

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

**DARĪJUMU INTELIGENCES SISTĒMA FINANŠU
ATSKAITĒM FINREP FORMĀTĀ**

MAĢISTRA DARBS

Autore: **Olga Medvedeva**

Studenta apliecības Nr.: om10009

Darba vadītājs: Dr.dat. Laila Niedrīte

RĪGA 2017

ANOTĀCIJA

Maģistra darbā ir izskatītas FINREP (Financial Reporting) finanšu datu atskaišu standarta pielāgošanas iespējas darījumu inteligences sistēmām. FINREP standartu izstrādāja Eiropas Banku iestāde, to izmantošana ir reglamentēta Eiropas Savienības kredītiestādēm, kas atbilst noteiktām prasībām.

Maģistra darba mērķis ir izpētīt FINREP finanšu atskaišu veidošanas iespējas Oracle Hyperion Financial Management maksas rīkā, kā arī izpētīt pieejamos atvērtā koda darījumu inteligences risinājumus. Izvēloties vienu no atvērtā koda sistēmām, izpētīt to pielāgošanas iespējas FINREP atskaitēm, secinot vai ir iespējams panākt līdzīgo funkcionalitāti kā Oracle Hyperion rīkā.

ATSLĒGVĀRDI: darījumu inteligence, FINREP standarts, Oracle Hyperion, Pentaho, OLAP

ABSTRACT

Master's thesis: Business intelligence system for financial reports in FINREP format

Master thesis explores the possibilities of development of FINREP financial reporting standard in business intelligence systems. FINREP standard is developed by the European Banking Authority, and its use is indicated for the credit institutions that meet the requirements set.

Master thesis goal is to investigate FINREP financial reports development possibilities in the commercial Oracle Hyperion Financial Management tool, as well as to explore the available open source business intelligence solutions. Choosing one of the open-source systems, explore customization options for FINREP reports, concluding whether it is possible to achieve similar functionality as in Oracle Hyperion tool.

KEYWORDS: business intelligence, FINREP standard, Oracle Hyperion, Pentaho, OLAP

AUTOREFERĀTS

Darba ietvaros autore izpētīja un aprakstīja FINREP finanšu datu atskaišu izstrādes iespējas komerciālajā rīkā Oracle Hyperion Financial Management, kā arī atvērtajā koda rīkā Pentaho, ar mērķi secināt vai atvērtā koda risinājums ir salīdzināms ar komerciālo rīku.

Tika izpētīts vai dotais standarts ir aktuāls Latvijā, kā arī kādi līdzīgie risinājumi pastāv un ir pieejami. Tika secināts ka divas organizācijas piedāvā tirgū gatavus risinājumus Hyperion rīkā, toties gatava atvērtā koda bezmaksas risinājuma tādām vai līdzīgam standartam nepastāv. Dažas kompānijas piedāvā arī risinājumus bāzētus uz Microsoft Excel, bet tie ir vērsti uz XBRL datnes veidošanu, nevis datu analīzi ar darījumu inteliģences sistēmu palīdzību.

Darba kopumā tika izpētīti 56 literatūras avoti, kas sastāv no 49 interneta avotiem, 6 grāmatām un 1 raksta. Visiem izmantotajiem literatūras avotiem ir norādīta atsauce. Papildus atsauces uz tīmekļa resursiem ir ievietotas darba tekstā kā zemsvītras atsauces, kur ir definēta papildus informācija, kas varētu būt interesanta lasītājam.

Literatūras analīzes mērķis bija izpētīt problēmas aktualitāti, pastāvošus risinājumus, populārākos darījumu inteliģences rīkus starp pieejamajiem, kā arī iepazīties ar datu noliktavu izstrādes specifiku katrā no izskatāmajiem rīkiem. Apskatāmo rīku popularitāte tika secināta no pieejamajiem informācijas tehnoloģiju tirgu pētījumiem par korporatīvas veiktspējas vadības un darījumu inteliģences rīkiem.

Darba ietvaros bija aprakstīta datu noliktavu izveide, un to pamatojums, priekš abiem darījumu inteliģences rīkiem. Atvērtā koda risinājumam ir norādīti trūkumi, kā arī ir piedāvātas idejas to iespējamajam risinājumam. Maģistra darba izstrādei tika veltīti aptuveni četri mēneši.

Darba teksta ievades un pareizrakstības kļūdas ir pārbaudītas ar Microsoft Word teksta programmu.

SATURS

Apzīmējumu saraksts	7
Ievads	8
1. FINREP atskaites un eksistējošie risinājumi.....	10
2. Darījuma inteliģences rīku izvēle.....	14
3. Izvēlēto darījumu inteliģences rīku teorētiskais apskats	17
3.1. Hyperion rika funkcionalitātes teorētiskais apskats	17
3.1.1. HFM konfigurēšanas elementi.....	18
3.1.2. HFM dimensijas.....	19
3.1.3. HFM datu attēlošanas iespējas.....	21
3.1.4. Datu un lietotāju darbību audits	22
3.1.5. Datu ievade sistēmā	22
3.1.6. HFM procesu kontrole	22
3.2. Atvērtā koda rīku teorētiskais apskats	23
3.2.1. JasperSoft Community Edition	23
3.2.2. Pentaho Community Edition.....	24
3.2.3. Palo (Jedox Base).....	25
4. Eksperimenta realizācija izvēlēto darījumu inteliģences rīkos	27
4.1. Eksperimenta realizācija HFM rīkā	27
4.1.1. HFM rīka instalācija	27
4.1.2. HFM lietotnes izveidošana	31
4.1.3. HFM metadatu definēšana	32
4.1.3.1. “Account” dimensijas definēšana.....	34
4.1.3.2. “Custom1” dimensijas definēšana.....	41
4.1.3.3. “Custom2” un “Custom3” dimensijas definēšana	43
4.1.3.4. “Entity” dimensijas definēšana.....	43
4.1.3.5. “Scenario” dimensijas definēšana	44
4.1.3.6. “Currency” definēšana	45

4.1.4.HFM datu ievades formas	45
4.1.5.HFM noteikumu definēšana.....	46
4.2. Eksperimenta realizācija Pentaho Community Edition rīkā.....	48
4.2.1. Pentaho CE rīka instalācija	49
4.2.2. Datu noliktavas fiziskās shēmas izveide	52
4.2.3. Datu noliktavas Mondrian shēmas izveide	55
4.2.4. Datu atskaišu veidošana.....	61
Rezultāti un secinājumi	64
Izmantotā literatūra un avoti	67
Pielikumi	72

APZĪMĒJUMU SARAKSTS

Atvērtā koda programmatūrā - programmatūra, kuras pirmkodu ir iespējams atklāti izmantot gan izstrādātājiem, gan lietotājiem.

CPM (ang. „Corporate performance management”, vai citādi arī EPM ang. ”Enterprise Performance Management”) programmatūra – rīku komplekss, kas atbalsta biznesa procesus (parasti tas ir – plānošanas, konsolidācijas, atskaišu un finanšu datu pārvaldību procesi).

Konsolidācija - uzņēmumu kāda perioda pārskatu datu apvienošana.

COREP – Eiropas Centrālās Bankas komisijas izstrādāts finanšu atskaišu standarts.

ERP (Enterprise resource planning – Organizācijas Resursu Plānošana) – programmu sistēma, kas atbalsta organizācijas galvenos procesus un palīdz pārvaldīt tos centralizēti.

Eiropas Banku iestāde (ang. EBA) - neatkarīga ES iestāde, kas nodrošina efektīvu un atbilstīgu uzraudzības regulējuma Eiropas banku sektorā.

XML (ang. eXtensible Markup Language) – iezīmēšanas valodas standarts.

XBRL (ang eXtensible Business Reporting Language) – XML bāzēta iezīmēšanas valoda, finanšu datu drošai apmaiņai.

OLAP (Online analytical processing) - tiešsaistes analītiskā datu apstrādes tehnoloģija.

ETL (Extract Tranform Load) – datu integrācija no citiem avotiem - atlase, pārveide un augšupielāde.

Kubs – daudzdimensiju datu masīvs.

Dimensija – datu kuba strukturālais elements, kas pieder vienai kategorijai.

MDX (Multidimensional Expressions) – vaicājumu valoda priekš datu atlasēm no daudz dimensiju datu noliktavām.

SQL (structured query language) – vaicājumu valoda priekš datu atlasēm un modifikācijām relāciju datubāzēs.

Internet Information Services – tīmekļa serveris integrēts Windows NT operētājsistēmās.

IEVADS

Mūsdienas vairākām kompānijām, pārvaldot savas finanšu aktivitātes, radās problēma, saistīta ar to, ka ir daudz datu, bet maz informācijas, ko var izsecināt no šiem datiem - jo lielāka ir kompānija, jo lielāka paliek šī problēma. Darījumu inteliģences rīki palīdz kombinēt produktus, tehnoloģijas un metodes, lai pārvērstu datus par informāciju, ko varētu izmantot dziļākai analīzei un kas var palīdzēt uzlabot kompānijas sniegumu. Veidojot dažādas atskaites ar biznesa inteliģences rīku palīdzību var kvalitatīvi redzēt tendences finanšu datos un izdarīt vajadzīgos secinājumus. [1]

Nepieciešamība veidot finanšu atskaites var būt atkarīga ne tikai no kompānijas iekšējiem noteikumiem, bet arī no ārējiem faktoriem, piemēram, prasības no finanšu audita organizācijām, vai likumā noteikti standarti. Viens no tādiem standartiem ir FINREP (Financial Reporting), kas ir finanšu datu atskaites standarts, ko izstrādājusi Eiropas Banku Iestāde. Šī standarta galvenais uzdevums ir panākt saskaņotus finanšu atskaišu sagatavošanas noteikumus kredītiestādēm. FINREP standarts sastāv no 64 tabulām un vairāk nekā tūkstoš validācijām, kas pārbauda tabulu šūnu vērtības. Noteiktām kredītiestādēm šādas atskaites katru ceturksni ir jāiesniedz regulējošai iestādei (piemēram, valsts Centrālajai Bankai) vienotā XBRL formātā.

Vairākas kompānijas tirgū piedāvā Microsoft Excel risinājumus šādu atskaišu noformēšanai. Šāda pieeja var apgrūtināt raportējošām organizācijām pareizo vēsturisko datu pārvaldi un analīzi, jo dati nav saglabāti caurskatāmā formātā. Šī iemesla dēļ dažas kompānijas piedāvā risinājumus ar informāciju sistēmu izmantošanu, lai dati būtu vieglāk sagatavojami atskaitēm. Brīvi pieejamas informācijas par šādiem risinājumiem internetā nav publiskas daudz, jo visi šie rīki ir par maksu. Atvērtā koda vai bezmaksas risinājumi šādam nolūkam internetā nav atrodam.

Informācijas sistēmas izmantošana šādām finanšu atskaitēm atvieglo pārskatu sagatavošanu, padarot datus caurskatāmākus, kā arī atvieglojot datu konsolidāciju, analīzi, validāciju un vēsturisko datu uzturēšanu. Kamēr lielām kredītiestādēm ir iespēja izmantot dārgas sistēmas, nelielām organizācijām varētu būt interesanta atvērtā koda bezmaksas sistēmas ieviešana un adaptēšana.

Lielas kompānijas, kuru ienākumi pieļauj ievest dārgas sistēmas, liek uzsvāru uz lielām sistēmām, lai standartizētu procesus, kas notiek visos uzņēmuma departamentos, izmantojot vienu sistēmu datu ievadīšanai, glabāšanai un apstrādei. Vienas sistēmas izmantošana palīdz arī

saglābāt atbilstību reglamentējošām prasībām [2]. Pārsvārā šiem nolūkiem tiek izmantoti lieli un dārgi korporatīvās veiktspējas vadības darījumu lietojumprogrammu kompleksi (CPM – ang. „Corporate performance management” vai EPM ang. „Enterprise Performance Management”), kas palīdz sekot organizācijas datiem attiecībā uz tādiem faktoriem, kā ieņēmumi, ieguldījumi, darba izmaksas, kā arī tādiem organizācijas procesiem, kā finanšu datu atskaite, plānošana, uzskaitē.[3] Viena no tādām sistēmām ir Oracle Hyperion, kurā ir iekļauts arī Hyperion Financial Management – finanšu konsolidācijas un atskaišu rīks. Tas nodrošina datu uzticamību un caurskatāmību, kā arī iespēju ātri konsolidēt un nodot finanšu atskaites, uzturēt datus pēc normatīvajām prasībām [4].

Faktori, kāpēc iestādes var izvēlēties ieviest risinājumus, kas ir bāzēti uz bezmaksas sistēmām, var būt dažādi, sākot no kompānijas politikas attiecībā uz maksas rīku izmantošanu, pārvaldīto un uzkrāto datu apjomu un beidzot ar kompānijas ienākumiem, naudas apgrozījumu, kas var neļaut dārgas sistēmas pirkšanu (piemēram, Oracle Hyperion ERP sistēmas cena uz 2016.gadu sākas no 14 445 USD, un to atsevišķas daļas Financial Management – 5 200 USD [19]).

Maģistra darba mērķis ir parādīt finanšu atskaites veidošanas, konsolidācijas un apstrādes iespējas komerciālajā Oracle Hyperion Financial Management rīkā, izmantojot FINREP datu atskaites standartu, kā arī izpētīt, kādi atvērtā koda darījuma inteliģences rīki ir pieejami tādām uzdevumiem. Izvēloties vienu no atvērtā koda rīkiem, pielāgojot to FINREP finanšu datu atskaitēm, izsecināt, vai ir iespējams panākt līdzīgu funkcionalitāti kā komerciālajā finanšu konsolidācijas rīkā - Hyperion.

Mērķa sasniegšanai ir noteikti šādi uzdevumi:

1. Izpētīt situāciju tirgū attiecībā uz piedāvātiem FINREP atskaites uzturēšanas risinājumiem.
2. Izmantojot literatūru un autores empīrisko pieredzi aprakstīt komerciālas Oracle Hyperion Financial Management (HFM) sistēmas iespējas.
3. Pielāgot Oracle HFM sistēmu FINREP atskaites standartam.
4. Izmantojot literatūru un pieejamos informācijas tehnoloģiju tirgu pētījumus (ko veic tādas organizācijas kā Gartner, Butler Analytics un citi) apskatīt populārākus atvērtā koda darījumu inteliģences risinājumus un to iespējas.
5. Izvēlēties vienu no atvērtā koda risinājumiem un pielāgot to atskaitēm FINREP standartā.

1. FINREP ATSKAITES UN EKSISTĒJOŠIE RISINĀJUMI

Noteikumi kredītiestādēm nodot ceturkšņa un gada FINREP finanšu datu pārskatus tika izveidoti ar Eiropas Centrālās Bankas padomi 2014.gadā. No sākuma šādas prasības tika izvirzītas tikai attiecībā uz vairāk nozīmīgām iestādēm (bankām), kas atrodas Eiropas Savienības valstīs. Kopš 2014.gada atskaišu standarts visu laiku tika papildināts – tika ievestas jaunās tabulu rindas, jaunas tabulas un jaunas validācijas, daļa no sākotnējām bija arī dzēsta. 2015.gadā noteikumi bija papildināti ar prasībām nodot atskaites arī individuālajām iestādēm, kurām pirmais atsauces datums atskaišu sniegšanai ir 2017.gada 30 jūnijs; un mazāk nozīmīgām iestādēm – 2017.gads 11 februāris [14, 15, 17]. Iestāžu klasificēšana notiek atsevišķi katrai iestādei attiecīgi uz „PADOMES REGULU (ES) Nr. 1024/2013” 6.pantu 4.punktu, ņemot vērā iestādes lielumu, nozīmīgumu Eiropas Savienības vai iesaistītās dalībvalsts ekonomikā, aktīvu vērtību un citus kritērijus un faktoros [18].

Latvijā šādu atskaišu nodošanu regulē Finanšu un Kapitāla Tirgus Komisija (FKTK), kas 2016.jūlijā arī pieņēma "Uzraudzības finanšu pārskatu sagatavošanas normatīvos noteikumus", kas nosaka datu iesniegšanu individuālajā līmenī un ir attiecināmas uz „kredītiestādēm un citās dalībvalstīs reģistrēto kredītiestāžu un ārvalstu kredītiestāžu filiālēm” [16]. FKTK noteikumos katram kredītiestāžu tipam ir pateikts nodot atskaites tikai par lielāko daļu no visām 64 FINREP tabulām (piemēram mazāk nozīmīgām iestādēm – 43 tabulas) [17].

Eiropas Centrālās Bankas tīmekļa vietnē var atrast finanšu statistiku kas saucas „caurskatāmības pārskats”, un satur sevī noteiktas tabulas, kur šūnu vērtības ir kombinētas no COREP¹ un FINREP atskaitēm². Pēdējā tāda veida caurskatāmības pārskata atskaite sastāv no datiem nodotiem uz ECB periodā no 2015. decembra līdz 2016. jūnijam.

Ir arī norādījumi datu integrācijai no abiem standartiem, kas attēlo uz kuru tabulu šūnām datiem no kuriem standartiem ir jāiet (katrai šūnai atbilst kāds datu punkts attēlots kā standarta pirmais burts tabulas numurs_rinda_kolona. Piemēram – 1.1.attēlā, var redzēt, ka uz tabulu “Peļņa un zaudējumi” pirmām divām rindām jāiet tikai datiem no tabulām F 02.00 un F 16.01 no FINREP standarta.

¹ <http://www.datatracks.eu/blog/corep-finrep-implementation-challenges-part-ii/>

² <https://www.eba.europa.eu/risk-analysis-and-data/eu-wide-transparency-exercise/2016/results>

	As of 31/12/2015	As of 30/06/2016
(mln EUR)		
Interest income	F 02.00_010_010	F 02.00_010_010
Of which debt securities income	F 16.01.a_020_010	F 16.01.a_020_010

1.1. att. Datu integrācija no FINREP atskaites tabulām uz tabulu „P&L” („Peļņa un zaudējumi”) ECB „caurskatāmības vingrinājumā” [12]

Analizējot šo statistiku var secināt ka no Latvijas piedēroņāsajām bankām pagaidām datus nodod tikai ABLV. Ir arī dati par bankām, kuru filiāles ir Latvijā (piemēram SwedBank, DnB, Nordea), bet tas tiek bāzētas ārvalstīs [12].

FINREP standarts ietver sevī vairākus dokumentus, īpaši daudz attiecība uz XBRL datnes struktūru (kuru ir jānodod FKTK). Nozīmīgākie no tiem ir:

1. Datu validācijas dokuments, kurā tiek pārskaitītas visas datu validācijas, definētas FINREP standartā (1.2.attēls). Ir definēta tāda informācija, kā validācijas identifikators, versija kurā notika izmaiņas un izmaiņu tips (dzēsta, pievienota, izmainīta), validācijas tips (bloķē datu nodošanu vai nē), validētas tabulas, tas rindas un kolonas, ka arī validācijas formulējums.

	A	B	C	J	K	L	R	S	T	U	
1	Changed in framework release			Prerequisites - apply rule if ALL tables are reported (or for module containing the table if table							
2	ID	Replaces	Severity	T1	T2	rows	columns	sheets	Formula		
639	v0895_m	2.0 (2013/09)	Blocking	F 12.00		{090}			{r530} = {r010} + {r020} + {r160} + {r300} + {r330} + {r470} + {r500}		
640	v0896_m	2.0 (2013/09)	Blocking	F 15.00.a		{010-190}			{c020} + {c030} <= {c010}		
641	v0897_m	2.0 (2013/09)	Blocking	F 15.00.a	F 01.01				sum({F 15.00.a, r020, {{c010}, {c080}}}) <= {F 01.01, r070, c010}		

1.2. att. FINREP tabulu šūnu validācijas [21]

2. Dokuments kas satur visu tabulu veidnes (1.3.attēls). Tajā ir visas FINREP tabulas, grupētas pēc numuriem Excel lappusēs (piemēram F 16.01 – F 16.07 atrodas lappusē nr.16).

	A	B	C	D	E	F
1	1. Balance Sheet Statement [Statement of Financial Position]					
2						
3	1.1 Assets					
4						
5				References	Breakdown in table	Carrying amount
6						010
7	010	Cash and cash balances at central banks		IAS 1.54 (i)		
8	020	Cash on hand		Annex V, Part 2.1		
9	030	Cash balances at central banks		Annex V, Part 2.2	4	
10	040	Other demand deposits		Annex V, Part 2.3	4	
11	050	Financial assets held for trading		IFRS 7.8(a)(ii); IAS 39.5, AG 14		
12	060	Derivatives		IAS 39.9	10	
13	070	Equity instruments		IAS 32.11	4	

1.3. att. Annex III dokuments ar visām FINREP tabulu veidnēm [20]

3. DPM (ang. Data Point Model – Datu Punktu Modelis) – dokuments kurā katras tabulas šūnām papildus ir piešķirts unikāls identifikators (1.4.att.). Vairākām tabulām varbūt kopējie datu punktu identifikatori, kas norāda, ka datiem nodotiem tajās šūnās ir jābūt vienādiem.

Code	Description	Value (€)
010	Cash, cash balances at central banks and other demand deposits	10888
020	Cash on hand	112758
030	Cash balances at central banks	112750
040	Other demand deposits	112754

1.4. att. DPM dokuments ar visiem datu punktiem iekļautiem FINREP tabulās [22]

Informācijas tehnoloģiju tirgū pastāv vairākas kompānijas, kas specializējas uz darījumu intelīģences sistēmu veidošanu priekš FINREP un otrā līdzīga standarta (COREP) atskaitēm. Visi šī risinājumi ir maksas un prasa papildus pielāgošanu kredītiestādes vajadzībām un tehniskajām iespējām.

Kompānija CFORS piedāvā kombinēto risinājumu FINREP un COREP atskaitēm, kas ir bāzēts uz Oracle HFM [6]. CFORS izstrādāta sistēma nodrošina gan datu manuālo ievadi, gan augšupielādēšanu sistēmā no citiem avotiem (datu integrācija notiek izmantojot FDM rīku), gan arī vairākas atskaites gan par datiem, gan par raportēšanas procesu kopumā [6]. Kombinētu risinājumu priekš atskaitēm šos divos standartos, ar iespējām integrēt datus no dažādiem avotiem, Oracle Hyperion sistēmā (izmantojot Essbase un HFM) piedāvā arī kompānija Amosca [10]. Amosca piedāvā arī „Report Authority” rīku priekš datu konvertēšanas XBRL formātā [10,11].

Publiski pieejamos tīmekļa avotos nav iespējams atrast informāciju par kādu Latvijas kompāniju, kas piedāvātu kādu risinājumu FINREP atskaitēm. Ir zināms tikai ka Latvijas banka ABLV Bank, priekš savas filiāles Luksemburgā, 2014. gadā, nopirka FINREP atskaišu sistēmu Summix, kas ir piedāvāts Latvijas tirgū, no Nīderlandu kompānijas Wolters Kluwer Financial Services [7]. Šī sistēmā arī apvieno sevī atskaites ne tikai FINREP, bet arī COREP un citas atskaites un nodrošina gan datu augšupielādēšanu no citiem avotiem, gan glabāšanu (sistēmas datuvē) un manuālas datu korekcijas, gan to validāciju un konsolidāciju [8, 9]. No visas pieejamas informācijas nevar secināt par to ar kādam tehnoloģijām šī sistēma ir izstrādāta.

Nav iespējams arī atrast kādu informāciju par atvērta koda (vai arī vienkārši bezmaksas) informāciju sistēmām priekš FINREP, vai kāda līdzīga finanšu standarta atskaitēm. Ir pieejami tikai risinājumi, kas atbalsta XBRL datnes noformēšanu, validāciju pēc atskaišu standartiem, ko

piedāvā dažas organizācijas, piemēram Arelle [13]. Šādi risinājumi, protams, neatrisina problēmu kā apkopot un uzglabāt datus. Ir arī vairākas norādes uz to ka ir iespējams izmantot dažas atvērta koda ERP (ang. „Enterprise Resource Planning” – Organizācijas Resursu Plānošanas) sistēmas priekš organizācijas finanšu datu pārvaldību pēc IFRS ieteikumiem (piemēram ADempiere [14]). Šāda veida risinājumi ir vairāk attiecināmi uz kompānijām, kas ražo dažādus produktus, un izmanto iekšējo ERP sistēmu produktu informācijas, maksājumu un pasūtījumu pārvaldīšanai, un palīdz pārvaldīt grāmatvedības procesus par pārdotajiem un saražotajiem produktiem. Šeit ir nepieciešams pieminēt, ka ir daži pētījumi ERP sistēmu sfērā, kā piemēram [44], kas pēta atvērta koda (šajā gadījumā Pentaho) pielāgošanu organizāciju vajadzībām, kas varētu saņemt ERP sistēmas datus, un palīdzēt lietotājiem lēmumu pieņemšanā un datu analīzē.

Var secināt ka nav gatava atvērta koda bezmaksas risinājuma priekš FINREP, vai kāda līdzīgā standarta atskaitēm. Komerciālo piedāvājumu, kas izmanto informācijas sistēmas priekš šādām atskaitēm ir diezgan daudz, bet informācija par tiem ir slēgta. Tikai divas kompānijas pārdod risinājumus, par kuriem publiski rakstīts, ka tie ir balstīti uz Hyperion Financial Management.

2. DARĪJUMA INTELIGENCES RĪKU IZVĒLE

Visi rīki, apskatītie šajā darbā ir izvēlēti balstoties uz to popularitātes publiski pieejamos informācijas tehnoloģiju tirgu pētījumus (piemēram no Gartner, Butler Analytics un citiem resursiem). Viens no papildus iemesliem izvēlēties Oracle Hyperion Financial Management rīku ir tas, ka autorei ir darba pieredze ar to.

Oracle Hyperion Financial Management (tālāk HFM) ir viena no lielās CPM sistēmas Oracle Hyperion daļām, parējie nozīmīgākie rīki, kas ietilpst šajā sistēmā ir:

- Oracle Hyperion Planning – risinājums priekš finanšu un budžeta prognozēšanas.
- Oracle Essbase – daudzdimensiju OLAP (Online Analytical Processing) serveris.
- Gartner ievietoja Oracle Hyperion sistēmu līderos Maģiskajā kvadrantā par finanšu CPM sistēmām, kas tika izdots 2016.gada 31.maijā [5] (2.1.att.).



2.1. att. Gartner Maģiskais Kvadrants CPM sistēmām[5]

Veidojot šo apkopojumu, tika novērtētas CPM sistēmu īpašības kā vērtības ko sistēma dod grāmatvedības procesam, produkta stratēģijā (cik loģiski ir visi rīki kas ir iekļauti visa CPM kompleksā), tirgus saprašanā un klientu apmierinātībā [5]. Novērtējot Oracle Hyperion, tika atzīmēts, ka viens no veiksmes faktoriem ir tas, ka Oracle izdodas pārdot rīku kompleksus, nevis tikai atsevišķas programmas. Tā piemēram 34% no pircējiem atzinās ka pērc Oracle

programmatūru tikai tāpēc ka viņiem jau bija veiksmīgi izmantotas citas Oracle lietotnes [5]. Savukārt viens no galvenajiem brīdinājumiem bija par to ka Hyperion sistēmas instalēšana un uzstādīšana prasa ievērojamas tehniskas iemaņas un tehnisko aprīkojumu [5]. Tādēļ daļa no klientiem izvēlas arī Oracle piedāvātus Hyperion mākoņu risinājumus [5].

Šeit ir jāpiemin, ka visas minētas šajā Maģiskajā Kvadrantā sistēmas ir komerciālas. Dažām tomēr ir bezmaksas izmēģinājumu periodi, vai speciālas versijas ar ļoti ierobežoto funkcionalitāti (kā piemēram SAP HANA Express Edition versija, kuras datubāzes izmērs ir ierobežots ar 32 GB [25]).

Oracle Hyperion ir starp līderiem arī „The Forrester” ASV pētnieciskajā kompānijas izveidotajā CPM (EPM) rīku pārskatā „The Forrester Wave”, kas bija publicēts 2016.gada oktobrī – ”Enterprise Performance Management, Q4 2016” (2.2.att.).



2.2. att. Forrester Wave EPM sistēmu sadalījums [26].

Ja Gartner un Forrester vairāk izskata komerciālas sistēmas, tad, savukārt Eiropas analītiskā kompānija Butler Analytics, vairāk uzmanības vērš arī atvērta koda risinājumiem.

Tā „Butler Analytics Yearbook”, kura ir iekļauts darījumu inteliģences sistēmu pārskats un novērtējums par 2015.gadu, ir nodaļa arī par atvērta koda risinājumiem. Šeit starp pirmajiem līderiem ir Eclipse Birt, JasperSoft, Palo un Pentaho Community sistēmas. Tālāk tiek pieminētas arī ReportServer un SpagoBI [24]. Interesanti arī ka JasperSoft un Pentaho šajā pārskatā parādās arī nodaļā par uzņēmumu atskaišu platformām, pie tam JasperSoft iet uzreiz pēc līdera, kas ir IBM Cognos, kas ir Gartner Maģiskajā Kvadrantā ievietots kā vienīga izaicinājumu pārņemoša CPM sistēma [24, 5].

Ir nozīmīgi arī ka Pentaho un JasperSoft (tā ražotājs Tibco) sistēma bija iekļauti arī iekš Gartner Maģiskajā Kvadrantā par darījumu inteligences platformām, kas bija publiskots 2016.gada februārī (2.3.att.).



2.3. att. Gartner Maģiskais Kvadrants darījumu inteligences sistēmām[23].

Šajā Gartner pārskata tikai nebija sadalījuma starp šo programmu atvērtā koda „Community” versijām un komerciālas versijas, kurai protams, ir vairāk papildus iespēju. Neskatoties uz „Community” versijas ierobežojumiem, kompānija Adaptive Insights, izveidoja un pārdod EPM sistēmu, kas ir izveidota uz Pentaho Community platformas [27]. Rakstā no 2006.gada bija minēts ka ir pieejama arī šīs sistēmas Express Edition bezmaksas versija ar ierobežoto funkcionalitāti [27], tomēr šobrīd, pārmeklējot adaptiveinsights.com tīmekļa lapu nevienas norādes uz šādu versiju vairs nevar atrast.

Ņemot vērā ka darba mērķis ir izpētīt ceturkšņa atskaišu ģenerēšanas iespējas, kas ir apkopojums visiem datiem, kas ir uzņēmumā sakrājušies dažādos avotos pēdējo trīs mēnešu laikā, ETL un OLAP ir galvenās prasības attiecībā uz atvērtajiem rīkiem. Butler Analytics līderis – Eclipse Birt, nav savietojams ar OLAP. Tas ir spējīgs pieņemt un apstrādāt datus no dažādiem avotiem, bet OLAP sistēma tajā nav iebūvēta [40].

Vadoties pēc iepriekšminētajiem pētījumiem, no atvērtā koda darījumu inteligences sistēmām tika izvēlēts apskatīt JasperSoft, Palo un Pentaho Community sistēmas.

3. IZVĒLĒTO DARĪJUMU INTELIĢENCES RĪKU TEORĒTISKAIS APSKATS

Šajā nodaļā tiks veikts Oracle Hyperion Financial Management rīka, kā arī izvēlēto darījumu inteliģences rīku funkcionalitātes teorētiskais apskats.

3.1. Hyperion rīka funkcionalitātes teorētiskais apskats

Oracle HFM pamatā ir daudzdimensiju datubāze, kas nodrošina vairākas finanšu datu apstrādes funkcionalitātes, tajā skaitā datu konvertēšanu dažādās valūtās, datu kalkulācijas pēc nepieciešamiem noteikumiem (kodu šādiem aprēķiniem var rakstīt Visual Basic valodā), atskaišu ģenerēšanu, transakciju novienādošanu. Tas ir tīmekļa bāzēta sistēma, kas palīdz ātri un droši pārvaldīt organizācijas finansiālo informāciju [28].

Galvenokārt šī sistēma tiek izmantotā –

- Finanšu datu pārvaldīšanai, konsolidēšanai no dažādiem avotiem,
- Finanšu atskaišu ģenerēšanai (ik-mēneša, ceturkšņa vai gadā pārskatiem).

HFM rīkam ir 3 līmeņu arhitektūra. Tas sastāv no sekojošajiem līmeņiem [31]:

1. Klientu līmenis – šeit atrodas lietotāja interfeiss. Šis līmenis arī ir atbildīgs par metadatu attēlošanu un pārvaldīšanu.
2. Lietotnes līmenis – šeit atrodas tīmekļa serveris, atskaites. Līmenis satur datu pārvaldīšanas funkcionalitāti un saikni ar datubāzi.
3. Datubāzes līmenis – šeit atrodas relāciju datubāze un rīka metadati.

Visi procesi HFM sistēmā ir piesaistīti kompānijas hierarhijai. Dati tiek ievadīti pamata kompānijās (filiālēs) un pēc tam konsolidēti uz augstāko līmeni. Parasti biznesa procesi iekš HFM notiek šādos soļos (3.1.1. att) – dati tiek augšupielādēti sistēma (vai arī ievadīti manuāli), pēc tam dati ir kalkulēti (šajā solī tika kalkulētas piemēram, validācijas, un šūnas kurām ir noteiktas speciālas darbības – datu kopēšana no vienas tabulas uz citu, saskaitīšanas vai reizināšanas operācijas), tālāk ir iespējam pārbaudīt vērtības un labot tos, pēc vajadzības [30]. Pēc tam kad persona atbildīga par šo līmeni (filiāli) nodrošina ka dati ir kārtībā, attiecīgais līmenis ir slēgts un dati tiek konsolidēti uz augšu. Konsolidācijas laikā notiek valūtas translācijas. Ja kāda filiāle pieder augstākajai kompānijai tikai daļēji, tad konsolidācijas laikā dati tiks saskaitīti uz augšu ņemot vērā tos daļas procentus uz kurām tā pieder augstākajai kompānijai. Pēc

tam dati tiek pārbaudīti arī augstākajā līmenī, ja nepieciešams, arī atkal pamainīti zemākajās kompānijās hierarhijā. Kad dati ir kārtībā, periods (mēnesis) ir slēgts. Ja nepieciešams nodot datus tālāk kādai pārbaudošai iestādei, tie tiek eksportēti no sistēmas un publiskoti [30].



3.1.1. att. Procesi HFM sistēmā [30].

3.1.1. HFM konfigurēšanas elementi

Galvenie elementi, kurus var izmantot lai konfigurētu HFM sistēmu ir šādi:

1. **Metadati** – definē dimensiju sastāvdaļas un saturu. Vienas no svarīgākajām dimensijām ir „Account” – kas parasti tiek izmantota lai definētu tabulu rindas, „Entity” – kas ir izmantota priekš kompānijas hierarhijas definēšanas, „Custom” dimensijas kas var būt vairākas, piemēram lai definētu valūtu, valstis, datu avotu. Metadatos var norādīt tādas svarīgas sistēmas uzstādījumus kā pamata valūtu, validācijas kontu („Account” dimensijas elementu, kura vērtība nosāka vai būtu iespējams tagad aizvērt noteiktajam kompānijas līmenim datu ievadi vai nē, 0 nozīmē ka var, citas vērtības nozīmē ka nevar).

Metadati var būt izmainīti neierobežoti daudz reižu, augšupielādēti sistēmā, vai arī lejupielādēti no tas. Tas glabājas APP (struktūra tāda pati kā CSV) vai XML datnē. HFM veic integritātes pārbaudi katru reizi kad metadati tiek augšupielādēti sistēmā, ja datnē ir kļūdas, tad augšupielādēšana nav iespējama.

2. **Noteikumi** (“Rules”) – kods kas nosāka iekšējās sistēmas kalkulācijas, Visual Basic valodā. Šajā kodā var izmantot HFM specifiskās funkcijas, priekš datu manipulēšanas (piemēram HS.Noinput(), kas aizver noteiktus dimensiju krustpunktus). Noteikumi galvenokārt tiek izmantoti:
 - a. datu aprēķināšanai,

- b. dažu dimensiju krustpunktu rakstīšanas aizvēršanai/atvēršanai,
- c. konsolidācijas , kalkulācijas īpašiem gadījumiem.

Noteikumi glabājas .RLE failā, kas var būt rediģēts kā parastais TXT fails. Tie ir sakārtoti nodaļās – apakš funkcijās (piemēram „Sub Consolidate()”, “Sub Calculate()”).

3. **Drošības uzstādījumi** – kompānijām un kontiem („Entity” un „Account” dimensijas) tiek glabāti .SEC failā, kuru arī var modificēt kā parasto teksta failu un atjaunot sistēmā, ielādējot datni sistēmā. Metadatos dimensijas elementiem var piešķirt drošības klases, un vairākiem elementiem var būt vienādas drošības klases. Drošības failā lasīšanas vai rakstīšanas piekļuves tiesības šādām drošības klasēm var dot gan lietotāju grupām, gan lietotājiem. Ja piekļuves tiesības ir definētas grupai, lietotājiem var definēt šo grupu. Lietotājiem var definēt tiesības mainīt datus uz dažādām līmeņiem – kad lietotājs slēdz kompāniju priekš sava līmeņa, šūnas paliek joprojām atvērtas ievadei priekš lietotāja ar vairākām tiesībām (tas parasti attēlo darbinieku hierarhiju kompānijā).
4. **Locekļu saraksts** („Memberlist”) – sistēmā var veidot papildus sarakstus ar dažādu dimensiju speciālajiem elementiem, kas piemēram nav apvienoti pēc metadatu hierarhijas, bet pēc kādām speciālajām prasībām. Tādus sarakstus var izmantot piemēram noteikumos, lai definētu kalkulācijas kādam speciālajiem elementiem. Var izmantot arī datu ievades formās, lietotāju ērtībai, lai vieglāk būtu izvēlēties kādas zināmas vērtībās.

3.1.2. HFM dimensijas

Definējot to, kādi dati būs sistēmā un kādas būs dimensijas, ir jāveic uzstādījumus HFM metadatu failā. Katrs datu punkts katrā tabulā ir kādu dimensiju krustpunkts. Tādu visu dimensiju krustpunktu kopumu sauc par skatu punktu.

Kopumā HFM ir šādas sistēmā definētas dimensijas:

Scenārijs – var būt Aktuāls, Budžets vai Prognozēšanas. Scenārija izvēle ir atkarīga no biznesa prasībām sistēmai. Aktuāls piemēram atzīme, ka sistēma tiks glabāti dati no pagājušajām un aktuālajām operācijām [29].

Gada un Perioda dimensijas. Perioda dimensijā mēneši ir grupēti kvartālos un pēc tam gadā. Gada dimensija vienkārši satur visu gadu sarakstu, kas tiks izmantoti sistēmā (piemēram 2015, 2016, 2017).

Skats. Šajā dimensijā var izvēlēties veidu par kādu periodu izvēlēties skatīties datus. Ir pieejamas opcijas „Year to date”, „Quarter to date” – kas atspoguļos datu summu katrā šūna kopš perioda sākuma līdz šodienai [29]. Ir arī „Periodic”, kas attēlos datus tikai par izvēlēto mēnesi.

Value. Šajā dimensijā ir iespējas izvēlēties kādā valūtā izņemt datus. Var izvēlēties ne tikai kādu konkrētu valūtu bet piemēram opciju „Vecāka valūta”, kas parādīs datu tajā valūtā, kas ir definēta priekš vecākas kompānijas hierarhijā. Var apskatīties arī datu novienādotas vērtības. Tas nozīmē, kā piemēram ja starp divām meitu kompānijām bija viena transakcija, bet informācija par to nav vienāda abās kompānijās, tad starpībā parādīsies kā novienādotā vērtība uz vecākas kompānijas. Šis mehānisms ir ērts priekš organizācijām ar lielu hierarhiju, kur ir daudz filiāļu.

Entitijas dimensija. Kā jau bija minēts augstāk, šī dimensija ir atbildīga par kompānijas iekšējo struktūru. Šeit ir jābūt definētām visām organizācijas daļām hierarhiskā secībā. Var definēt arī kādu neeksistējošo kompāniju, ja tas palīdz iekšējās datu kalkulācijās, vai piemēram var arī sadalīt vienu kompāniju vairākos līmeņos, ja ir gaidīts ka datus par to var ievadīt dažādi cilvēki, kuriem nav jāredz citus datus un jādod iespēja veikt datu kalkulāciju tikai ar saviem ievadītājiem datiem.

ICP („InterCompany”) dimensija. Šī dimensija, parasti ir kopija no Entitijas dimensijas, un to uzturēšana notiek sistēmā automātiski. Šeit ir visas kompānijas ar kurām var ievadīt transakcijas. Parasti tas ir bāzes līmeņa kompānijas, kur var ievadīt datus (jo visās vecāk kompānijas dati vairs jau nav ievadīti manuāli, bet konsolidēti no apakšējiem līmeņiem).

Iekš kompānijas transakcijas likvidējas vienas kompāniju grupas ietvaros. Piemēram ja mums ir kompānijas ar kodiem 8000Group, zem tas ir 8001, 8002, un ārpus, citās hierarhijās ir vēl kompānijas 9000, 4000, uz 8000Group transakcijas iekš kompānijām 8001 un 8002 jau nebūs redzamas:

- **8000Group**, transakcija ar **9000** = 20, transakcija ar **4000** = 30, [ICP TOP] = 50
 - **8001** – transakcija ar 8002 = 10, transakcija ar **9000** = 20, [ICP TOP] = 30
 - **8002** – transakcija ar 8001 = -10, transakcija ar **4000** = 30, [ICP TOP] = 20

Kontu dimensija. Šeit var definēt tabulu rindas. Katram elementam ir iespējams norādīt vairākus tipus – piemēram aktīvi, ieņēmumi, izdevumi. Katram tipam pēc noklusējumā sistēmā būs dažāda uzvedība [29]. Piemēram vērtības no kontiem ar tipu „izdevumi” tiks kalkulētas ņemot vērā absolūtas vērtības (ignorējot vērtību zīmi).

HFM ir arī tā saucamas papildus dimensijas, kuru skaits var būt mazāks vai lielāks, atkarībā no noteiktas sistēmās vajadzībām. Šādas dimensijas sauc par „Custom” un tālāk seko

secības numurs (piemēram „Custom1”, „Custom2”). Šajās dimensijās var tikt glabāta piemēram informācija par valstīm, produktiem, datu avotiem [29].

3.1.3. HFM datu attēlošanas iespējas

Lai attēlotu datus lietotājam HFM sistēmā var veidot:

1. Datu ievades formas. Formas ir pieejamas no tīmekļa HFM lapas, un tajos var uzreiz ievadīt, vai modificēt datus uz bāzes dimensiju līmeņiem. Formā katrai šūnai var rakstīt komentāru par ievadītām vērtībām. Komentāri nekonsolidējas uz augstākajām kompānijām. FINREP standartā ir tabulas, kurām ir definētas šūnas ar tekstuāļajām vērtībām, kurām jākonsolidējas, lai uz vecākas entitijas varētu saņemt atskaiti par vērtībām katrā kompānijā.
2. Datu atskaites ar diagrammām, summārām rindām un kolonām, kā arī ar vēlamo dizainu. Dažādi objekti var tikt izmantoti vairākās atskaitēs (kā galvene, kājene). Atskaitēs veidošanā ir pieejamas vairākas funkcijas (līdzīgas MS Excel – SUM(), AVG() un citas). Tiesa gan, atskaišu veidošana nav iebūvēta HFM rīkā – atskaites var veidot Hyperion Financial Reporting Studio rīkā, kas nāk kopā ar Hyperion EPM sistēmu. Katra atskaite tiek saglabāta .des failā, kuram iekšā ir XML struktūra. Atskaites var brīvi lejupielādēt no sistēmas un augšupielādēt tajā, kā arī var definēt piekļuves tiesības – lietotājiem, vai lietotāju grupām var atļaut atskaites vai nu lasīt, vai nu modificēt.
3. Kolekcijas no atskaitēm, ar ko var uzreiz atvērt vairākas atskaites PDF vai Excel formātā.
4. Datu izņemšanu no sistēmas ar Microsoft Excel SmartView papildinājumu, kas arī ir iekļauts Hyperion EPM sistēmā. SmartView pievienojumprogramma atbalsta ne tikai datu lasīšanu no HFM datubāzes, bet arī datu rakstīšanu atpakaļ, kā arī krustpunktu statusu pārbaudi (vai ir slēgta, atvērta priekš datu ievades).
5. Datu atlasītu no sistēmas datnē (katrs datu punkts ir definēts kā dimensiju krustpunkts). Var arī datus ielādēt no datnes, piemēram datni ar atlasītajiem datiem ielādēt atpakaļ.

3.1.4. Datu un lietotāju darbību audits

HFM atbalsta ļoti detalizētu lietotāju darbību auditu. Katrai šūnai tiek glabāta informācija par iepriekšējām vērtībām un lietotāju kas tos mainīja. Priekš šādam atskaitēm var veidot speciāla audita vēstures atlases, pa stundām, dienām, mēnešiem, priekš katra iespējama krustpunkta.

Ir arī audits par lietotāju darbībām sistēmā kopumā, bet tas nav granulārs (piemēram var redzēt ka lietotājs atvēra datu formas un atskaites, bet nav zināms kurus).

3.1.5. Datu ievade sistēmā

Datus, HFM sistēmā var ievadīt gan manuāli (ar datu formu, vai SmartView palīdzību), gan arī augšupielādēt no citiem avotiem.

Visvienkāršākā datu augšupielāde ir no tas pašās sistēmas uz sevi. Šāda situācija var veidoties, piemēram taisot datu rezerves kopijas no sistēmās, un augšupielādējot tos, sistēmas avārijas gadījumā.

Var arī integrēt datus no citiem avotiem. Šī funkcionalitāte ir pieejama Hyperion EPM sistēmā, bet nav iebūvēta Financial Management rīkā. Ir iespējams izmantot rīku Oracle Financial Data Quality Management, kurā var definēt datu integrācijas noteikumus datu augšupielādei no kādas datnes uz kādu noteiktu HFM sistēmu.

3.1.6. HFM procesu kontrole

HFM procesu kontroles funkcionalitāte dod iespēju pārskatīt datu atskaišu sagatavošanas procesu noteiktajā periodā, priekš noteiktas kompānijas un tas bērnu kompānijām (ja tādi ir):

1. Lietotāji ar mazākām tiesībām var slēgt datu ievadi savas kompānijās priekš noteiktā perioda, tādā veidā parādot ka kompānijas datu ievade ir pabeigta, un nodrošinot, ka tikai lietotāji, ar lielākām tiesībām varēs mainīt datus šajā kompānijā, kādas korekcijas gadījumā.
2. Kad lietotājs slēdz datu ievadi:
 - a. Tiek pierakstīta vēsture (par to, kurš slēdza šo kompāniju, šajā periodā).
 - b. Tiek pierakstīts lietotāja komentārs, ko viņš ieraksta slēdzot kompāniju.
 - c. Ja datu validācijas, definētas sistēmā, kopā dod datu kļūdu skaitu lielāku par 0, priekš slēdzamas kompānijas, tad sistēma neļaus slēgt datu ievadi priekš šīs kompānijas.

- d. Ja tā ir kompānija, kurai ir bērnu kompānijas, un bērnu kompānijām vēl nav slēgta datu ievade, tad sistēma neļaus slēgt doto kompāniju.
3. Var pārskatīt vai dati ir kalkulēti (konsolidēti) uz vecāku kompānijām, vai tikai ir pagaidām ievadīti “bērnu” kompānijās. Datus var konsolidēt jebkurā laikā, tas nekā neietekmēs “bērnu” kompānijas, bet uzreiz kad “bērnu” kompānijā ir ievadīts kāds skaitlis statuss priekš vecāku kompānijas mainās uz nekonsolidētu.
4. Var pārskatīt kļūdu skaitu, no visām validācijām, kas definētas sistēmā priekš noteiktām tabulu šūnām.

Kopumā procesu kontrole nodrošina kontroli un auditu visiem biznesa procesiem, kas ir parādīti 3.1.1. attēlā.

3.2. Atvērtā koda rīku teorētiskais apskats

Šajā nodāļā tiks apskatīta atvērtā koda darījumu inteliģences rīku funkcionalitāte, balstoties uz literatūras avotiem un publiski pieejamas dokumentācijas. Salīdzinot ar visu Hyperion piedāvātu funkcionalitāti vienas no galvenajām prasībām tādiem rīkiem būtu:

1. ETL iespējas datu integrācijai no dažādiem avotiem.
2. OLAP serveris, kas atļautu datu konsolidāciju un sarežģītākus datu aprēķinus.
3. Atskaišu ģenerēšanas iespējas.
4. Drošības uzstādījumu iespējas – lai varētu definēt lietotāju piekļuves iespējas.
5. Iespējamība modificēt datus pēc tos augšupielādēšanas sistēmā.
6. Iespējamība lejupielādēt datus no sistēmas (piemēram atskaišu nodošanai tālāk).
7. Atbalsta un dokumentācijas pieejamība.

Daudzās darījumu inteliģences sistēmās ir prasības lai programmatūrā būtu iekļauta arī funkcionalitāte priekš mobilajam ierīcēm. Tomēr ceturkšņa finanšu pārskatu sagatave neprasa mobilitāti. Tas ir vairāk par datu apkopojumu no vairākiem datu avotiem, datu pārbaudē pēc validācijām un nelielas datu modifikācijas, ja nepieciešams. Līdz ar to, šeit prasībās priekš atskaišu rīkiem nav definētas mobilas lietotnes, bet uzsvars ir uz ETL un OLAP funkcionalitāti.

3.2.1. JasperSoft Community Edition

JasperSoft sistēmai ir piedāvātas divas versijas – komerciāla (kurai ir 30 dienu bezmaksas izmēģinājums) un „Community” versija.

Tāpat kā komerciālajā versijā atvērtā koda versija ir pieejams: atskaišu dizaineris un dzinējs; interaktīvais atskaišu pārskats (var kārtot, filtrēt, formatēt kolonas); viens repozitorijs priekš atskaišu, lietotāju profilu un citu datu glabāšanas; lietotāju piekļuves tiesību pārvaldīšana katrai atskaipei. Ir arī mobila versija, datu integrācijas iespējas no citiem avotiem, kā arī OLAP serveris (kas iekļauj sevī Mondrian ROLAP, kā arī pievienošanas iespējas Microsoft SQL Server servisiem) [31]. „Community” versija iet ar GNU AGPL licenci, kas padara obligātu koda modifikācijas publiskošanu, ja programmatūra ar tādu modifikāciju kļūst publiski pieejama [32].

Atbalsts bezmaksas versijai arī ir ierobežots – tikai publiski pieejama dokumentācija un publiskais forums. Komerciālajai versijai vēl ir pieejams izstrādātāju atbalsts un mācību portāls tiešsaistē [31]. Tomēr dokumentācija izskatās diezgan pilnīga, pat ar instrukcijām un ekrān attēliem, un forumā ir diezgan daudz aktivitātes – izskatās ka nepieciešamības gadījumā darbā ar šo programmatūru būs iespējas kur meklēt pamācības.

Neskatoties uz atvērtā koda versijas ierobežojumu ir kompānijas, kas piedāvā savus pakalpojumus šādas versijas pielāgošana. Piemēram „Helical IT Solutions”, kas piedāvā arī alternatīvus risinājumus funkcionalitātei, kas nav pieejama „Community” versijā – tā lai aizvietotu ekspromtvaicājumu funkcionalitāti, kas var atvieglot datu analīzi gala lietotājiem, tiek piedāvāts izmantot Pentaho SAIKU pievienojumprogrammu [33].

Vēl var atzīmēt, ka ir pieejamas arī 21 pievienojumprogramma JasperSoft funkcionalitātes paplašināšanai – piemēram PHP klients, Google Earth Visualization, DynamicReports (dinamisko atskaišu veidošanai) [34].

Var secināt ka JasperSoft ir pieejamas gandrīz visas opcijas no prasītām – ir gan ETL dzinējs, gan OLAP, gan drošības uzstādījumi, gan rīka detalizēta dokumentācija. Vienīgi nevar atrast nekādas informācijas par to, vai ir iespējama datu modificēšana pēc ielādēšanas sistēmā.

3.2.2. Pentaho Community Edition

Pentaho Community BI suite ietver sevī tādu funkcionalitāti kā ETL, OLAP, datu atskaites iespējas, kā arī informācijas paneļus [24, 36]. Ir arī atbalstīta drošība – piemēram lietotāju lomas OLAP serverī [39].

Tas satur vairākus atvērtā koda rīkus, kas strādā kopā, lai nodrošinātu sarežģītus biznesa procesus un atskaites [24, 35]. Ir iekļauti tādi rīki kā piemēram:

1. Kettle priekš datu integrācijas. Tas satur gan ETL dzinēju, gan lietotāju grafisko interfeisu, kas ļauj lietotājiem definēt datu integrācijas uzdevumus [35].
2. Mondrian priekš OLAP (līdzīgi kā JasperSoft).

3. Pentaho Report Designer, kas palīdz ģenerēt atskaites uzreiz no datu integrācijas dzinēja, vairākos formātos (PDF, Excel, CSV) [35].
4. Metadata Editor, kas palīdz veidot datu integrācijas noteikumus, pārvēršot datubāzes fizisko struktūru uz loģisko biznesa modeļi [37].
5. Pentaho Aggregation Designer, kas palīdz veidot kalkulētas tabulas, lai uzlabotu Mondrian veiktspēju [35].
6. Schema Workbench, kas palīdz veidot un testēt Mondrian kubu shēmas [35].

Pentaho komerciālajā versijā ir pieejami vairāki papildus rīki, kas padara darbību ar rīku ērtāku, piemēram AES paroļu atbalsts, mobilas ierīces lietotne, darbu palaišana pēc plāna datu integrācijas serverī un citas.

Pēc Pentaho izstrādātāju vārdiem, vienīga nozīmīga starpība starp atvērtā koda un komerciālas sistēmas ir lietotāju atbalsts un atjauninājumu atnākšana. Tā Pentaho speciālistu atbalsts ir pieejams tikai komerciālajai versijai, kurai arī atnāk visādi mazi atjauninājumi un problēmu labojumi apmēram trīs reizes kvartālā. „Community” versijai savukārt atjauninājumi atnāk tikai divas reizes gadā [36], un lietotāju atbalsts tiek realizēts ar dokumentācijas un lietotāju foruma palīdzību.

Ir jāatzīmē arī Pentaho Marketplace vietni, kur ir ievietotas vairākas pievienojumprogrammas, ko var pieinstalēt Pentaho BI Suite rīkiem. Pieejamas pievienojumprogrammas nāk vai no „Klientu joslas” (kurām Pentaho nodrošina atbalstu), vai no „Kopienas joslas” [38].

Var secināt ka Pentaho arī ir pieejamas gandrīz visas prasības – ETL, OLAP, vairākas atskaišu iespējas, lietotāju atbalsts. Tomēr, salīdzinot ar JasperSoft piedāvāto funkcionalitāti, Pentaho ir vairāk papildus opciju – piemēram plašāks pievienojumprogrammu klāsts, metadatu pārvaldība, kas padara darbību ar OLAP serveri ērtāku. Nav daudz informācijas par to, vai būtu iespējams ne tikai saņemt datus no sistēmas, bet arī modificēt tos, tomēr vienā avotā [42] ir norādīts, ka Kettle ir iespējams komunicēt ar OLAP serveriem un rakstīt tajos datus. Tomēr vairākos citos avotos, kā arī [42] ir norādīts arī ka Mondrian OLAP neatbalsta datu rakstīšanu – lai modificētu datus ir jāmodificē tos avotos, un tad jāpārlādē OLAP serverī.

3.2.3. Palo (Jedox Base)

Palo atvērtā koda darījumu intelīģences sistēma, apstrādā datus izmantojot Palo ETL Server un OLAP sistēmu, datu attēlošanai lietotājiem tiek izmantots Microsoft Excel (arī OpenOffice), kā arī piedāvā tīmekļa interfeisu. Administratori var veidot datu atskaites Palo tīmekļa lapā,

turpat šīs atskaites būs pieejamas arī gala lietotājiem. Sistēma arī atbalsta savienojumus ar vairākām datubāzēm [24].

Palo ir alternatīvais nosaukums Jedox Base atvērta koda sistēmai, pastāv arī komerciālā versija Jedox Premium [24, 41]. Abām versijām ir izmantots viens un tas pats OLAP serveris, pie tam Base versijas datubāzes izmērs un lietotāju skaits nav ierobežots. Tomēr atšķirības starp versijām ir diezgan lielas – Premium versija piedāvā daudz vairāk funkciju Excel programmā, tīmekļa vietnē, ETL funkcionalitātē, kā arī lietotāju pieejas kontroles mehānismus [41].

Tomēr viena no pozitīvajām lietām par Jedox ir ka tas atbalsta datu modificēšanu, atšķirībā no vairākām OLAP sistēmām [43]. Pie tam, Jedox OLAP atbalsta 250 dimensiju izveidošanu [43].

Kas attiecas uz dokumentāciju – pati mājaslapā ir diezgan daudz informācijas par Premium versiju un tikai viens raksts par Base (rakstīts 2013.gadā). Ir lietotāju forums³, bet tajā ir maz aktivitātes – attiecīgi maz informācijas no kuras varētu mācīties par iespējamām problēmām darbā ar šo programmu.

Secinot, var teikt Jedox Base daudz priekšrocību OLAP iespēju daļā. Lietotāju drošības efektīva pārvaldīšana un lietotāju atbalsts, paliek uz Premium daļu. Ar visu ierobežoto Base versijas funkcionalitāti, liekas ka tajā ir daudz mazāk iespēju nekā Pentaho Community Edition.

³ <http://forum.jedox.com/index.php>

4. EKSPERIMENTA REALIZĀCIJA IZVĒLĒTO DARĪJUMU INTELIĢENCES RĪKOS

4.1. Eksperimenta realizācija HFM rīkā

Šajā nodaļā ir aprakstīts praktiskais darbs ar Oracle Hyperion Financial Management rīku, kura ietvaros bija izmēģināta FINREP standarta pielāgošana. Tādēļ ka autorei ir praktiskā pieredze darbā ar HFM, uz tas ir bāzēts vairums no aprakstiem zemāk, bez ievietotas atsauces uz oficiālo HFM lietotāja dokumentāciju.

4.1.1. HFM rīka instalācija

Oracle Hyperion licence, norāda, ka šo produktu var bezmaksas lejupielādēt, instalēt un izmantot priekš testēšanas, vai lietotņu demonstrēšanas. Tomēr brīdi, kad lietotne, kas ir izveidota šajā sistēmā, tiks izmantota produkcijā ir jāparaksta vienošanas ar Oracle Korporāciju un jāpērk licence. Produkts ir dārgs, bet pēc autores pieredzes, Oracle nodrošina labu lietotāju atbalstu (piemēram ja sistēmā ir kļūdas, ko traucējummeklēšanas ieteikumi nepalīdz, Oracle atbalsts dublē klienta lietotni un uzstādījumus savā vidē un meklē kļūdu iemeslus paši).

Tā kā maģistra darba ietvaros HFM produkta lejupielādēšanas mērķis ir tikai notestēt un novērtēt to, autore izmantoja produkta versiju bez licences un vienošanām.

Nepieciešamus produktus var lejupielādēt gan no Oracle mājaslapas⁴, gan no speciālas krātuves⁵. Abos gadījumos ir nepieciešams izveidot lietotāja kontu Oracle sistēmā, tam, galvenokārt, ir nepieciešams tikai darbojošs epasts.

Tā kā HFM ir tikai viens modulis no visa Hyperion produkta (vēl ir piemēram Essbase un Planning), priekš tas instalēšanas ir nepieciešams Hyperion pamats – Foundation services kopā ar WEBLogic serveri.

Gan HFM, gan Foundation services var lejupielādēt kā atsevišķas instalācijas (no sākuma lejupielādēt un pieinstalēt servisus un serveri, un tad to pašu izdarīt ar HFM), gan ka vienu instalāciju ar visiem Hyperion produktiem (bet tad ir jāizvēlas uzstādīšanas brīdī kurus produktus nav jāinstalē). Tomēr, ka paradīja prakse, dažādu komponentu atsevišķa instalācija var radīt problēmas (piemēram ar portiem – WebLogic serveris izmanto portu 9000, un HFM jāinstalē uz servera kad serveris darbojas, bet instalācijas laikā uzstādīšanas rīks nepieņēma vērtību 9000

⁴ www.oracle.com

⁵ <https://edelivery.oracle.com>

priekš uzstādīšanas porta, jo ports jau ir aizņemts ar serveri). Tādēļ no otrā mēģinājuma, tika izvēlēts lejupielādēt visu sistēmu - Oracle Hyperion Financial Management Plus, no Edelivery lapaspuses.

Visi faili aizņem 15.8 GB atmiņas:

Release	Selected Item	Applicable Terms & Restrictions	Size	Published Date
Oracle Enterprise Performance Management System 11.1.2.4.0 for Microsoft Windows x64 (64-bit), 19 files	Oracle Hyperion Financial Management Plus	Oracle Standard Terms and Restrictions	15.8 GB	Jun 01, 2016
Oracle Enterprise Performance Management System Release Client Installers (11.1.2.4.0), 1 file			2.8 GB	
Oracle Enterprise Performance Management System Release Installation Documents and Readmes (11.1.2.4.0), 1 file			956.4 MB	
Oracle Enterprise Performance Management System Release Oracle HTTP Server (11.1.2.4.0), 1 file			1.7 GB	
Oracle Enterprise Performance Management System Release Part 1 (11.1.2.4.0), 1 file			1.7 GB	
Oracle Enterprise Performance Management System Release Part 2 (11.1.2.4.0), 1 file			1.5 GB	
Oracle Enterprise Performance Management System Release Part 3 (11.1.2.4.0), 1 file			1.4 GB	
Oracle Enterprise Performance Management System Release Part 4 (11.1.2.4.0), 1 file			2.1 GB	
Oracle Enterprise Performance Management System Release Part 5 (11.1.2.4.0), 1 file			2.6 GB	
Oracle Identity Management (11.1.1.7.0), 1 file			1.8 GB	
Oracle Smart View for Office (11.1.2.5.500), 1 file			126.7 MB	

4.1.1.1. att. Visi faili iekļautie HFM instalācijā

Salīdzinājumam – divi atsevišķi komponenti – HFM un Foundation services instalācijas faili kopā aizņēma tikai 8 GB.

Sistēmai arī ir nepieciešama relāciju datubāzes serveris. Hyperion ir savietojams ne tikai ar Oracle datubāzi, bet arī ar IBM DB2 un Microsoft SQL Server. Maģistra darbā ir izmantots Microsoft SQL Server 2008 Express Edition [46].

Pirms uzsākt instalāciju ir jāpārbauda ka operētājsistēmā (ja instalācija notiek Windows sistēmā) ir uzstādīti un iespējoti Microsoft IIS (Internet Information Services). Soļi, nepieciešamie priekš šī uzdevuma ir aprakstīti Microsoft tīmekļa lapā⁶. Pēdējo IIS versiju var lejupielādēt arī no msdn lapaspuses⁷. Hyperion ir savietojams ar 6. līdz 8.5 versijai [46].

Pirms instalācijas vēl ir jāuzstāda operētājsistēmas lietotāja konts, ar kuru tiks palaisti Windows servisi. Šo kontu ir jānorāda operētājsistēmā zem Local Security Policy/Local Policies/User Rights Assignment priekš iespējas ieslēgt kā servisu (Log on as a service).

Hyperion instalēšanas soļi ir aprakstīti Oracle lietotāja grāmatā⁸. Tomēr ir jāpiemin, ka sistēmas uzstādīšanas un konfigurēšanas rīks ir diezgan informatīvs un ar to pārsvarā var darboties intuitīvi.

Piemēram ja IIS nav iespējoti, parādās šāda kļūda, kā 4.1.1.2. attēlā.

⁶ [https://msdn.microsoft.com/en-us/library/ms181052\(v=vs.80\).aspx](https://msdn.microsoft.com/en-us/library/ms181052(v=vs.80).aspx)

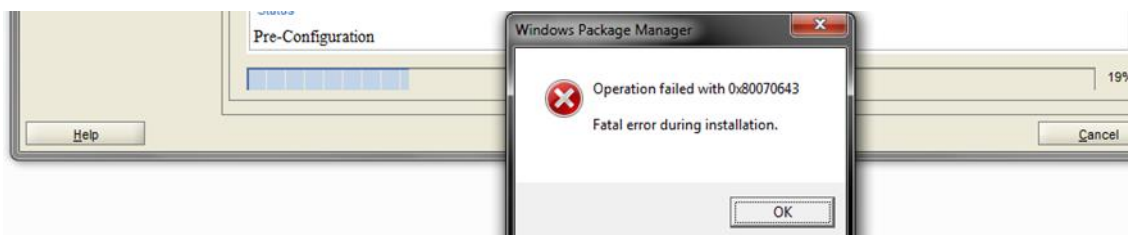
⁷ <https://www.microsoft.com/en-us/download/details.aspx?id=34679>

⁸ https://docs.oracle.com/cd/E57185_01/EPMSD/ch06s02.html



4.1.1.2. att. Kļūda HFM instalēšanas laikā, ja nav iespējoti IIS

Tomēr ir arī gadījumi, kad kļūdas nav lasāmas. Piemēram, instalācijas pēdējā solī parādījās kļūda kā attēlā zemāk (4.1.1.3. att.).



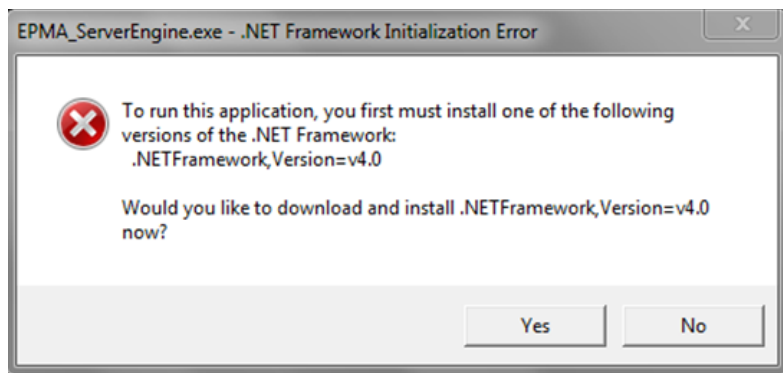
4.1.1.3. att. Kļūda HFM instalēšanas laikā, saistīta ar problēmām palaist EPM serveri

Analizējot žurnāl failus Oracle direktoriņā tika atrasts fails (pēc adreses C:\Oracle\Middleware\user_projects\epmsystem2\diagnostics\logs\epma), kura modifikācijas laiks sakrīta ar laiku kad tika izdota kļūda. Ieraksts žurnāl failā skaidroja, ka nebija iespējams palaist EPMA_ServerEngine.exe (4.1.1.4. att.).

```
[2017-02-25T10:34:38.126+02:00] [Error]
Hyperion.DimensionServer.Interface.Exceptions.EPMAServiceException:
Process 'C:\Oracle\Middleware\EPMSysstem11R1\products\Foundation\BPMA\
AppServer\DimensionServer\ServerEngine\bin\EPMA_ServerEngine.exe' could not be started.
```

4.1.1.4. att. Instalācijas kļūdas pieraksts žurnāl-datnē

Palaižot EPMA_ServerEngine.exe atsevišķi, bija saņemta lasāmākā kļūda, kā attēlā zemāk (4.1.1.5. att.).



4.1.1.5. att. Kļūda palaizot EPMA_ServerEngine.exe

Pēc .NETframework 4.versijas instalēšanas, Hyperion instalācija tika pabeigta veiksmīgi.

Pēc instalācijas, pirms uzsākt konfigurāciju ir nepieciešams izveidot lietotāju un datubāzi Microsoft SQL serverī.

Definējot lietotāju ir jānorāda SQL Servera iebūvēto autentifikācijas veidu. Definējot datubāzi priekš Hyperion sistēmas ir jāizpilda arī šādas komandas [45]:

1. SET READ_COMMITTED_SNAPSHOT ON – kas norāda, ka lasīšanas operācijas nebloķē datu atjauninājumus, un otrādi – datu izmaiņas operācijas nebloķē datu lasīšanu.
2. COLLATE SQL_Latin1_General_CP1_CS_AS – kas norāda kā SQL Serverim jākārtoti dati.

Pēc tam ir jāveic papildus soli attiecībā uz SQL Serveri, kā piemēram XA transakciju iespējošana, kas ir detalizēti aprakstīts Oracle Hyperion lietotāja dokumentācijā, attiecīgajā nodaļā⁹.

Tomēr ja arī pēc visiem iepriekšminētiem soļiem Hyperion konfigurēšanas laikā radīsies kļūda, ka nevar pievienoties SQL Serverim ir jāpārbauda sekojošs SQL tīmekļa konfigurācijā iekš SQL Server Configuration Manager:

1. Ir jāpārbauda vai zem protokoliem ir norādīts TCP/IP vai nē.
2. Ir arī jāpārbauda ka nekas nav norādīts priekš TCP dinamiskajiem portiem un TCP ports ir uzstādīts uz 1433 (vai kāda cita vērtība, kuru tad ir jāizmanto Hyperion uzstādīšanas rīkā).

Viena no pēdējām kļūdām, kas tika saņemta, parādījās HFM konfigurēšanas brīdī. Konfigurējot HFM, WEBLogic serverim jābūt ieslēgtam. Tomēr ieslēdzot WEBLogic serveri parādās kļūda, kā attēlā zemāk (4.1.1.6. att.).

```
weblogic.security.SecurityInitializationException:  
The loading of OPSS java security policy provider failed due to exception,  
see the exception stack trace or the server log file for root cause.  
If still see no obvious cause, enable the debug flag
```

4.1.1.6. att. Kļūda HFM konfigurēšanas laikā

Lai to risinātu ir jāpārkopē fails system-jazn-data.xml no direktorijas C:\Oracle\Middleware\oracle_common\modules\oracle.jps_11.1.1\domain_config uz C:\Oracle\Middleware\user_projects\domains\EPMSys\config\fmwconfig [47].

⁹ <http://www.oracle.com/webfolder/technetwork/tutorials/obe/hyp/FCM11.1.2.3-SingleNodeInstall/FCMSingleNodeInstall.htm#t13>

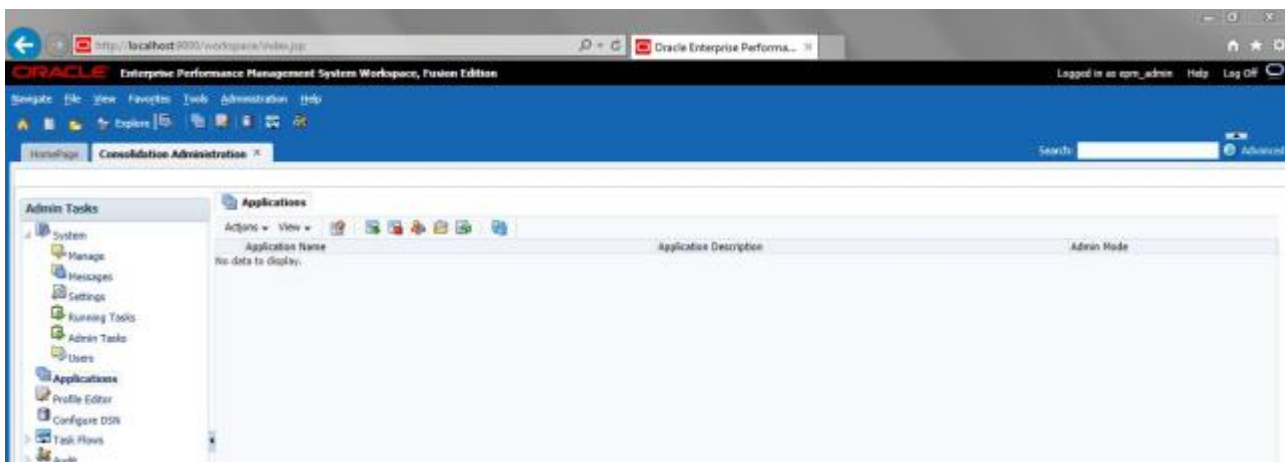
Pēdējā kļūda ar ko nākas saskarties bija saistīta ar HFM moduļa pazušanu no Hyperion tīmekļa lapas darba vietas (Workspace), kas tagad ir pēc adreses <http://localhost:9000/workspace/index.jsp>.

Tas bija saistīts ar to, ka WEBLogic serveris bija sakonfigurēts agrāk nekā HFM modulis. Problēma tika atrisināta ar servera otrreizējo konfigurāciju. Pēc tam serveris bija pārstartēts, kā arī tika pārstartētā darba vieta (ir jāpariet pēc adreses <http://localhost:9000/workspace/refresh>, jāievada administratora vārds un parole, un jāpārlicinās, ka tiks saņemts veiksmes paziņojums [46] (4.1.1.7. att.).



4.1.1.7. att. HFM tīmekļa darba virsmas atjaunojums

Kad HFM modulis ir pievienots serverim, darba vietā var atvērt “Navigate/ Administration/ Consolidation Administration” sadaļu, kā attēlots attēlā zemāk (4.1.1.8. att.).



4.1.1.8. att. HFM rīka servera vietne

Var secināt ka HFM instalācija ir samēra ilgs un sarežģīts process. Ne visas kļūdas ir aprakstītas Oracle dokumentācijā, daļa no kļūdām ir atrodama tīmekļa resursos, kamēr daļa, kuru nekur nevar atrast, paliek kā servera administratora atklādošanas uzdevums.

4.1.2. HFM lietotnes izveidošana

Lai izveidotu HFM lietotni, to ir jāpiereģistrē „Workspace” lapā, atzīmējot ka tiek veidota konsolidācijas lietotne, un ir jāielādē lietotnē „importformat” dokuments.

Datne „importformat” tiek veidotā „Financial Management” programmā, kas ir pieejama mapē „Client Installers” pakotnē Oracle Financial Management Plus, kas satur visus Hyperion

instalēšanai nepieciešamus servisu. Šeit tiek norādīts kāda ir lietotnes atskaišu periodiskums (visur tika norādīta ceturkšņa periodiskums "QTD" jo atskaites ir ik ceturksni) un pielāgotās dimensijas, kā arī kādas dimensijas izmantot priekš valūtas konvertēšanas.

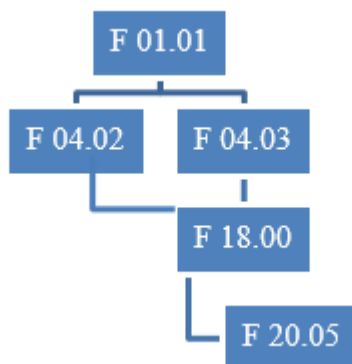
Definējot metadatus priekš HFM FINREP (4.1.2.1.. att.) lietotnes, autore izvēlējās 3 papildus pielāgotas ("Custom") dimensijas – Custom1 (tabulu kolonnas), Custom2 (valstis), Custom3 (datu avots, lai varētu izšķirt kalkulētus un manuāli ievadītus datus).

	Custom Dimension Name	Custom Dimension Alias	Custom Dimension Size	Use For Currency
1	AllCustom1	Custom1	Large	From Currency
2	AllCustom2	Custom2	Large	To Currency
3	AllCustom3	Custom3	Medium	N/A

4.1.2.1. att. Pielāgoto dimensiju definēšana jaunajai lietotnei

4.1.3. HFM metadatu definēšana

Praktiskam HFM pētījumam, tika izvēlētas 5 FINREP tabulas, kas ir savstarpēji saistītas ar datu validācijām. Attiecības starp šīm tabulām varētu apzīmēt, kā attēlā zemāk (4.1.3.1. att.).

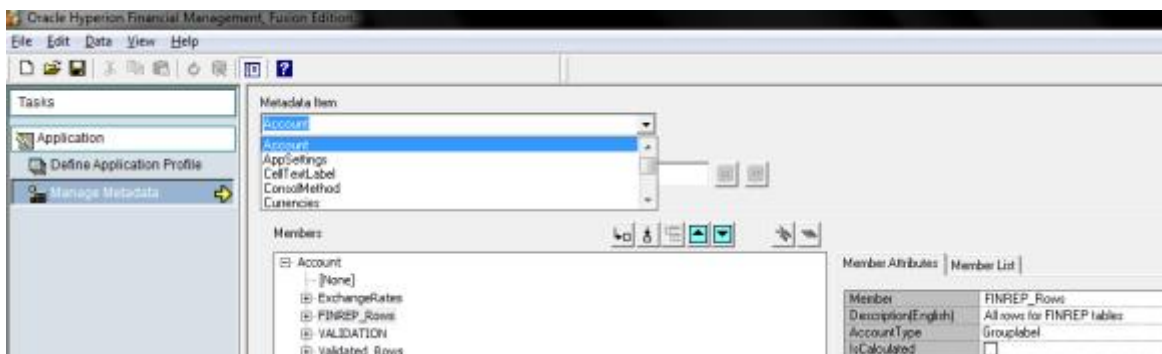


4.1.3.1. att. Attiecības starp izvēlētajām FINREP tabulām

Tālāk tika definēti metadatu dimensiju locekli, noteikumi un izveidotas attiecīgas datu atskaites. Šeit ir svarīgi atcerēties, ka locekļu hierarhija katrā dimensijā ir tiešā veidā saistīta ar datu summēšanu augstāk. Tādā veidā pēc noklusējuma, visi locekļi, kas ir hierarhijas zemākajos līmeņos (hierarhijas lapas) ir atvērti priekš datu ievades, savukārt locekļi, kuriem ir vismaz viens bērns saturēs datu summu no šī bērna. Lai samazinātu HFM datu kalkulācijas un agregācijas laiku, izstrādātāji ieteic neveidot locekļus, kuriem būtu tikai viens bērns, ja tas nav speciāli prasīts no biznesa puses. Tādas vecāku-bērnu locekļu attiecības var apiet un modificēt, lai dati nekalkulētos augstāk, un lai vecāku locekļi būtu atvērti datu ievadei, par to tiks arī pastāstīts zemāk. Svarīgi ir uzsvērt arī to, ka bērnu locekļi (lapas, vai arī vecāki) var būt arī kopējie

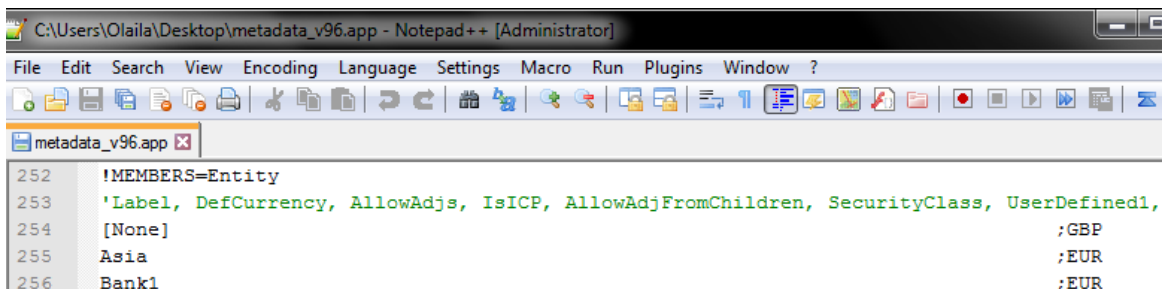
vairākām hierarhijām vienā dimensijā. Šajā gadījumā dati no šādiem locekļiem kalkulēsies augstāk uz visām hierarhijām, kur tas parādās.

Metadatu tehniskajai definēšanai, līdzīgi kā priekš import formāta datnes definēšanas, tiek izmantota Oracle Hyperion piedāvātā programma – „Financial Management” (4.1.3.2. att.). Šajā programmā var apstrādāt metadatu datni XML vai APP formātā.



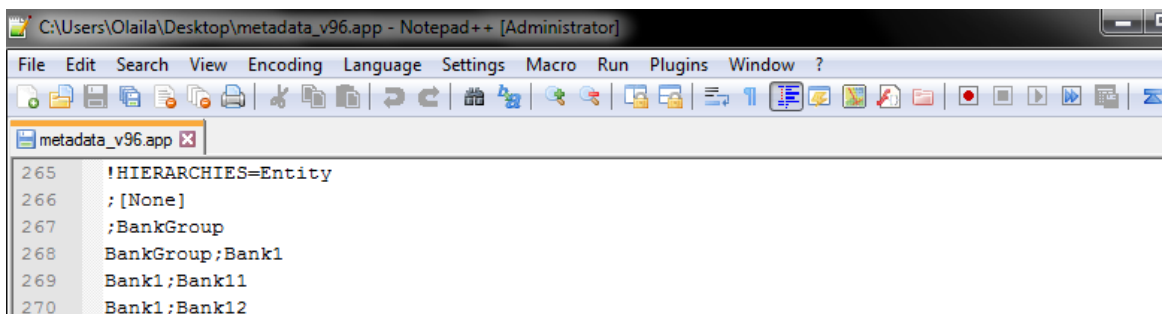
4.1.3.2. att. Financial Management programmas sadaļa metadatu definēšanai.

Apstrādājot metadatu datni, tas nolasa dimensiju skaitu un to nosaukumu, kas ir definēti datnē, kā arī atribūtu nosaukumu, kas ir rakstīts katras dimensijas definēšanas sākumā (4.1.3.3. att.).



4.1.3.3. att. "Entity" dimensijas piemērs, ar definētiem atribūtiem no metadatu datnes.

Locekļu hierarhija ir noteikta izejot no hierarhijas attiecībām, kas arī tiek pierakstīti datnē priekš katras hierarhijas (4.1.3.4. att.).



4.1.3.4. att. "Entity" dimensijas hierarhijas piemērs.

Apstrādāt līdzīgus dimensijas locekļus, definējot tiem atribūtus, vai veidojot jaunus locekļus, var ērti arī Microsoft Excel programmā, līdz tam nokopējot dimensijas informāciju no Notepad programmas.

Metadatu datnē var definēt arī lietotnes uzstādījumus, kopā ar validācijas slēgšanas kontiem pēc fāzēm. Validācijas slēgšanas konts ir „Account” dimensijas loceklis kurā tradicionāli ieraksta kādu skaitļi, ja citām datu validācijām ir sakalkulētas datu kļūmes. Ja dati tur nav vienādi ar nulli, tad nevar aizvērt fāzi, kurai šis konts ir reģistrēts kā slēdzes konts.

HFM var organizēt datu nodošanu pa fāzēm, tādā veidā ierobežojot lietotājiem laiku, kad var aizpildīt noteiktas tabulas. Kad noteiktā daļa ir aizvērtā noteiktajai kompānijai datu ievadē nav iespējama tabulu rindās, kas pieder pie dotas fāzes. Tā kā FINREP standartā tabulas ir sadalītas pa četrām daļām, kad tas ir jānodod, tika nolemts šīs daļas atspoguļot fāzēs. Tādā veidā ir definēti četri slēgšanas konti, kur katram atbilst viena fāze. Tā, piemēram, pēdējā kvartālā, kad ir jānodod visas četras daļas (tas ir visas FINREP tabulas), lietotāju darbu var organizēt tā lai katra daļā būtu secīgi aizpildīta un aizvērtā.

4.1.3.1. “Account” dimensijas definēšana

“Account” dimensijā bija nolemts definēt locekļus priekš tabulu rindām, kas ir tradicionāli HFM sistēmām. Dimensijas locekļu nosaukumiem ir tikai tehniskie ierobežojumi, viss cits ir pilnīgi atkarīgs no sistēmas projektētāja. Tā piemēram nosaukumam ir jābūt īsākam par 80 zīmēm, unikālam, ir jāsakās ar burtu, var saturēt zemsvītras zīmes, bet nav jāsaturs tādas zīmes kā “;”, “:”, atstarpes u.c.

Lai “Account” dimensijas locekļus varētu ātri izšķirt starp citiem dimensiju locekļiem (piemēram datu ekstraktā no sistēmas, vai iekš SmartView datu izvilkšanas), tika nolemts ka katra locekļa nosaukums sakas ar “Row”. Tālāk iet tabulas un rindas apzīmējums – piemēram “Row_0101_010_0000_000” apzīmēs tabulu F 01.01, rindu “010”.

Nosaukumu unikālītāti ir viegli apmierināt izmantojot EBA standartu, jo pēc FINREP katrai tabulai ir unikāls nosaukums, kā arī katrai rindai tabulā ir savs unikāls numurs (piemēram “010”, “020” utt.). Kā jau tika atzīmēts FINREP datu tabulas ir savstarpēji saistītas ar datu validācijām, bet standartā atšķir vēl 3 bāzes tabulas, kas ir saistītas ar visām tabulām sistēmā. Bāzes tabulas (F 01.01 – F 02.00) atspoguļo summāros datus no vairākām tabulām. Viena bāzes tabula ir arī starp autores izvēlētājām tabulām – F 01.01. Starp citām autores izvēlētājām tabulām šajā tabulā ir rindas, kas ir kalkulētas no F 04.02 un F 04.03. Tādēļ, lai būtu vieglāk atsekot tādas kalkulētas rindas, tam nosaukuma beigās tiek pievienots otras tabulas nosaukums un rindas numurs – piemēram rinda, kas tiek kalkulēta no tabulas F 04.02 var tikt nosaukta kā

“Row_0101_090_0402_010”. Locekļiem, kas nav tieši saistīti ar citām tabulu rindām beigās ir nulles.

Tādēļ kā validācijas slēgšanas loceklis tradicionāli pieder pie “Account” dimensijas, arī citas lietotāju, vai standartu definētas datu validācijas parasti asociējas ar šo dimensiju. FINREP definētajām validācijām ir īss un unikāls nosaukums, kas nesatur “aizliegtas” zīmes, līdz ar to, šos nosaukumus var ērti izmantot definējot validāciju locekļus.

Uzsverot visu iepriekš aprakstīto “Account” metadatos tika izveidotas šādas hierarhijas –

1. “FINREP_Rows”, kur ir definētas tabulu rindas, grupētas pēc tabulām, kas ir savukārt grupētas pēc “daļām” (kas nosāka kāda gada laika šos tabulas ir jāraportē). Rindas, kas ir kopējas tabulai F 01.01 un tabulām F 04.02 un F 04.03 atrodas abās hierarhijās, kas atļauj tām summēties gan tur, gan tur.
Rindu hierarhijas noteikšanai tika izmantotas FINREP validācijas nosakošas rindu attiecības iekš tabulām, kas ļoti atvieglo sistēmas projektēšanu – visu darbu var izpildīt bez biznesa puses iesaistīšanas.
2. “VALIDATION”, kur ir definēti validāciju konti, grupēti pēc “daļām”, kurās ir tabulas priekš atskaišu nodošanai, pārbaudītas ar šīm validācijām. Šeit ir definēti arī validāciju slēgšanas konti.
3. “Validated_Rows”, kur ir izveidoti mākslīgie kontu apvienojumi. Tādi konti palīdzes validāciju, kas pārbauda vairākas rindas, nesaistītas pēc hierarhijas, automatizācijā. Tādā veidā konti no vairākām tabulām var būt apvienoti zem kopēja locekļa, kas glabāts šajā hierarhijā. Nosaukums tādiem mākslīgiem kontiem sastāv no pirmās tabulas pirmās rindas un pēdējās tabulas, pēdējās rindas, kas ir pārbaudītas validācijā. Piemēram validācija v1220_m, kuras noteikums ir sekojošs: “F 04.02, c(010), r010 >= sum(r030-050)” (kur “c” apzīmē kolonu, un “r” apzīmē rindu), var pārbaudīt kontu “Row_0402_030_0402_050”, lai kalkulētu vērtības no labas vienādojuma puses.
4. Pēc klasiskās pieejas, “Account” dimensijā, arī tika definēti 3 locekļi priekš valūtas translācijas kalkulēšanas – likme kas bija perioda sākuma, beigās un vidēja likme.

Pēc hierarhijas noteikšanas ir jādefinē arī locekļu atribūti. “Account” dimensijas locekļiem ir vairāk atribūtu, nekā citām dimensijām.

1. **Apraksts.** Ierobežojumi konta aprakstam ir līdzīgi, kā konta nosaukumam – 80 zīmes garumam, un ir aizliegts izmantot dažādas zīmes, kā piemēram “:” vai “;”.

2. **Konta tips.** Konta tips ir vērtība, no kuras ir atkarīgs tas, vai zemāk esošie konti sasummēsies ar citiem savos vecākos, vai atņemsies, kā arī kā vērtības tur akumulēsies nākotnē [48].

Visas rindas izvēlētas tabulās ir atzīmētas kā “Assets” (aktīvi). Šo tipu var izmantot pēc noklusējuma kontiem, kuriem nav izņēmumu kalkulācijās. Kalkulējot valūtu translācijas, šiem kontiem tiks izmantoti dati likmes kontā, kas bija norādīts “DefaultRateforBalance” uzstādījumos, lietotnes definēšanas laikā. Konta tipu var izsecināt arī pēc tabulu/rindu nosaukumiem – tā piemēram F 01.01 (kura nosaukums ir “1.1 Assets”), F 04.02 (kura nosaukums ir “4.2 Financial assets designated at fair value through profit or loss”) un F 04.03 (“4.3 Available-for-sale financial assets”) nosaukumos jau figurē šis vārds, kas nosāka ka šīs tabulas dati pieder pie aktīviem. “Assets” tipa konti pieskaitīs savas vērtības klāt vecāku kontu datiem, ja tiem ir tāds pats vai “Expense” (izdevumi) tips, un automātiski tiks atņemti no “Liability” (saistības) un “Revenue” (ieņēmumi) vecāku kontiem. Attiecības starp dažādiem kontu tipiem ir aprakstītas matricā lietotājā dokumentācijā¹⁰. Šeit var pieminēt, ka gan “Expense”, gan “Revenue” kontu tipiem notiek datu summēšana no iepriekšējiem periodiem – tas ir pēc “Year-to-date” principa.

Dažas palīg-rindas, kas ir ievestas hierarhijās kontu grupēšanas nolūkos, (piemēram “Part1”, “VALIDATION”), ir atzīmētas ar tipu “GroupLabel”. Tas nozīmē ka ievadīt datus šajā kontā nevar, un dati no zemākajam rindām nesummēsies augstāk ne uz šo vecāku, ne uz augstākajiem vecākiem.

Validāciju konti ir atzīmēti ar tipu “Balance”, kas uzvedas līdzīgi “Assets” tipam, bet kura vērtības nav konvertētas citās valūtās. Priekš validācijām tas arī nav vajadzīgi, jo tas vērtības tiks kalkulētas, izejot no vērtībām tabulās.

Kontu tipi priekš valūtas likmēm ir atzīmēti ar tipu “CurrencyRate”, kas nosaka, ka tiem nebūs veiktas nekādas kalkulācijas ar citiem kontiem, un datu summēšanas no iepriekšējiem periodiem.

3. **Konts ir kalkulēts vai nav** (“IsCalculated”). Šī vērtība nosāka, vai hierarhijas lapas loceklis tiks atvērts priekš datu ievades, vai ir aizvērts un vērtības tajā tiks kalkulētas sistēmās noteikumos.

¹⁰

https://docs.oracle.com/cd/E17236_01/epm.1112/hfm_admin_11.1.2.2.300/frameset.htm?ch04s01s01.html

Šī vērtība ir atzīmēta tikai priekš diviem kontiem – slēgšanas validāciju kontiem priekš pirmās un otrās fāzes, jo to vērtība tiks kalkulētā noteikumos.

4. **Konts ir konsolidēts vai nav** (“IsConsolidated”) uz vecākām dimensijām. Šī vērtība ir atzīmēta tikai bāzes tabulu rindu kontiem. Validācijas vērtības un vecāku rindu vērtības ir kalkulētas priekš katras kompānijas atsevišķi.
5. **Kontam var būt vai nebūt starp-kompāniju transakcijas** (“ICP” – “inter-company transactions”). Šis atribūts apzīmē ka vērtības var tikt ievādītas uz “Entity” dimensijas locekļiem un visiem “ICP” (kas satur “Entity” dimensijas bāzes kompānijas) dimensijas locekļu krustojumam, pretējā gadījumā vērtības kompānijām var būt tikai uz [ICP None].
6. **Pievienojum-konts** (“PlugAcct”). Šis konts satur neatbilstības starp vērtībām starp-kompāniju transakcijās. Tādā veidā ja vienai kompānijai ir transakcija ar otro kompāniju (vērtības uz “Entity” un “ICP” dimensiju krustojuma), kas atšķiras pēc vērtības ar datiem ievadītiem otrajā kompānijā, tad vērtības kalkulētas pievienojum-kontā izlīdzinās vērtības uz kompānijas, kas ir šo abu kompāniju vecāks [49].
7. **Pielāgoto dimensiju locekļi** (“CustomTop”). Specifificēti šeit pielāgoto dimensiju locekļi ir derīgi priekš datu apstrādes uz šo locekļu krustojuma ar doto kontu. Visām tabulu rindām šeit ir specificēts konkrētas tabulu kolonas vecākais loceklis (“Custom1” dimensija), un vecākais no datu ievades veida dimensijas loceklis (“Custom3” dimensija), tādā veidā nodrošinot, ka dati var nonākt rindās no jebkāda datu avota.
“Custom2” dimensijas loceklis ir specificēts tikai rindām no tabulas F 20.05, jo šajai tabulai datiem ir jābūt sadalītiem pa valstīm. Visām parējām rindām šis lauks ir tukšs, nodrošinot, ka dati var atrasties tikai “Custom2.[None]” loceklī, pēc noklusējuma.
Validāciju un validāciju slēgšanas kontiem ir specificēts tikai “Custom1” dimensijas loceklis “ValidationSettings”, zem kura ir locekļi, kas tiek izmantoti, lai veiktu validāciju uzstādījumus.
8. **Decimālo ciparu skaits** (“NumDecimalPlaces”). Šeit ir jānorāda cik daudz decimālo skaitļu attēlot pēc noklusējumā, piemēram datu ievades formās, vai SmartView datu izņemšanā.
9. **Rindas vienību izmantošana** (“UsesLineItems”). Šajā eksperimentā nav paredzēts izmantot rindas vērtības datu korekcijai.

10. **Pielāgoto dimensiju agregācijas iespējošanā** (“EnableCustomAggr”). Šī atribūti attēlo vai konts var kalkulēties ar pielāgotām dimensijām vai nē. Šādu atribūtu ir tik daudz, cik ir norādīts pielāgoto dimensiju iekš lietotnes [49].
Tā kā visi definētie locekļi, gan summārie, gan kalkulētie, gan lapas ir izveidotas lai uz tiem glabātu datus un pārskatītu tos, šī atribūti ir atspējoti tikai kontiem atzīmētiem kā “GroupaLabel”.
11. **Lietotāja definēti atribūti** (“UserDefined”). Šajos laukos var pierakstīt jebkādu informāciju 80 zīmju garumā, ko pēc tam var izmantot sistēmas noteikumos. Pavisam ir pieejami trīs lauki priekš šādas informācijas.
Izmantojot šos laukus var automatizēt validāciju kalkulācijas. Tā kā katras validācijas noteikumi izskatās kā vienādībā, katrā no lietotāja definētiem atribūtiem var pierakstīt rindas un kolonas locekļus, kas tiek pārbaudīti no kreisas un no labas vienādojuma puses. Trešajā atribūta var pierakstīt vienādības zīmi, kas tik izmantots validācijā.
12. **XBRL birkas**. Šeit var specificēt XBRL birkas, garumā līdz 225 zīmēm. Tā kā darba mērķis nav saistīts ar gala XBRL datnes izveides, šis lauks ir atstāts tukšs priekš visiem locekļiem [49].
13. **Drošības klases** (“SecurityClass”). Konta nosaukums, kas būs izmantots pieslēdzot lietotāja piekļuves tiesības datiem. Drošības klases nosaukumiem ir tādi paši ierobežojumi kā kontu nosaukumiem.
Šajā darbā kontu drošības klases nosaukums ir veidots pievienojot “SA” (“SecurityClass Account”) un rindas nosaukuma, piemēram “SA_Row_0101_110_0402_010”. Šādā veidā var ierobežot lietotāju piekļuvi tabulām, vai tabulu atsevišķām rindām.
Validāciju drošības klasēm modificēšanas piekļuve ir tikai administratoriem, kas varēs mainīt validāciju uzstādījumus, visiem pārējiem lietotājiem ir tikai lasīšanas piekļuve.
14. **Derīgs “ICP” dimensijas loceklis** (“ICPTopMember”). Līdzīgi kā ar locekļiem no pielāgotām dimensijām, šeit var specificēt “ICP” dimensijas locekli kurš kopa ar saviem bērniem būs derīgs dotajam kontam, ierobežojot kompānijas kurām var būt reģistrētas transakcijas uz dotas rindas. Tā kā šī funkcionalitāte neietekmē FINREP standarta loģiku, bet vairāk biznesa puses loģiku, šis atribūts ir atstāts tukšs visām dimensijām, atļaujot veidot “ICP” transakcijas ar jebkādu kompāniju.

15. **Datu audita ieslēgšana** (“EnableDataAudit”). Šis atribūts nosāka vai vajag reģistrēt darbības kontā (kāds lietotājs ievadīja kādus datus). Lai samazinātu auditējamas informācijas daudzumu šis atribūts ir atzīmēts tikai tabulu rindām.

16. **Kalkulācijas atribūts** (“CalcAttribute”). Tas ir 80 zīmju teksts, kas apraksta kalkulācijas, definētas noteikumos priekš dota locekļa, kas tiek attēlots sistēmā, kā komentāri rindai. Visiem locekļiem šis lauks ir atstāts tukšs [49].

17. **HFM fāzes numurs** (“SubmissionGroup”). Tas ir fāzes numurs, kuram pieder loceklis – ja fāze tiek slēgta, ievade šajā rindā tiek slēgta arī.

Validācijām ir norādīta pēdējā iespējamā fāze, atkarībā no pārbaudītām tabulām. Tā jā validācija pārbauda tabulu F 18.00 un F 20.05, tas fāzes numurs ir tāds pats kā tabulai F 20.05 - 2. Tapāt ir arī pievienotas fāzes validāciju palīg-kontiem. Kontiem, arī ir norādīta fāze, atkarībā no tā kurai daļai pieder tabula – tas ir jā tabula F 20.05 pieder 2.daļai, tad visi konti no šīs tabulas ir atzīmēti priekš 2. fāzes. Valūtas likmes kontiem ir norādīta pirmā fāze, lai pēc pirmās daļas slēgšanas, valūtas likmes vērtības vairs nevarētu mainīt (lai nebūtu situācijas, kad daļa no atskaitēm jau ir gatava, tiek mainīta likme un to dati pēc kalkulācijas kļūst neprecīzi).

18. **Vecāks pēc noklusējuma** (“DefaultParent”). Parasti šajā atribūtā norāda vecāku, kas iet uzreiz virs dota locekļa. Visiem kontiem ir norādīts vecāka nosaukums.

Kad “Account” dimensijas definēšana ir pabeigta, tā izskatās kā attēlā zemāk (4.1.3.1.2.att.).

The screenshot shows the 'Account' dimension hierarchy on the left and the 'Member Attributes' table on the right.

Account Hierarchy:

- Account
 - [None]
 - ExchangeRates
 - FINREP_Rows
 - Part1
 - F_01_01
 - F_04_02
 - Row_0402_020_0000_000
 - Row_0402_030_0000_000
 - Row_0402_040_0000_000
 - Row_0402_050_0000_000
 - Row_0101_100_0402_190
 - F_04_03
 - F_18_00
 - Part2
 - VALIDATION
 - ValPart1
 - ValPart2
 - VLockACPart2
 - v3081_m
 - v3082_m
 - v3083_m
 - v3138_m
 - v3139_m
 - v3965_s
 - v4361_s

Member Attributes Table:

Member	Member List
Member	Row_0402_050_0000_000
Description(English)	of which non-financial corporations
AccountType	ASSET
IsCalculated	<input type="checkbox"/>
IsConsolidated	<input checked="" type="checkbox"/>
IsICP	Y
PlugAcct	
Custom1TopMember	Carr_am
Custom2TopMember	
Custom3TopMember	AllCustom3
NumDecimalPlaces	0
UsesLineItems	<input type="checkbox"/>
EnableCustom1Aggr	<input checked="" type="checkbox"/>
EnableCustom2Aggr	<input checked="" type="checkbox"/>
EnableCustom3Aggr	<input checked="" type="checkbox"/>
UserDefined1	
UserDefined2	
UserDefined3	
XBRLTags	
SecurityClass	
ICPTopMember	
EnableDataAudit	<input checked="" type="checkbox"/>
CalcAttribute	
SubmissionGroup	1
Default Parent	Row_0402_030_0402_050

4.1.3.1.2.att. Vienas rindas piemērs ar atribūtiem.

Validācijas konta piemērs, ar definētiem lietotāja atribūtiem, priekš to kalkulācijas automatizācijas ir attēlots 4.1.3.1.3.attēlā.

Member	v1220_m
Description(English)	F 04.02, c(010), r010 >= sum(r030-050)
AccountType	Balance
IsCalculated	<input type="checkbox"/>
IsConsolidated	<input type="checkbox"/>
IsICP	N
PlugAcct	
Custom1TopMember	AllCustom1
Custom2TopMember	AllCustom2
Custom3TopMember	AllCustom3
NumDecimalPlaces	0
UsesLineItems	<input type="checkbox"/>
EnableCustom1Aggr	<input checked="" type="checkbox"/>
EnableCustom2Aggr	<input checked="" type="checkbox"/>
EnableCustom3Aggr	<input checked="" type="checkbox"/>
UserDefined1	A#Row_0101_110_0402_010.C1#Carr_am
UserDefined2	>=Summary
UserDefined3	A#VRow_0402_030_0402_050.C1#Carr_am
XBRLTags	
SecurityClass	
ICPTopMember	
EnableDataAudit	<input type="checkbox"/>
CalcAttribute	
SubmissionGroup	1
Default Parent	ValPart1

4.1.3.1.3.att. Validācijas konta piemērs ar atribūtiem

Atribūtu definēšana “Account” locekļiem var būt diezgan sarežģīts process, īpaši lai varētu saņemt lielāko izdevību no HFM sistēmas izmantošanas. Tomēr ja dizains ir gatavs, definēt visus atribūtus vienam loceklim var dažu minušu laikā.

Vienas FINREP tabulas hierarhijas projektēšana “Account” dimensijā ar validāciju analīzi var aizņemt 2-3 stundas. Vienas tabulas rindu atribūtu dizaina izstrāde jaunā sistēmā, vai ar specifiskām prasībām var aizņemt apmēram 3-4 stundas, tomēr ja taisīt pēc analogijas ar citām tabulām, tad var aiziet līdz 1 stundai. Tehniski pievienot visus locekļus pēc hierarhijas ar atribūtiem metadatos vienai tabulai aizņem apmēram 1 stundu. Tādā veidā, vairākums no kontu apstrādes iespējām ir jau iebūvētas sistēmā, un tas padara izstrādes laiku diezgan ātro, prasot no sistēmas administratora lielāko darbu tikai dizaina fāzē.

Validāciju pievienošana ir sarežģītākā, jo ir jāanalizē validācijas prasības, un ja nepieciešams, arī jāizveido papildus locekļi, kas apvienos pārbaudītas rindas. Tādā veidā vienas validācijas definēšana un pārskats var aizņemt līdz 0.5 stundai. Kopumā HFM darbības iespēju pārbaudei tika definētas 15 validācijas – ka pārbauda divus datu punktus no vienādojuma, kas

pārbauda rindas zem definētajiem datu punktiem (hierarhijas validācijas), un kas pārbauda vai rindas ir pozitīvas vai negatīvas.

4.1.3.2. *“Custom1” dimensijas definēšana*

Iekš “Custom1” dimensijas ir veidotas tabulu kolonas. Lai šīs dimensijas locekļus varētu ātri izšķirt starp citām dimensijām, tiem ir piemērots šāds nosaukums: “Col_TabulasNumus_KolonasNumurs”, piemēram “Col_0402_020”. Pēc FINREP standarta arī tabulu kolonām ir unikāls identifikators. Locekļa nosaukumam un aprakstam šajā dimensijā ir tādi paši ierobežojumi, kā “Account” dimensijā.

“Custom1” dimensijā ir izveidotas sekojošas hierarhijas:

1. “FINREP_Columns”, kur ir definētas tabulu kolonas, grupētas pēc tabulām, kas ir savukārt grupētas pēc “daļām”. Kolona, kas ir kopēja tabulai F 01.01 un tabulām F 04.02 un F 04.03 atrodas abās hierarhijās, kas atļauj tai summēties gan tur, gan tur, līdzīgi kā bija ar kontu hierarhiju. Tomēr, tādēļ kā bāzes tabulām ir tikai viena kolona, un tabula F 01.01 tiek salikta arī no vērtībām kas iet no citām tabulām, kas nav aprakstīti šeit, tai ir dots unikāls nosaukums (“Carr_am”, jo kolonas nosaukums ir “Carrying amount”).

Līdzīgi, kā rindu, kolonu locekļu hierarhijas noteikšanai tika izmantotas FINREP validācijas.

2. “Validated_Columns” – kas satur locekļus mākslīgai kolonai apvienošanai, lai atvieglotu to pārbaudi pēc validācijām. Locekļu nosaukums šeit ir līdzīgi ka „Account” dimensijā, sākas ar „VCol” un tālāk seko tabulas nosaukums un kolonu nosaukums (no kolonas, līdz kolonai).
3. “ValidationSettings” – kur atrodas locekļi, kas palīdzēs veikt validāciju uzstādījumus, vienkārši ievadot datus “Account” un “Custom1” dimensiju krustojumā. Šeit ir tādi locekļi, kā:
 - a. “ValidationIsSoft” – kas palīdzēs noteikt, ka vērtību kļūdas pēc noteiktās validācijas nepieskaitīsies klāt kopējam kļūdu skaitam, un atļaus slēgt fāzi.
 - b. “ChangePhase” – kas pieskaitīs validācijas kļūdu arī pie tas fāzes, kas ir noteikta šajā loceklī. Tādā veidā, ja pēc kādām vajadzībām lietotāji pieprasīs lai validācijas kļūda liegtu aizvērt fāzi ne tikai fāzē kurai pieder validācija, bet arī agrākajā fāzē (piemēram fāzē, kurai pieder pirmā no pārbaudāmam tabulām), to varētu sasniegt ierakstot nepieciešamas fāzes numuru validācijas kontu un šīs kolonas krustojumā.

- c. “WriteToCellText” – skaitlis šajā locekļī pierakstīs pie validācijas komentāriem vienādojumu ar skaitļiem, pēc kurā noteikumos tika sakalkulēta validācija. Šāda veida uzstādījums var palīdzēt lietotājiem atklūdot validācijas, kur ir daudz pārbaudāmo locekļu. Šī iespēja piederēs tikai bāzes kompānijām, lai vecāko kompāniju konsolidācija neveidotu liekus komentārus uz visām zemāk esošajām kompānijām.
- d. „FailureAmount” – kur pierakstīt sistēmāš noteikumos kopējo skaitļi, kas attēlo kāda ir validēto skaitļu starpība.

Pēc hierarhijas definīcijas, ir jādefinē locekļu atribūti. Vairāki no tiem ir līdzīgi “Account” dimensijai, piemēram – apraksts, vecāks pēc noklusējuma (“DefaultParent”), ir kalkulēts vai nav (“IsCalculated”), drošības klase, fāzes numurs, lietotāja definēti atribūti (arī 3 lauki).

Viens atribūts, kas atšķiras no citiem ir “SwitchSignForFlow” – zīmes maiņa noteiktiem kontu tipiem, kas krustojas ar šo dimensijas locekli, noteicot, vai tas pieder pie debeta vai kredīta kontiem. Līdzīgi uzvedas “SwitchTypeForFlow”, kas maina konta tipu (piemēram “Assets” uz “Expense”). Tā kā tādas kalkulācijas neattiecas uz bāzes FINREP standarta prasībām, bet vairāk uz speciālām biznesa prasībām, šo divu lauku vērtības nav atzīmētas nevienam loceklim.

Pēc hierarhijas un locekļu atribūtu definēšanas “Custom1” dimensija izskatās kā attēlā zemāk.

The screenshot shows the SAP Customizing tool interface. On the left, a tree view displays the hierarchy of the Custom1 dimension. The root is Custom1, which contains several nodes: [None], AllCustom1, [None], FINREP_Columns, Part1_Columns, Part2_Columns, Validated_Columns, and ValidationSettings. Under Part1_Columns, there are nodes F_01_01, F_04_02, F_04_03, and F_18_00. Under Part2_Columns, there is node F_20_05. Under ValidationSettings, there are nodes ValidationIsSoft, ChangePhase, WriteToCellText, and FailureAmount. On the right, the Member Attributes tab is active, showing a table with the following data:

Member	Part1_Columns
Description(English)	Part 1 Columns of FINREP tables
IsCalculated	<input type="checkbox"/>
SwitchSignForFlow	<input type="checkbox"/>
SwitchTypeForFlow	<input type="checkbox"/>
UserDefined1	
UserDefined2	
UserDefined3	
SecurityClass	
SubmissionGroup	1
Default Parent	AllCustom1

4.1.3.2.1.att. Custom1 dimensijas piemērs

4.1.3.3. “Custom2” un “Custom3” dimensijas definēšana

„Custom2” un “Custom3” dimensiju locekļu atribūti ir tādi paši kā „Custom1”.

„Custom2” dimensijā (4.1.3.3.1.att.) tiek glabātas valsts, sadalītas pa reģioniem, lai varētu sadalīt datus pēc valsts, priekš tabulas F 20.05. Šeit ir arī [None] loceklis, kur var glabāt datus priekš visām parējām tabulām.

The screenshot shows the Custom2 dimension hierarchy on the left and its member attributes on the right. The hierarchy is as follows:

- Custom2
 - [None]
 - AllCustom2
 - [None]
 - AllCountries
 - Europe
 - FR
 - DE
 - DK
 - Asia
 - JP
 - CN
 - TH

The member attributes table is as follows:

Member Attributes	Node Attributes	Member List
Member		AllCountries
Description(English)		All countries and regions
IsCalculated		<input type="checkbox"/>
SwitchSignForFlow		<input type="checkbox"/>
SwitchTypeForFlow		<input type="checkbox"/>
UserDefined1		
UserDefined2		
UserDefined3		
SecurityClass		
SubmissionGroup		1
Default Parent		AllCustom2

4.1.3.3.1.att. Custom2 dimensijas piemērs

„Custom3” dimensijā tika izveidoti locekļi priekš datu atšķiršanas pēc datu avotiem. Reālas organizācijas datu avoti varētu būt dažādi, dati var nākt no dažādām sistēmām. Augšupielādēt tos HFM sistēmā varētu izmantojot Oracle Financial Data Management rīku, veicot nepieciešamos datu integrācijas uzstādījumus. Šajā piemēram ir izveidoti trīs locekļi – priekš manuālas ievades, augšupielādētajiem datiem sistēmā un kalkuletājiem datiem (4.1.3.3.2.att.).

The screenshot shows the Custom3 dimension hierarchy on the left and its member attributes on the right. The hierarchy is as follows:

- Custom3
 - [None]
 - AllCustom3
 - [None]
 - ManualInput
 - LoadedData
 - Calc

The member attributes table is as follows:

Member Attributes	Node Attributes	Member List
Member		ManualInput
Description(English)		Manual input
IsCalculated		<input type="checkbox"/>
SwitchSignForFlow		<input type="checkbox"/>
SwitchTypeForFlow		<input type="checkbox"/>
UserDefined1		
UserDefined2		
UserDefined3		
SecurityClass		
SubmissionGroup		1
Default Parent		AllCustom3

4.1.3.3.2.att. Custom3 dimensijas piemērs

Kā vienmēr, loceklis [None] arī tika izveidots, priekš sistēmas iekšējām kalkulecijām, kas nav saistītas ar datu ievadi – validācijas, konsolidācija, valūtas likmes utt.

4.1.3.4. “Entity” dimensijas definēšana

„Entity” dimensijā tiek glabātas kompānija vai nodaļa, kas ietilpst organizācijas struktūrā.

Eksperimentam tika izveidotas 4 mazas kompānijas, divas no kurām ir vienas kompānijas filiāli. Ir izveidota arī papildus hierarhija, kas grupē kompānijas arī pēc ģeogrāfiskas lokācijas.

Šeit locekļu atribūti arī ir daļēji līdzīgi citu, jau apskatīto dimensiju atribūtiem. Atribūti kas atšķiras ir šādi – kompānijas valūta pēc noklusējuma „DefCurrency”, izmaiņu iespējošana (ja šis atribūts ir atzīmēts, tad kompānijai var rediģēt datus, pievienojot informāciju arī ar žurnāliem¹¹), izmaiņu iespējošana no bērniem (vai žurnāli bērnos var vai nē ietekmēt datus vecākajās kompānijās). Katram lapas locekļiem ir iespējota datu koriģēšana ar žurnāliem, kā arī datu ievade krustojumā ar „ICP” dimensiju („IsICP atribūts”). Katram loceklim ir arī pievienota drošības klase, kuras nosaukums veidojas pievienojot „SE_” („SecurityClass” + „Entity”) klāt kompānijas nosaukumam (piemērs – 4.1.3.4.1. att. zemāk).

The image shows two parts of the Oracle EPM interface. On the left is a tree view of the 'Entity' dimension, showing a hierarchy: Entity -> [None] -> BankGroup -> Bank1 -> Bank11, Bank12; Bank2; Bank3; GeoLocations -> Europe -> Bank11, Bank2; Asia -> Bank12. On the right is a 'Member Attributes' table for the 'Bank11' member.

Member Attributes	Member List
Member	Bank11
Description(English)	Bank nr11
DefCurrency	EUR
AllowAdj	<input checked="" type="checkbox"/>
IsICP	<input checked="" type="checkbox"/>
AllowAdjFromChildren	<input type="checkbox"/>
SecurityClass	SE_BANK11
UserDefined1	
UserDefined2	
UserDefined3	
HoldingCompany	
SecurityAsPartner	
Default Parent	Bank1

4.1.3.4.1.att.Entity dimensijas piemērs

4.1.3.5. “Scenario” dimensijas definēšana

Lietotnei tika izvēlēts scenārijs „Actual”, lai atspoguļotu, ka ievade notiek pašreizējā periodā ar pašreizējiem datiem.

Vadoties pēc lietotāja dokumentācijas¹², šajā scenārijā tika norādītas šādas vērtības atribūtiem - 4.1.3.5.1.attēls.

¹¹ https://docs.oracle.com/cloud/latest/epm-common/SVPBC/about_oracle_journals_for_financial_management.htm#SVPBC-journals_1

¹² https://docs.oracle.com/cd/E57185_01/HFMAD/ch04s05.html

Scenario	
ACTUAL	
Member Attributes	Member List
Member	ACTUAL
Description(English)	Actual
DefaultFreq	QTD
DefaultView	QTD
ZeroViewForNonadj	QTD
ZeroViewForAdj	QTD
ConsolidateYTD	<input checked="" type="checkbox"/>
UserDefined1	
UserDefined2	
UserDefined3	
SupportsProcessManagement	Y
SecurityClass	
MaximumReviewLevel	5
UsesLineItems	<input type="checkbox"/>
EnableDataAudit	0
DefFreqForCTrans	QTD
PhasedSubStartYear	
Default Parent	#root

4.1.3.5.1. att.Scenario dimensijas piemērs

Atskaišu periodiskums visur tika norādīts kā „QTD”, lai attēlotu, kas tas ir ceturkšņa bāzētas atskaites.

4.1.3.6. “Currency” definēšana

„Currency” uzstādījumi ietver sevī visas sistēmā definētas valūtas. Locekļiem šeit ir papildus atribūti, tādi kā – mērogs („Scale”), kas attēlo kādā formātā ir ievadītas valūtas likmes (0 apzīmē vienības); konvertēšanas operators (izvēlētais D apzīmē kā lai aprēķinātu naudu šajā valūtā ir jāsadala vērtības ar valūtas likmi); un iespējamība izvēlēties valūtu definējot iekš-kompānijas transakciju (arī tika atzīmēts priekš visām valūtām [50]).

	Currency	Description(English)	Scale	TranslationOperator	DisplayInICT
1	CAD	Canadian Dollar	0	D	<input checked="" type="checkbox"/>
2	CNY	Chinese Yuan	0	D	<input checked="" type="checkbox"/>
3	DKK	Denmark Crona	0	D	<input checked="" type="checkbox"/>
4	EUR	Euro	0	D	<input checked="" type="checkbox"/>

4.1.3.6.1.att. Currency dimensijas piemērs

4.1.4.HFM datu ievades formas

Kad metadati ir definēti un augšupielādēti sistēmā, var pārbaudīt kā izskatās visi definētie locekļi, un vai visi nepieciešamie locekļi ir atvērti uz krustojuma ar citām dimensijām.

Šo var izdarīt izveidojot datu ievades formas. Tas ir tabulas, pieejamas „Workspace” lapaspusē, kur tos var arī uztaisīt un rediģēt. Ievadot datus tur, tie uzreiz tiek pierakstīti datubāzē. Virs tabulas ir dimensiju saraksts, no kura var izvēlēties kuru dimensiju krustojumā ievadīt datus. Tabulas var veidot ar iebūvēto „vilkt-un-mest” funkcionalitāti, vai arī rakstīt ar skriptu. Šūnas kas ir atzīmētas ar dzelteno krāsu ir atvērtas datu ievadei.

Ir iespējams arī veidot saistītas formas, lai ievadītu datus nevis pa tiešu uz doto tabulu, bet uz saistīto tabulu. Tā var ievadīt datus F 01.01 formas rindās, kas ir saistītas ar F 04.02 un F 04.03 tabulām.

4.1.5.HFM noteikumu definēšana

HFM noteikumos ir sarakstītas kalkūlācijas, kas notiek starp dimensiju locekļiem. Šeit ir aprakstītas validāciju kalkūlācijas, Visual Basic valodā.

„Account” un „Custom1” dimensijas locekļi, definētie validāciju „UserDefined1/3” atribūtos, tiek pārbaudīti pēc zīmes un noteikumiem definētiem „UserDefined2” atribūtā.

Piemēram, noteikumi, kas izskaitļo starpības validācijās, kas pārbauda divus datu punktus vai tie ir lielāki viens par otru, vai nē (4.1.5.1.att.).

```
Sub Calculate()
' Run for base level entities only
If HS.Entity.IsBase("", "") Then

    vallist = HS.Account.List("VALIDATION", "[Base]") 'getting list of all validations
    For each val in vallist
        ud2List = HS.Account.UD2(val) 'getting UserDefined property

        Dim udArray() As String
        udArray = Split(ud2List, ".") 'splitting sign of equality of validations
        if udArray(2) = "Summary" then 'for validations checking data on summary level
            'checking difference between two part of equation
            HS.Clear "A#" & val & ".C1#FailureAmount.C2#[None].C3#[None].I#[ICP NONE]"
            difference = HS.Account.UD1(val) & "." & ALLTOPSXCL - HS.Account.UD3(val) & "." & ALLTOPSXCL
            if difference <> 0 then
                if udArray(1) = ">=" then
                    if HS.Account.UD1(val) & "." & ALLTOPSXCL < HS.Account.UD3(val) & "." & ALLTOPSXCL then
                        'write difference to the amount of failure
                        HS.EXP "A#" & val & ".C1#FailureAmount.C2#[None].C3#[None].I#[ICP NONE] = " & difference
                    end if
                elseif HS.Account.UD1(val) & "." & ALLTOPSXCL > HS.Account.UD3(val) & "." & ALLTOPSXCL then
                    HS.EXP "A#" & val & ".C1#FailureAmount.C2#[None].C3#[None].I#[ICP NONE] = " & difference
                end if
            end if
        end if
    Next val
End Sub
```

4.1.5.1.att. HFM noteikumu piemērs validācijām, kas pārbauda summāras vērtības

Izpildoties sistēmā, šāds kods pārbaudīs validācijas, kurām iekš “UD” atribūtiem ir definēti noteiktie pārbaudījumi. Piemēram, ja iekš “UserDefined2” atribūta ir definēts teksts “>=.Summary”, tad sadalot šo tekstu pēc punkta, uz divām masīva daļām tiks pārbaudīts vai pirmā daļa, ko pārbauda validācija ir “>=”, nekā otrā vai nē. Zinot otro “Summary” daļu, kas ir definēta atribūtā, kods izmantos gan kreiso, gan labo vienādojuma no “UserDefined1” un no “UserDefined2”, izmantojot tos kā ārējās atslēgas. Tā piemēram, ja validācijas “UserDefined1” ir “A#Row_0101_110_04_02_010.C1#Carr_am”, un “UserDefined2” ir “A#VRow_0402_030_0402_050.C1#Carr_am”, kods pārbaudīs vai rinda no pirmās tabulas ir lielāka vai vienāda par rindu no otrās tabulas (abām rindām tiks pārbaudīta kolona “Carr_am” (“Carrying amount”)). Šeit var atzīmēt vēl ka rinda VRow_0402_030_0402_050 ir mākslīga – izveidota speciāli priekš validācijas, jo sākas ar “V”.

Pārbaude, kad validācijas formulā ir definēts pārbaudīt nevis tieši divas, vai summāras šūnas, bet salīdzināt katru šūnu no kolonas, izmanto vārdu “Hierarchy”, nevis “Summary” iekš “UserDefined2” atribūta. Šajā gadījumā kods ciklā pārbauda visus atribūtus, kad pēc hierarhijas atrodas zem rindas, kas ir definēta iekš “UserDefined1” vai “UserDefined3” atribūtiem.

Var secināt ka izmantojot HFM hierarhiskās attiecības starp dimensiju locekļiem, un lietotāja definētus atribūtus var samērā ērti pārbaudīt datu validācijas.

Tomēr ir jānorāda, ka koda atklūdošana iespējas nav plašas. Nav pieejama koda redaktora, tādēļ kodu var rakstīt piemēram Notepad++ programmā. Kļūdas tiek atrastas tikai kad kods ir palaists HFM sistēmā. Ja kodā ir daļas, kas piemēram pārbauda kompāniju uz kuras tika palaista kalkulācija ar „if” konstrukciju vērsās pie locekļa, kura nav sistēmā, kods nekad neizpildīsies, un nekur nebūs par to kļūdas pieraksta. Tādēļ, rakstot noteikumus HFM sistēmā ir jāpievērš daudz uzmanības koda pārskatam un testam.

4.2. Eksperimenta realizācija Pentaho Community Edition rīkā

Eksperimenta tika izmantota Pentaho CE versija 7, kurā ietilpst arī Mondrian OLAP servera versija 3. Atšķirībā no Mondrian 4 versijas, kas ir iekļauta Pentaho Enterprise Edition versijā, datu noliktavas shēmas (datnes ar .XML paplašinājumu) priekš Mondrian 3 var modificēt izmantojot Schema Workbench. Mondrian 4 versijas shēmas pagaidām var modificēt tikai teksta redaktora programmās. Mondrian 4 ir vairāki uzlabojumi salīdzinot ar Mondrian 3 (piemēram “What-if” analīze), tomēr Mondrian 4 versija bija pēdējo reizi izlaista 2015.gadā, kamēr pēdējais atjauninājums Mondrian 3 versijai iznāca 2016.gada Novembrī¹³.

Vajag atzīmēt ka Mondrian 4 “what-if” analīze varētu simulēt lietotāju datu ievadi tabulu šūnās. Tā ļauj Saiku pievienojumprogrammā izvēlēties vaicājuma scenārija funkcionalitāti un modificēt datus šūnas, pie tam Mondrian pārkalkulēs šūnas, kas ar atkarīgas no izmainītām šūnām [51]. Šādā pieeja varētu atvieglot lietotājiem datu verificēšanas procesu, ļaujot saprast kādas vērtības ir jāpamaina avotu datus, lai datu noliktavā validācijas atzīmētu šūnas par pareizajām.

No sākuma eksperimentu bija plānot veikt integrējot Pentaho ar Palo OLAP serveri, kas atļauj datu rakstīšanu. Tomēr šāda integrācija neizdevās. Palo serveris ir pieejams kopā ar Jedox darījumu inteliģences rīku. Jedox bezmaksas versija – Jedox Base pēc instalācijas izdeva kļūdu, ka licence nav kārtībā. Izradījās, ka lai izmantoti Jedox Base rīku ir jāsniedz savas kompānijas dati Jedox organizācijai, un jāgaida kamēr viņi piezvanīs lai apspriestu Jedox Base licences sniegšanu. Bija izmēģināts arī instalēt Jedox pilno versiju, ar izmēģinājuma licenci, tomēr šajā gadījumā arī bija konstatēta kļūdā – pēc instalācijas tika parādīts paziņojums ka licences derīguma termiņš bija līdz 2016.gadam. Abos gadījumos visi mēģinājumi pievienoties Palo serverim neizmantojot Jedox beidzas ar kļūdu ka licence nav derīga.

Papildus šim bija konstatēts kā pieejamais atvērtā koda rīks darbam ar Palo – JPalo pēdējo reizi tika atjaunots 2008.gadā¹⁴. Kamēr pats Palo kā atsevišķais atvērtā koda serveris pēdējo reizi tika atjaunots 2010.gadā¹⁵.

Tādēļ ņemot vērā visu augstākminēto tiks nolemts veikt eksperimentu izmantojot Mondrian serveri, versiju 3.

¹³ <https://sourceforge.net/projects/mondrian/files/mondrian/mondrian-4.3.0/>

¹⁴ <https://sourceforge.net/projects/jpalo/files/jpalo%20client/JPalo%20Client%202.0/>

¹⁵ <https://sourceforge.net/projects/palo-etl-server/files/etlserver/>

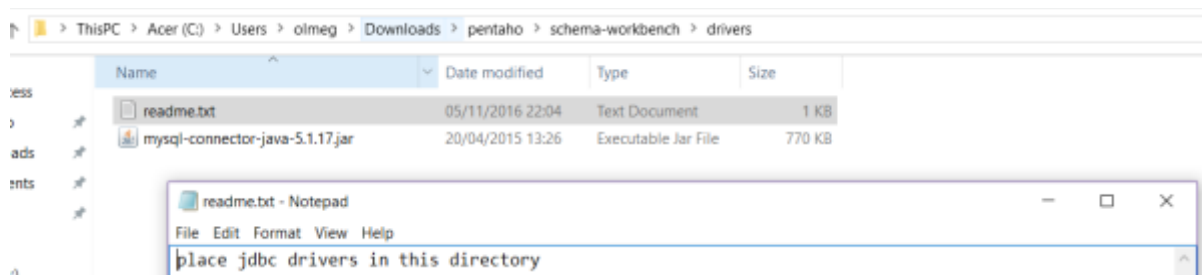
4.2.1. Pentaho CE rīka instalācija

Pentaho rīka instalācija ir daudz vieglākā nekā HFM. Serveris, kā arī citi izstrādātāju ieteikti darbā ar Pentaho rīki ir pieejami “Sourceforge.com” mājaslapā savukārt sakārtotās saites uz tiem var atrast Pentaho mājaslapā¹⁶. Visi rīki ir lejupielādējamie datu arhīvos un rīku palaišanai ir jāpalaiž datne ar .bat paplašinājumu (priekš Windows operētājsistēmas, un .sh attiecīgi pie Linux bāzētām sistēmām). Servera instalācijas datnēs ir iekļauts arī Mondrian OLAP un Tomcat serveris.

Pirms rīku palaišanas ir jāuzstāda sistēmas mainīgie. Pentaho dokumentācijā ir norādīts tikai mainīgais “PENTAHO_HOME”, tomēr tikai šī mainīgā uzstādīšanas nepietiek – ir jādefinē vēl “JAVA_HOME” un “JRE_HOME”. Šos uzstādījumus var veikt automātiski – izmantojot .bat datni “set-pentaho-env.bat”, kas ir arī iekļauta servera arhīvā.

Vēl viena lieta, ko būtu labi izdarīt uzstādot Pentaho ir to savienošana ar datubāzi. Pēc noklusējuma, Pentaho strādā ar datubāzi Hibernate, kas palaižas katru reizi ar servera startu. Lai efektīvi izmantotu servera resursus Pentaho ir jāuzstāda darbam ar kādu no datubāzēm. Šeit der gan relāciju, gan NoSQL datubāzes. Viena no lietotāju kopienes populārām datubāzēm ir atvērta koda MySQL relāciju datubāze. Tā kā autoriem jau bija pieredze darbā ar to, MySQL tika izvēlēts integrācijai ar Pentaho. Tādēļ papildus solis Pentaho instalācijā bija vēl MySQL lejupielādē un instalācija.

Lai Pentaho serveris varētu pievienoties MySQL datubāzei ir jāiekopē datne “mysql-connector-java.jar” uz “drivers” mapi, kur jau tiek glabāta datne ar attiecīgo ieteikumu lietotājam . Līdzīgais darbs ir jāpaveic arī ar citiem Pentaho rīkiem, respektīvajās mapēs (piemēram iekš Schema Workbench rīka (4.2.1.1.att.)).



4.2.1.1. att. Schema workbench direktorijas mapē, kur ir jāievieto MySQL .jar datne

Pēc tam ir jāveic arī Pentaho sistēmas datubāzu izveide MySQL datubāzē, palaižot SQL skriptus no Pentaho mājaslapas. Procesa aprakstu, kā veikt šādus uzstādījumus ir pieejams

¹⁶ <http://community.pentaho.com/>

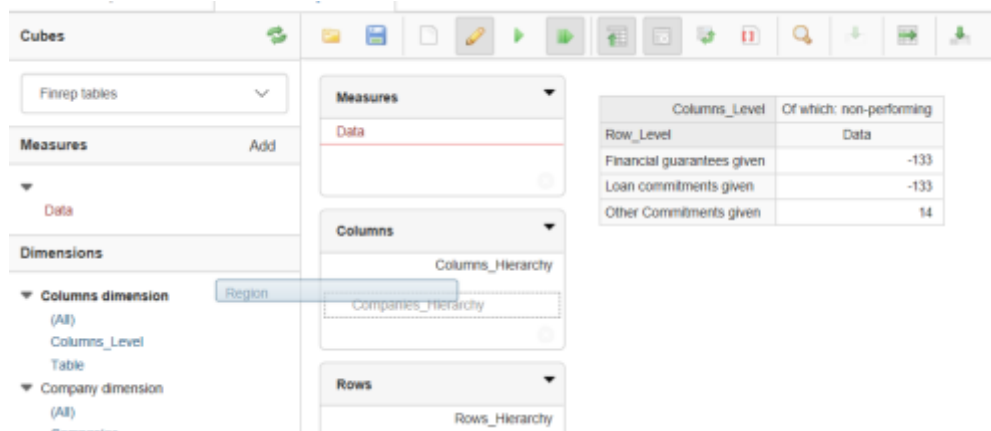
Pentaho mājaslapā¹⁷. Kopā ar datubāzēm tiek izveidoti arī to lietotāji, kas ir norādīti arī Pentaho “.properties” datnēs, tādēļ, ja serveris tiks izmantots produkcijā, šos lietotājus un to paroles labāk ir nomainīt (4.2.1.2.att).

```
backup.sql applicationContext-spring-security-hibernate.properties
1 jdbc.driver=com.mysql.jdbc.Driver
2 jdbc.url=jdbc:mysql://localhost:3306/hibernate
3 jdbc.username=Olga
4 jdbc.password=test123
5 hibernate.dialect=org.hibernate.dialect.MySQL5Dialect
```

4.2.1.2. att. piemērs MySQL uzstādījumiem un lietotāja datu uzstādījumiem Pentaho .properties datnē

Pentaho pēc noklusējuma būs pieejams pēc adreses - <http://localhost:8080/pentaho/> , tādēļ pirms servera palaišanas ir arī jāpārlicinās ka 8080 ports ir pieejams. Vairāk servera instalācijai nekādu soļu veikt nevajag – ir jāpalaiž “start-pentaho.bat” un serveris būs pieejams.

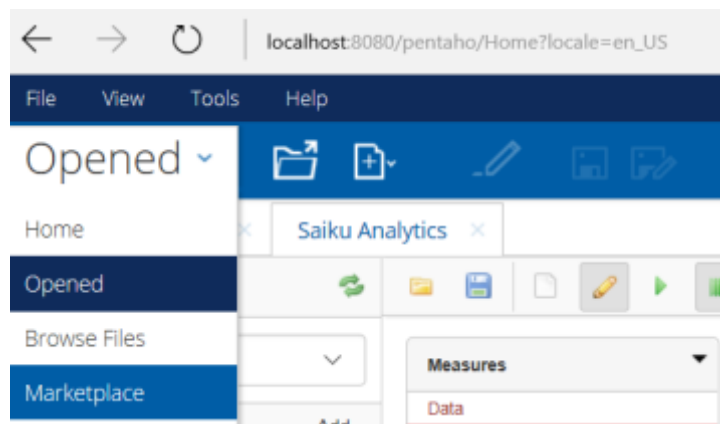
Papildus serverim, tika veikta arī Saiku Analytics atvērtā koda pievienojumprogrammas instalēšana. Saiku pievienojumprogramma tiek izmantota lai varētu ērti veidot datu atskaites no datu noliktavas. Pentaho maksās Enterprise Edition versijā šim nolūkam ir pieejams Pentaho Analyzer [52]. Pentaho CE versijā pēc noklusējuma ir tikai JPivot, kas atbalsta bāzes datu atskaites, un nav tik ērts izmantošanā kā Saiku. Saiku pievienojumprogrammā var veikt datu atlases pēc dimensijām, dimensiju hierarhijām un līmeņiem izmantojot “vilkt un nomest” funkcionalitāti (4.2.1.3. att.), kā arī saglabāt datu atlases vaicājumus, kas ir MDX valodā, kurus lietotāji varēs vēlāk atvērt un izmantot kā datu atskaites.



4.2.1.3. att. piemērs Saiku pievienojumprogrammas darbības piemērs

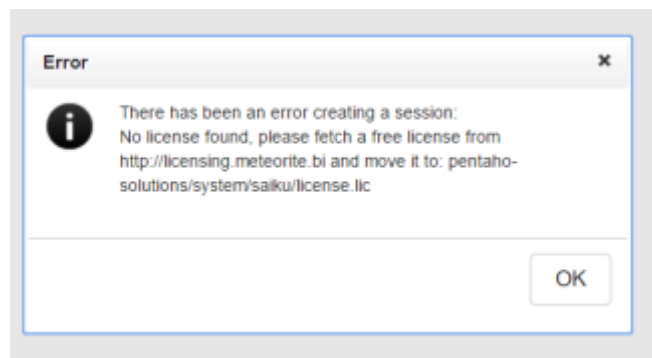
Saiku ir iespējams ērti instalēt no Pentaho/Marketplace vietnes, pieejamas izvēlnē instalētājā servera vietnē (4.2.1.4. att).

¹⁷“Initialize MySQL DI Repository Database”
<https://help.pentaho.com/Documentation/6.1/OF0/OP0/030/020>



4.2.1.4. att. Pentaho Marketplace saite servera vietne

Uzreiz pēc instalācijas tiks atspoguļotā kļūda, kā trūkst licences, ko ir jāizveido pēc adreses: <http://licensing.meteorite.bi/login> (4.2.1.5. att). Licences izveidei ir nepieciešams darbojošs epasts, servera nosaukums, kompānijas nosaukums (kas nekā nav pārbaudīts) un maksimālais lietotāju skaits, kas izmantos šo pievienojumprogrammu.



4.2.1.5. att. Saiku licences kļūda

Pentaho instalācijas beigās var pārbaudīt arī savienojuma izveidi ar MySQL datubāzi. Priekš tam ir nepieciešams ievadīt tikai servera adresi kur datubāze atrodas (šajā gadījumā “localhost”) datubāzes nosaukumu, lietotāja datus pieslēgšanai ar datubāzi. Porta vērtība jau ir definēta, bet pēc nepieciešamības to var izmainīt. Pentaho ir pieejamas ne tikai MySQL bet arī citu datubāžu priekš-definētie uzstādījumi (4.2.1.6. att.), kas var palīdzēt lietotājam ātri izveidot savienojumu ar nepieciešamo datubāzi.

4.2.1.6. att. Pentaho pieejamie datubāžu savienojumi

Datu noliktavas izstrādes ērtībai var iespējot Mondrian SQL un MDX vaicājumu pierakstu žurnāl-datnē [53]. Lai to paveiktu ir jāmodificē log4j.xml datni atkomentējot nepieciešamas rindas.

4.2.2. Datu noliktavas fiziskās shēmas izveide

Atšķirībā no Oracle HFM rīka, kur vairums no dimensiju ir jau priekš-definēts, un nav jā rūpējas par faktu tabulu, shēmas izveide Pentaho rīkā ir pilnīgi atkarīga no datu noliktavas izstrādātāja. Veidot fiziskās tabulas MySQL datubāzē var gan izmantojot Pentaho datu integrācijas rīku Kettle, vai arī izmantojot MySQL rīkus, kā piemēram MySQL Workbench.

Shēma ir izstrādāta sniegpārslas veidā, ar vienu fakta tabulu un vairākām mākslīgām faktu tabulām, kas ir nepieciešamas papildus informācijas grupēšanai.

Daļa no dimensijām tika izveidota līdzīga kā HFM rīkā:

1. Account (priekš tabulu rindām), ar iekšējo hierarhiju Tabula ← Rinda.
2. Custom1 (priekš tabulu kolonām), ar iekšējo hierarhiju Tabula ← Kolona.
3. Custom2 (priekš valstīm), ar iekšējo hierarhiju Reģions ← Valsts.
4. Time (priekš laika periodiem), Gads ← Ceturksnis ← Mēness.
5. Custom3 (priekš datu avotiem), bez iekšējas hierarhijas.
6. Entity (priekš kompānijām), Reģions ← Kompānija (šeit reģions ir ģeogrāfiskā lokācija kur atrodas kompānija, kas var būt atšķirīgs no reģiona Custom2 dimensijā kur ir domāts reģions par darījumiem ar kuru kompānija var raportēt datus).

7. Currency (priekš datu konvertēšanas citās valūtās), bez iekšējas hierarhijas.

Dažām dimensijām (kā Entity, Account, Custom1) ir pievienota kolona “ChildOf”, lai varētu attēlot vecāku-bērnu attiecības starp atribūtiem. Tālāk tādas attiecības tiks nodemonstrētas uz Entity dimensijas piemēra.

Papildus FINREP tabulu rindu un kolonu definīcijām, informācija šeit tiks sadalīta arī pēc datu punktiem. Tā kā katrs ieraksts faktu tabulā ir attiecināms uz vienas šūnas vērtību dimensiju nogriezumā, tika lemts izmantot datu punktu definīcijas, pieejamas FINREP standartā (piemērs 4.2.2.1. att).

F 20.05.a - Geographical breakdown of off-balance sheet items subject to credit risk by residence of the counterparty (a)					
Sheet per Country of residence of the counterparty		(RCP:GA) Residence of counterparty (GA:GAS_1 ex. default ('all')) <Key value>			
		Columns			
		Nominal amount	Of which: debt forbearance	Of which: non- performing	
		010	022	025	
Rows	Loan commitments given	010	39714 €€\$	110962 €€\$	110972 €€\$
	Financial guarantees given	020	39711 €€\$	€€\$	110970 €€\$
	Other commitments given	030	39728 €€\$	€€\$	110977 €€\$

4.2.2.1. att. Tabulas F 20.05a definētu datu punktu piemērs, no dokumenta “DPM table layout and data point categorisation.zip / Annotated Table Layout 250-FINREP 2016-A.xlsx”¹⁸

Tas atrisinās vairākas validācijas, saistītas ar vienādiem datu punktiem vairākās tabulās. Tādā veidā ja dati ir ielādēti vienā datu punktā, tie var atspoguļoties vairākās tabulās, un var neveidot sistēmā validācijas, kas pārbauda ka dati šajās šūnās ir vienādi. HFM tāds piegājiens nebija ērts, jo tur ir tradicionāls piegājiens dimensiju krustojumiem – gan datu atskaites, gan datu formas tiek deklarētas kā rindas reiz kolonām, nevis kā atsevišķu datu punktu virkne.

Validācijas ir pievienotas arī atsevišķajā dimensijā, ar to identifikācijas nosaukumu, to datu pārbaudes formulu un vienādības zīmi.

Mākslīgajās faktu tabulās tiek glabāta informācija priekš :

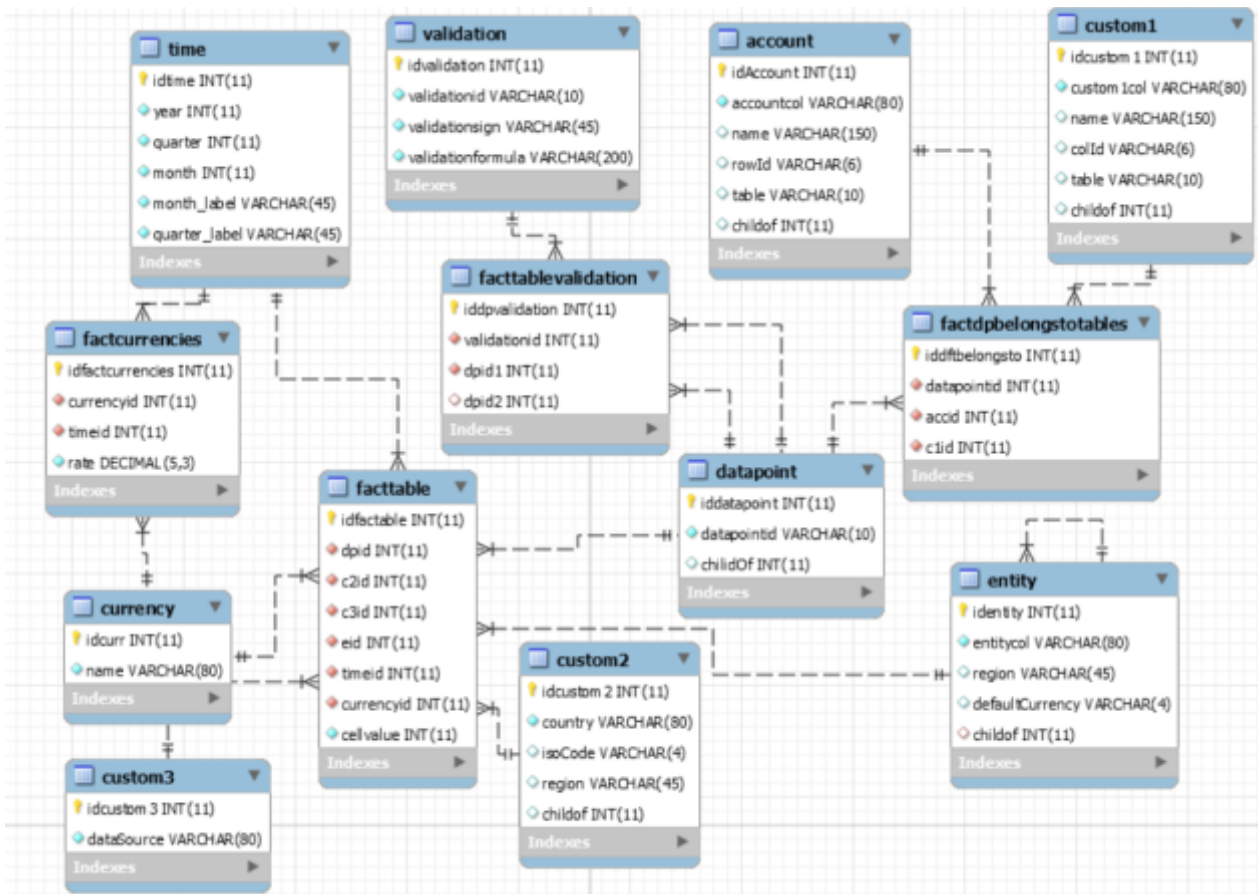
1. Valūtu likmēm pa laikā periodiem (tabula “factCurrencies”). Apvienojot FINREP datu kubu un valūtas kubu viena virtuālajā kubā (kas ir vairāku kubu apvienojums), var attēlot datus vairākās valūtās.
2. Rindu un kolonu apvienojumam ar datu punktiem (tabula “factDPBelongsToTables”). Tādā veidā, lai varētu datus attēlot lietotājam saprotamā veidā, ar kolonu un rindu nosaukumiem pēc tabulām nevis vienkārši

¹⁸ <https://www.eba.europa.eu/risk-analysis-and-data/reporting-frameworks/reporting-framework-2.5>

datu punktu identifikatoriem, izmantojot Mondrian iebūvētu JOIN funkcionalitāti datus var saņemt nogriezumā Finrep rinda /kolona, nevis tikai pa datu punktiem.

- Validāciju un to pārbaudīto locekļu apvienojuma (tabula “factTableValidation”). Tādā veidā katru datu punktu apvienojumu kas ir norādīts šajā faktu tabulā var pārbaudīt pēc vienādojuma zīmes, kas ir glabāts pie validāciju definīcijām. Ja validācija pārbauda datu punktu summu pa vairākām tabulām, var veidot mākslīgus datu punktus, kas apvienos sevī datu punktus, pārbaudītus no vienādojuma labas un kreisās puses.

Pilna datu noliktavas shēma ir atspoguļotā 4.2.2.2. attēlā.



4.2.2.2. att. MySQL fiziskā datu noliktavas shēma

Lai parādītu validācijas piemēru tika izveidots viens skats, kas apvieno sevī visas validācijas, kas pārbauda šūnas, kurām ir jābūt lielākam vai vienādām ar nulli. Datu skatus var veidot arī Mondrian shēma, tomēr glabātu datu atlasē vienā vietā un atvieglotu shēmas lasīšanu, skats tika saglabāts MySQL datubāzē. Tā kā dati faktu tabulā ir sadalīti pa laika periodiem, bet validācijas informācija ir universālā (vieni un tie paši datu punkti ir pārbaudīti gan vienā mēnesī, gan otrā), skats kalpo kā savā veidā faktu tabula, apvienojot sevī validācijas, laiku, datu punktu

vērtības kas ir zemāk par nulli, kā arī citas dimensijas. Lai Mondrian varētu skaitīt cik datu punktu neizgāja validācijas pārbaudi, skatā ir pievienota kolona ar unikālām vērtībām, kas ir visu iekļauto dimensiju rindu unikālo identifikatoru apvienojums (piemērs 4.2.2.3. attēlā).

idvalc	validabo	validationformule	celvalue	table	row	col	year	quart	month_label	timeid	eid	c2id	c3id	rowid
4	v3965 s	{F 20.05.a} >= 0	-133	F 20.05	020 Financial o...	025 Of w...	2017	O2	April	4	9	2	2	49226
4	v3965 s	{F 20.05.a} >= 0	-12	F 20.05	010 Loan com...	025 Of w...	2017	O1	Januarv	1	8	2	2	48225
4	v3965 s	{F 20.05.a} >= 0	-121	F 20.05	010 Loan com...	025 Of w...	2017	O1	Januarv	1	8	1	1	48115

4.2.2.3. att. Validācijas skata datu atlasē piemērs

Vairums no informācijas dimensijās ir tehnisks, un ir aizpildāms ar datu noliktavas administratoru – kā piemēram rindu un kolonu nosaukumi, validācijas definēšana utt. Veids kā automatizēt šādas informācijas ielādēšanu uz datu noliktavu ir pamats turpmākajam pētījumam un nav piedāvāts šajā darbā. Manuālā datu aizpilde ir līdzīga pēc laika HFM rīkam – tā piemēram, lai aizpildītu informāciju par vienu validāciju un to pārbaudāmajiem datu punktiem var būt nepieciešams līdz 0.5 stundai.

Definējot lietotāju piekļuvi datubāzes objektiem ir jānodrošina, ka rakstīšanas tiesības šādām tabulām ir tikai administratoriem. Darbā nav atspoguļots arī datu augšupielādes veids uz datu noliktavu. Tā kā datu avotu skaits un informācijas pieejamība ir atkarīgs no konkrētas iestādes resursiem, datu avotu tabulā ir pievienoti tikai daži potenciālie locekļi. Lai augšupielādētu datus no pieejamajiem resursiem, varētu tikt izmantots Pentaho datu integrācijas rīks Kettle, kurā var definēt vairākus avotus, tajā skaitā datnes, un vairākus galamērķa punktus, kā piemēram datubāžu tabulas.

4.2.3. Datu noliktavas Mondrian shēmas izveide

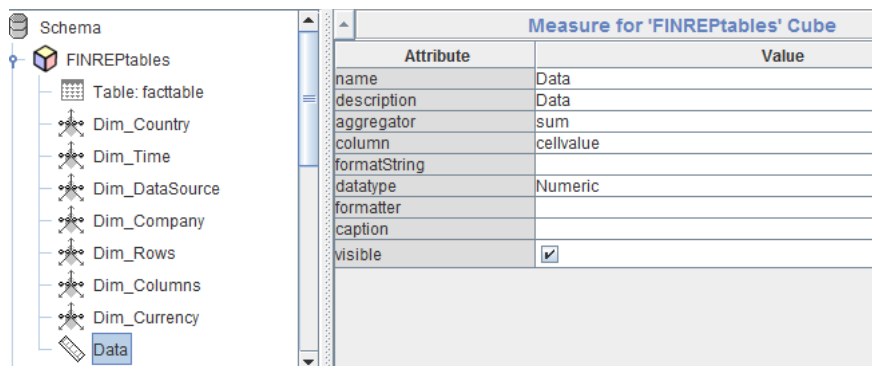
Mondrian datu noliktavas shēma, ko izmanto Pentaho datu vaicājumu veidošanai ir .XML formātā. Kā jau bija minēts augstāk, darbā tika izmantota 3.versija Mondrian OLAP serverim, kuras shēmas veidošanai ir iespējams izmantot rīku Schema Workbench. Pirms sākt definēt shēmu no fiziskās datubāzes, ir jānodrošina kā ir iespējams pievienoties šajā datubāzei. Schema Workbench šī funkcionalitāte pieejama zem “Options/Connections..” un ir līdzīga kā Pentaho servera tīmekļa vietnē.

Veidojot shēmu un ievadot tajā izmaiņas, ir ieteicams veikt to rezerves kopiju. Shēmas apraksta atribūta, vai anotācijā var norādīt arī kurš un kad veica pēdējās izmaiņas tajā. Tāda pieeja var palīdzēt shēmas ātrai atgūšanai, gadījumā ja tika ieviesta kāda kļūda [54].

Visas dimensijas attēlotas MySQL tabulās ir arī atspoguļotas kā Mondrian dimensijas.

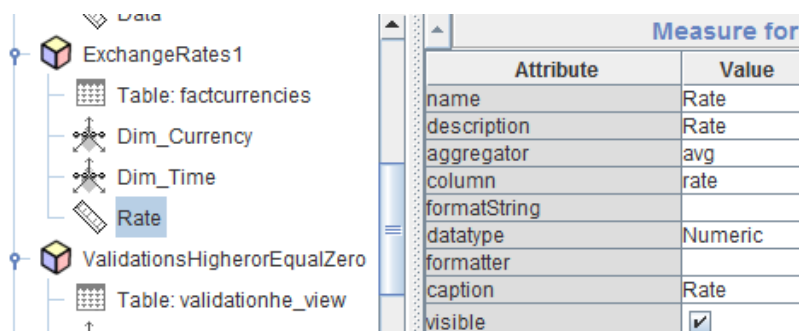
Ir pievienoti šādi datu kubi (visu datu kumu shēmas ir darba 1.pielikumā):

1. FINREPTables – kas ir faktu tabulas “factTable” atspoguļojums. Mērījums datu agregācijai ir norādīts kā Mondrian iebūvēta funkcija “sum”, tādā veidā veidojot atskaites pa vairākām hierarhijas līmeņiem, dati tiks summēti, lai atspoguļotu totālas vērtības (4.2.3.1. att.).



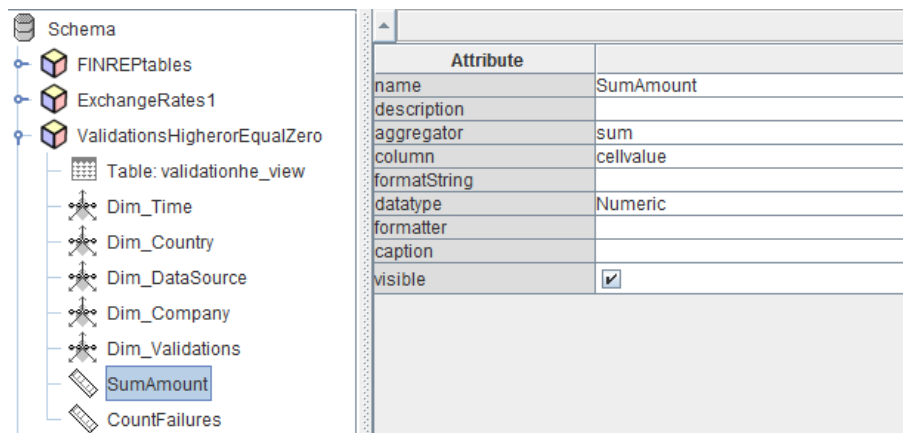
4.2.3.1. att. **FINREPTables kuba saturs**

2. ExchangeRates – kas satur sevī laika un valūtas dimensijas, kas ir tabulas “factCurrency” atspoguļojums (4.2.3.2. att.). Šo kubu var izmantot lai apskatītu valūtas pa laika periodiem. Valūtas dimensijā nav definēti līmeņi hierarhiskām attiecībām, kā ir Laika dimensijā (Gads/ceturksnis/mēness). Datu atlasēm, kas izmantos Laika dimensijas hierarhijas, šajā kubā ir norādīt mērījums “avg”, kas atspoguļos valūtas kursa vidējo aritmētisko pa ceturkšņiem un gadiem. Šādā veidā, pēc nepieciešamības varēs atsekt valūtas kursu izmaiņas pa gadiem.



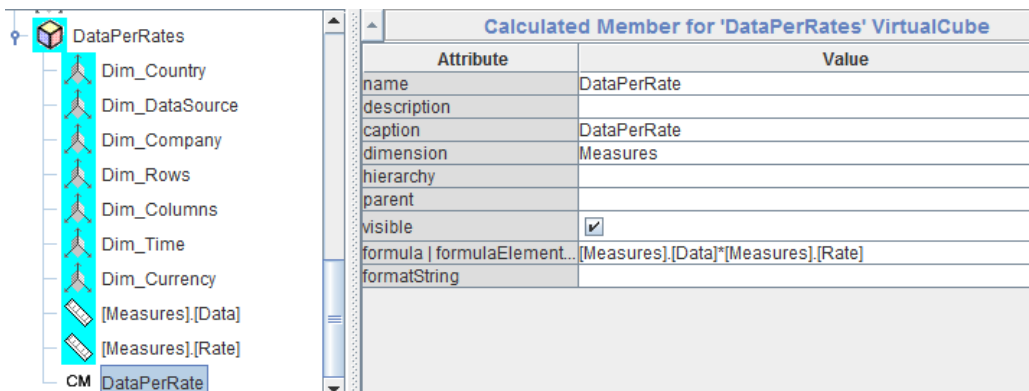
4.2.3.2. att. **ExchangeRates kuba saturs**

3. ValidationHigherOrEqualZero – kas satur dimensijas validāciju pārbaudei (4.2.3.3. att.). Šeit ir pievienoti arī divi mērījumi, izmantojot Mondrian priekš-definētas agregācijas funkcijas – “sum” priekš datu summēšanas, kas neizgāja pārbaudi, kā arī “count”, lai skaitītu datu punktus ar negatīvajām vērtībām. Šajā kubā kā faktu tabula ir norādīts skats, veidots speciāli priekš šādām validācijām, kas tika aprakstīts iepriekšējā nodāļā.



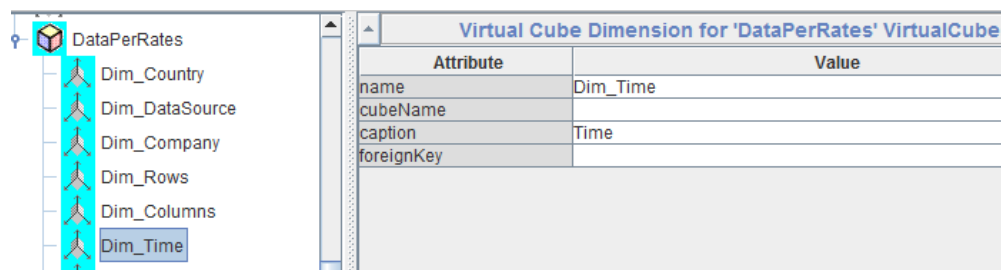
4.2.3.3. att. ValidationHigherOrEqualZero kuba saturs un “SumAmount” mērijuma uzstādījumi.

Ir pievienots arī viens virtuāls kubs, priekš datu atspoguļojuma pēc valūtas kursiem. Tas iekļauj sevī visas dimensijas priekš datu atlasēs, kā arī mērijumus no abiem kubiem – FINREPTables un ExchangeRates. Kubam pievienots viens kalkulēts loceklis, priekš datu konvertācijas pēc valūtu kursa (4.2.3.4. att).



4.2.3.4. att. DataPerRates kuba saturs

Visas dimensijas šajā kubā, izņemot laika un valūtas dimensijas, ir definētas kā piederošas kubam FINREPTables. Laika un valūtas dimensijas, kas pieder abiem kubiem ir atstātas bez tiešas kuba definēšanas (4.2.3.5. att.).



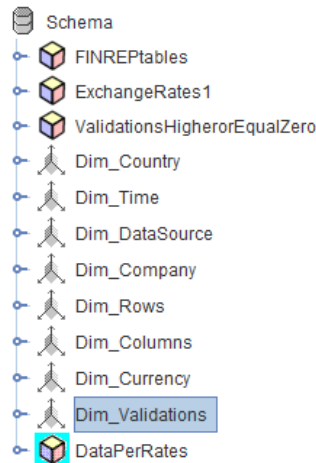
4.2.3.5. att. Laika dimensijas definēšana virtuālajā kubā.

Tomēr šī risinājuma nepietiek pilnvērtīgai datu atspoguļošanai pēc valūtu kursa, jo šajā gadījumā, dati tiek parādīti atskaitē tikai ja datu atlasē tiks izvēlēta vienīgi priekš Valūtas un Laiks dimensijām – nevar veikt detalizēto atlasu pa citām dimensijām. Piemērs tādai atlasei ar visiem definētiem mērījumiem ir atspoguļots 4.2.3.6. attēlā.

Measures						
Rate						
Data						
DataPerRate						
Quarters	Q1			Q2		
Currency	Rate	Data	DataPerRate	Rate	Data	DataPerRate
DKK	2	1	2	4	-119	-476
EUR	1	-133	-133	1	-	-
SEK	3	16	48	5	-	-

4.2.3.6. att. Datu atlase no kuba DataPerRates

Kopumā Mondrian shēma ar visām dimensijām un kubiem ir atspoguļota 4.2.3.7. attēlā.



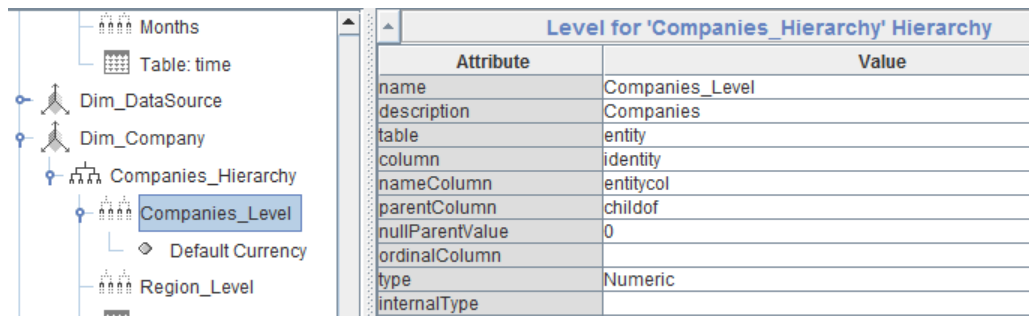
4.2.3.7.att. Mondrian shēma priekš Pentaho FINREP atskaitēm.

Visas dimensijas hierarhijas, ir atspoguļotas Mondrian shēmā, kā līmeņi. Dimensijām ir pievienoti arī atribūti (tabulu kolonas no fiziskās shēmas), tomēr pagaidām Saiku un JPivot pievienojumprogrammas datu atlasēm nevar izmantot dimensiju atribūtus, tādēļ viena un tā pati kolona no fiziskās tabulas var būt attēlota gan kā atribūts, gan kā dimensijas līmenis – piemēram valsts dimensija šādā veidā ir atspoguļots ģeogrāfiskais reģions, kuram pieder valsts (4.2.3.8.att.).

Measure for 'ExchangeRates1' Cube	
Attribute	Value
name	Rate
description	Rate
aggregator	avg
column	rate
formatString	
datatype	Numeric
formatter	
caption	Rate
visible	<input checked="" type="checkbox"/>

4.2.3.8.att. Valsts dimensija Mondrian shēmā.

Dimensijā Entity ir parādīts piemērs kā var atspoguļot vecāku-bērnu attiecības, izmantojot Mondrian iebūvēto funkcionalitāti (4.2.3.9.att.). Sadaļā “parentColumn” ir jāatzīmē arēja atslēga, kas norādīs uz vecāka elementa. Ir jānorāda arī kāda šūnas vērtība tiks uzskatīta par vērtību, kas attēlo ka vecāks neeksistē (var būt “null” - tukšums, vai “0”) [55]. Šajā gadījumā ir norādīta “0” vērtībā.



4.2.3.9.att. **Kompānijas dimensijas vecāka-bērna attiecības Mondrian shēmā.**

Šāda veida attiecību definēšanai ir arī savi trūkumi. Tā kā Mondrian neveido tukšas vietas dimensiju krustojumos kuba definēšanas laikā, bet atspoguļo tikai aktuālus datus, kas atrodas faktu tabulā, ja datu nav vecākajā locekļī, tad tas arī neparādīsies datu atskaitē, atlasot visas kompānijas. No otras puses, ja kaut viena šūnā faktu tabulā ir norādītā kāda vērtība priekš vecākā locekļa, tas tiks attēlots Saiku ģenerētājā atskaitē. Piemēram ja nulle ir ierakstīta faktu tabulā priekš kompānijas “Bank1”, kas ir “Bank11” un “Bank12” vecāks, un kompānijā “Group”, kas ir visu kompāniju vecāks, Saiku atgriezīs atskaiti kā 4.2.3.10.attēlā.

	All Companies		
	Group		
	Bank1		
	Bank11	Bank12	
	Of which: non-performing	Of which: non-performing	
	Q1		Q2
	Denmark	Sweden	Sweden
Row_Level	DataSource1	DataSource4	DataSource4
Financial guarantees given	-	-	-114
Loan commitments given	-121	-12	-
Other Commitments given	14	-	-

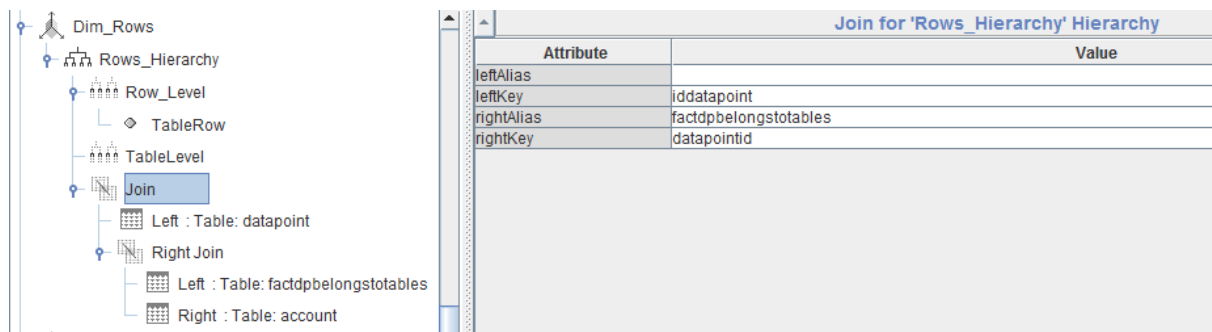
4.2.3.10.att. **Kompānijas dimensijas vecāka-bērna locekļu atlasē piemērs.**

Tādā veidā, ja iestādei ir hierarhiskā kompāniju struktūra un ir nepieciešams, to atspoguļot datu atlasē, ir jāpārdomā kā aizpildīt faktu tabulu. Varētu piemēram veidot mākslīgus ierakstus katrā mēnesī, kur ir ielādēti dati, ierakstot nulles vecākajos locekļos, ja dati ir augšupielādējami zemākajās kompānijās. Protams, faktu tabulas aizpilde ar nullēm, mākslīgi palielinot to apjomu

nav labs risinājums, jo lielāk kļūs datubāze, jo vairāk laika aizņems vaicājumu apstrādē. Šis jautājums var arī tikt risināts turpmākajos pētījumos.

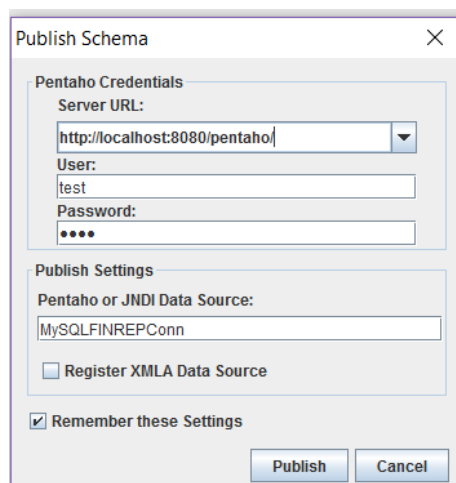
Ir nepieciešams arī pieminēt, ka Mondrian dokumentācijā lielo datu apjomu vaicājumiem ir ieteikts izmantot mākslīgas ierobežošanas tabulas (“closure table”) ģenerēšanu. Šajā tabulā ir atzīmēti locekļu identifikatori, kas atspoguļo vecāku-bērnu attiecības, kā arī dziļumu (cik locekļu attālumā bērns ir no vecāka). Dokumentācijā ir attēlots kā izmantot rīku Kettle tādas tabulas aizpildīšanai, no eksistējošas tabulas ar vecāku identifikatoru kolonu, kā arī dots SQL procedūras skripts šādam uzdevumam MySQL datubāzē (ja nav iespējams izmantot Kettle [56]). Tomēr darbības rezultāts ar tādas tabulas izmantošanu ir tāds pats kā izmantojot Mondrian atribūtu “parentColumn”.

Vēl viens izņēmums-gadījums ir attēlots Mondrian dimensijās FINREP tabulu rindām un kolonām. Fiziskā datubāzē šīs dimensijas ir saistītas ar faktu tabulu caur tabulu ar datu punktiem. Tādēļ šīs Mondrian dimensijas ir veidotas no abu tabulu apvienojuma, izmantojot Mondrian un Scema Workbench JOIN funkcionalitāti. Piemēram FINREP tabulu rindu dimensijā tiek pievienots JOIN un RIGHT JOIN lai saņemtu rindu nosaukumus priekš katra datu punkta (4.2.3.11.att.). Šāda veidā informācija ies uz faktu tabulu no tabulām “dataPoint” ← “factBelongsToTable” ← “custom1”.



4.2.3.11.att. JOIN attiecības FINREP tabulu rindu dimensijā, Mondrian shēmā.

Kad shēmas definēšana ir pabeigta to ir jāpublicē Pentaho serverī (iekš Schema Workbench “File/Publish”). Šajā solī ir jādefinē saite uz Pentaho servera vietni, Pentaho lietotāja dati, kā arī MySQL pievienošanas nosaukums (kas tika iepriekš izveidots Pentaho serverī), kā attēlots 4.2.3.12.attēlā.



4.2.3.12.att. Schema Workbench loģiskas datu noliktavas shēmas publicēšanas parametri

Ja shēmas publicēšanas laikā Mondrian atrod tajā kļūdas, paziņojumi tiks attēloti tikai komandu uzvednē. Grafiskajā saskarnē vienmēr tiek radīts paziņojums par shēmas veiksmīgo publicēšanu. No otrās puses, kļūdas komandu uzvednē ir gandrīz vienmēr diezgan viegli saprotamas – piemēram attēlā 4.2.3.13. ir kļūda kas nosaka ka Valsts dimensijā jaunajai hierarhijai ir jābūt vismaz viens līmenis.

```
16:20:59,973 ERROR [Workbench] Exception : Schema file C:\Users\olmeg\Downloads\FINREP2.xml is invalid.Mondrian Error:Hierarchy '[Dim_Country.New Hierarchy 0]' must have at least one level.
mondrian.olap.MondrianException: Mondrian Error:Hierarchy '[Dim_Country.New Hierarchy 0]' must have at least one level.
    at mondrian.resource.MondrianResource$Def0.ex(MondrianResource.java:984)
    at mondrian.rolap.RolapHierarchy.<init>(RolapHierarchy.java:250)
    at mondrian.rolap.RolapDimension.<init>(RolapDimension.java:130)
    at mondrian.rolap.RolapCube.getOrCreateDimension(RolapCube.java:784)
    at mondrian.rolap.RolapCube.<init>(RolapCube.java:195)
```

4.2.3.13.att. Mondrian atrasta kļūda shēmas publicēšanas laikā.

Pēc shēmas publicēšanas tai ir jābūt pieejamai gan priekš JPivot, gan priekš Saiku pievienojumprogrammām. Šis process var aizņemt 1-3 minūtes. Lai paātrinātu shēmas pieejamību Pentaho serverī, var izmantot funkcionalitāti, kas atjauno Mondrian shēmas kešatmiņu (servera tīmekļa vietnē “Tools/Refresh/Mondrian Schema Cache”). Saiku pievienojumprogrammā var izmantot līdzīgu funkcionalitāti – “Kubu atjaunošana”, kas arī pārlādē shēmas informāciju.

4.2.4. Datu atskaišu veidošana

Atskaišu veidošana Pentaho, izmantojot Saiku pievienojumprogrammu ir diezgan viegla, līdzīgi kā HFM, to var izmantot intuitīvi.

Izmantojot hierarhijas Tabula → Rinda, un Tabula → Kolona, vienu un to pašu atskaiti var filtrēt pēc dažādām tabulām – kā piemēram 4.2.4.1.att. priekš tabulas F 20.05.

Columns_Level			Nominal amount	Of which: debt forbearance	Of which: non-performing	Provisions for commitments and guarantees given
Table			F 20.05	F 20.05	F 20.05	F 20.05
Row_Level	Table	Quarters	Data	Data	Data	Data
Financial guarantees given	F 20.05	Q1	11	-	15	18
Loan commitments given	F 20.05	Q1	3	13	-119	17
Other Commitments given	F 20.05	Q1	12	-	30	-

4.2.4.1. att. FINREP tabulas F 20.05 atskaites piemērs

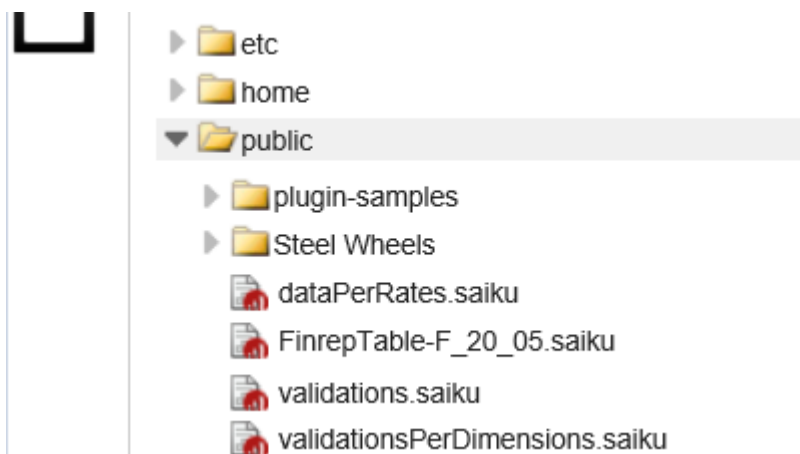
Var veikt arī detalizētu datu atlasī, neievietojot dimensijas filtrā, bet pievienojot tos uz rindu un kolonu asīm – kā piemēram att. kur ir tabula F 20.05, ar datu detalizāciju pa dimensijām (4.2.4.2. att.).

Columns_Level							Nominal amount	Of which: debt forbearance	Of which: non-performing	Provisions for commitments and guarantees given
Table							F 20.05	F 20.05	F 20.05	F 20.05
Row_Level	Table	Quarters	DataSource_Level	Country_Level	Region	Companies	Data	Data	Data	Data
Financial guarantees given	F 20.05	Q1	DataSource1	Denmark	Europe	Bank11	11	-	-	-
			DataSource4	Sweden	Europe	Bank11	-	-	15	-
Loan commitments given	F 20.05	Q1	DataSource1	Denmark	Europe	Bank11	2	-	-121	-
				Sweden	Europe	Bank12	-	-	-	1,700
				Norway	Europe	Bank11	-	-	-1,182	-
			DataSource4	Japan	Asia	Bank11	-	-	-	130
				Denmark	Europe	Bank12	-	-	14	-
				Norway	Europe	Bank11	-	13	-	-
Other Commitments given	F 20.05	Q1	DataSource1	Sweden	Europe	Bank12	12	-	-	-
				Norway	Europe	Bank11	-	-	16	-
			DataSource4	Denmark	Europe	Bank11	-	-	14	-

4.2.4.2. att. FINREP tabulas F 20.05, datu atlase pa pieejamajām dimensijām.

Protams, lietotāji nav pasargāti no kļūdas izvēlēties rindas no vienas tabulas, un kolonas no otrās tabulas, tāpēc varētu būt ērti saglabāt vaicājumus, kas attēlo datus pa katrai FINREP tabulai.

Visi vaicājumi tiek saglabāti .saiku formātā un pēc atvēršanas Saiku pievienojumprogrammā ir pieejami, kā datu atskaites (4.2.4.3. att.).



4.2.4.3. att. Saiku glabātie MDX vaicājumi, pieejamie lietotājam

Līdzīgi var veidot arī validāciju atskaites – gan pa visiem datiem kopumā (att.), gan ar noteikto detalizācijas pakāpi (4.2.4.4. att.).

Years	2017			
Quarters	Q1		Q2	
Months	January		April	
Validations	CountFailures	SumAmount	CountFailures	SumAmount
v3965_s	2	-1,321	1	-119

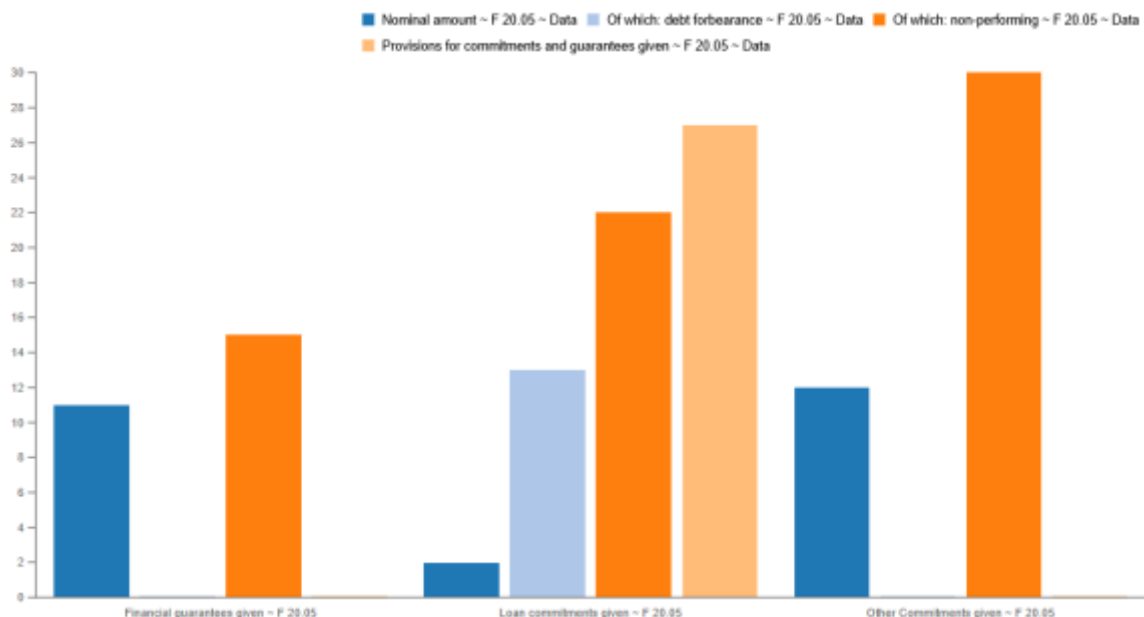
4.2.4.4. att. Validāciju atskaite pa mēnešiem

Tāda veida atskaite, var palīdzēt gan vērot datu atskaišu procesu, gan noteikt kurā datu punktā ir kādas kļūdas (4.2.4.5.att.).

Quarters	Q1				Q2	
Companies	Bank11				Bank11	
DataSources	DataSource1				DataSource1	
Countries	Denmark		Norway		Sweden	
Validations	CountFailures	SumAmount	CountFailures	SumAmount	CountFailures	SumAmount
v3965_s	1	-121	1	-1,200	1	-119

4.2.4.5.att. Validāciju atskaite pa visām dimensijām.

Pēc nepieciešamības datu atskaites var noformēt arī ar grafiku un diagrammu palīdzību, vairums no pieejamajām diagrammām ir līdzīgs tiem, kas ir piedāvāti Microsoft Excel programmā (4.2.4.6. att.).



4.2.4.6.att. F 20.05 tabulas piemērs izmantojot Saiku datu attēlošanas iespējas

Atskaite var eksportēt no Saiku arī uz Excel datnēm, gan uz CSV datnēm, un turpināt datu analīzi Microsoft Excel programmā.

REZULTĀTI UN SECINĀJUMI

Darba pirmajā daļā tika parādītās FINREP atskaišu standarta definīcija un to prasības, kā arī ieskats tirgus piedāvātājos risinājumos. No publiski pieejamās informācijas var secināt, ka tikai divas kompānijas piedāvā jau gatavus risinājumus, kuru pamatā ir Oracle Hyperion. Ir arī piedāvājumi, kas balstīti uz citām sistēmām, bet nav iespējams uzzināt, kā tie ir veidoti un ar kādām tehnoloģijām. Var secināt, ka atvērtā koda publiski pieejami risinājumi šim vai arī kādam līdzīgam standartam nav, kā arī nav atvērtā koda pieejamo CPM sistēmu.

Darba otrajā nodaļā tika izpētīti darījumu inteligences rīku līderi, balstoties uz pieejamajiem pētījumiem par informācijas tehnoloģiju tirgu. Lai izpētītu, kā var izmantot CPM (korporatīvas veiktspējas vadības) sistēmu finanšu atskaitēm, tika izvēlēta CPM sistēma, ko Gartner ievietoja līderu starpā savā 2016. gada pētījumā – Oracle Hyperion. Savukārt, lai izvēlētos atvērtā koda rīku FINREP pielāgošanai, balstoties uz tirgus pētījumiem par atvērtā koda darījumu inteligences sistēmām, tika izvēlēti trīs līderi - Pentaho Community, JasperSoft, Jedox Palo. Tika definēti kritēriji labāka rīka izvēlēšanai, balstoties uz FINREP atskaišu specifiku un piedāvātajām Hyperion CPM iespējām.

Darbā trešajā nodaļā tika apskatīta Oracle HFM funkcionalitāte no teorētiskas puses, kā arī pieejamās iespējas, kas ir atvērtā koda rīku līderiem. Var secināt, ka Hyperion piedāvā plašu funkcionalitāti finanšu atskaišu uzturēšanai, sākot no iespējām augšupielādēt datus sistēmā un modificēt tos, beidzot ar biznesa procesu atbalstu, kad meitu uzņēmumus var slēgt datu ievadei un nodot datus tālākai apstiprināšanai konsolidētājai kompānijai.

Atvērtā koda rīki pamatā piedāvā OLAP, ETL un atskaišu veidošanas iespējas. Lai veidotu funkcionalitāti, kas varētu atbalstīt biznesa procesus līdzīgi kā CPM, ir nepieciešams veikt izvēlēta rīka pielāgošanu, izstrādājot tajā atbilstošu datu noliktavu. Salīdzinot trīs darījumu inteligences rīkus, var secināt, ka vispiemērotākais rīks šādam uzdevumam ir Pentaho. Vienīgā problēma ir, ka tā izmantotais Mondrian OLAP serveris neatbalsta datu modificēšanu, datus var tikai augšupielādēt tajā. Tas nozīmē, ja FINREP noteiktajās validācijās parādīsies, ka atskaitēs ir kļūda (piemēram, bieži šādu veidu atskaitēs dati nesakrīt noapaļojuma dēļ), tad dati ir jāmodificē avota datubāzēs, kas varbūt ne tik ērti ceturkšņa atskaišu veidošanā. Datu modificēšanu atbalsta Jedox Base (Palo), bet šajā rīkā pietrūkst citu lietu, ko piedāvā Pentaho (piemēram, metadatu pārvaldība, pieejamā informācija par rīka darbību, plašs pievienojumprogrammu piedāvājums).

Darba ceturtajā nodaļā tika veikts eksperiments, kā var pielāgot FINREP atskaites standartu darījumu intelīģences rīkos. Var secināt, ka HFM piedāvā diezgan ērtu pieeju kā pārvaldīt metadatus. Vairums no dimensijām jau ir priekš-definēti, kā arī ir priekš-definēti to atribūti – piemēram valūtas dimensija ir definēta pēc noklusējuma, un administratoram ir tikai jānorāda kādas valūtas tiks izmantotas lietotnē. Tomēr tādām dimensiju gatavām iestatījumiem ir arī negatīvas puses, jo locekļu definēšana tādās dimensijās var prasīt diezgan lielo sistēmas arhitekta darba ieguldījumu, jo ir jāpārziņa katras dimensijas specifisko atribūtu apstrādi. Kalkulācijas, kuras nevar atrisināt ar locekļu hierarhisko sakārtojumu metadatos, var definēt noteikumus. Izmantojot trīs lietotāja brīvi definētos atribūtus (“UD1” – “UD3”) var arī automatizēt validāciju kalkulācijas priekš datu šūnām, izmantojot tos kā ārējās atslēgas uz citiem dimensiju locekļiem. Metadatos var definēt drošības klases priekš katras dimensijas, kurus var vēlāk izmantot sistēmā lai definētu lietotāja piekļuves tiesības datu objektiem.

Vienā no negatīvajām lietām iekš HFM ir lielas problēmas ar to instalāciju. Kļūdas, kas radās instalēšanas gaitā ne vienmēr ir viegli atklājamas, un nav apskatīti dokumentācijā, kas ir pieejama no Oracle.

Veicot eksperimentu ar atvērtā koda rīku Pentaho, tika pamēģināts integrēt to kopā ar Palo OLAP serveri, kas atbalsta datu rakstīšanu datu noliktavā. Tomēr Palo instalācija nebija veiksmīga, jo dotais OLAP serveris ir pieejams kopā ar Jedox Base rīku, kuram ir problēmas gan ar brīvas piekļuves licenci, gan ar rīka pilnas versijas izmēģinājuma licenci. Pie tam atvērtā koda programmatūra darbam ar Palo – JPalo pēdējo reizi tika atjaunota 2008.gadā, un pats Palo serveris – 2010.gadā. Tādēļ tika nolemts veikt eksperimentu uz Pentaho iebūvētā Mondrian OLAP servera.

Pentaho rīks bija integrēts ar MySQL datubāzi, kur tika izveidota datu noliktavas fiziskā shēma. Pēc tam Mondrian serverī bija izveidota arī loģiskā datu noliktavas shēma, kuru Pentaho varētu izmantot priekš datu atlases vajadzībām. Datu atlases iespējas tika nodemonstrētas uz FINREP tabulas F 20.05, kurai pēc standarta ir definēts datu sadalījums pa valstīm.

Lai attēlotu datu validāciju apstrādes iespējas ir izveidotas pārbaudes priekš validācijām kas pārbauda vai šūnas ir negatīvas vai nē. Uz Kompānijas (Entity) dimensijas piemēra tika attēlots kā varētu attēlot hierarhiskās attiecības starp dimensijas locekļiem, izmantojot Mondrian iebūvēto funkcionalitāti.

Tika apskatīts vēl kā varētu veikt valūtas konvertāciju, tomēr šādai metodei ir savi trūkumi, jo var redzēt tikai summārus datus pa valūtām.

Salīdzinot ar HFM, netika attēlots ICP dimensijas izveide priekš iekš-kompānijas transakcijām. Tāda dimensija pēc noklusējuma ir pieejama HFM rīkā, tomēr tādas dimensijas esamība datu noliktavā, var būt atkarīga no iekšējām kompānijas biznesa prasībām. Pēc nepieciešamības tādu dimensiju ir iespējams veidot no Kompānijas (Entity) dimensijas lapas locekļiem (izmantojot jau definētas hierarhiskās attiecības starp kompānijām).

Atšķirībā no HFM, risinājums ar atvērtā koda rīku, neatbalsta rakstīšanu datu noliktavā. Priekš šī uzdevuma varētu integrēt Pentaho ar Mondrian 4.versiju, tomēr tas atbalsta datu modifikācijas tikai atskaitēs, nemodificējot datu noliktavas datus.

Var secināt ka atvērtā koda rīkā ir attēlota funkcionalitāte FINREP standarta pamatprasībām – kā attēlot datus, atlasīt tos atskaitēs pēc nepieciešamām dimensijām, kā var veikt datu validāciju pārbaudes, un kā varētu veikt valūtas konvertēšanu. Kopumā novērtējot Pentaho darbību var secināt, ka tajā ir iespējams veidot līdzīgu funkcionalitāti kā Oracle HFM rīkā – pēc nepieciešamības dimensijās var veidot papildus atribūtus, kurus var apstrādāt Mondrian shēma izmantojot MDX vaicājumu valodu.

Turpmākajam pētījumam paliek jautājumi kā veikt valūtas konvertēšanu zemākajos datu atlases līmeņos, izmantojot Mondrian funkcionalitāti; kā attēlot hierarhiskās attiecības bez nepieciešamības saistīt vecāku locekļus ar faktu tabulu (kubu), kā arī kādā veidā iespējot datu rakstīšanu datu noliktavā izmantojot Mondrian serveri.

IZMANTOTĀ LITERATŪRA UN AVOTI

[1] Stīvs Villiams, Nensija Villiams, *The Profit Impact of Business Intelligence*, Morgan Kaufmann, 2006.

[2] Pauls Harmons, *A Business Process Management Guide for Managers and Process Professionals, Third Edition*, Morgan Kaufmann, 2014.

[3] R. Kelly Rainer Jr., Brad Prince, Casey Cegielsk, *Introduction to Information Systems: International Student Version, Fifth Edition*, John Wiley & Sons, 2015.

[4] IT Central Station “Oracle HFM vs Longview CPM”, tiešsaiste – atsauce 24.01.2017. Pieejams: https://www.itcentralstation.com/products/comparisons/longview-cpm_vs_oracle-hfm

[5] John E. Van Decker, Christopher Iervolino, “Magic Quadrant for Financial Corporate Performance Management Solutions”, 31 May 2016, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.gartner.com/doc/reprints?id=1-38C8M9Z&ct=160601&st=sb>

[6] CFORS, FINREP un COREP implementācijas apraksts, 2013, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://cfors.eu/wp-content/uploads/2013/11/crd-iv-challenge.pdf>

[7] Vasīlijs Kravcevs, “Sistēmas Summix ieviešana ABLV Bank Luxembourg”, 2014., tiešsaiste – atsauce 24.01.2017. Pieejams: http://www.reportum.lv/collateral/2-Seminar-26-Feb-2014-Sistemas_Summix_ieviesana_ABLV_Banka.pdf

[8] Wolters Kluwer, “EU Financial Market Regulatory Reporting Solutions”, Rīga, 26 February 2014., tiešsaiste – atsauce 24.01.2017. Pieejams: http://www.reportum.lv/collateral/1-Seminar-26-Feb-2014-Wolters_Kluwer_Financial_Services.pdf

[9] “Risinajumi Eiropas finanšu tirgus regulatoru prasībām”, *DB.LV*, 2014. gada 20. marts, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.db.lv/reklamraksti/risinajumi-eiropas-finansu-tirgus-regulatoru-prasibam-412041>

[10] “AMOSCA develop solution for COREP and FINREP reporting“, *Amosca*, tiešsaiste – atsauce 24.01.2017. Pieejams: http://www.amosca.co.uk/pdfs/datasheets/COREP_FINREP_Reporting.pdf

[11] Michael Grieger, Josef Macdonald, “Utilising Hyperion EPM for COREP & FINREP reporting”, *Amosca*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.amosca.co.uk/news/amosca-develop-solution-corep-finrep-reporting/>

[12] “2016 EU-wide transparency exercise results”, *EBA*, 18 November 2016, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.eba.europa.eu/risk-analysis-and-data/eu-wide-transparency-exercise/2016/results>

[13] “Arele Concept”, *Arele*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://arelle.org/>

[14] Eiropas Savienības Oficiālais Vēstnesis, “EIROPAS CENTRĀLĀS BANKAS REGULA (ES) Nr. 468/2014, ar ko izveido vienotā uzraudzības mehānisma pamatstruktūru Eiropas Centrālās bankas sadarbībai ar nacionālajām kompetentajām un norīkotajām iestādēm (VUM pamatregula)”, 2014. gada 16. aprīlis, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://eur-lex.europa.eu/legal-content/LV/TXT/?uri=CELEX:32014R0468>

[15] Eiropas Savienības Oficiālais Vēstnesis, “EIROPAS CENTRĀLĀS BANKAS REGULA (ES) 2015/534 par uzraudzības finanšu informācijas sniegšanu (ECB/2015/13)”, 2015. gada 17. marts, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://eur-lex.europa.eu/legal-content/LV/TXT/?uri=CELEX:32015R0534>

[16] Finanšu un kapitāla tirgus komisija, “FKTK apstiprina vairākus tirgus dalībniekiem saistošus noteikumus”, Rīgā, 06.07.2016., tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.fktk.lv/lv/mediju-telpa/citas-publicacijas/2016/501-mediju-telpa/pazinojumi-masu-informacijas-l/2016/5856-fktk-apstiprina-vairakus-tirgus-dalibniekiem-saistosus-noteikumus.html>

[17] Finanšu un kapitāla tirgus komisijas normatīvie noteikumi Nr. 119, “Uzraudzības finanšu pārskatu sagatavošanas normatīvie noteikumi”, Rīgā 2016. gada 6. jūlijā, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://likumi.lv/ta/id/283431-uzraudzibas-finansu-parskatu-sagatavosanas-normativie-noteikumi>

[18] Eiropas Savienības Oficiālais Vēstnesis, “PADOMES REGULA (ES) Nr. 1024/201 ar ko Eiropas Centrālajai bankai uztic īpašus uzdevumus saistībā ar politikas nostādņēm, kas attiecas uz kredītiestāžu prudenciālo uzraudzību”, 2013. gada 15. oktobris, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://eur-lex.europa.eu/legal-content/LV/TXT/?uri=celex:32013R1024>

[19] “Oracle Business Intelligence Applications Global Price List”, *Oracle*, September 1, 2016., tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.oracle.com/us/corporate/pricing/price-lists/business-intelligence-price-list-071347.pdf>

[20] “Annex III - FINREP templates IFRS”, *EBA*, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.eba.europa.eu/regulation-and-policy/supervisory-reporting/implementing-technical-standard-on-supervisory-reporting>

[21] “EBA Validation Rules v2.6.0.0”, *EBA*, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.eba.europa.eu/regulation-and-policy/supervisory-reporting/implementing-technical-standard-on-supervisory-reporting-data-point-model>

[22] “DPM Database 2.6.0.0”, *EBA*, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.eba.europa.eu/regulation-and-policy/supervisory-reporting/implementing-technical-standard-on-supervisory-reporting-data-point-model>

[23] Josh Parenteau, Rita L. Sallam, Cindi Howson, Joao Tapadinhas, Kurt Schlegel, Thomas W. Oestreich, “Magic Quadrant for Business Intelligence and Analytics Platforms”, 04 February 2016., tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.gartner.com/doc/reprints?ct=160204&id=1-2XXET8P>

[24] BUTLER Analytics, “Business analytics yearbook 2015”, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.butleranalytics.com/wp-content/uploads/2014/12/yb2015v2.pdf>

[25] “What is SAP HANA, express edition?”, *SAP*, tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.sap.com/cis/developer/topics/sap-hana-express.html>

[26] „The Forrester Wave: Enterprise Performance Management, Q4 2016”, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://reprints.forrester.com/#/assets/2/465/'RES118246'/reports>

[27] Craig Schiff , “Free BPM: Open Source Comes to Performance Management“, August 15, 2006., tiešsaiste – atsauce 24.01.2017. Pieejams: http://www.b-eye-network.com/blogs/schiff/archives/2006/08/free_bpm_open_s.php

[28] “Oracle Hyperion Financial Management”, *Oracle, 2015.*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.oracle.com/us/products/middleware/bus-int/064105.pdf>

[29] Peter John Fugere, Jr., „Oracle Hyperion Financial Management Tips & Techniques Design, Implementation & Support” (p 15.) Oracle Press

[30] Rupam Majumdar et al., „Oracle Hyperion Financial Management 11.1.1.3 Beginner’s Guide Part I”, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.slideshare.net/alooa2/oracle-hfm-beginners-guide-part-i>

[31] “BUSINESS INTELLIGENCE SOFTWARE EDITIONS”, *JASPERSOFT*, 2016., tiešsaiste – atsauce 24.01.2017. Pieejams: <https://www.jaspersoft.com/editions>

[32] “Jaspersoft Community Edition vs. Commercial Edition, A guide for choosing the TIBCO Jaspersoft edition that’s right for you”, *JASPERSOFT*, 2016., tiešsaiste – atsauce 24.01.2017. Pieejams:

https://www.jaspersoft.com/sites/default/files/confirmation_files/jaspersoft_community_vs_commercial_web.pdf

[33] “Jaspersoft CE v/s EE / Having Jaspersoft EE functionality in Jaspersoft CE”, *Helicaltech*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://helicaltech.com/blogs/jaspersoft-ce-versus-ee-having-jaspersoft-ee-functionality-in-jaspersoft-ce/>

[34] “Extensions”, *JasperSoft Community*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://community.jaspersoft.com/exchange/extensions/index>

[35] “Pentaho Community & Enterprise Edition Product Comparison”, *Pentaho*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://community.pentaho.com/>

[36] “Pentaho CE vs EE”, *Atolcd*, 2016., tiešsaiste – atsauce 24.01.2017. Pieejams: https://www.atolcd.com/fileadmin/Evenements/Pentaho_Day_2016/Pentaho_CE_vs_EE_-_comparison_4-12-16.pdf

[37] “A Conceptual Overview of the Pentaho Metadata Editor”, *Pentaho*, 2016., tiešsaiste – atsauce 24.01.2017. Pieejams: <https://help.pentaho.com/Documentation/6.1/0N0/110/020>

[38] “Pentaho Marketplace”, *Pentaho* <http://www.pentaho.com/marketplace/>

[39] “How to Design a Mondrian Schema”, *Pentaho*, tiešsaiste – atsauce 24.01.2017. Pieejams: http://mondrian.pentaho.com/documentation/schema.php#Access_control

[40] “BIRT Review”, *Innovent Solutions*, tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.innoventsolutions.com/birt-review.html>

[41] “Jedox in two versions: Base and Premium”, *Jedox*, 2013., tiešsaiste – atsauce 24.01.2017. Pieejams: <http://www.jedox.com/en/press/jedox-two-versions-base-preiumedox-two-versions-base-premium/>

[42] Matt Casters et al, “Pentaho Kettle Solutions: Building Open Source ETL Solutions with Pentaho”, (p.273), Wiley Publishing Inc

[43] “How Jedox works”, *Jedox Knowledge Base*, January 20, 2017 ., tiešsaiste – atsauce 24.01.2017. Pieejams: <https://knowledgebase.jedox.com/knowledgebase/jedox-works/>

[44] Ines Martins, Nuno Datia, Helder Pita: “*Business Intelligence framework for financial support*”, Instituto Superior de Engenharia de Lisboa , January 2010 ., tiešsaiste – atsauce 20.05.2017. Pieejams: https://www.researchgate.net/profile/Nuno_Datia/publication/259706982_Business_Intelligence_framework_for_financial_support/links/00b7d52d6afcd4bf48000000.pdf

[45] “Appendix B: Configuring Microsoft SQL Server” , *Oracle*, tiešsaiste – atsauce 20.05.2017. Pieejams:

<http://www.oracle.com/webfolder/technetwork/tutorials/obe/hyp/FCM11.1.2.3-SingleNodeInstall/FCMSingleNodeInstall.htm#t13>

[46] “Oracle Enterprise Performance Management System Release 11.1.2.x (Excel)”, *Oracle*, tiešsaiste – atsauce 20.05.2017. Pieejams: <http://www.oracle.com/technetwork/middleware/bi-foundation/hyperion-supported-platforms-085957.html>

[47] Anjum Ara, “javax.xml.stream.XMLStreamException: Premature end of file encountered”, tiešsaiste – atsauce 20.05.2017. Pieejams: <http://learninghyperion.blogspot.com/2014/03/the-loading-of-opss-java-security.html>

[48] “Account Type Behavior”, *Oracle*, tiešsaiste – atsauce 20.05.2017. Pieejams: https://docs.oracle.com/cd/E17236_01/epm.1112/hfm_admin_11.1.2.2.300/frameset.htm?ch04s01s01.html

[49] “Defining Accounts”, “Table 5. Account Member Attributes”, *Oracle*, tiešsaiste – atsauce 20.05.2017. Pieejams: https://docs.oracle.com/cd/E17236_01/epm.1112/hfm/admin/frameset.htm?ch04s01.html

[50] “Financial Management 11.1.2.4.100 Administrator's Guide”, “Defining Currencies”, *Oracle*, tiešsaiste – atsauce 20.05.2017. Pieejams: https://docs.oracle.com/cd/E57185_01/HFMAD/ch04s09.html

[51] William D. Back, Nicholas Goodman, Julian Hyde, *Mondrian in Action*, (“11.2 What-if analysis”, p238), Manning Publications

[52] “Pentaho”, *open-source-guide*, tiešsaiste – atsauce 20.05.2017. Pieejams: <http://www.open-source-guide.com/en/Solutions/Applications/Business-intelligence-suites/Pentaho>

[53] “How to create an individual log file for Saiku Adhoc on Pentaho BI Server”, *Anonymous Business Intelligence*, tiešsaiste – atsauce 20.05.2017. Pieejams: <https://anonymousbi.wordpress.com/2013/03/15/how-to-create-an-individual-log-file-for-saiku-adhoc-on-pentaho-bi-server/>

[54] “2.1 Annotation”, *Pentaho*, 2011, tiešsaiste – atsauce 20.05.2017. Pieejams: <http://mondrian.pentaho.com/documentation/schema.php#Annotation>

[55] “5.2 Parent-child hierarchies”, *Pentaho*, 2011, tiešsaiste – atsauce 20.05.2017. Pieejams: http://mondrian.pentaho.com/documentation/schema.php#Parent_child_hierarchies

[56] “5.2.2 Closure tables”, *Pentaho*, 2011, tiešsaiste – atsauce 20.05.2017. Pieejams: http://mondrian.pentaho.com/documentation/schema.php#Closure_tables

PIELIKUMI

1. Pielikums. Kubu un virtuālā kuba attēlojums Mondrian XML shēmā

```
<Cube name="FINREPTables" caption="Finrep tables" visible="true"
description="FINREP tables" cache="true" enabled="true">
  <Table name="facttable">
  </Table>
  <DimensionUsage source="Dim_Country" name="Dim_Country" visible="true"
foreignKey="c2id" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Time" name="Dim_Time" visible="true"
foreignKey="timeid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_DataSource" name="Dim_DataSource"
visible="true" foreignKey="c3id" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Company" name="Dim_Company" visible="true"
foreignKey="eid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Rows" name="Dim_Rows" visible="true"
foreignKey="dpid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Columns" name="Dim_Columns" visible="true"
foreignKey="dpid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Currency" name="Dim_Currency"
caption="Currency" visible="true" foreignKey="currencyid"
highCardinality="false">
  </DimensionUsage>
  <Measure name="Data" column="cellvalue" datatype="Numeric"
aggregator="sum" description="Data" visible="true">
  </Measure>
</Cube>
<Cube name="ExchangeRates1" caption="Exchange Rates1" visible="true"
description="Exchange rates per time" cache="true" enabled="true">
  <Table name="factcurrencies">
  </Table>
  <DimensionUsage source="Dim_Currency" name="Dim_Currency" visible="true"
foreignKey="currencyid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Time" name="Dim_Time" visible="true"
foreignKey="timeid" highCardinality="false">
  </DimensionUsage>
  <Measure name="Rate" column="rate" datatype="Numeric" aggregator="avg"
caption="Rate" description="Rate" visible="true">
  </Measure>
</Cube>
<Cube name="ValidationsHigherorEqualZero" caption="Validations
HigherorEqualZero" visible="true" description="Validations HigherorEqualZero"
cache="true" enabled="true">
  <Table name="validationhe_view">
  </Table>
  <DimensionUsage source="Dim_Time" name="Dim_Time" visible="true"
foreignKey="timeid" highCardinality="false">
  </DimensionUsage>
  <DimensionUsage source="Dim_Country" name="Dim_Country" visible="true"
foreignKey="c2id" highCardinality="false">
```

```

    </DimensionUsage>
    <DimensionUsage source="Dim_DataSource" name="Dim_DataSource"
visible="true" foreignKey="c3id" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Dim_Company" name="Dim_Company" visible="true"
foreignKey="eid" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Dim_Validations" name="Dim_Validations"
visible="true" foreignKey="idvalidation" highCardinality="false">
    </DimensionUsage>
    <Measure name="SumAmount" column="cellvalue" datatype="Numeric"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="CountFailures" column="rowid" datatype="Numeric"
aggregator="count" visible="true">
    </Measure>
</Cube>
<VirtualCube enabled="true" name="DataPerRates" caption="Data per currency
rates" visible="true" description="Data per currency rates">
    <VirtualCubeDimension cubeName="FINREPTables" caption="Country"
visible="true" highCardinality="false" name="Dim_Country">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="FINREPTables" caption="DataSource"
visible="true" highCardinality="false" name="Dim_DataSource">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="FINREPTables" caption="Company"
visible="true" highCardinality="false" name="Dim_Company">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="FINREPTables" caption="Rows"
visible="true" highCardinality="false" name="Dim_Rows">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="FINREPTables" caption="Columns"
visible="true" highCardinality="false" name="Dim_Columns">
    </VirtualCubeDimension>
    <VirtualCubeDimension caption="Time" visible="true"
highCardinality="false" name="Dim_Time">
    </VirtualCubeDimension>
    <VirtualCubeDimension caption="Currency" visible="true"
highCardinality="false" name="Dim_Currency">
    </VirtualCubeDimension>
    <VirtualCubeMeasure cubeName="FINREPTables" name="[Measures].[Data]"
visible="true">
    </VirtualCubeMeasure>
    <VirtualCubeMeasure cubeName="ExchangeRates1" name="[Measures].[Rate]"
visible="true">
    </VirtualCubeMeasure>
    <CalculatedMember name="DataPerRate" caption="DataPerRate"
formula="[Measures].[Data] * [Measures].[Rate]" dimension="Measures"
visible="true">
    </CalculatedMember>
</VirtualCube>
</Schema>

```

Maģistra darbs: **Darījumu intelīģences sistēma finanšu atskaitēm FINREP formātā**

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā elektroniskā kopija atbilst izdrukai.

Autors: _____ / Olga Medvedeva/

(Autora paraksts)

Ar savu parakstu apliecinu, ka esmu lasījis augstāk minēto maģistra darbu un atzīstu to par **pieņemot** / **nepieņemot** (nevajadzīgo izsvītrot) aizstāvēšanai Latvijas Universitātes datorzinātņu maģistrā.

Darba vadītājs: _____ / Laila Niedrīte, Dr.sc.comp/

(Vadītāja paraksts)

Darbs iesniegts maģistratūras sekretariātā _____.

(Iesniegšanas datums)

Ar šo apliecinu, ka darba elektroniskā versija ir augšupielādēta LU informatīvajā sistēmā.

Studiju metodiķe: _____.

(Metodiķes paraksts)

Recenzents: _____

(Akad.amats, zin.grāds, vārds, uzvārds)

Darbs aizstāvēts maģistra gala pārbaudījuma komisijas sēdē

_____ prot.Nr. _____

(Darba aizstāvēšanas datums)

Komisijas sekretārs: _____

(Sekretāra paraksts)