājs>,<kolonna2>,<primārās_atslēgas_kolonnas>)> from
<faktu_tabula> F, <ierobežotā_dimensija>_skats D where
F.<ārējā_atslēga_ierobežotā_dim> = D.<primārā_atslēga>
```

(5)

```
Create view <dimensija>_skats as select <primārā_atslēga>,
<funkc(<lietotājs>,<primārā_atslēga>,<kolonna1>,<dimensija>,
<ierobežotās_kol_vārds>)>,...,
<funkc(<lietotājs>,<primārā_atslēga>,<kolonna2>,<dimensija>,
<ierobežotās_kol_vārds>)> from <dimensija> where <primārā_atslēga> in (select
<ārējā_atslēga_dim> from <faktu_tabula>_skats)
```

(6)

Kodā (7) doti divu dinamisko skatu veidošanas piemēri, kas atbilst (skat. att.65.) piemēram – dimensijai 'Persona' un faktu tabulai 'Faktu_tabula'. Citas skatu definīcijas nav nepieciešamas, jo var pieņemt, ka dimensija 'Laiks' pieejama visiem, bet dimensijai 'Departaments' neeksistē vertikālais ierobežojums. Kodā izmantotie apzīmējumi kol_fakti_numb nozīmē faktu tabulas funkcijas paveidu skaitliska datu tipa vērtībām, bet kol_dim_varch - dimensiju funkcijas veidu varchar tipa vērtībām.

```
Create view fakti_skats as select F.Laiks_id, F.pers_id, F.dep_id,
kol_fakti_numb(user, 'Fakts1', F.Laiks_id, F.pers_id, F.dep_id) as Fakts1,
kol_fakti_numb(user, 'Fakts2', F.Laiks_id, F.pers_id, F.dep_id) as Fakts2 from
faktu_tabula F, departaments_skats D where F.dep_id=D.dep_id
```

```
Create view persona_skats as select pers_id,
kol_dim_varch(user,id, 'vards', 'Persona', 'kolonnas') as vards,
kol_dim_varch(user,id, 'uzvards', 'Persona', 'kolonnas') as uzvards,
kol_dim_varch(user,id, 'vecums', 'Persona', 'kolonnas') as vecums,
kol_dim_varch(user,id, 'gimenes_stav', 'Persona', 'kolonnas') as gimenes_stav
from persona where id in (select pers_id from fakti_skats)
```

(7)

Skatu lasīšanas tiesību piešķiršana

Pēc dinamisko skatu izveidošanas tiek piešķirtas šo skatu lasīšanas tiesības datu noliktavas lietotājiem. Tiek izveidota atbilstoša procedūra, kura dod lietotājiem skatu lasīšanas tiesības. Ja kāds skats ir tukšs, tad lietotājam tiesības lasīt šo skatu netiek dotas, pretējā gadījumā tiek dotas. Galvenie šīs procedūras soļi ir redzami kodā (8):

```
For visi_lietotāji exception if owner do rewoke select on skata_vārds from
  lietotājs;
For visi_lietotāji exception if owner do
  If eksistē_dati(skata_vārds, lietotājs) then
    grant select on skata_vārds to lietotājs
```

(8)

Kodā (8) tiek lietota funkcija *eksistē_dati*, kuras atgrieztā vērtība ir 'true', ja lietotājam attiecīgais skats nav tukšs.

Ja tiek izveidots jauns lietotājs, tas jāapraksta pieejas tiesību glabāšanas datu struktūrās - definē lietotāju un pievieno tam atbilstošu lomu, un izpilda procedūru, kura dod tiesības lasīt skatus. Jauni skati jāveido tikai pieejas kontroles mehānisma izveides sākumā, vai arī, ja tiek izveidotas jaunas datuves, kurām nepieciešams kontrolēt pieeju. Lietotāju puses rīkos papildus tiesības nav nepieciešams definēt.

6.2.3.3. Modeļa realizācija ar vairākām ierobežotām dimensijām

Ja ir vairākas ierobežotās dimensijas, tabula 'Lomas' tiek lietota, lai aprakstītu vertikālo ierobežojumu tāpat kā vienas ierobežotās dimensijas gadījumā, bet tabulā 'Lietotaji' ir nepieciešamas izmaiņas - jāizveido tik kolonnas, cik nepieciešamas horizontālā ierobežojuma aprakstīšanai. Tā, piemēram, ja jāierobežo pēc dimensijām 'Departaments' un 'Laiks', tad tabulā 'Lietotaji' nepieciešamas divas kolonnas 'Rindas_Dep' un 'Rindas_Laiks'. Skatu definēšanas algoritms saglabājas līdzīgs. Vispirms tiek definēts skats faktu tabulai. Ja datuvē ir divas ierobežotās dimensijas, tad faktu tabulai atbilstošā dinamiskā skata vispārīgā forma ir redzama kodā (9):

```
Create view <faktu_tabula>_skats as select <primārās_atslēgas_kolonnas>,
<func(<lietotājs>,<kolonnā>,<primārās_atslēgas_kolonnas>)>,...,
<func(<lietotājs>,<kolonnā>,<primārās_atslēgas_kolonnas>)> from
<faktu_tabula> where <ārējā_atslēga_ierobežotā_dim1> in (select
<primārā_atslēga> from <ierobežotā_dim1> where
hor_func(<ierobežotās_kol_vert1>,<lietot_vards>,<hor_ties_kol_nosaukums1>)=1)

Intersection

select <primārās_atslēgas_kolonnas>,
<func(<lietotājs>,<kolonnā>,<primārās_atslēgas_kolonnas>)>,...,
<func(<lietotājs>,<kolonnā>,<primārās_atslēgas_kolonnas>)> from <faktu_tabula>
where <ārējā_atslēga_ierobežotā_dim2> in (select <primārā_atslēga> from
<ierobežotā_dim2> where
hor_func(<ierobežotās_kol_vert2>,<lietot_vards>,<hor_ties_kol_nosaukums2>)=1)
```

(9)

Ja definē skatu, kas atbilst iepriekš aplūkotajam piemēram (skat. 65. att.), ierobežojot arī dimensiju 'Laiks', tad to var izdarīt ar (10) koda DDL teikumu.

```

Create view Fakti_loma3 as select Laiks_id,Pers_id,Dep_id,
kol_fakti_numb(user,'Fakts1', Laiks_id, pers_id, dep_id) as Fakts1,
kol_fakti_numb(user,'Fakts2', Laiks_id, pers_id, dep_id) as Fakts2 from
faktu_tabula where dep_id in (select dep_id from Departamets where
hor_func(kods,lietotajs,rindas_dep)=1)

Intersection

select Laiks_id,pers_id,dep_id,
kol_fakti_numb(user,'Fakts1',Laiks_id, pers_id, dep_id) as Fakts1,
kol_fakti_numb(user,'Fakts2', Laiks_id, cilv_id, strukt_id) as Fakts2 from
faktu_tabula where laiks_id in (select laiks_id from laiks where
hor_func(gads,lietotajs,rindas_laiks)=1)

```

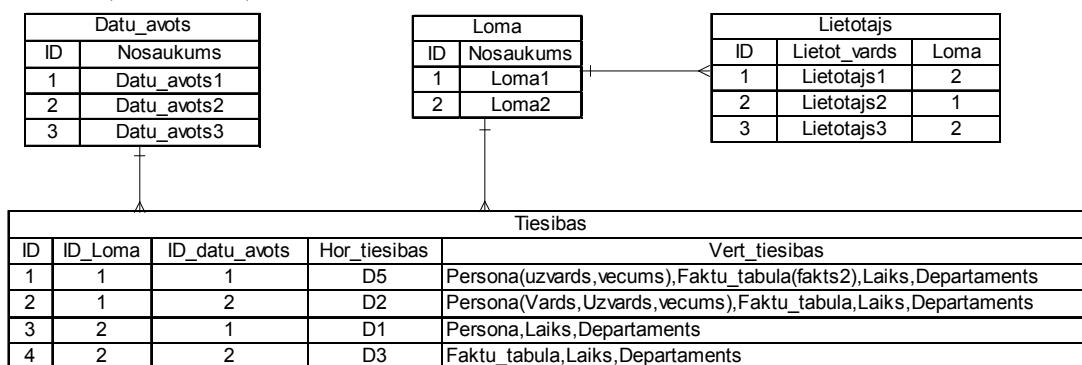
(10)

Visām dimensijām atbilstošie dinamiskie skati tiek definēti, izmantojot (6) kodā doto formu.

Ja skatu definīcijas netiek veidotas automātiski, tad nav nepieciešama tik komplicēta skata definīcija kā dota (9) kodā. Būtībā šī skata vaicājums var būt arī viens SELECT teikums, kurā iekļautas trīs tabulas – faktu tabula un abas ierobežotās dimensijas. Skatu vispārīgās formas ir veidotas ar mērķi padarīt šos dinamiskos skatus uzbūves ziņā līdzīgus, līdz ar to piemērotus automatizētai uzturēšanai.

6.2.3.4. Modeļa realizācija ar vairāk kā viena datu avota tiesībām

Datu struktūras, kurās iespējams glabāt datu pieejas tiesības no vairākiem datu avotiem, ir dotas attēlā (skat.67.att.).



67.att. Vairāku datu avotu tiesību glabāšana

Vairāku avotu gadījumā faktu tabulai atbilstošā dinamiskā skata vispārīgā forma ir līdzīga jau kodā (9) dotajai, izņemot to, ka šķēluma vietā jālieto apvienojums. Jāmaina arī funkcijas, kas realizē vertikālo ierobežojumu. Šīm funkcijām ir jāatgriež šūnas vērtība tad un tikai tad, ja šī šūna ietilpst kādā no lomas <hor_tiesibas, vert_tiesibas> pāriem. Pretējā gadījumā šīm funkcijām ir jāatgriež tukša šūna.

Lai to izdarītu, jāmaina (4) kodā dotie vertikālo ierobežojumu veidojošo funkciju pamatsoļi, kā rezultātā ir iegūstams kods (11).

```

1. select kolonna into šūnas_vērtība from bāzes_tabula where
   primārā_atšķēga=primārās_atšķēgas_kolonnas;
2. Cursor C1 is select * from LOMA, LIETOTAJŠ, TIESĪBAS where
   lietot_vards=lietotājs and lietotajs.loma=loma.id and ID_loma=LOMA.id;
3. For katram_kursora_ierakstam do
   if satur(kolonna,primārās_atšķēgas_kolonnas,hor_tiesības,vert_tiesības)
   then return šūnas_vērtība else return null;

```

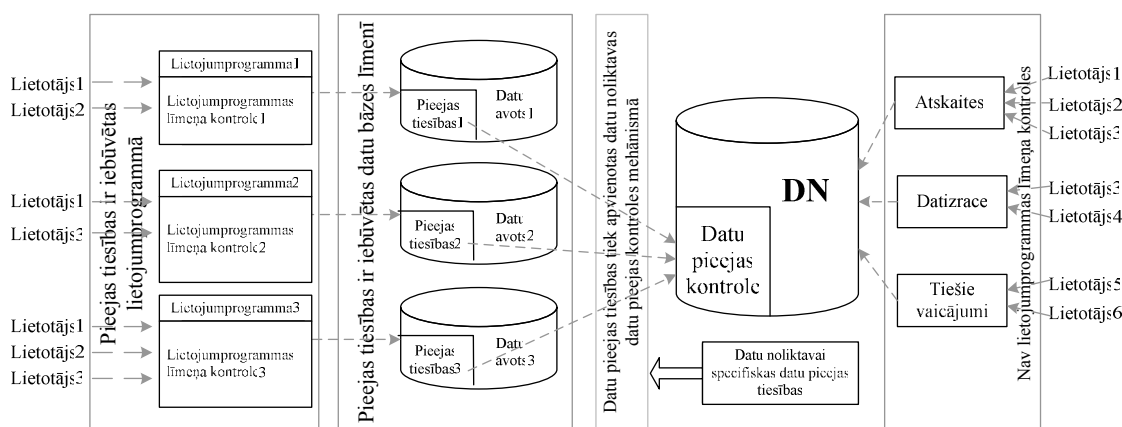
(11)

Katram_kursora_ierakstam ir cikla nosacījums, kas ļauj izpildīt ciklu, kamēr eksistē neapstrādāti ieraksti. Funkcija *satur* pārbauda, vai attiecīgā šūna pieder lomas kopai, kuru apraksta visi lomas pāri <hor_tiesības, vert_tiesības>.

Visas dimensijas tiek definētas, izmantojot (6) kodā doto formu.

6.2.3.5. Datu avotu tiesību izmantošanas iespējas datu noliktavu pieejas kontrolē

Piedāvātais pieejas kontroles mehānisms nodrošina iespēju izmantot datu avotu sistēmu pieejas kontroles mehānismus. Kopējā shēma, kas atspoguļo datu avotu datu bāzēs realizēto autorizācijas mehānismu izmantošanu, lai definētu datu noliktavas lietotājiem datu pieejas tiesības redzama attēlā (skat.68.att.). Dotā shēma ietver nepieciešamo komponentu papildus datu noliktavas specifisko tiesību definēšanai un uzturēšanai. Piedāvāto tiesību mantošanas iespēju var izmantot, ja datu avotos eksistējošie pieejas kontroles mehānismi ir izveidoti, izmantojot skatus, vai arī, ja katra lietotāja tiesības datu piekļuvei datu avotos glabājas datu bāzes tabulās.



68.att. Datu avotu tiesību izmantošana datu noliktavas pieejas kontroles mehānismā

Metodes aprobācijas projektā LU datu noliktavā datu pieejas tiesību definēšanai tiek izmantotas lietotāju pieejas tiesības no viena datu avota.

LU datu noliktavas gadījumā, realizējot promocijas darbā piedāvāto pieejas kontroles mehānismu, tabulas 'Lietotaji' dati tiek ņemti no datu avota (LUIS) tabulas LUSER, kas apraksta LUIS lietotāju tiesības. Par pamatu tiesību definēšanai LUIS tiek ņemtas vai nu struktūrvienības, vai studiju programmas. Datu noliktavā par pamatu horizontālā ierobežojuma definēšanai tiek ņemti struktūrvienību kodi (ierobežojošā kolonna).

Aprakstītās shēmas izmantošana nodrošina tiesību saskaņotību starp datu avotiem un datu noliktavu, kā arī daudzu lietotāju gadījumā atvieglo datu noliktavas pieejas kontroles automatizētas uzturēšanas procesu.

6.3. Metode dalīto faktu izmantošanai lēni mainīgu faktu gadījumā

6.3.1. Lēni mainīgo faktu jēdziens

Tipiska datu noliktavas atskaite reprezentē kāda objekta izmaiņas laika gaitā. Ja šīs izmaiņas notiek lēni, tad datu noliktavas daudzdimensiju modeļa elementu - faktu, kas mēra šīs izmaiņas, var saukt par lēni mainīgu faktu.

Lēni mainīga fakta jēdzienu ieviesa [CCY99]. Viņi apgalvoja, ka gadījumā, ja fakta vērtība paliek nemainīga kādu laika periodu, tad mērījumu vērtības atkārtojas katrā laika vienībā visa perioda garumā. Rezultātā faktu tabulā tiek glabātas daudzas un, pēc autoru domām, liekas un atkārtojošas vērtības. Kā šīs problēmas risinājums tiek piedāvāts veidot "transakciju-orientētu" faktu tabulu, kur fakta tabulas ieraksts reprezentē transakcijai līdzīgu struktūru. Tās vērtība ir nemainīgā fakta vērtība, bet transakcijas sākuma un beigu laiki atbilst tā laika perioda sākumam un beigām, kurā fakta vērtība ir nemainīga. Laika periods dažādiem ierakstiem var ievērojami atšķirties, sākot no dažām dienām vai mēnešiem, beidzot ar vairākiem gadiem. "Transakciju-orientētas" faktu tabulas ir piemērotas, lai atbildētu uz jautājumiem, kas attiecas uz pašu mākslīgo transakciju, piemēram, kāds ir vidējais transakcijas garums. Pieejas trūkums ir tāds, ka daži vaicājumi, piemēram, par mērījumu vērtību dinamiku, no "transakciju-orientētas" faktu tabulas nav atbildami.

Jāņem vērā, ka šīs "transakciju-orientētas" faktu tabulas būtībā glabā mākslīgi izveidotas transakcijas, kas datu avotu sistēmās neeksistē. Reālās transakciju līmeņa faktu tabulas, kas reprezentē datu avotu sistēmas transakcijas, tiek analizētas [KRRT98] kā arī tās ir vienas no plašāk lietotajiem faktu tabulu veidiem. Otrs faktu tabulu veids, saskaņā ar [KRRT98] ir momentuzņēmumu faktu tabulas, kas glabā mērījumus par biznesa notikumu (faktu) noteiktos laika momentos (reizi nedēļā, reizi mēnesī vai līdzīgi).

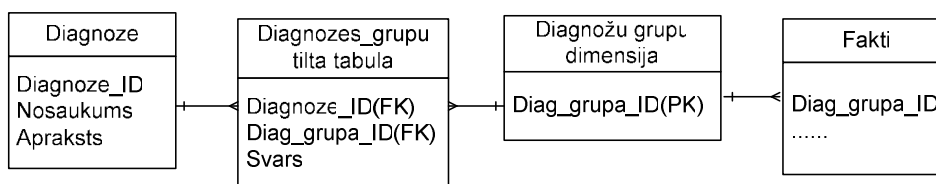
6.3.2. Faktu dalīšanas un svarošanas pielietošanas situāciju apskats

Neatkarīgi no faktu tabulas veida, dimensiju tabulas parasti ir saistītas ar faktu tabulu ar attiecību $1 : m$, tomēr datu noliktavās var būt situācijas, kad tiek runāts par $m : n$ attiecībām starp dimensiju tabulu un faktu tabulu. Lēni mainīgam faktam, ja tas tiek realizēts kā "transakciju-orientēta" faktu tabula, vērtība ir spēkā ilgāku laika periodu, kas ietver vairākas laika dimensijas laika vienības, t.i., laika dimensijas detalizācijas pakāpes. Tātad arī lēni mainīgi fakti var tikt uzskatīti par $m : n$ attiecību starp faktu tabulu un laika dimensiju.

Dažādu apsvērumu dēļ var būt nepieciešams saglabāt nemainīgu zvaigznes shēmas struktūru, piemēram, ātrdarbības risinājumu dēļ, zvaigznes shēmas modeļa vienkāršības vai citu iemeslu dēļ. Paliek atklāts jautājums, kā glabāt faktus zvaigznes shēmā, ja reāli starp faktu tabulu

un dimensiju attiecība ir $m:n$ nevis $m:1$, un kādi risinājumi ir piemēroti lēni mainīgo faktu tabulu veidošanai kā vienam no $m:n$ attiecību speciālgadījumiem.

Viens no veidiem, kā risināt $m:n$ attiecību starp faktu tabulu un dimensiju tabulu zvaigznes shēmas ietvaros, ir tilta tabulas, kas tiek izmantotas, lai līdzīgu objektu grupu elementiem aprakstītu to svarus grupas iekšienē. Piemērs šādai tilta tabulai ir pacientu diagnožu grupu tilta tabula [KR02], [SRM+01] (skat.69.att.). Fakti šajā gadījumā reprezentē maksājumus par procedūrām, bet viena veiktā procedūra, par ko ir veikts maksājums, var ārstēt uzreiz vairākas diagnozes. Šajā gadījumā $m:n$ attiecība ir starp diagnozēm un faktiem. Tiek izveidota Diagnožu grupu dimensija, kur katra grupa izveidota tā, lai tās attiecība ar faktu tabulu būtu $1:m$, bet tilta tabula apraksta $m:n$ attiecību starp Diagnozēm un Diagnožu grupām. Tiek ieviesti svāri, kas raksturo, piemēram, pamatdiagnozes un blakus diagnozes. Svaru summai ir jābūt 1. Atbilstoši katras diagnozes svaram grupā var tikt dalīti arī maksājumu fakti. Tomēr šāds risinājums nav viegli saprotams lietotājiem, ja viņiem ir pieejami datu noliktavas metadati, lai pašiem būtu iespējams definēt atskaites.



69.att. Tilta tabulas izmantošana, pēc [KR02], [SRM+01]

[SRM+01] apraksta sešus iespējamus risinājumus $m:n$ attiecībām datu noliktavās. Divi no tiem saistīti ar faktu tabulas detalizācijas pakāpes maiņu. Gadījumā, kad jaunie fakti nav pieejami, fakta vērtības tiek izrēķinātas, dalot sākotnējo fakta vērtību atbilstoši svāriem un tiek glabāti vai nu vienā faktu tabulā vai arī divās atsevišķās faktu tabulās, katrā ar savu detalizācijas pakāpi. Būtībā tā sauktās "tilta tabulas" vērtības - svāri tiek pārnesti uz faktu tabulām. Tā vietā, lai iepriekš izrēķinātu un glabātu detalizētus faktus, [EKK02] piedāvā ieviest transformācijas funkciju no vienas datalizācijas faktiem uz jaunās detalizācijas faktiem.

Autoru [SRM+01], [KR02], [EKK02] piedāvātie risinājumi attiecas uz reālu transakciju faktu tabulām. Piedāvātajos risinājumos lēni mainīgo faktu gadījums netiek aplūkots.

6.3.3. Piedāvātās metodes apraksts

Problēmas lēni mainīgu faktu gadījumā rodas, ja no vienas un tās pašas faktu tabulas ir nepieciešams veidot gan dinamiskas atskaites par lēni mainīgiem faktiem, gan arī atskaites par objektu kopējo stāvokli uz kādu noteiktu laika momentu. Kā piemēru var minēt faktus, kuri ir spēkā vairākus gadus, bet faktu dinamiskas atskaitē šī fakta vērtība jāiekļauj katru gadu (neatkarīgi no tā, ka fakta vērtība nav mainīta un nav veidots jauns ieraksts).

Lai risinātu šo problēmu, promocijas darbā tiek piedāvāta metode, kas pielieto faktu dalīšanu, tos svarojot. Tiek pieņemts, ka lēni mainīgie fakti ir $m:n$ attiecība starp faktu tabulu un laika dimensiju, un ka svarotie fakti tiek glabāti vienā faktu tabulā ar lēni mainīgajiem

faktiem, lai varētu aprēķināt gan faktu atribūtu dinamiku, gan situāciju uz kādu konkrētu laika momentu.

Jāņem vērā, ka faktus iespējams dalīt, lietojot dažādus svarošanas principus. Vienam lēni mainīgajam faktam ar vērtību f , sadalot to n dalītajos faktos, dalīto faktu vērtības f_i , šīs grupas ierakstiem var aprēķināt dažādos veidos, piemēram:

- kā vidējo vērtību uz faktu grupas ierakstu skaitu - $f_i = f/n$;
- atkarībā no katra dalītā fakta f_i spēkā būšanas intervāla garuma.

Ar faktu svarošanu lēni mainīgajam faktam tiks turpmāk saprasta sākotnējā fakta sadalīšana vairākos faktos, kuru spēkā būšanas laika apvienojums sakrīt ar sākotnējā ieraksta spēkā būšanas laiku. Spēkā būšanas laika intervāli sadalītajiem faktiem nepārklājas. Eksistē funkcija, ar kuru, pielietojot to jauniegūtajām jauno sadalīto faktu vērtībām, iespējams aprēķināt sākotnējā fakta vērtību.

Runājot par faktu dalīšanu, izmantojot jebkādu svarošanas principus, ir jāaplūko dažādi faktu veidi: aditīvi, semiaditīvi un neaditīvi, jo pielietotie risinājumi katram faktu veidam var būt atšķirīgi.

Promocijas darbā ir apskatīti divi faktu svarošanas veidi - proporcionāla faktu svarošana dinamikas veidošanai un faktu svarošana dinamikas veidošanai atbilstoši spēkā būšanas intervālam.

6.3.3.1. Proportcionāla faktu svarošana dinamikas veidošanai

Ar proporcionālu faktu svarošanu tiek apzīmēta sākotnējās fakta vērtības izdalīšana ar jaunās faktu grupas ierakstu skaitu, neatkarīgi no šiem ierakstiem atbilstošo spēkā būšanas laiku intervālu garumiem.

Ar intervālu $[ft1::ft2]$ apzīmē laika posmu, kurā ir spēkā sākotnējā fakta vērtība f un $ft1 < ft2$. Vērtība $ft2$ var būt nezināms nākotnes moments. Situācijas vienkāršošanai pieņemsim, ka $ft2$ ir zināms, kā tas arī ir lielā daļā gadījumu.

Ar dinamikas intervālu tiks apzīmēts laika periods, kura ietvaros ir nepieciešams atspoguļot fakta vērtību, lai to salīdzinātu ar šī fakta vērtībām nākošos pēc kārtas sekojošos tāda paša garuma laika periodos, kuros šī fakta vērtība arī ir spēkā.

Ar intervālu $[d_{i1}::d_{i2}]$ apzīmē dinamikas intervālu robežas, kur $d_{i1} < d_{i2}$ un $1 \leq i \leq n$; n - nepieciešamais dinamikas intervālu skaits, lai pārklātu sākotnējā fakta spēkā būšanas intervālu $[ft1::ft2]$. Pirmais dinamikas intervāls tiek izvēlēts tā, ka $d_{11} \leq ft1 \leq d_{12}$, bet pēdējam dinamikas intervālam ir spēkā nosacījums $d_{n1} \leq ft2 \leq d_{n2}$.

Ar Δdt apzīmē intervāla $[d_{i1}::d_{i2}]$ garumu laika dimensijas zemākajās granularitātes vienībās – mēnešos, dienās, stundās, minūtēs – atkarībā no konkrētās situācijas (datu analīzes prasībām). Visi dinamikas intervāli ir vienāda garuma.

Ja $n \geq 2$ tad $d_{i+11} = d_{i2} + 1$ un $d_{i+12} = d_{i1} + \Delta dt$, kur 1 ir laika dimensijas zemākās detalizācijas pakāpes laika vienība.

Ja spēkā nosacījums, ka $d_{11} \leq ft1 < ft2 \leq d_{12}$, t.i. $n=1$, tad fakta dalīšana nav nepieciešama, pirmajam un šajā gadījumā vienīgajam dinamikas intervālam ir spēkā pats lēni mainīgais fakts.

Fakta dalīšana, dalīto faktū laika robežu intervālu noteikšana un dalīto faktū skaita noteikšana notiek pēc sekojoša algoritma:

```

Int_pirmajam_dalit_faktam:=[ft1::d1t1]
i:=1
loop
    exit when ft2 > d1t2
    i:=i+1
    d1t1:=d1t1+1
    d1t2:=d1t1+Ad
    int_i:=[d1t1::d1t2]
end loop
i:=i+1
int_pedejam_dalit_faktam:=[d1t1::ft2]
skaits:=i;

```

(12)

Svarotās fakta vērtības fs katram dalītajam fakta ierakstam aprēķināmas pēc formulas {1}.

$$fs=f/skaits$$

{1}

Ja visas faktū vērtības ir vai nu '0' vai arī '1', tad tāds fakts tiek lietots objektu vai ar tiem saistīto notikumu skaitīšanai. Šāda veida fakts turpmāk tiks saukts par vienības faktū, savukārt šāda fakta sadalīšanu saskaņā ar formulu {1} un kodu (12) par vienības dalījumu.

Dalīto faktū veidošana un to vērtību aprēķināšana ir lietojama gadījumos, kad fakti ir aditīvi. Dalīto faktū tabulas piemērs ir dots attēlā (skat. 70.att.). Piemērā ir izmantota fakta dalīšana ar dinamikas intervālu "gads".

Faktu_tabula					
Sakotnejais_no	Sakotnejais_lidz	Dalitais_no	Dalitais_lidz	Fakts	Dalitais_fakts
15.04.2003	30.12.2004	15.04.2003	31.12.2003	10	5
15.04.2003	30.12.2004	01.01.2004	30.12.2004	10	5
24.01.2005	18.08.2005	24.01.2005	18.08.2005	20	20

70.att. Aditīvu faktū dalīšana, izmantojot svarus

Ja fakts ir semiaditīvs, sākotnējai faktū tabulai ir jāpievieno kolonna ar vienības faktū. Jāsvaro ir vienības fakts. Ja faktū aplūko pret aditīvu dimensiju, fakta vērtību sareizina ar sākotnējo vērtību, pretējā gadījumā ir jālieto sākotnējā fakta vērtība. Semiaditīvas faktū tabulas faktū dalījuma piemērs ir dots attēlā (skat. 71.att.). Faktū tabula būtībā sakrīt ar to, kas dota attēlā (skat. 70.att.), pieņemot, ka šajā gadījumā fakti ir semiaditīvi.

Faktu_tabula					
Sakotnejais_no	Sakotnejais_lidz	Dalitais_no	Dalitais_lidz	Fakts	Vienibas_dalijums
15.04.2003	30.12.2004	15.04.2003	31.12.2003	10	0,5
15.04.2003	30.12.2004	01.01.2004	30.12.2004	10	0,5
24.01.2005	18.08.2005	24.01.2005	18.08.2005	20	1

71.att. Semiaditīvu faktū dalīšana, izmantojot svarus

Faktu dalīšanu, izmantojot vienības fakta kolonnu, var uzskatīt par vieglāk saprotamu lietotājiem, jo būtībā fakti tabulā ir divi dažādi fakti, kas atsevišķos gadījumos var tikt sareizināti.

6.3.3.2. Faktu svarošana dinamikas veidošanai atbilstoši spēkā būšanas intervālam

Ne vienmēr iespējams fakti vērtību dalīšanai izmantot dinamikas intervālu skaitu, jo ir būtiski ņemt vērā dalīto fakti spēkā būšanas laika intervālu garumus. Lai varētu atrast katras sadalītā fakta vērtības lielumu atkarībā no tā spēkā būšanas laika intervāla garuma, ir jāizveido papildus funkcija *diff()*, kas atrod laika dimensijas zemākās granularitātes objektu skaitu norādītā laika intervālā. Ja šāda funkcija eksistē, tad sākotnējam faktam katrā dalītā fakta vērtību iespējams atrast pēc formulas {2}.

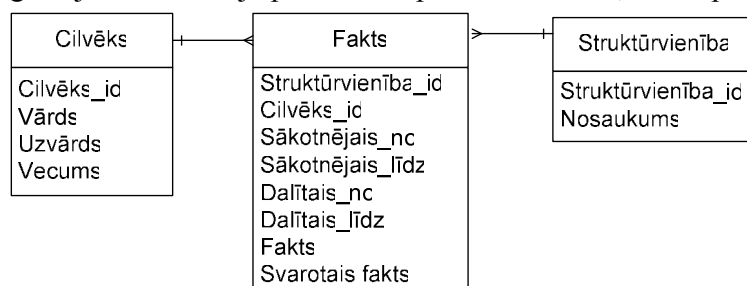
$$f_d = \frac{f}{diff([ft1 :: ft2])} \cdot diff([ts :: tb]) \quad \{2\}$$

Ar *[ts::tb]* tiek apzīmēts laika intervāls, kurā atbilstošais dalītais fakts ir spēkā. Šī intervāla garums vispārīgā gadījumā var būt, sākot no vienas laika dimensijas zemākās granularitātes vienības līdz pat dinamikas intervālā ietilpstošo laika dimensijas zemāko granularitātes vienību skaitam.

6.3.4. Dalīto un svaroto fakti izmantošanas piemērs

Lai demonstrētu dalīto un svaroto fakti izmantošanu, kā piemēru apskatīsim LU datu nolikativu. LU datu nolikativā dalītie un svarotie fakti tiek lietoti, lai veidotu atskaites par darbiniekiem. Piemērs dinamikas atskaitē, kurā tiek izmantoti dalītie un svarotie fakti, ir darbinieku skaita maiņas dinamika pa akadēmiskajiem vai kalendārajiem gadiem. Darba līgums šajā gadījumā ir uzskatāms par lēni mainīgo fakti, jo tas lielā daļā gadījumu ir spēkā vairākus gadus. Šim lēni mainīgajam faktam tiks pielietota fakti svarošana un dalīšana.

Attēlā (skat.72.att.) dots darbinieku datu zvaigznes shēmas modelis ar divām dimensijām, kurās glabājas informācija par cilvēka personas datiem, kā arī par struktūrvienībām.



72.att. Datu struktūras lēni mainīga fakta piemēram

Piemēra fakti tabulā glabājas vienības fakts (Fakts) –ar vērtību ‘1’ un vienības fakta dalījums (Svarotais fakts). Piemēra vienkāršībai fakti tabulā tiek glabāti datumi, nevis ārējās

atslēgas uz laika dimensiju. Pārējie piemēra faktu tabulas lauki ir ārējās atslēgas uz cilvēku un struktūrvienību dimensijām.

No attēlā (skat. 72.att.) redzamā modeļa iespējams iegūt divu tipu atskaites:

- 1) atskaite par stāvokli uz noteiktu datumu,
- 2) dinamikas atskaite par fakta izmaiņām.

Atskaite par stāvokli uz noteiktu datumu

Apskatāmā piemēra gadījumā tā būs atskaite par darbinieku skaitu uz noteiktu datumu. Atskaitē pamatā ir paši lēni mainīgie fakti. Šāda veida atskaites ir momentuzņēmumu tipa atskaites. (13) kodā ir dots piemērs vaicājumam, kas atrod darbinieku skaitu, piemēram, uz 2004. gada 1. jūliju.

```
Select nosaukums, round(sum(svatrotais_fakts)) from Struktūrvienība s, Fakts f
where f.struktūrvienība_id=s.struktūrvienība_id and '01.07.2004' between
(sākotnējais_no and sākotnējais_līdz) group by nosaukums
```

(13)

Dinamikas atskaite par fakta izmaiņām

Otra veida datu noliktavu atskaite, kuru ļauj iegūt šādi veidota faktu tabula, apskatāmajā piemērā ir darbinieku skaits pa struktūrvienībām dinamikā. Tas nozīmē, ka jāiegūst darbinieku skaits uzņēmumā pa struktūrvienībām uz katra gada noteiktu datumu. Kodā (14) ir dots piemērs vaicājumam, kas noskaidro darbinieku skaita dinamiku pa struktūrvienībām katra gada 1. jūlijā.

```
Select nosaukums, sum(fakts), year(dalītais_no) from Struktūrvienība s, Fakts f
where f.struktūrvienība_id=s.struktūrvienība_id and '01.07' between
(day_month(dalītais_no) and day_month(dalītais_līdz)) group by nosaukums,
year(dalītais_no)
```

(14)

Konstruējot šāda veida vaicājumu, ir nepieciešams lietot laika funkcijas. Nepieciešama funkcija `year`, kas no pilna datuma noskaidro gadu, kā arī funkcija `day_month`, kas no pilna datuma noskaidro datumu un mēnesi. Funkcijai `between` arī ir jābūt definētai tā, lai tā strādā ar datumu bez gada. Ja faktu tabulā laika vietā glabāsies ārējās atslēgas uz laika dimensiju, tad skata definīcija būs atšķirīga.

Lietojot faktu dalīšanu, palielinās faktu tabulas apjoms. Tas, cik daļās fakti tiek dalīti, ir atkarīgs no katras konkrētās datu noliktavas. Dati var tikt dalīti pa dinamikas intervāliem gads, mēnesis, diena (iepriekš zinot, kādas dinamikas atskaites būs nepieciešamas). Paši lēni mainīgie fakti var būt spēkā gan dažas stundas, gan dažas dienas, gan arī vairākus gadus. LU datu noliktavas projektā dalot faktus - darbinieku darba līgumus pa dinamikas intervāliem viens gads, ierakstu skaits faktu tabulā palielinājās gandrīz 3 reizes.

6.4. Nodaļas secinājumi

Promocijas darbā tika piedāvāti risinājumi divām problēmām datu noliktavu realizācijā - viena metode ir pieejas kontrolei datu noliktavās, otra metode ir saistīta ar lēni mainīgu faktu glabāšanas un izmantošanas problēmu.

Secinājumi par piedāvāto pieejas kontroles modeli un mehānismu:

- piedāvātais pieejas kontroles modelis piemērots datu noliktavai, ja ir nepieciešams kontrolēt pieejas tiesības pēc datu elementu satura,
- piedāvātais pieejas kontroles mehānisms izmantojams, ja ir liels potenciālo lietotāju skaits, kas nosaka nepieciešamību pēc automatizējama, viegli uzturama risinājuma, kas būtu realizēts datu bāzes līmenī,
- otra situācija, kas nosaka šāda mehānisma lietošanu, ir tad, ja lietotāji datu noliktavai lielā mērā pārklājas ar datu avotu lietotājiem, jo mehānisms piedāvā iespēju izmantot gan datu avotu tiesību ierobežojumus, gan definēt jaunas tiesības, kas specifiskas pašai datu noliktavai.

Rakstā [NG05] ir piedāvāti divi iespējamie pieejas kontroles mehānismi: lomu bāzēts mehānisms un iepriekš promocijas darbā apskatītais dinamisko skatu mehānisms. Lomu bāzētais mehānisms, kā to pierādīja eksperimenti, kas atspoguļoti arī [NG05] rakstā, ir mazāk efektīvs gan uzturēšanas, gan ātrdarbības ziņā nekā dinamisko skatu mehānisms. Tādēļ promocijas darbā tiek piedāvāts tikai modelis, kas realizēts ar dinamiskajiem skatiem. Arī metodes aprobācijas projektā - LU datu noliktavā - realizēts un tiek izmantots promocijas darbā piedāvātais pieejas kontroles modelis, kas realizēts ar dinamiskajiem skatiem.

Secinājumi par piedāvāto metodi lēni mainīgu faktu glabāšanas un izmantošanas problēmu risināšanai:

- metode lietojama lēni mainīgiem faktiem,
- metode lietojama, ja lēni mainīgu faktu gadījumā no vienas un tās pašas faktu tabulas ir nepieciešams veidot gan dinamiskas atskaites par lēni mainīgiem faktiem, gan arī atskaites par objektu kopējo stāvokli uz kādu noteiktu laika momentu,
- metodes ierobežojums - lietojot faktu dalīšanu, palielinās faktu tabulas apjoms. Prognozējot faktu tabulas apjoma pieaugumu, jāņem vērā dinamikas intervāla garums un lēni mainīgo faktu spēkā būšanas intervāla garums, abi šie parametri nosaka, cik dalītajos faktos būs jāsadala lēni mainīgais fakts.

NOBEIGUMS

Promocijas darba ietvaros veiktas literatūras studijas, lai konstatētu situāciju datu noliktavu konceptuālo modeļu jomā. Tika pētītas eksistējošas metodes, kā datu noliktavu konceptuālie modeļi ir iegūstami. Darba autore nonāca pie secinājuma, ka esošās metodes katra ir lietojama noteiktā datu noliktavas konceptuālo modeļu izstrādes situācijā. Nepieciešams situāciju raksturot ar situāciju raksturojošiem faktoriem, lai varētu izvēlēties piemērotu gatavu metodi, vai lai varētu ātri noskaidrot vajadzību pēc jaunas metodes un, izmantojot metožu inženierijas principus, šo jauno metodi arī konstruētu atbilstoši situācijai.

Promocijas darbā, pielietojot metožu inženierijas principus, tiek piedāvātas trīs jaunas metodes, katra no tām pārstāv atšķirīgu pieeju datu noliktavu konceptuālo modeļu izstrādē.

Lietotāju vadītas metodes LVM nepieciešamību nosaka plašais analīzes vajadzību spektrs, daudzie potenciālie lietotāji. Tātad pastāv vajadzība noskaidrot analīzes prioritātes. Nav zināms arī konkrēts analīzes mērķis, piemēram, vajadzība mērīt noteiktu procesu vai dažu saistītu procesu apakškopu. Metode balstās uz intervijām un to apstrādi, intervijas izmanto gan analīzes vajadzību noskaidrošanai, gan apstrādājot interviju atbildes tiek noteiktas prioritātes.

LVM galvenie jauninājumi saistīti ar interviju apstrādi prioritāšu noteikšanai.

Datu vadītas metodes DVM nepieciešamību nosaka tas, ka ir zināms process, ko vajag vērtēt, bet nav zināmi rādītāji, kā procesu mērīt. Risinājums - meklēt iespējamus rādītājus no datiem, noskaidrot, ko iespējams izrēķināt.

Datu vadītām metodēm ir raksturīga pretruna starp datu uzkrāšanu operatīviem mērķiem un to jauno lietošanas veidu datu analīzē. Tādēļ tās netiek uzskatītas par labākajām, kaut arī ir situācijas, kad nav alternatīvas, piemēram, nav citu veidu kā noskaidrot indikatorus.

DVM galvenie jauninājumi ir datu vadītas pieejas vispārināšana vairāku datu avotu integrētam datu modelim un lietotāju skatu definēšana, lai formalizētu iespējamus datu analīzes veidus, kādi izriet no datiem

Mērķu vadītas metodes MVM lietošanu arī nosaka tas, ka zināms process, ko vajag mērīt, bet nav zināmi rādītāji. Šajā situācijā varētu balstīties uz datiem un datu modeļiem. Tā kā datu vadītas pieejas negarantē procesu mērīšanai atbilstošāko modeli un indikatori tiek piedāvāti tādi, kas izriet no datiem, nevis objektīvi nepieciešami, tad atbilstošāks ir cits risinājums - mērķu vadīta pieeja. Mērķu vadītas metodes ir jaunāks virziens datu noliktavu konceptuālo modeļu izstrādē. Tiek iegūts kompānijas mērķiem atbilstošāks rezultāts. Iegūtais abstraktais modelis ir apakškopa no datu avotu modeļiem, klāt var nākt jauni atribūti. Tātad var būt vajadzība pēc jauniem datiem, kas vēl jāuzkrāj.

MVM trūkums, salīdzinot ar datu vadīto metodi, ir mērķu vadītas metodes sarežģītība. MVM pielietoto kā komponentu GQ(I)M metodi, tās ietvaros ir jāveic intervijas, lai noskaidrotu

mērķus. Mērķa vadītas metodes gaitā jāveido abstraktais modelis, jādefinē OCL. MVM metodes beigās hierarhijas tāpat tiek noteiktas no avotu datu modeļiem.

MVM galvenie jauninājumi saistīti ar organizācijas mērķiem visatbilstošāko indikatoru noteikšanu, jo balstās uz mērķu analīzi, izmantojot GQ(IM) metodi. Tiek būvēts abstraktais modelis, modelim atribūti tiek vākti mērķu analīzes laikā. Abstraktais modelis kalpo par pamatu indikatoru formālai definēšanai.

Izstrādātās metodes sekmīgi aprobētas Latvijas Universitātes datu noliktavas projektā, katra no tām atsevišķu datuvju izstrādē un situācijās, ko noteica metodei specifiskie situācijas faktori.

Metodes izvēli katrā konkrētā gadījumā, lai atkārtoti pielietotu, nosaka **situācijas faktori**. Katrai no metodēm, kas piedāvātas promocijas darbā, situācijas faktoru analīze tika dota atbilstošas apakšnodaļas secinājumos (5.3.3., 5.4.3., 5.5.4.). Kopsavilkums un salīdzinājums noskaidrotajiem situācijas faktoriem dažādās metodēs dots nodaļas secinājumos 5.6. apakšnodaļā.

Izstrādājot trīs metodes, konstatēti kopā 11 dažādi situāciju faktori:

- process pazīstamība konkrētajai organizācijai;
- analizējamo (mērāmo) procesu skaits;
- process ir vai nav identificēts (izvēlēts), kurš jāmēra;
- intervējamo skaits,
- analīzes prasību spektrs,
- prasību prioritātes,
- integrējamo datu avotu skaits,
- datu avotu modeļi un to integrēšanas iespējas,
- mērāmie indikatori,
- indikatoru analīzes dimensijas,
- analīzes veids .

Šie faktori nosaka metodes izvēli, kā arī, ja nepieciešams konstruēt jaunu metodi, nosaka metodes komponenta izvēli no eksistējošu metožu komponentiem, to adaptācijas nepieciešamību un veidu vai jauna komponenta izstrādes nepieciešamību.

Promocijas darbā tika piedāvāti arī risinājumi divām problēmām datu noliktavu realizācijā - viena metode ir pieejas kontrolei datu noliktavās, otra metode ir saistīta ar lēni mainīgu faktu glabāšanas un izmantošanas problēmu.

Galvenie secinājumi par piedāvāto **pieejas kontroles modeli un mehānismu**:

- piedāvātais pieejas kontroles modelis piemērots datu noliktavai, ja ir nepieciešams kontrolēt pieejas tiesības pēc datu elementu satura,
- piedāvātais pieejas kontroles mehānisms izmantojams, ja ir liels potenciālo lietotāju skaits, kas nosaka nepieciešamību pēc automatizējama, viegli uzturama risinājuma, kas būtu realizēts datu bāzes līmenī,

Galvenais secinājums par piedāvāto **metodi lēni mainīgu faktu glabāšanas un izmantošanas** problēmu risināšanai:

- metode lietojama, ja lēni mainīgu faktu gadījumā no vienas un tās pašas faktu tabulas ir nepieciešams veidot gan dinamiskas atskaites par lēni mainīgiem faktiem, gan arī atskaites par objektu kopējo stāvokli uz kādu noteiktu laika momentu.

Darbs turpināms pie situācijas faktoru analīzes un noteikšanas. Piemēram, LU datu noliktavas projektā apzināta vēl viena iespējama pieeja jaunu metožu izstrādē - metode, kas balstās uz izstrādājamās datu noliktavas līdzību ar jau eksistējošu datu noliktavu, līdz ar to nepieciešams pētīt, kādās situācijās šī metode lietojama, kādi ir situāciju raksturojošie faktori.

SAĪSINĀJUMI

Saīsinājums, nosaukums	Skaidrojums
ADAPT	Application Design for Analytical Processing Technologies
API	Application Programming Interface
CWM	Common Warehouse Metamodel
DAG	Directed Acyclic Graph
DaWaK	Data Warehousing and Knowledge Discovery (konference)
DBPS	Datu bāzu pārvaldības sistēma
DFM	Dimensional Fact Model
DIS	Data Item Set
DMDW	Design and Management of Data Warehouses (konference)
DOLAP	Data Warehousing and OLAP (konference)
EDUCAUSE	ASV augstskolu bezpeļņas organizācijas nosaukums
ER	Entity - Relationship
EUNIS	European Universities Information Systems (konference)
ETL	Extract, Transform, Load
GQM	Goal Question Measure
GQ(I)M	Goal Question Indicator Measure
IS	Informācijas sistēma
LMS	Learning Management System
LU	Latvijas Universitāte
LUIS	LU Informācijas Sistēma
LAIS	Latvijas Augstskolu Informācijas sistēma
ME/R	Multidimensional Entity/Relationship model
MOF	Meta Object Facility

OCL	Object Constraint Language
OLAP	On-line Analytical Processing
OLTP	On-line Transaction Processing
OMG	Object Management Group
PODS	Principles of Database Systems (konference)
SIGMOD	Special Interest Group on Management of Data (konference)
SPJ	Select-projection-join
UML	Unified Modeling Language
URL	Unified Resource Locator
VLDB	Very Large Data Bases
WebCT	e-studiju vides nosaukums
XMI	XML Metadata Interchange
XML	Extensible markup language

LITERATŪRA

- [AASY97] D. Agrawal, A. El Abbadi, A. Singh, T. Yurek "Efficient View Maintenance at Data Warehouses", *Proc. of the ACM SIGMOD Int. Conf. on Management of data*, Tucson, Arizona, USA, 1997, pp. 417- 427.
- [AF03] D. E. Avison, G. Fitzgerald, "Where now for development methodologies?" *Communications of the ACM* , Vol. 46, No1, 2003, pp. 78- 82.
- [AGS97] R. Agrawal, A. Gupta, S. Sarawagi, "Modeling multidimensional databases", *Proc. of 13th Int. Conf. on data Engineering (ICDE)*, IEEE Press, 1997, pp. 232-243.
- [Art05] J. Artz, "Data Driven vs. Metric Driven Data Warehouse Design", *Encyclopedia of Data Warehousing and Mining*, Idea Group, 2005, pp. 223 – 227.
- [ASS01] A. Abello, J. Samos, F. Saltor, "A Framework for the Classification and Description of Multidimensional Data Models", *Proc. of 12th Int. conf. on Database and Expert Systems Applications (DEXA)*, LNCS 2113, Springer, 2001, pp. 668-677.
- [BCCFP01] A. Bonifati, F. Cattaneo, S. Ceri, A. Fuggetta, S. Paraboschi, "Designing Data Marts for Data Warehouses", *ACM Transactions on Software Engineering and Methodology*, Vol. 10, No. 4, 2001, pp. 452-483.
- [BCDS01] A. Bonifati, F. Casati, U. Dayal, M.-C. Shan, "Warehousing Workflow Data: Challenges and Opportunities", *Proc. of the Int. Conf. on Very Large Data Bases (VLDB2001)*, 2001, pp. 649-652.
- [BCR94] V. Basili, G. Caldeira, H. D. Rombach, "The Goal Question Metric Approach", *In J. Marciniak (ed.), Encyclopedia of Software Engineering*, Wiley, 1994.
- [Ben06] M. Benantar, *Access Control Systems: Security, Identity Management and Trust Models*. Springer Verlag, 2006, p.261.
- [Ber03] J. Berk, "Learning Measurement: It's Not How Much You Train, But How Well", *In: The Elearning Developers Journal*, Vol. 3, 2003, pp. 1-8.
- [Bis02] M. Bishop, *Computer Security: Art and Science*, Pearson Education, 2002, p.1084.
- [BN04] J. Benefelds, L. Niedrite, "Comparison of Approaches in Data Warehouse Development in Financial Services and Higher Education", *ICEIS*, Vol.1, Porto, 2004, pp. 552-557.
- [BM01] P. Brusilovsky, P. Miller, "Course Delivery Systems for the Virtual University", *In: T. Tschang, T. Della Senta, (Eds.): Access to Knowledge: New Information Technologies and the Emergence of the Virtual University*, Elsevier Science, 2001, pp. 167-206.
- [BR88] V. Basili, H.D. Rombach, "The TAME Project: Towards Improvement-Oriented Software Environments", *IEEE Transactions on Software Engineering*, Vol. 14, No. 6 ,1988, pp. 758-773.
- [Bri96] S. Brinkkemper, "Method engineering: engineering of information systems development methods and tools", *Information and Software Technology*, Vol. 38, No 4, 1996, pp. 275- 280.

- [BRK99] M. Bajec, R. Rupnik, M. Krisper, "Using Data Warehouses in University Information Systems", *Proc. of conf. EUNIS99*, Finland, 1999, pp. 115-121.
- [Bru04] P. Brusilovsky, "KnowledgeTree: A Distributed Architecture for Adaptive E-Learning", *In: Proc. of the 13th Intl. World Wide Web Conference (WWW 2004)*, ACM Press, 2004, pp. 104-113.
- [BSHD98] M. Blaschka, C. Sapia, G. Hofling, B. Dinter, "Finding your way through multidimensional data models". *In Proc. of 9th Int. Conf. on Database and Expert Systems Applications (DEXA)*, LNCS 1460, Springer, 1998, pp. 198-203.
- [BS05] E. Bertino, R. Sandhu, "Database Security - Concepts, Approaches, and Challenges", *IEEE Transactions on Dependable and Secure Computing*, Vol. 2, No. 1, January-March , 2005, pp. 2-19.
- [BU99] M. Boehnlein, A. Ulbrich-vom-Ende, "Deriving Initial Data Warehouse Structures from the Conceptual Data Models of the Underlying Operational Information Systems", *Proc. of the ACM 2nd Int. Workshop on Data Warehousing and OLAP, DOLAP'1999*, ACM, 1999, pp. 15-21.
- [BU00] M. Boehnlein, A. Ulbrich-vom-Ende, "Business Process Oriented Development of Data Warehouse Structures", *Proc. of Int. Conf. Data Warehousing 2000*, Physica Verlag, 2000, pp. 3-16.
- [Bul98] D. Bulos, S. Forsman, "Getting Started with ADAPT", whitepaper, Symmetry Corp, San Rafael, 1998.
- [CCS93] E.F. Codd, S.B. Codd, C.T. Salley, "*Providing On-line Analytical Processing to User-Analysts: An IT Mandate*", white paper, E.F. Codd and Associates, 1993.
- [CCY99] C. Chen, M. Cochinwala, E. Yueh, "Dealing with slow-evolving fact: a case study on inventory data warehousing", *In Proc. of the 2nd ACM Int. Workshop on Data Warehousing and OLAP (DOLAP '99)*, ACM Press, 1999, pp. 22-29.
- [CD97] S. Chaudhuri, U. Dayal, "An Overview of Data Warehousing and OLAP Technology", *SIGMOD Records*, Vol.26, No1, 1997, pp. 65-74.
- [Che98] D. Cheung, "Developing a student evaluation instrument for distance teaching", *In: The American Journal of Distance Education*, Vol. 19, No.1, 1998, pp. 23-42.
- [Chen76] P.P-S. Chen. "The Entity Relationship Model – Towards a Unified View of Data", *ACM Transactions on Database Systems (TODS)*, Vol. 1, No. 1, 1976, pp. 9-36.
- [CLF] Apache Http Serwer Log Files, [tiešsaiste]. – [atsauce 10.12.2007.]. Pieejams: <http://httpd.apache.org/docs/logs.html>
- [CT98] L.Cabbibo, R.Torlone, "A logical approach to multidimensional databases", *Proc. of int. conf. Advances in Database Technology - EDBT'98*, LNCS 1377, Springer, 1998, pp. 183-197.
- [CT04] E. Cardoso, M.J. Trigueiros, "Decision Support Systems for University effectiveness", *Proceedings of conference EUNIS'04*, 2004.
- [Des01] J.F. Desnos, "A National Data Warehouse Project for French Universities", *Proc. of conf. EUNIS2001*, Berlin, 2001, pp. 146-152.

- [DR99] G. David, L. Ribeiro, "Getting Management support from an University Information System", *Proc. of conf. EUNIS99*, Finland, 1999, 132-137.
- [EDUC] EDUCAUSE - nonprofit association to advance higher education by promoting the intelligent use of information technology, [tiešsaiste]. – [atsauce 10.12.2007.]. Pieejams: <http://www.educause.edu/>
- [EKK02] J. Eder, C. Koncilia, H. Kogler, "Temporal data warehousing: business cases and solutions", *Proc. of the Int. Conf. on Enterprise Information Systems (ICEIS'02)*, Spain, 2002, pp.81-88.
- [FF99] B. Fitzgerald, G. Fitzgerald, "Categories and contexts of information systems development: making sense of the mess", *In Proc. of European Conf. on Information Systems*, 1999, pp.194-211.
- [FTVP04] E. Ferrandez-Medina, J. Trujillo, R.Villaroel, M. Piattini, "Extending UML for Designing Secure Data Warehouses", *Proc. of Int. Conf. on Conceptual Modeling (ER 2004)*, Springer, 2004, pp. 224–237.
- [FW95] N.E. Fenton, R. Whitty, "Introduction". *Software Quality Assurance and Measurement. A Worldwide Perspective*, International Thomson Computer Press, 1995, pp. 1-19.
- [GBC+06] F. Garcia, M. Bertoa, C. Calero, A. Vallecillo, F. Ruiz, M. Piattini, M. Genero, "Towards a consistent terminology for software measurement", *Information and Software Technology*, Vol. 48, Issue 8, August 2006, pp. 631-644.
- [GM95] A.Gupta, I.S. Mumick, "Maintenance of materialized views: Problems, techniques and applications", *IEEE Data Engineering Bulletin. Special Issue on Materialized Views and Data Warehousing*, Vol.18, No.2, June 1995. pp. 3-18.
- [GMR98a] M. Golfarelli, D. Maio, S. Rizzi: "The Dimensional Fact Model: a Conceptual Model for Data Warehouses", In: *International Journal of Cooperative Information Systems*, Vol. 7, No. 2&3, 1998, pp. 215-247.
- [GMR98b] M. Golfarelli, D. Maio, S. Rizzi, "Conceptual Design of Data Warehouses from E/R Schemes", *Proc. of Hawaii Int. Conf. on System Sciences*, vol. VII, Kona, Hawaii, 1998, pp. 334-343.
- [GN06] L. Grundmane, L. Niedrite, "Splitting facts using weights", *In Proc. of the 8th Int. Confe. of Enterprise Information Systems (ICEIS 2006)*, Paphos, Cyprus, 2006.
- [Goe05] M. Goeken, "Anforderungsmanagement bei der Entwicklung von Data-Warehouse-Systemen. Ein sichtenspezifischer Ansatz", Eds. R. Winter, J.Schelp, *Auf dem Weg zur Integration Factory: Proceedings der DW 2004 - Data Warehousing und EAI*, 2005, pp. 167 - 186.
- [GRG05] P.Giorgini, S.Rizzi, M.Garzetti, "Goal-Oriented Requirement Analysis for Data Warehouse Design", *Proc. of the 8th ACM Int. Workshop DOLAP*, 2005, pp. 47-56.
- [Har91] Harrington, J. H., *Business Process Improvement - The breakthrough strategy for total quality, productivity, and competitiveness*, McGraw-Hill, 1991, p. 274.
- [Har97] F. Harmsen, "Situational method engineering", Dissertation Thesis, University of Twente, Moret Ernst & Young Management Consultants, The Netherlands, 1997.

- [IEEE Std. 610.12-1990] *IEEE Standard Glossary of Software Engineering Terminology*, IEEE, 1990, p. 96.
- [IGG03] C. Imhoff, N. Gallemmo, J.G. Geiger, *Mastering Data Warehouse Design. Relational and Dimensional techniques*, Wiley Publishing, 2003, p. 456.
- [Inm02] W.H. Inmon, *Building the Data Warehouse*, 3rd ed., Wiley Computer Publishing, 2002, p. 428.
- [ISO15939] *ISO/IEC 15939: 2002, Software Engineering – Software Measurement Process*, ISO, 2002.
- [ISOVIM] *ISO VIM, International Vocabulary of Basic and General Terms in Metrology*, ISO, 1993.
- [JLK00] M. Jarke, T. List, J. Koller, "The challenge of process data warehousing", *Proc. of 26th Int. Conf. VLDB 2000*, 2000, pp. 473-483.
- [JLVV03] M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis, *Fundamentals of Data Warehouses*, 2nd Ed., Springer Verlag, 2003, p. 219.
- [Kat99] R.N. Katz, "Competitive Strategies for Higher Education in the Information Age", *Proc. of Conf. EUNIS'99*, Espoo, Finland, 1999.
- [KHRS05] P.Kueng, C. Hagen, M.Rodel, S.Seifert, "Business Process Monitoring & Measurement in a Large Bank: Challenges and Selected Approaches", *Proc. of 16th International Workshop DEXA 2005*, 2005, pp. 955-961 .
- [KM00] R. Kimball, R. Merz, *The Data Webhouse Toolkit: Building the Web-Enabled Data Warehouse*, John Wiley and Sons, Inc., 2000, p. 416.
- [KN96] R.S. Kaplan, D.P. Norton, *The Balanced Scorecard*, Harvard Business School Press, 1996.
- [KO04] C.Kaldeich, J.Oliveira, "Data Warehouse Methodology: A Process Driven Approach", *Proc. of Int. Conf. CAISE*, LNCS, Vol. 3084, 2004, pp. 536 – 549.
- [KPMG02] KPMG Consulting, *Managing Customer Relationships in Higher Education*, KPMG, 2002.
- [KRRT98] R Kimball, L.Reeves, M. Ross, W. Thornthwite, *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses*, John Wiley, 1998, p. 771.
- [KR02] R. Kimball, M. Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, John Wiley, 2002, p. 464.
- [KS96] G. Kotonya, I. Sommerville, "Requirements Engineering With Viewpoints", *Software Engineering Journal*, Vol.11, Issue 1, Jan 1996, pp. 5 - 18.
- [KWL01] P.Kueng, T. Wettstein, B. List, "A Holistic Process Performance Analysis through a Process Data Warehouse", *Proc. of the American Conf. on Information Systems*, 2001, pp. 349-356.
- [KW92] K. Kumar, R. J. Welke, "Method engineering: a proposal for situationspecific methodology construction", *In: Eds.W.W. Cotterman, J.A. Senn, Systems Analysis and Design: A Researc Agenda*. Wiley, 1992, 257-268.

- [LBMS02] B. List, R. M. Bruckner, K. Machaczek, J. Schiefer, "A Comparison of Data Warehouse Development Methodologies. Case Study of the Process Warehouse", *Proc. of Int. conf. DEXA 2002, LNCS 2453*, Springer-Verlag, 2002, pp. 203–215.
- [Leh98] W. Lehner, "Modeling large scale OLAP scenarios", *Advances in Database Technology - EDBT'98*, LNCS 1377, Springer, 1998, pp. 153-167.
- [LK05] B. List, B. Korherr, "A UML 2 Profile for Business Process Modelling", *Proc. of ER 2005 Workshops AOIS, BP-UML, CoMoGIS, eCOMO, and QoIS*, Springer, LNCS Vol. 3770, 2005, pp. 85-96.
- [LM04] B.List, K.Machaczek, "Towards a Corporate Performance Measurement System", *Proc. of the ACM symposium SAC '04*, ACM Press, 2004, pp. 1344-1350.
- [LPD03] X. Lei, C. Pahl, D. Donnellan, "An Evaluation Technique for Content Interaction in Web-based Teaching and Learning Environments", *Proc. of the 3rd IEEE Intl. Conf. on Advanced Learning Technologies (ICALT'03)*, 2003, pp. 294-295.
- [LTS02] S. Lujan-Mora, J. Trujillo, I. Song, "Extending the UML for Multidimensional Modeling", *Proc. of 5th Int. Conf. on the Unified Modeling Language (UML 2002)*, LNCS 2460, Springer, 2002, pp. 290-304.
- [LTV04] S. Lujan-Mora, J. Trujillo, P.Vassiliadis, "Advantages of UML for Multidimensional Modeling", *Proc. of the 6th Int. Conf. on Enterprise Information Systems (ICEIS 2004)*, Vol. 1, ICEIS Press, Porto (Portugal), 2004, pp. 298-305.
- [LVP07] M. Leppanen, K. Valtonen, M. Pulkkinen, "Towards a Contingency Framework for Engineering an EAP Method", *Proceedings of the 30th Information Systems Research Seminar in Scandinavia IRIS2007*, 2007.
- [LVT04] S. Lujan-Mora, P. Vassiliadis, J. Tujillo, "Data Mapping Diagrams for Warehouse design with UML", *In Proc. 23rd Int. Conf. on Conceptual Modeling (ER 2004)*, 2004, pp. 8-12.
- [MCJ+01] J. McGarry, D. Card, C. Jones, B. Layman, E. Clark, J. Dean, F. Hall, *Practical Software Measurement: Objective Information for Decision Makers*, Addison-Wesley, 2001, p. 304.
- [MD04] R. Mazza, V. Dimitrova, "Visualising Student Tracking Data to Support Instructors in Web-Based Distance Education", *In: Proc. of the 13th Intl. World Wide Web conference on Alternate track papers and posters*, ACM, 2004, pp. 154-161.
- [Med05] I. Medvedis, *Pieteikšanās uz studiju kursiem un reģistrēšanās studijām. Darba plūsmas pārstrukturēšanas atskaite*, LU, 2005.
- [MIT] MIT Data Warehouse, [tiešsaiste]. – [atsauce 10.12.2007.]. Pieejams: <http://web.mit.edu/ist/warehouse/>
- [MP04] V. Mahnič, M. Poženel, "Analyzing enrolment and graduation records using a data warehouse", *Proceedings of conference EUNIS'04*, Ljubljana, Slovenia, 2004.
- [MR05] I. Mirbel, J. Ralyte, Situational method engineering: combining assembly-based and roadmap-driven approaches, *Requirements Engineering*, Vol. 11, No.1 , Dec., 2005., pp. 58-78.

- [MT02] E. Medina, J. Trujillo, "Representing Conceptual Multidimensional Properties Using the Common Warehouse Metamodel (CWM)", *In Proc. of the 3rd int. Conf. on Advances in Web-Age information Management*, LNCS 2419, Springer-Verlag, 2002, pp. 259-270.
- [Mue01] M. zur Muehlen, "Process-driven Management Informations Systems - Combining Data Warehouses and Workflow Technology", *Proc. Of the Fourth Int'l Conf. On Electronic Commerce Research (ICECR-4)*, 2001, pp. 550-566.
- [Mun03] J. Munson, *Software Engineering Measurement*, Auerbach Publications, 2003, p. 468.
- [NG05] L. Niedrite, L. Grundmane, "Controlling access to Data Warehouse data within the database", In: Eds. J.Barzdins and A.Caplinskas. *Databases and Information Systems, Selected papers from the 6th International Baltic Conference DB&IS'2004*, Riga, Latvia, IOS Press, FAIA (Frontiers in Artificial Intelligence and Applications), Vol. 118, 2005, pp. 126-140.
- [Nie04] L. Niedrite, "Requirements and Options for Data Warehouses at Universities", *Latvijas Universitātes raksti, 669. sējums "Datorzinātne un Informācijas tehnoloģijas"*, 2004, pp. 117- 125.
- [NN03] L. Niedrite, A. Niedritis, "The Development and Implementation of Unified Information System for Universities in Latvia – a Retrospective View and Conclusions", *Proc. of the 9th Int. Conf. of European University Information Systems*, Amsterdam, 2003, pp. 480 - 483.
- [NSTN07a] Niedrite L., Solodovnikova D., Treimanis M., Niedritis A., „The Development Method for Process-Oriented Data Warehouse”, In: *WSEAS Transactions on Computer Research*, Issue 2, Vol. 2, February 2007, pp. 183 – 190.
- [NSTN07b] Niedrite L., Solodovnikova D., Treimanis M., Niedritis A.: “Goal-Driven Design of a Data Warehouse-Based Business Process Analysis System”, In: *Proceedings of the 6th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering And Data Bases (AIKED '07)*, Corfu, February 2007.
- [OMGa] Object Management Group: UML 2.0 Superstructure, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://www.omg.org/cgi-bin/doc?formal/05-07-04>
- [OMGb] Object Management Group: OCL 2.0 Specification, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://www.omg.org/cgi-bin/doc?ptc/2005-06-06>
- [OMGc] Object Management Group (OMG), *Common Warehouse Metamodel (CWM)*, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://www.omg.org/cgi-bin/doc?ad/2001-02-01>, 2000.
- [OMGd] Object Management Group (OMG), *Software Process Engineering Metamodel Specification (v1.0)*. [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://www.omg.org/cgi-bin/doc?formal/2005-01-06> , 2002.
- [OM04] L. Olsina, M. Martín, Ontology for Software Metrics and Indicators, *Journal of Web Engineering*, Rinton Press, US, Vol.2 No.4, pp. 262-281, 2004.
- [PCTM03] J.Poole, D.Chang, D.Tolbert, D. Mellor, *Common Warehouse Metamodel Developers Guide*, Wiley Publishing, 2003, p. 704.

- [PD02] C.Phipps, K.C.Davis, "Automating data warehouse conceptual schema design and evaluation", *Proc. of the 4th Int. Workshop DMDW'2002*, CEUR-WS.org, Vol. 28., on-line: <http://CEUR-WS.org/Vol-28/>, 2002, pp. 23-32.
- [Ped00] T. B. Pedersen, "Aspects of Data Modeling and Query Processing for Complex Multidimensional Data", PhD Thesis, Faculty of Engineering and Science, Aalborg University, 2000.
- [PGF96] R.E. Park, W.G. Goethert, W.A. Florac, "Goal-Driven Software Measurement – A Guidebook". *Technical Report, CMU/SEI-96-HB-002*, Software Engineering Institute, Carnegie Mellon University, 1996.
- [PMBOK] Project Management Institute, *A Guide to the Project Management Body of Knowledge (PMBOK Guides)*, Project Management Institute, 2004, p. 388.
- [PR95] V. Poe, L. Reeves, *Building a Data Warehouse for Decision Support*, Prentice Hall, 1995, p. 210.
- [PSK01] L.A. Plugge, S. Schoenmakers, P.A. Kirschner, "Electronic environment for management of learning systems", *In: The International Journal of Technologies for the Advancement Knowledge and Learning*, Vol.3, No.1, 2001.
- [PP00] T. Priebe, G. Pernul, "Towards OLAP Security Design – Survey and Research Issues", *Proc. 3rd ACM International Workshop on Data Warehousing and OLAP (DOLAP 2000)*, USA, 2000, pp. 33-40.
- [PP01] T. Priebe, G. Pernul, "A Pragmatic Approach to Conceptual Modeling of OLAP Security", *Proceedings of the 20th international Conference on Conceptual Modeling: Conceptual Modeling* (November 27-30, 2001). LNCS, Vol. 2224, Springer-Verlag, 2001, pp. 311-324.
- [RALT06] S.Rizzi, A.Abelló, J.Lechtenbörger, J.Trujillo, "Research in data warehouse modeling and design: dead or alive?", *Proceedings of the 9th ACM Int. Workshop on Data Warehousing and OLAP (DOLAP '06)*, ACM Press, 2006, pp. 3-10.
- [RDR03] J. Ralyte, R. Deneckere, C. Rolland, "Towards a generic model for situational method engineering", *J. Eder and M. Missikoff (Eds.) Proc. of the 15th Int. Conf. on Advanced Information Systems Engineering (CAISE'03)*, LNCS 2681, Springer-Verlag, 2003, 95-110.
- [Riz03] S. Rizzi, "Open Problems in Data Warehousing: Eight Years Later", In: Eds. H.J. Lenz et.al. *Proc. of the 5th Intl. Workshop DMDW'2003*, CEUR-WS.org, Vol.77, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://CEUR-WS.org/Vol-77/keynote.pdf>
- [RK99] M. Rahkila, M. Karjalainen, "Evaluation of learning in computer based education using log systems", *Proc. of 29th ASEE/IEEE Frontiers in Education Conf.*, 1999, pp. 16-21.
- [Rol97] C. Rolland, "A Primer for Method Engineering", *Proc. of the INFormatique des ORganisations et Syst'emes d'Information et de D'ecision (INFORSID'97)*, Toulouse, France, 1997.
- [RR01] J.Ralyté, C.Rolland, "An approach for method reengineering," *Proc. of the 20th Int. Conf. on Conceptual Modelling (ER2001)*, LNCS 2224, Springer-Verlag, 2001, pp. 471-484.

- [SBHD98] C. Sapia, M. Blaschka, G. Höfling, B. Dinter, "Extending the E/R Model for the Multidimensional Paradigm", *Proc. of Advances in Database Technologies, ER '98 Workshops on Data Warehousing and Data Mining, Mobile Data Access, and Collaborative Work Support and Spatio-Temporal Data Management*, LNCS 1552, Springer, 1998, pp. 105-116.
- [SCHT03] J. Sheard, J.Ceddia, J. Hurst, J. Tuovinen, "Inferring Student Learning Behaviour From Website Interactions: A Usage Analysis", *Education and Information Technologies*, Vol. 8, No.3, Kluwer Academic Publishers, 2003, pp. 245-266.
- [SGG05] A. Silberschatz, P.B. Galvin, G. Gagne, *Operating System Concepts*, 7th Edition, John Wiley & Sons. Inc, 2005, p.921.
- [Smi99] K. Smith-Gratto, "Best Practices and Problems North Carolina A & T State University", Report to the Distance Education Evaluation Task Force Distance Education, 1999, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://qed.ncat.edu/ir&p/report.htm>
- [SN05] D. Solodovnikova, L. Niedrite, "Using Data Warehouse Resources for Assessment of E-Learning Influence on University Processes", In: *Proceedings of the 9th East-European Conference on Advances in Databases and Information Systems*, Tallinn, Estonia , 2005, pp. 233. – 248.
- [SNN01] J. Stonis, L. Niedrite, A. Niedritis, "The influence of the Information System on administrative processes at the university of Latvia", *Proc. of the 7th Int. Conf. of European University Information Systems "The Changing Universities - The Role of Technology"*, Berlin, 2001, pp.119-126.
- [SRM+01] I.-Y.Song, W. Rowen, C. Medsker, E. Ewen, "An Analysis of Many-to-Many Relationships Between Fact and Dimension Tables in Dimensional Modeling", In *Proc. of the Int. Workshop on Design and Management of Data Warehouses (DMDW'2001)*, CEUR-WS, 2001.
- [SS96] K. van Slooten, B. Schoonhoven, "Contingent information systems development", *Journal of Systems and Software*, Vol. 33 , No 2, May., 1996, 153 - 161.
- [Sto03] J.Stonis "Augstākā izglītība kā pakalpojumu tirgus dalībnieks", LU konference, 2003.
- [SV01] D.R. Silva, M.T.P. Vieira, "An ongoing assessment model for distance learning", In: *Proc. of the Intl. Conf. on Internet and Multimedia Systems and Applications (IMSA2001)*, USA , 2001, pp. 17-21.
- [SV02] D.R. Silva, M.T.P. Vieira, "Using Data Warehouse and Data Mining Resources for Ongoing Assessment of Distance Learning", *Proc. of IEEE Intl. Conf. on Advanced Learning Technologies (ICALT 2002)*, 2002, pp. 40-45.
- [TBC99] N.Tryfona, F.Busborg, J.G.B. Christiansen, "StarER: A conceptual model for data warehouse design", *Proc. of ACM 2nd. Int. Workshop on Data warehousing and OLAP (DOLAP)*, USA, 1999, pp. 3-8.
- [TCS03] E. Tittel, M. Chapple, J.M. Stewart, *Certified Information Systems Security Professional Study Guide*, Sybex, 2003, p.783.
- [Tho97] C.R. Thomas, "Information Architecture: the Data Warehouse Foundation", *CAUSE/EFFECT*, Vol. 20, No. 2, Summer 1997, pp. 31-40.

- [Tho00] S. Thorne. "The Data Warehouse: Keeping It Simple", *Educause Quarterly*, No.3, Vol.23, 2000, [tiešsaiste] – [atsauce 10.12.2007.], Pieejams: <http://www.educause.edu/ir/library/pdf/EQM0033.pdf>
- [TPGS01] J. Trujillo, M. Palomar, J. Gomez, I. Song, "Designing Data Warehouses with OO Conceptual Models", *Computer*, Vol. 34, No 12, IEEE Computer Society Press, Dec. 2001, pp. 66-75.
- [Vas00] P.Vassiliadis, "Gulliver in the Land of Data Warehousing: Practical Experiences and Observations of a Researcher", In: Eds. M. Jeusfeld et.al., *Proc. of the 2nd Int. Workshop DMDW'2000*, CEUR-WS.org, Vol. 28., [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: <http://CEUR-WS.org/Vol-28/>
- [Wes01] P. Westerman, *Data Warehousing using the Wal-Mart Model*, Morgan Kaufmann, 2001, p. 297.
- [Web] Webopedia: *On-line encyclopedia dedicated to computer technology*, [tiešsaiste] – [atsauce 10.12.2007.]. Pieejams: http://www.webopedia.com/TERM/e/e_learning.html
- [WFMC99] Workflow Management Coalition, *Terminology and Glossary*, 3rd Edition, Document Nr. WFMC-TC-1011, 1999, p. 65.
- [Wid95] J. Widom, "Research Problems in Data Warehousing", *Proc. of Int. Conf. on Info. and Knowledge management (CIKM)*, 1995, pp. 25-30.
- [WME+01] E. Weippl, O. Mangisengi, W. Essmayr, F. Lichtenberger, W. Winiwarter, "An Authorization Model for Data Warehouses and OLAP", *Proc. of the Workshop on Security in Distributed Data Warehousing*, 2001.
- [WS03] R.Winter, B.Strauch, "A Method for Demand-driven Information Requirements Analysis in Data Warehousing Projects", *Proceedings of the 36th Hawaii Int. Conf. on System Sciences*, IEEE, 2003, p. 231.
- [Zas99] M.R. Zastrocky, "The Information- and Technology- Enabled University of 2004", *Proceedings of conference EUNIS99*, Espoo, Finland, 1999.
- [ZGMW98] Y. Zhuge, H. Garcia-Molina, J.L. Wiener, "Consistency Algorithms for Multi-Source Warehouse View Maintenance", *Distrib. Parallel Databases*, Vol 6, No. 1, Kluwer Academic Publishers, Jan. 1998, pp. 7-40.
- [ZL01] O.R. Zaiane, J. Luo, "Towards Evaluating Learners' Behaviour in a Web-Based Distance Learning Environment", In: *Proc. of Intl. Conf. on Advanced Learning Technologies ICALT'01*, IEEE, 2001, p.357.

PIELIKUMI

1. pielikums Promocijas darba autores publikācijas

Publikācijas apkopotas atsevišķā sējumā, kas pievienots promocijas darbam.

2. Pielikums. Augstskolu datu noliktavas - situācijas raksturojums

Pielikums atspoguļo promocijas darbā veikto pētījumu sākuma posmu, situācijas izpēti, kas kalpoja par motivāciju tālākiem pētījumiem datu noliktavu konceptuālo modeļu izstrādes metožu jomā. Pilnīgāks izklāsts atrodams publikācijā [Nie04], kas ietverts arī atsevišķajā 1. pielikumā.

Augstākās izglītības iestādes, tāpat kā citas organizācijas var runāt gan par to veicamajiem procesiem, gan pārvaldes procesiem. Par galvenajām universitāšu funkcijām var uzskatīt izglītības un zinātnes funkcijas. Par galveniem atbalsta procesiem var uzskatīt pārvaldības procesus, piemēram, cilvēku resursu pārvaldības un finanšu resursu pārvaldības procesus. Aplūkojot tuvāk pārvaldības procesus dažādos organizācijas struktūrvienību hierarhijas ietvaros, var detalizēt to funkcijas, piemēram, aplūkot atsevišķi mācību procesa organizācijas un citas funkcijas.

Pēdējo desmit gadu laikā pieaugusi universitāšu savstarpējā konkurence, palielinoties universitāšu un studiju programmu skaitam. Tajā pašā laikā jāpiekrīt [Kat99], ka universitātes un koledžas reti formulē savas stratēģijas, nolūkus un praksi, izmantojot ar konkurenci saistītus jēdzienus. Tomēr apkārtējās realitātes spiediens uz tradicionālajiem resursiem, ko pastiprina tehnoloģiju bāzētas izglītības piedāvājums, liek domāt par konkurenci.

Līdz ar domāšanas maiņu konkurences virzienā tiek runāts par izglītības procesiem biznesa procesu kontekstā un tādā gadījumā ir jādefinē arī galvenie jēdzieni šajā kontekstā. Universitātēm kā izglītības piegādātājiem jādefinē, kas ir produkts, kādi ir resursi, kāda ir produkta cena. Studentiem kā klientiem arī ir vairāki jautājumi - no kuras universitātes pirkt un tieši kādu produktu. Bez tam klients grib zināt, kāda ir produkta cena un kvalitāte, lai izdarītu izvēli.

Ir daudzi piemēri biznesa terminoloģijas lietošanai augstākajā izglītībā. Plašāk zināmais piemērs ir Fēniksas universitāte, kas fokusējas pieaugušo izglītības jomā. Pastāv tendences lielu korporāciju izglītības iestāžu attīstībā, ka tās kļūst par universitātēm ar akreditētām studiju programmām, piemēram, Motorolas universitāte.

Būtiskākie momenti, kam būtu jāpievērš uzmanība augstskolu vadībai konkurences apstākļos, ir [Sto03]: sabiedriskās attiecības, konkurence, ieskaitot ārzemju augstskolas, tirgus analīze, stratēģiskā plānošana, ieņēmumu/izdevumu analīze, kooperācija ar tradicionālām biznesa organizācijām.

Vienlaikus ar jaunu stratēģiju izstrādi konkurences apstākļos universitātes meklē jaunas metodes, kas tām palīdzētu un atbalstītu arī klientus pareizu lēmumu pieņemšanā. IT būtiski ietekmē, kā institūcijas atrisina problēmas, ar kurām tās saskaras konkurences apstākļos [Zas99]. Viens no līdzekļiem, ko izmanto arī universitātes, cerot, ka tie palīdzēs risināt problēmas, ir datu noliktavas.

Datu noliktavas izglītības sfērā tiek lietotas tieši tādu pašu iemeslu dēļ kā citās biznesa sfērās - padara infomāciju pieejamāku, integrē datus no vairākiem avotiem.

Tālāk tiks apskatīti daži scenāriji datu noliktavas izmantošanai universitātēs, kas sastīti gan ar tradicionāliem augstskolas procesiem (3.un 4. scenārijs), gan ar jauniem biznesa procesiem (1. un 2. scenārijs). Tālāk apskatītie scenāriji daļēji realizēti Eiropas un Amerikas augstskolu universitāšu datu noliktavās [BRK99], [DR99], [Des01], [CT04], [MP04].

1. scenārijs orientēts uz labu studentu piesaistīšanu, datu noliktavas papildina augstskolu informācijas sistēmas, nodrošinot universitātes ar iestāšanās rezultātu analīzi.

Piemēram [BRK99], ja uzņemšanas algoritms balstās uz skolas eksāmenu atzīmēm un izvēlētām studiju programmām prioritāšu secībā, tad informācija par iepriekšējo gadu uzņemšanas procesiem var noderēt, izvēloties programmu atbilstoši reflektanta zināšanu vērtējuma līmenim.

2.scenārijs orientēts uz klientu attiecību pārvaldības atbalstu [KPMG02]. Lai augstskolām būtu iespējams efektīvi strādāt ar dažādu kategoriju studentiem, piemēram, ar dažādām vecuma grupām, ar tālmācības studentiem, ir nepieciešams zināt to vajadzības un prasības, nodrošināt saikni un atbilstošus pakalpojumus studējošajiem visā studenta dzīves cikla garumā.

3. scenārijs orientēts uz studiju procesa uzlabošanu [DR99], piemēram, studiju kursu novērtēšanu, izmantojot dažādus indikatorus, kā piemēram, reģistrēto, novērtēto studentu skaitu, vidējo atzīmi studentu grupai u.c., lai konstatētu potenciālas problēmas, pirms tās iestājas, un saprastu, kāds ir produkts, ko studenti pērk. Vēl iespējams līdzīgi vērtēt arī studiju programmu, izmantojot reflektantu skaitu, studentu "atbirumu", beigušo skaitu u.c. līdzīgus rādītājus.

4. scenārijs orientēts uz resursu plānošanu universitātē, piemēram, plānojot mācību procesu, nosaka nepieciešamo telpu, studentu grupu skaitu uz noteikto kursu, par pamatu ņemot iepriekšējo gadu statistiku. Līdzīgi indikatori palīdz novērtēt arī auditoriju aizpildījumu un pasniedzēja noslodzi. Cits piemērs par cilvēku resursu plānošanu saistīts ar personāla karjeras plānošanu, analizējot darbinieku darba "vēsturi". Finanšu resursu jomā var būt nepieciešams analizēt izdevumus saistībā ar studentu skaitu, pētniecības rezultātiem u.c. indikatoriem. Mazāk acīmredzams piemērs šī scenārija ietvaros ir mērīt augstskolas starptautisko atpazīstamību un tās veicināšanai nepieciešamos ieguldījumus [Des01].

[KPMG02] piedāvā arhitektūru, kas izmantojama arī universitātēs, kurā ietverta arī datu noliktava. Uzņēmuma lietotņu integrācijas slānis nodrošina piekļuvi datiem no dažādām uzņēmuma iekšējām sistēmām. Kā integrācijas līdzeklis tiek izmantota datu noliktava, bet piekļuve datiem tiek realizēta ar dažādiem risinājumiem - portālu, klientu attiecību pārvaldības rīkiem u.c.. Datu noliktava tiek izmantota arī datu analīzei un datizracei.

Esošās datu noliktavu realizācijas universitātēs ir ļoti atšķirīgas gan izvēlēto analizējamo biznesa procesu, gan arhitektūras, gan izmantoto rīku ziņā. Pēc [THO97] datiem gandrīz 25% ASV universitāšu izmanto vai plāno izmantot datu noliktavas. Izmantoto rīku klāsts ir ierobežots, pārsvarā atkarīgs no citos universitātes projektos izmantotām platformām. Populārākā datu bāze -

Oracle, klienta puses risinājumiem - tīmekļa risinājumi. Runājot par biznesa procesiem, tikai reta universitāte izmanto visus esošos datu avotus, lielākais vairums sāk ar studentu datu analīzi. Eiropā nav līdzīga pieredzes apkopojuma, bet par pietiekoši reprezentatīvu informācijas avotu var uzskatīt ikgadējo EUNIS (Eiropas universitāšu informācijas sistēmas) konferenci, kurā katru gadu piedalās vairāk nekā 100 dažādu valstu universitāšu pārstāvji, tomēr pēdējo 7 gadu laikā ziņojumu par datu noliktavām ir bijis maz, piemēram, 1999. un 2004. gadā par Slovēnijas (Ljubļanas) [BRK99], [MP04] un Portugāles (Porto) [DR99] universitāšu datu noliktavām, 2001. gadā par Francijas starpaugstskolu datu noliktavu [Des01].

3. pielikums. LU datu noliktavas raksturojums

Būtiski, uzsākot datu noliktavu projektu, ir novērtēt organizācijas gatavību datu noliktavai [KRRT98]. Uzsākot LU datu noliktavas izveidi, tika veikta datu noliktavas iespējamības izpēte LU, novērtējot vairākus faktoros. Viens no faktoriem ir datu avotu esamība ar pietiekoši ilgu datu uzkrāšanas vēsturi un šo datu kvalitāte. Kā potenciālie datu avoti, uzsākot datu noliktavas projektu, tika apzināti LU Studentu informācijas sistēma LUIS un grāmatvedības sistēma "Apvārsnis". LUIS izstrādāta, izmantojot Oracle RDBPS un *Oracle Application Server*. Studentu dati tiek uzkrāti kopš 1997. gada, studiju maksas kopš 1998. gada, studentu reģistrācija uz kursiem no 2000. gada, reflektantu uzņemšana kopš 1998. gada. Tiesa, datu kvalitātes novērtējumā varēja saskatīt likumsakarību, ka pirmajos datu uzkrāšanas gados datu kvalitāte bija zemāka nekā turpmākajos gados. Otrs faktors datu noliktavas iespējamībai bija nepieciešamība datu analīzei, ko noteica pieaugošā konkurence augstskolu starpā. Jāatzīmē, ka datu izmaiņu vēstures analīzes iespējas datu avotos bija ierobežotas, kā arī nepastāvēja iespēja veidot integrētu datu analīzi, izmantojot vairākus datu avotus.

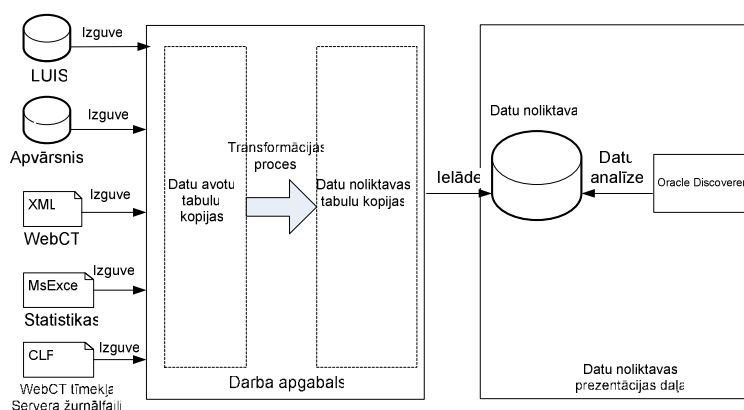
LU datu noliktavu raksturo sekojoši aspekti:

Mērķis: LU datu noliktavas veidošanas mērķis ir analizēt datus arī vēsturiski, nevis tikai konkrētajā brīdī pieejamos, kā tas ir OLTP sistēmās vai statistikas failos. Agrāk LU informācijas sistēmas LUIS ietvaros lietotājiem datus analizēt vēsturiski tikpat kā nebija iespējams, to varēja izdarīt tikai uz pašreizējo dienu. Pēdējās informācijas sistēmas LUIS izmaiņas ļauj veikt nelielu vēsturisku datu analīzi, tomēr LUIS vēsturiskos datus uzkrāj tikai daļēji, nereti izmantojot žurnāla tabulas, tādēļ lietotājiem veikt pilnvērtīgu vēsturisku datu analīzi nav iespējams. Vēsturisko datu trūkums neļauj analizēt informāciju dažādos laika momentos, salīdzināt to. Piemēram, līdz šim nebija iespējams automātiski analizēt informāciju par studentu skaitu fakultātē pa gadiem vai arī apskatīties informāciju par studentiem konkrētā laika momentā pagātnē. Tāpat nebija iespējams salīdzināt LU darbinieku skaitu pa gadiem, kā arī tādu darbinieku skaitus, kas vienlaicīgi strādā gan LU, gan arī institūtos.

Datu avoti, kas iepriekš nebija integrēti: līdz šim izstrādātajās datuvēs galvenie datu avoti ir LU informācijas sistēmas LUIS un Apvārsnis, kā arī dažādi WebCT faili un programmā MsExcel apkopotās statistikas uz kādu noteiktu laika momentu.

Apgrūtināta informācijas pieejamība: piemēram, līdz šim, ja kādam bija nepieciešama izziņa no grāmatvedības, bija jāiet uz grāmatvedību un grāmatvedības darbiniekiem jālūdz izdrukāt nepieciešamo informāciju. Šo informāciju varēja iegūt tikai papīra formātā. Kopš grāmatvedības datuves izveidošanas datu noliktavā, šiem datiem ir iespējams piekļūt daudz ātrāk. Lietotājs var atlasīt interesējošos un nepieciešamos datus.

Datu noliktavas uzbūve, ieskaitot datu avotus, attēlota zīmējumā (skat. 73.att.).



73.att. LU datu noliktavas uzbūve

LU datu noliktavas projekts tika uzsākts **2003. gadā**. Veikta prasību analīze, projektējuma izstrāde un „Darbinieku komandējumu” datuves izstrāde.

2004. gads

Pirmā datu vitrīna “Augstskolas statistika” izveidota no EXCEL formāta statistikas pārskatiem “Augstskola- 1”, kur dati ir pieejami, sākot no 1998. gada (studenti un doktoranti), kā arī darbinieku statistikas, sākot no 1997.gada. Izveidotas atskaites gan tabulu, gan grafiku veidā, salīdzinot pa gadiem dažādus statistikas rādītājus. Šīs atskaites pilna laika studentu datiem ir publiski pieejamas LU portāla kanālā “LU atskaites”. Pārējās atskaites (nepilnam laikam un personāla dati) ir pieejamas autorizētiem lietotājiem.

Otrā datu vitrīna ir “Studentu statistika”, kas tiek rēķināta no pamatdatiem, kur par datu avotu kalpo LUIS. Šeit, atšķirībā no LUIS, statistiku var aprēķināt arī uz jebkuru datumu pagātnē un arī apskatīties, kādi pamatdati veido izrēķināto skaitli. Būtībā ir restaurēta, cik to pieļauj pieejamās informācijas kvalitāte, studentu studiju vēsture – statusu maiņa laika gaitā. Izrēķinātās statistikas, bet bez pieejas pamatdatiem, arī ir publiski pieejamas portālā.

Trešā datu vitrīna “Studentu kustība” apkopo sevī virkni atskaišu, kas saistītas ar studentu pāreju no vienas studiju programmas uz citu, pāreju no viena studiju līmeņa uz citu, finansējuma maiņu, kā arī eksmatrikulācijas statistika pa iemesliem.

Ceturta datu vitrīna “Kontu apgrozījums” ir paredzēta, lai fakultāšu izpilddirektori un citi lietotāji – projektu vadītāji varētu piekļūt grāmatvedības programmas “Apvārsnis” datiem – ir izveidotas atskaites, kur pa projektiem var apskatīt gan summēti, gan detalizēti izdevumus un ieņēmumus.

Piektā datu vitrīna – “E-kursu uzbūve un izmantošana” nodrošina gan e-kursu izstrādes gaitas analīzi (cik un kādas kursa komponentes uz konkrēto datumu ir pievienotas), gan e- kursu izmantošanas analīzi.

2005. gads

Dati par LU Institutu darbiniekiem - statistika, rādītāju dinamika pa gadiem.

E-universitātes kursu izmantošanas datuves 2. versija – detalizēta WebCT lietošana transakciju līmenī, ko raksturo studenta klikšķis uz kursa komponentes (tīmekļa lapas) e-kursā un izmantošanas laika pievienošana. Datuves izveides mērķis – noskaidrot kursa izmantošanu līdz

komponenšu līmenim. Var noteikt klikšķu skaitu uz komponentes, izmantošanas laiku komponentei, sesijas laiku.

Reģistrācija uz semestri un uz studiju kursiem – datuve minēto procesu mērīšanas atbalstam.

Datu noliktavas “izmantošanas” datuve – noslodzes analīzei un populārāko atskaišu noteikšanai, lai uzlabotu ātrdarbību.

2006. gads.

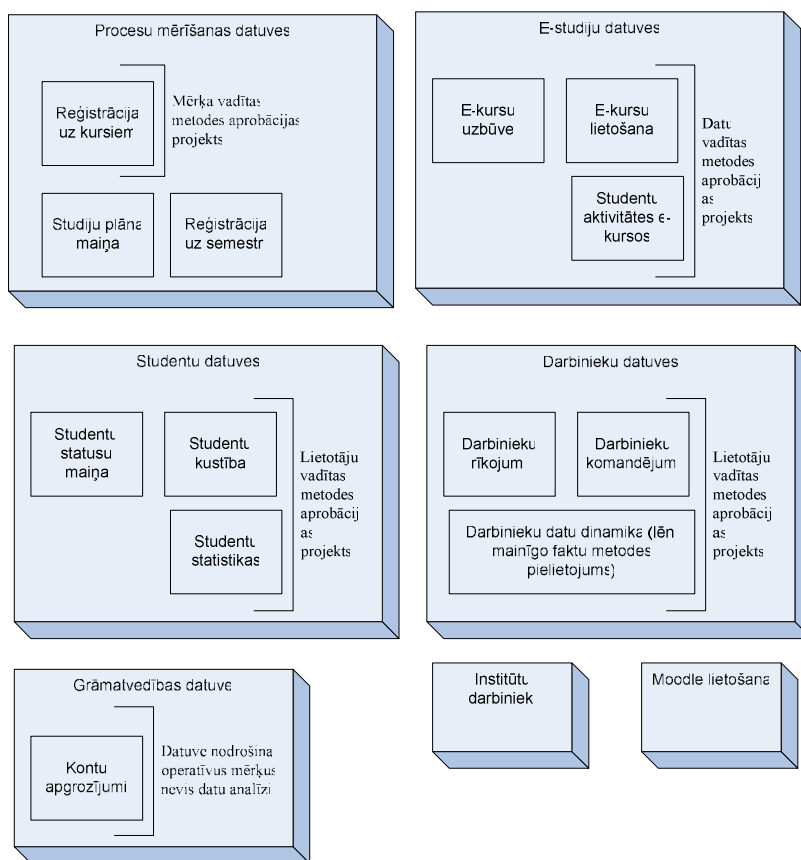
Jaunas datuves netiek izstrādātas, tiek attīstītas e-studiju un procesu mērīšanas datuves, nodrošinot jaunu atskaišu izstrādi.

2007. gads.

E-studiju vides Moodle lietošanas datuve, kas tika veidota, analizējot pieejamo datu avotu līdzību ar jau eksistējošo WebCT lietošanas datuves datu avotiem. Mērķis bija nodrošināt iespēju aprēķināt tādus pašus indikatorus, kas jau bija zināmi un akceptēti iepriekšējās e-studiju vides WebCT lietošanas vērtēšanai.

Jaunas bezlicenču saskarnes izstrādes uzsākšana, lai nodrošinātu plašāku pieeju datu noliktavas datiem.

Kopējo stāvokli datu noliktavā - datuves un to izstrādē pielietotās metodes attēlo zīmējums (skat. 74. att.).



74.att. LU datu noliktavas datuves

4. pielikums. GQ(I)M metodes soļu detalizēts apraksts

GQ(I)M metode [PGF96] ir modificēta GQM [BR88], [BCR94] metode. Metode aprakstīta promocijas darbā 4.4. apakšnodaļā un tās komponenti izmantoti 5.5. apakšnodaļā kā mērķu vadītas metodes komponenti.

Šajā pielikumā detalizētāk aprakstīti GQ(I)M metodes 10 soļi Būtbā pirmie 4 soļi ir sagatavošanās, lai varētu pielietot GQM paradigmu. G(goals – mērķi) šajā paradigmā ir mērījumu mērķi. M – mērījums, kas atbilst mērījumu teorijā lietotai mērījumu definīcijai, kas dota promocijas darbā apakšnodaļā 4.2.

1.solis. Identificē biznesa mērķus un to prioritātes.

2.solis. Identificē, ko vēlas zināt, lai saprastu, novērtētu, prognozētu vai uzlabotu aktivitātes, kas saistītas ar mērķu sasniegšanu.

Katram biznesa mērķim nosaka, kas ir iesaistītās personas vai lomas un katrai no tām uzzīmē atbilstošu procesu modeli. Sastāda katras lomas pārvaldīto procesa komponentu (entītiju) sarakstu, vadoties no iepriekš apskatītajiem procesu elementu 4 veidiem. Katrai entītijai dokumentē jautājumus, kuru atbildes var palīdzēt sasniegt biznesa mērķus.

3.solis. Identificē biznesa apakšmērķus.

Līdzīgus jautājumus katras entītijū-jautājumu sagataves ietvaros grupē, nepievēršot uzmanību, par kādu entītijū jautājums uzdots, bet gan kādai kopējai tēmai jautājumi atbilst. Piemēram, analizē, vai jautājums atbilst procesa pārvaldībai, izmaiņu pārvaldībai, klientu attiecību pārvaldībai. Katrai šādai jautājumu grupai formulē apakšmērķi.

4.solis. Identificē entītijū atribūtus atbilstoši iepriekš noteiktajiem apakšmērķiem.

Tiek attīstīts tālāk sākotnējais procesu modelis. Jautājumi iepriekš tika uzdoti par entītijām un tajos slēptā vai atklātā veidā ir ietverti entītijū atribūti.

5.solis. Formalizē mērījumu mērķus.

Līdz šim solim var runāt par biznesa mērķiem un biznesa procesu analīzi. Tālāk, izmantojot biznesa apakšmērķus un precizēto procesu modeli (galvenokārt interesē atribūti), definē mērījumu mērķus.

Mērījumu mērķa formulējumā parasti ir 4 komponenti:

- intereses objekts – viss process, produkts, vide, kāda konkrēta entītijā, atribūts;
- mērķis - kuru formulē pēc sekojošas shēmas "lai <saprastu | prognozētu | kontrolētu | salīdzinātu ...>";
- perspektīva – kuru formulē pēc sekojošas shēmas: " no < vadības | procesa pārbūves komandas | klienta ...> viedokļa";
- vide – dažāda veida konteksts, kas var vēlāk palīdzēt mērījumu rezultātu interpretēšanai – cilvēkfaktori, resursu faktori, organizācijas struktūra u.c. .

Piemēram: "Analizēt reģistrācijas procesa izmaiņas, lai prognozētu maksimālo studentu pieslēgumu skaitu no procesa pārbūves komandas viedokļa".

6.solis. Identificē jautājumus un tiem atbilstošus indikatorus, kas palīdz atbildēt uz šiem jautājumiem.

GQM paradigma tiek papildināta ar starpposmu I – Indikators, rezultātā iegūst GQ(I)M. Indikators ir veids, kā attēlot lietotājam viena vai vairāku mērījumu rezultātus (būtībā komunikācijas veids).

Darbību algoritms šī soļa realizācijai ir sekojošs: 1)izvēlas vienu mērījumu mērķi, 2)formulē jautājumus, uz kuriem var sniegt skaitliskas atbildes, 3)izstrādā sagataves, kā šo jautājumu atbildes attēlot lietotājam (grafīku veidi, tabulas veida struktūras).

7.solis. Identificē datu elementus (mērījumus), ko nepieciešams vākt, lai varētu konstruēt indikatorus.

8.solis. Precizē mērījumu definīcijas.

Precizēt mērījumu definīcijas nozīmē aprakstīt, tieši kādi dati tiek iekļauti datu elementu vērtību noteikšanā. Piemēram, ja ir datu elements „1. kursa studentu skaits studiju programmā X”, tad nepieciešams precizēt, vai ar to domāts uz 1. kursu reģistrēto studentu skaits (statusi „1. semestris”, „2. semestris”) vai arī jāiekļauj studenti, kam statuss ir „akadēmiskais atvaļinājums”, bet kas paņemts laikā, kad students bija reģistrēts 1. kursam.

Pāreju no indikatoriem uz aprēķināmiem mērījumiem atspoguļo shēma:

9.solis. Identificē darbības, kas jāveic, lai implementētu šos mērījumus.

Galvenais uzdevums ir analizēt un identificēt esošo stāvokli ar datu elementiem - kādi dati ir jau pieejami, kādi nav, kādas darbības jāveic, kādas metodes jālieto, lai datu elementi būtu pieejami, kādi rīki jālieto, lai datus savāktu. Jādefinē, kad un cik bieži dati jāvāc, kas būs datu lietotāji, kā dati tiks analizēti.

10.solis. Izstrādā mērījumu implementēšanas plānu:

a) plānā jāapraksta mērķi, kas pamato mērīšanas procesu un kādi būs mērījumi; b) apraksta biznesa mērķus un mērījumu mērķus, kā arī mērījumus un kā tie saistīti; c) implementācijas plānā jāietver darbības, kas jāveic, lai implementētu nepieciešamos mērījumus, resursus, atbildīgās personas, laika plānojumu.

5. pielikums. Datu vadītas metodes aprobācijas projekta raksturojums

Šajā pielikumā dots:

- metodes aprobācijas projekta problēmu apgabala detalizētāks raksturojums nekā apakšpunktā 5.4.2.1,
- metodes aprobācijas projektā realizētās datuves raksturojums.

Problēmu apgabala detalizēts raksturojums

Metodes aprobācijas projekts saistīts ar e-studiju procesa analīzi. E-studijas ir ar tīklu nodrošināta zināšanu un iemaņu piegāde. E-studijas izmanto elektroniskas lietotnes un procesus, lai mācītos [Web]. Eksistē daudzas kursu pārvaldības sistēmas - *Learning Management Systems* (LMS), kas nodrošina e-studijas, kuru salīdzinājumi un īpašību analīze atrodami daudzu autoru darbos [BM01], [PSK01].

LMS var iedalīt divās grupās. Viena LMS grupa piedāvā standartizētu pieeju gan kursu izstrādei, gan izmantošanai. Otra LMS grupa izmanto mākslīgā intelekta metodes, lai nodrošinātu individualizētu mācību procesu, bet problēma ir šo rīku arhitektūra, kā minēts [Bru04], tā nenodrošina vienlaicīgi gan mācību, gan arī administrēšanas vajadzības.

Standartizēto rīku grupu pārstāv WebCT, kas ir viena no plašāk izmantotajām e-studiju vidēm pasaulē. E-studiju vides kā WebCT ieviešana universitātē ietekmē dažādus procesus – gan pārvaldes procesus, piemēram, studentu reģistrācijas kārtību uz kursiem, pasniedzēju slodzes, gan mācību procesus. Lai nodrošinātu kvalitatīvu mācību procesu, jāzin, kā notiek e-studijas – vai tās pilnībā aizstāj kāda kursa apmācību, vai tās tiek kombinētas ar tradicionālo apmācību. Nav skaidrs, kā šajā gadījumā vērtēt kursā iegūtās zināšanas un mācību procesa kvalitāti.

Tradicionālajā klātienē mācību procesā novērtēšanu iedala formālās iespējās - testi, kontroldarbi, eksāmeni, un neformālās iespējās – pasniedzēja novērojumi, kur ņem vērā piedalīšanos, interesi, ķermeņa valodu. Lai iegūtu priekšstatu par kursa kvalitāti, studentu attieksme pret iegūto pieredzi ir vissvarīgākais rādītājs [Che98], bet attieksmes novērtēšanu veic, izmantojot aptaujas.

Parasti ir grūtības atrast piemērotus rādītājus un metodes, kā vērtēt e-studijas un to ietekmi – var analizēt kvantitatīvus rādītājus, var analizēt kvalitātes aspektus [Smi99], [Ber03].

Tālmācībā e-studiju vidēs pasniedzēja tiešu novērojumu vietā iespējams vērtēt studenta mijiedarbību ar e-studiju vidi, mijiedarbību starp studentiem un pasniedzēju ar vides palīdzību un starp pašiem studentiem. Piemēram, interesi un piedalīšanos var vērtēt ar pieslēgumu skaitu un sūtīto e-pasta vēstuļu skaitu [SV01]. E-studiju vides piedāvā lietotāju darbību reģistrāciju žurnālfailā, ko var izmantot informācijas vākšanai par lietotāja darbībām un to analīzei [RK99].

LMS vidēs parasti ir iebūvēta iekšējā studentu monitorēšanas komponente, kas ļauj pasniedzējam apskatīt dažus statistikas datus - pirmo, pēdējo pieslēgumu, pieslēgumu skaitu, atzīmes uzdevumos, ievietotos ziņojumus. Būtiski ir noskaidrot, kādus no šiem rādītājiem iespējams izmantot studenta mijiedarbības ar e-studiju vidi vērtēšanā un kā tos interpretēt, un vai e-vidē esošie statistikas rādītāji ir pietiekoši, vai jāmeklē citas procesus raksturojošo rādītāju

ieguves iespējas. Ar tām novērtēšanas iespējām, kas ir rīkiem, bieži vien nepietiek. LMS uzkrātos datus nevar izmantot analīzē kopā ar studenta datiem no citiem avotiem, parasti tie ir orientēti uz pasniedzēja skatījumu – statistika tiek atspoguļota par vienu atsevišķu kursu. Izpaliel organizācijas vadības skatījums par LMS izmantošanu kopumā.

Lai novērtētu e-studijas, ir bijuši vairāki pētījumi. Pētījumā [MD04] e-studiju novērtēšanas mērķi bija divi - 1) konstatēt problēmas studentiem e-studijās - nelasa materiālus, daudz laika pavada diskusijās vai otrādi; 2) novērtēt, lai uzlabotu kursus – konstatēt tos kursu materiālus, ko neizmanto, testus, kur nepareizas visas atbildes. E-studiju vide, kura tika izmantota pētījumā, bija WebCT. Rezultātu analīzes atvieglošanai ir piedāvāta papildus komponente ar analīzes rezultātu vizualizācijas iespējām.

Cits pētījums [RK99] iesaka vērtēt e-studijas, lai novērtētu un kontrolētu mācību procesu. Novērtēšanai izmantoja žurnālfailu analīzi (pieslēgšanās laiku, skaitu). Pētījumā izmantotie žurnālfaili nebija tīmekļu serveru žurnālfaili, bet autori apraksta arī iespēju izmantot tīmekļa servera žurnālfailus studentu aktivitāšu pētījumos.

Tīmekļa žurnālfailu analīze ir izmantota darbā [SCHT03] ar mērķi vērtēt kursu izmantošanas efektivitāti, ar efektivitāti saprotot kursa izmantošanas intensitāti, veidu. Raksta autors apgalvo, ka pilnīgu priekšstatu par studenta tiešsaistes mācību procesu var iegūt, kombinējot trīs avotus – tīmekļa žurnālfailus, studenta demogrāfiskos datus un dažādu aptauju rezultātus.

Ir vairāki pētījumi par datizraces izmantošanu e-studiju novērtēšanā, kur kā datu avoti arī tiek izmantoti tīmekļa žurnālfaili. Rakstā [LPD03] analīzes mērķis ir noteikt kursu izmantošanas veidus (*patterns*), [ZL01] novērtē studentu aktivitātes.

Datu noliktavas tradicionāli tiek izmantotas vadības informācijas analīzē [Inm02], [KRRT98], [IGG03], bet tās izmanto arī tīmekļa datu analīzē [KM00]. Tikai vienā no pētījumiem par e-studiju analīzi tiek piedāvāts izmantot datu noliktavas iespējas, kas atspoguļotas vairākos rakstos [SV01], [SV02]. Šeit piedāvātais e-studiju vērtēšanas modelis paredz trīs līmeņus vērtēšanai: 1. līmenī - lietotāju mijiedarbības (ar vidi, savstarpēji), 2. līmenī – vērtējumi iepriekš plānotiem darbiem (uzdevumu, testu), 3. līmenī - datizraces pielietojums. Datu glabāšanai tiek piedāvāts datu noliktavas modelis, bet netiek novērtēta datu ziņā realizācijas iespējamība.

LU projekta mērķis bija piedāvāt veidu, kā analizēt e-studiju vides izmantošanas ietekmi uz dažādiem universitātes procesiem. Piedāvāto risinājumu raksturo faktori:

- izvirzītie mērķi e-studiju analīzei:
 - e-studijās lietotās LMS sistēmas lietošanas vērtēšana – vai studenti un pasniedzēji pietiekami lieto WebCT, ar mērķi noskaidrot tālāku e-kursu izstrādes nepieciešamību, kā arī lai laicīgi noskaidrotu problēmas neaktīvas izmantošanas gadījumā;
 - pasniedzēju aktivitātes vērtēšana, lai noskaidrotu pasniedzēju slodzes pieaugumu;
 - izstrādāto WebCT e-kursu komponentu lietošanas vērtēšana – kurus komponentus izmanto vairāk, vai izmanto WebCT kā LMS specifiskos komponentus vai tikai mācību satura piegādes iespējas;
- orientēšanās uz plaši lietotu standarta LMS, kāds ir WebCT;

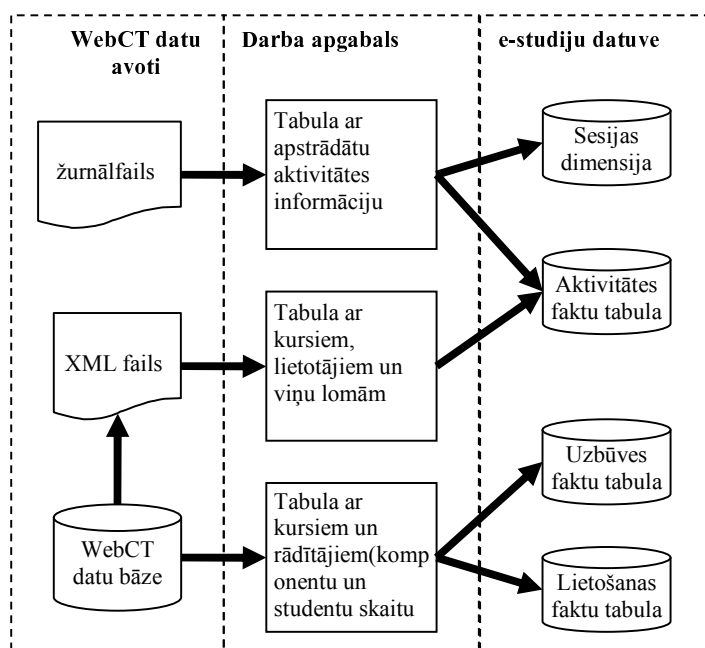
- datu noliktavas kā vides izvēli pamatoja lielais datu apjoms un nepieciešamība integrēt vairākus WebCT datu avotus ar studentu IS datiem analīzes mērķu noskaidrošanai, kā arī nepieciešamība analizēt datus visas universitātes mērogā;

- pieeja datu analīzei – skatu definēšana ar rādītājiem dažādās dimensiju hierarhiju detalizācijas pakāpēs, kas atspoguļo dažādas informācijas vajadzības atbilstoši šo datu izmantotāju biznesa funkcijām.

Datu vadības metodes aprobācijas projekta realizācija

Metodes aprobācijas projektam dati tika savākti par visiem e-kursu studentiem un docētājiem par 2004. gada rudens semestri no 01.09.2004. līdz 30.01.2005. Šajā laikā tika mācīti 274 e-kursi no 402 e-kursiem, kas universitātē tika izstrādāti WebCT vidē. Šos e-kursus izstrādāja un mācīja 213 pasniedzēji. Kopā Latvijas Universitātē 2004. gada rudens semestrī mācījās 29090 studenti. Datu noliktava realizēta *Oracle RDBMS*, kā datu piekļuves rīks tiek izmantots *Oracle Discoverer*, e-studiju datu atjaunošanas periods – vienu reizi nedēļā. Izstrādātā datu veide tiek ekspluatēta 2005., 2006., 2007. gada rudens un pavasara semestros.

Sekojošā zīmējumā (skat. 75.att.) ir attēlots kopējais datu iegūšanas, pārveidošanas un ielādes process no WebCT datu avotiem. Tas atspoguļo tikai WebCT datu avotus, pilnai datu noliktavai izmanto arī datus no LUIS.



75. att. Kopējais ielādes process

Aprobācijas projekta ietvaros datu apjoms dažādās datu struktūrās datu noliktavā ir redzams tabulā (skat.28.tabula).

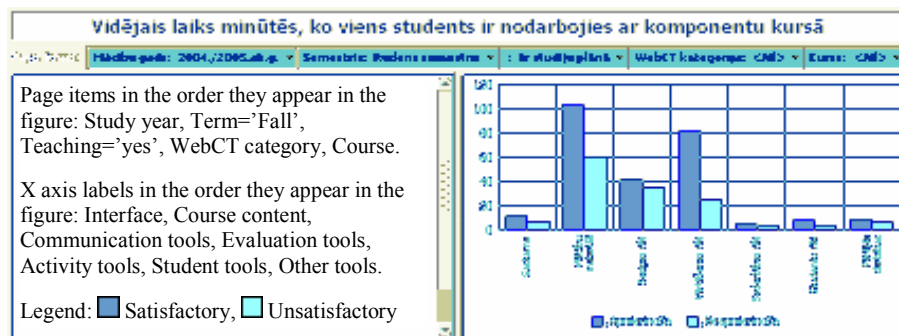
28. tabula

Dimensiju un faktu tabulu datu avoti

Dimensiju/faktu tabula	Datu avoti	Iegūtais ierakstu skaits
Uzbūve	WebCT iekšēja datubāze	110 624
Laiks		
Komponents	Savadīts ar roku	110
Kurss	LUIS, WebCT XML faili, faktu tabulas: Uzbūve, Lietošana	288
Ir Plānā	Savadīts ar roku	2
Lietošana	WebCT iekšēja datubāze	4 774
Aktivitāte	WebCT žurnālfaili, XML faili, LUIS	511 261
Persona	LUIS	4 512
Vērtējums	LUIS	8 584
Loma	WebCT XML faili	4
Programma	LUIS	97
Sesija	WebCT žurnālfaili	96 900

Ekrāna formu paraugi

Ekrāna formu paraugs redzams attēlā (skat. 76.att.), kas atspoguļo salīdzinājumu vidējam laikam minūtēs, ko ar dažādiem kursu komponentiem strādājis sekmīgi kursu nokārtojais un kursu nesekmīgi vai vispār nenokārtojais students. Attēls ir no promocijas darba autores raksta [SN05].



76 . att. Ekrāna formas paraugs e-studiju datuvei

Promocijas darbā piedāvātā datu vadītā metode atspoguļoti rakstā [SN05].

6. pielikums Indikatoru definēšana ar OCL

Pielikums atspoguļo promocijas darbā piedāvātajai Mērķu vadītajai metodei (skat. apakšnodaļu 5.5) metodes rezultātā noteikto indikatoru definīcijas ar OCL. Tabula (skat. 29.tabula) satur pilnā apjomā to indikatoru definīcijas, kuras ietekmē dimensiju modeļa izveidi, šī tabula ir tabulas (24. tabula) pilna versija. Indikatoru semantika apskatāma tabulā (skat.23.tabula).

29.tabula

Indikatoru definēšana ar OCL

I3	context Notional Model::I3():Integer body: Apply -> select(status='Applied' and can enrol='No') -> size()
I5	context Notional Model::I5():Integer body: Course Offering -> select(Apply -> select(status='Applied') -> Student -> asSet() -> size() > places and branch='Language courses') -> asSet() -> size()
I6	context Course Offering::I6():Real body: (Enroll -> select(status='Enrolled') -> Student -> asSet() ->size()) / (Apply -> select(status='Applied') -> Student -> asSet() -> size())
I8	context Notional Model::I8():Integer body: Course Offering -> select(cancelled='Yes') -> size()
I10	context Notional Model::I10():Integer body: Student -> select(financial debt='Yes') -> size()
I11	context Notional Model::I11():Integer body: Student -> select(academic debt='Yes') -> size()
I12	context Notional Model::I12():Set(Course Offering) body: Course Offering
I17	context Course Offering::I17():Integer body: Apply -> select(Study Advisor -> isEmpty()) -> Student -> asSet() -> select(study type='Part-time') -> size()
I18	context Notional Model::I18():Integer body: Apply.Student -> asset() -> size()
I19	context Notional Model::I19():Integer Apply.Student -> asSet() -> select(study type='Part-time') -> size()
I22	context Course Offering::I22():Integer body: Enrol -> select(status='Enrolled' and Study Advisor -> notEmpty()).Student -> asSet() -> size()
I26	context Course Offering::I26():Integer body: Cancel -> size()
I27	context Course Offering::I27():Integer body: Enrol -> size()
I28	context Course Offering::I28():Integer body: Cancel Enrolment -> size()
I30	context Notional Model::I30(f:String):Real body: E-course -> select(department=f).investment -> sum()
I31	context Notional Model::I31(f:String, t:Date):Integer body: E-course -> select(department=f).Register -> select(timestamp=t) -> size()
I32	context Notional Model::I32(f:String, t:Date):Integer body: E-course -> select(department=f).Register -> select(timestamp=t) -> E-course Student -> asSet() -> size()

Indikatoru definēšana ar OCL

I34	context Notional Model::I34(f:String, t:Date):Integer body: E-course -> select(department=f).Study -> select(timestamp=t) -> size()
I35	context Notional Model::I35(f:String, t:Date):Integer body: E-course -> select(department=f).Study -> select(timestamp=t).time spent -> sum()
I36	context Notional Model::I36(f:String, t:Date):Integer body: Study -> select(Tool.E-course.department=f and timestamp=t).session -> asSet() -> size()

7. pielikums. Mērķu vadītas metodes aprobācijas projekta raksturojums

Promocijas darba 5.5. apakšnodaļā piedāvātā mērķu vadītā metode pielietota Procesu mērīšanas sistēmas izstrādē, kura balstās uz šim nolūkam izstrādātu datu noliktavu. Mērāmais biznesa process ir „Studentu reģistrācija uz kursiem”. Šajā pielikumā doti procesu mērīšanas rezultāti studentu reģistrācijai uz kursiem.

Datu vākšana

Mērķu vadītas metodes aprobācijas projektā, pielietojot šo metodi, tika noskaidroti mērāmie indikatori. Pēc tam LUIS procedūrās, kas realizē mērāmo procesu, tika iestrādāta iespēja, ka katrs procedūras izsaukums tiek fiksēts žurnālfailā (skat. promocijas darba punktu 5.5.3.), kas īpaši izstrādāts procesu mērīšanas nolūkam (ar datu avotu izstrādātājiem saskaņots žurnālfaila formāts un uzkrājamās darbības - lai būtu iespējams no uzkrātajiem datiem aprēķināt interesējošos indikatorus).

Procesa mērīšanas perioda beigās uzkrātie dati tika ielādēti datu noliktavas datu struktūrās, kas izstrādātas atbilstoši dimensiju modelim, kas iegūts pielietojot promocijas darbā piedāvāto mērķu vadīto metodi.

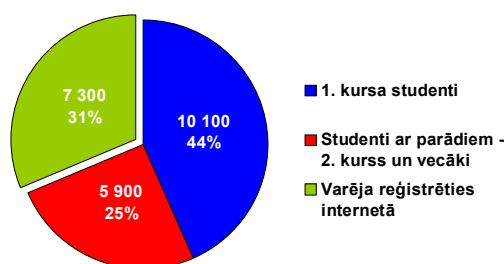
Pēc tam tiek izstrādātas atskaites un analizēti iegūtie rezultāti.

Procesa mērīšanas rezultāti

Reģistrācija uz kursiem tika analizēta vairākas reizes, šajā pielikumā izmantoti dati par divām procesu mērīšanām: 2005. gada rudens semestrī (29.08.2005 – 11.09.2005) un 2006. gada pavasara semestrī (31.01.2006. - 20.02.2006.); iegūtie rezultāti 2. reizē tika salīdzināti ar pirmo mērījumu reizi un varēja spriest par procesa izmaiņām. Tika analizēti gan operatīvie indikatori, gan mērījumi procesa beigās.

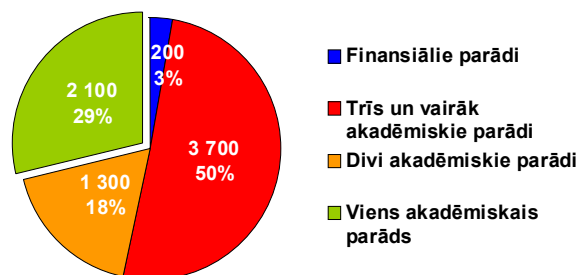
Galvenie secinājumi 2005. gadā par operatīvajiem indikatoriem - serveru noslodze (viena studenta reģistrācija uz kursu) bija vienmērīga, izņemot pirmo reģistrācijas dienu; vietu trūkums populārosursos – datus varēja izmantot papildus kursu plānošanai – izmaiņas kursu piedāvājumā notika visu reģistrācijas periodu.

Bet būtiski bija dati, kas tika iegūti no datu noliktavas pēc reģistrācijas procesa beigām, lai varētu veikt izmaiņas procesa organizācijā. Tika analizēta mērķauditorija – studenti. 1. kursa studentiem ir īpaša situācija – rudens semestrī nav iespējami akadēmiskie parādi, tādēļ tiek izdalīti īpaši (10 100 jeb 44%). Interesēja pārējo studentu sadalījums: 7300 jeb 31% teorētiski varēja reģistrēties internetā, 5900 jeb 25% nevarēja izmantot šo pakalpojumu parādu dēļ (skat. 77.att.). Tā kā mērķis ir palielināt studentu skaitu, kam šis pakalpojums ir pieejams, tad interesē divi aspekti, cik no minētajiem 31% reāli izmantoja un kā uzlabot situāciju ar parādnikiem (kāds ir precīzāks parādnieka raksturojums).



77.att. Procesa mērķauditorijas analīze

Sadalījums šiem 31% potenciāli iespējamiem procesa dalībniekiem tālāk bija 70% pret 30% (reāli izmantoja/neizmantoja reģistrāciju internetā). Attiecība labāka bija klātienē studentiem (79% pret 21%) un sliktāka neklātienē studentu vidū (53% pret 47%). Analizējot parādu veidus, kļuva skaidrs, ka lielāko daļu veido akadēmiskie parādnīki (tikai 3% bija finansu parādi), pie tam lielu daļu veido studenti ar 1 parādu – 29%, ar 2 parādiem 18%, ar 3 un vairāk 50% (skat. 78.att.).



78.att. Parādnīku raksturojums

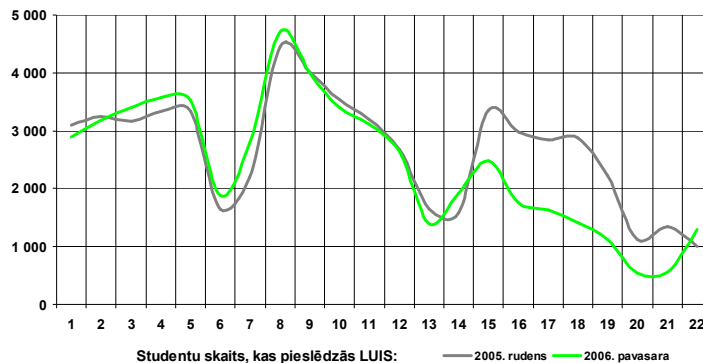
Analizējot reģistrācijas procesu, atklājās likumsakarība, ka daudzi studenti pieteicās uz kursiem procesa beigū posmā, jo kursu piedāvājumi studiju programmās tika mainīti, un ne tikai atbilstoši operatīvajiem datiem, t.i.ursos ar lielu pieprasījumu, bet bieži vienkārši tika pievienoti jauni kursi. Rezultātā studenti apstiprināja savu izvēli vēl, tā neļaujot citiem, pēc sekmēm vājākiem studentiem, laicīgi izdarīt savu izvēli.

Par izvēles apstiprināšanu var teikt, ka ļoti liela daļa studentu līdz galam nesaprata procesa uzbūvi, t.i., ka beigās izvēle kopumā vēl jāapstiprina.

Kā secinājumus no izdarītās analīzes var minēt, ka iesaistīto studentu īpatsvars ir augsts – ap 70%, bet lai to palielinātu, jāstrādā vakara un neklātienē nodaļās, kā arī dažās fakultātēs, kā arī varētu pieļaut studentu ar parādiem, piemēram, vienu akadēmisko parādu, piedalīšanos reģistrācijā. Vēl jāpanāk labāka termiņu ievērošana, piemēram, kursu piedāvājuma veidošanā, kā arī jāveic plašāks procesa izskaidrošanas darbs, lai tiktu līdz galam izpildīti visi procesa soļi.

Otrajā mērījumu reizē 2006. gada pavasarī, dati tika analizēti salīdzinājumā ar 2005. gada rudens semestra datiem.

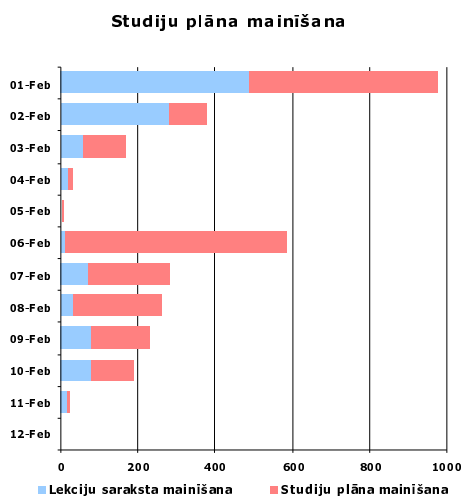
Serveru noslodze pa reģistrācijas perioda dienām (studentu aktivitāte) nav mainījusies, salīdzinot abus mērīšanas periodus 2005. un 2006. gadā (skat. 79.att.).



79. att. Studentu aktivitāte reģistrācijas procesā

Parādu veidi studentiem – studentu parādu skaits būtiski nav mainījies. Vērtējot pieteikumus uz izvēles kursiem, pieteikšanas aktivitāte ir nebūtiski palielinājusies. Pozitīva tendence vērojama studentu darbībās - pieteikšanās uz kursiem 2006. gadā noritējusi savlaicīgāk, neatstājot, piemēram, izvēles apstiprināšanu uz perioda beigām, tas nozīmē, ka procesa izpratne uzlabojusies.

Kā negatīvi vērtējama ir parādība, ka kursu piedāvājums joprojām (2006. gada pavasara semestra sākums) tiek mainīts reģistrēšanās procesa laikā (skat. 80.att.) .



80. att. Studiju plāna izmaiņas mērīšanas perioda laikā 2006. gadā

Kopumā izmaiņas rezultātos, salīdzinot procesa mērīšanas periodus 2005. un 2006. gadā, var vērtēt kā nelielas, atstājot iespējas vēl ievērojami uzlabot procesu.

Par procesu mērīšanas rezultātiem ziņots gan E-universitātes projekta uzraudzības padomes sēdēs, gan nosūtīti LU vadības un departamentu pārstāvjiem, gan ziņots 64. LU zinātniskajā konferencē informācijas tehnoloģiju sekcijā 2006. gada februārī.