

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatikainstituut

Infosüsteemide õppetool

**Andmeanalüüsi vahendite loomine Eesti
Rahvusringhäälingu
meediahaldussüsteemi näitel**

Magistritöö

Üliõpilane: Kairi Rüütel

Üliõpilaskood: 132279IABM

Juhendaja: Enn Õunapuu

Tallinn
2015

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

(kuupäev)

(allkiri)

Annotatsioon

Infoühiskonna ettevõtete ning organisatsioonide tegevuse käigus tekkivate andmete maht järjest suureneb ning sealjuures kasvab ka nende andmete võime anda organisatsioonile lisaväärtust. Üheks viisiks olemasolevaid andmeid targalt kasutada on luua hoiustamist väärt andmete tarvis andmeait ning nende analüüsimist võimaldav vahend, näiteks andmekuup.

Käesoleva töö eesmärgiks oli Eesti Rahvusringhäälingu meediahaldussüsteemi tarbeks välja arendada andmete analüüsimist võimaldavad andmeait ja –kuup, et saada ülevaade süsteemis tehtavatest arhiivmaterjali tellimustest ning tõsta antud andmete kasulikkust organisatsioonile.

Magistritöö põhitulemusi on kolm:

- 1) valmis süsteemi analüüs;
- 2) arendati välja andmeait, milles arhiivmaterjali tellimustega seonduvaid andmeid hoiustatakse;
- 3) arendati välja andmekuup, mille abil on võimalik andmeaidas olevat informatsiooni küsida.

Lisaks teostati andmekuubist andmete pärimiseks tarviliku klientrakenduse valik ning dokumenteeriti nii andmeaida- kui andmekuubi evitamise protsess, et uuenduste teostamist hõlbustada.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 71 leheküljel, 45 peatükki, 24 joonist, 10 tabelit.

Abstract

The amount of data, which is generated during the everyday business processes among today's organizations, is quickly increasing. In relation to that, the data's ability to give additional value to the organization increases as well. One way to raise the data's value for the organization is to store it in a data warehouse and develop a tool for analysing it, for example a data cube.

The purpose of the thesis was to analyse and develop a data warehouse and a data cube for media asset management system which is used by Estonian Public Broadcasting. The aim of the data warehouse and data cube is to enable end-user to create various queries and analyse the data which is related to media archive orders.

This thesis has three main outcomes:

- 1) an analysis of the created system;
- 2) a developed data warehouse, which is used to store the data related to media archive orders;
- 3) a developed data cube, which is used for accessing the information stored in the data warehouse.

Additionally, a client application for querying information from data cube was selected and the deployment processes for data warehouse and data cube projects were described and documented, to facilitate further developments.

The thesis is written in Estonian language and contains 71 pages of text, 45 chapters, 24 figures, 10 tables.

Lühendite ja mõistete sõnastik

Andmekuup *data cube*

Andmekuup on andmete esituse viis, mille puhul vaadeldakse andmeid mitmemõõtmeliselt. Andmekuubi abil analüüsitakse eelkõige selliseid andmeid, mille kaasabil on võimalik anda ettevõttele lisaväärtust (vt lisaks pt 4.4). (Acharya 2013, 5)

Dimensioonitabel *dimension table*

Dimensioonitabel on andmeidas eksisteeriv tabel, mis sisaldab olemit kirjeldavaid tekstilisi ning konstantseid numbrilisi väärtuseid. (Kimball, Ross 2002, 19)

ETL *Extract-Transform-Load*

Andmete andmeaita laadimise protsess, mille käigus andmed lähteandmebaasist laetakse, sobivale kujule teisendatakse ning seejärel andmeaita salvestatakse. (Junic 2006, 86)

Faktitabel *fact table*

Faktitabel on andmeaita keskne tabel, mis koondab endas olemit ärilisi arvulisi mõõdikuid ning viiteid temaga seonduvatesse dimensioonitabelitesse. (Vallaste, 2015)

MAM *Media Asset Management*

MAM on Eesti Rahvusringhäälingu telemajas kasutusel olev meediahaldussüsteem, mille abil toimub peamiselt telesaadetega seonduvate videofailide haldamine, arhiveerimine ning arhiivist tellimine.

OLAP***Online Analytical Processing***

OLAP on tehnoloogia, mille abil on võimalik teostada reaalajas andmete analüüsi. Antud lähenemist kasutatakse ärianalüüsi rakendustes, et võimaldada lõppkasutajale näiteks suuremahuliste kalkulatsioonide ning trendianalüüside koostamist. (What is OLAP ... 2015)

SSAS***SQL Server Analysis Services***

SSAS on Microsofti poolt arendatav komponent, mis võimaldab luua ärianalüüsi jaoks tarvilikke vahendeid, näiteks OLAP andmekuupe, mille abil organisatsiooni otsustusprotsessi toetada. (Analysis Services, 2015)

SSIS***SQL Server Integration Services***

SSIS on Microsofti poolt arendatav komponent, mille abil on võimalik luua keerukaid andmete integratsiooni ning teisenduse lahendusi, sh koostada ETL protsesse. (SQL Server Integration Services, 2015)

Jooniste nimekiri

Joonis 1. Andmeaida ülevaade Inmoni metoodikal.....	14
Joonis 2. Andmeaida ülevaade Kimballi metoodikal	15
Joonis 3. Kimballi elutsükkel	16
Joonis 4. Olemasoleva süsteemi arhitektuur.....	21
Joonis 5. Fakti- ning dimensioonitabelite asetsemine tähtskeemil.....	32
Joonis 6. Lumehelbe skeem.....	33
Joonis 7. Loodava süsteemi andmemudel	34
Joonis 8. Andmeaida tüüpiline arhitektuur.....	39
Joonis 9. Loodava süsteemi arhitektuur	41
Joonis 10. Intranet süsteemist kasutatavad andmed	43
Joonis 11. ETL protsessi andmete laadimise voog.....	44
Joonis 12. Tellimuste faktitabeli ETL protsess	45
Joonis 13. Kolmemõõtmeline andmekuup	49
Joonis 14. Andmekuubi lähteandmete vaade	50
Joonis 15. Andmekuubi dimensioonid	51
Joonis 16. Andmekuubi mõõdikud.....	51
Joonis 17. Andmekuubi dimensioonide ja mõõdikute vahelised seosed.....	52
Joonis 18. Andmekuubi projekti struktuur	53
Joonis 19. Nõude F01 väljundandmete näide.....	56

Joonis 20. Nõude F02 väljundandmete näide	56
Joonis 21. Nõude F03 väljundandmete näidis	57
Joonis 22. Nõude F04 väljundandmete näidis	57
Joonis 23. Nõude F05 väljundandmete näidis	58
Joonis 24. Nõude F06 väljundandmete näidis	58

Tabelite nimekiri

Tabel 1. Standardi ISO/IEC 9126 kvaliteedi mudel	29
Tabel 2. Tellimuste faktitabeli kirjeldus.....	34
Tabel 3. Tellimuste dimensioonitabeli kirjeldus	35
Tabel 4. Failide dimensioonitabeli kirjeldus	35
Tabel 5. Kasutajate dimensioonitabeli kirjeldus.....	35
Tabel 6. Klippide vahetabeli kirjeldus.....	36
Tabel 7. Klippide dimensioonitabeli kirjeldus	36
Tabel 8. Kuupäeva dimensioonitabeli kirjeldus	36
Tabel 9. Staatuste dimensioonitabeli kirjeldus	37
Tabel 10. Mittefunktsionaalsete nõuete ülevaatus kokkuvõte	59

Sisukord

1. Sissejuhatus	13
1.1 Taust ja probleem	13
1.2 Ülesande püstitus	13
1.3 Metoodika	14
1.3.1 Inmoni ülevalt-alla metoodika	14
1.3.2 Kimballi alt-üles metoodika	15
1.3.3 Metoodika valik	15
1.3.4 Kimballi elutsükkel	16
1.4 Ülevaade tööst	18
1.5 Töö sihtrühm	19
2. Olemasolev süsteem	20
2.1 Meediahaldussüsteem	20
2.1.1 Olemasoleva süsteemi arhitektuur	21
2.1.2 Arhiivmaterjali tellimise protsess	22
2.1.3 Failide nimetamine	22
2.2 Nõuded loodavale süsteemile	23
2.3 Olemasoleva olukorra eelised ja puudused	24
2.4 Andmeaida otstarbekuse hindamine	25
3. Süsteemi analüüs	27

3.1 Loodava süsteemi skoop ja kirjeldus.....	27
3.2 Nõuete analüüs	28
3.2.1 Funktsionaalsed nõuded	28
3.2.2 Infovajadused	28
3.2.3 Mittefunktsionaalsed nõuded.....	29
3.3 Andmemudeli disainimine.....	31
3.3.1 Tähtskeem ja lumehelbe skeem.....	31
3.3.2 Disainimustri valik	33
3.3.3 Süsteemi andmemudel.....	33
3.3.4 Tabelite kirjeldused	34
3.4 Andmete kvaliteet.....	37
4. Realisatsiooni teostamine	39
4.1 Loodava tehnilise lahenduse ülevaade	39
4.1.1 Andmeaida üldine arhitektuur	39
4.1.2 Loodava süsteemi arhitektuur.....	41
4.1.3 Arendamiseks valitud vahendid	41
4.2 Andmeait	43
4.2.1 Andmete laadimise protsess (ETL)	43
4.3 OLAP.....	47
4.4 Andmekuup	49
4.4.1 Andmekuubi arendusprotsess	50

4.5 Ärianalüüsi rakendus	53
4.6 Süsteemi testimine	55
4.7 Andmeaida ja -kuubi evitamine	60
4.8 Edasised arendusvõimalused	61
5. Kokkuvõte	62
Summary.....	64
Kasutatud kirjandus	65
Lisad	67
Lisa 1. Illustratsioon IBM lindirobotist	67
Lisa 2. Andmeaida andmebaasi ja -objektide loomise SQL laused	68
Lisa 3. Andmeaida projekti uuendamise juhend	70
Lisa 4. Andmekuubi projekti uuendamise juhend	71

1. SISSEJUHATUS

Järgnevalt tutvustatakse käesoleva magistritöö tausta ning lahendatavat probleemi. Seejärel antakse täpsem ülevaade ülesande püstitusest ja analüüsitakse, milline on selle lahendamiseks sobivaim meetodika. Viimaks antakse lühiülevaade töös käsitletavatest teemadest.

1.1 Taust ja probleem

Ettevõtted ja organisatsioonid väärtustavad järjest rohkem oma üht tähtsamat vara – infot. Mida aeg edasi, seda suuremaks muutub nimetatud vara maht ning sealjuures ka selle võime anda ettevõtte tegevusele lisaväärtust. Selleks, et organisatsiooni tegevuse käigus tekkinud andmeid oleks võimalik otstarbekal moel kasutada, tuleb luua informatsiooni hoiustamiseks, selleni jõudmiseks ja selle analüüsimiseks vastavad vahendid. Üheks võimalikuks viisiks on disainida ja modelleerida andmeait, millesse koondada oluline ning säilitamist väärt informatsioon. Sellise süsteemi loomine teeb võimalikuks olemasoleva teadmuse kaasamise otsustusprotsessi.

Käesoleva magistritöö raames luuakse Eesti Rahvusringhäälingu meediahaldussüsteemi tarvis andmete hoiustamiseks ning analüüsimiseks mõeldud tarkvara. Hetkel ei ole nimetatud süsteemi koosseisus andmete analüüsivahendeid, kuid vajadus ja huvi süsteemi omaja seisukohast olemasoleva info uurimiseks on olemas. Loodav süsteem võimaldab eelkõige saada ülevaate meedia arhiiviga seonduvast põhitegevusest: arhiivimaterjali tellimustest, mille põhjal on võimalik prioriseerida meedia digiteerimist.

1.2 Ülesande püstitus

Antud töö eesmärgiks on välja arendada andmete analüüsimist võimaldav andmeait ja -kuup, Eesti Rahvusringhäälingu meediahaldussüsteemi tarbeks, et saada ülevaade süsteemis tehtavatest meedia tellimustest ning tõsta sealsete andmete kasulikkust organisatsioonile.

Töö põhieesmärkideks on:

- 1) analüüsida, millised on loodava süsteemi infovajadused ning võimalused nende täitmiseks olemasolevate süsteemide toel;

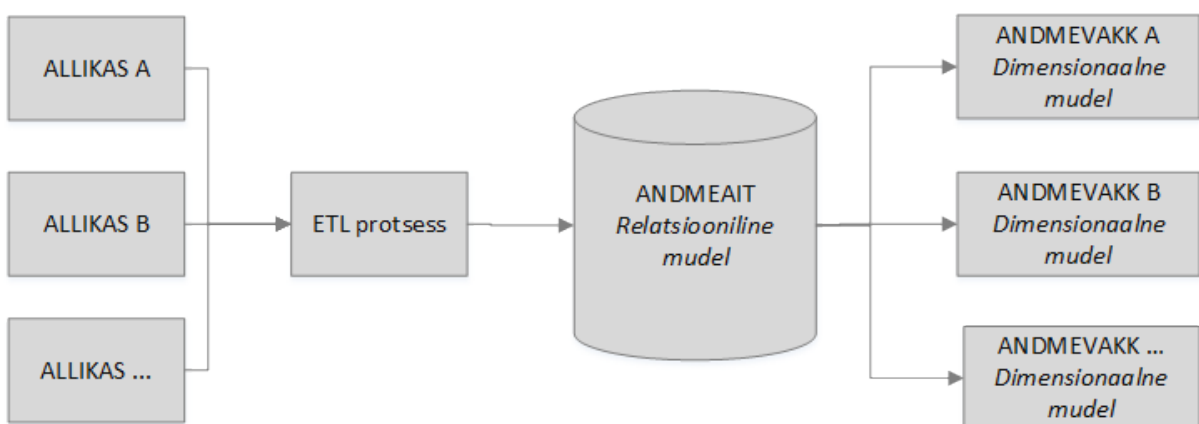
- 2) valmis arendada toimiv ning reaalselt kasu toov andmeanalüüsi vahend Eesti Rahvusringhäälingu meediahaldussüsteemi tarbeks.

1.3 Metoodika

Seatud eesmärkideni jõudmiseks uuritakse lähemalt, mis on meediahalduse infosüsteemi ning temaga seonduva arhiivi roll Eesti Rahvusringhäälingus ning milline on tema positsioon ülejäänud organisatsioonis olevate süsteemide suhtes, et omada üldpilti hetkeolukorrast. Seejärel selgitatakse välja millised on ootused ning nõuded sealsete andmete pärimiseks lõppkasutajate seisukohast, ehk uuritakse välja, milliseid funktsioone peab tulevane süsteem täitma. Selleks, et olemasolevaid andmeid süsteemi kasutajatele mõistetaval viisil esitada ning võimaldada neid mitmeti filtreerida arendatakse andmeait ning andmekuup. Projekti arendamise metoodika valikul lähtuti eelkõige püstitatud eesmärkidest. Metoodika valik viiakse läbi kahe lähenemise: Inmoni metoodika ja Kimballi elutsükli (ing. k. *Kimball Lifecycle*) vahel.

1.3.1 Inmoni ülevalt-alla metoodika

Inmoni metoodika on Bill Inmoni poolt 1990ndatel välja arendatud andmeaida arendamise metoodika. Selle põhiselt tuleb süsteemi arendamisele läheneda kogu organisatsiooni vaatest ning tulemuseks on relatsioonilist lähenemist kasutades arendatud andmeait, mis hõlmab kogu organisatsiooni tegevust. Süsteemi keskset andmeaita täiendavad väiksemad dimensionaalselt modelleeritud andmevakad, mille abil toimub süsteemist andmete pärimine (vt joonis 1). (Junic 2006, 86)



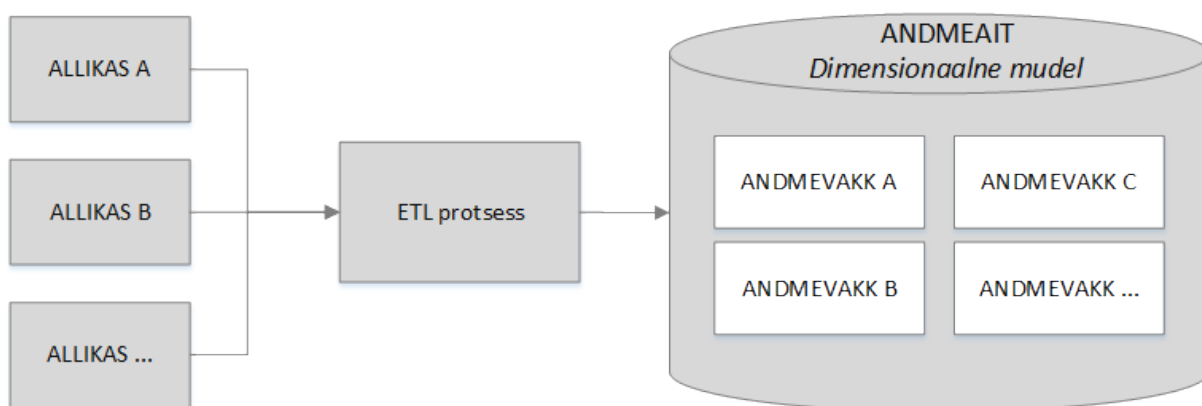
Joonis 1. Andmeaida ülevaade Inmoni metoodikal

Allikas: (Junic 2006, 86)

Andmete allikateks võivad olla erinevad organisatsioonis kasutusel olevate infosüsteemide andmebaasid. Seejärel viiakse läbi andmete laadimise protsess, mille tegevuse üldnimetusena on kasutusel termin ETL (ing. k. *Extract, Transformation and Load*), mille raames kogutakse andmed algallikatest, teisendatakse sobivale kujule ning seejärel laetakse need andmeaita. Kesksest andmeaidast küsivad omakorda andmeid dimensionaalselt modelleeritud andmevakad, mis oma olemuselt on ühe olemi põhised, näiteks tellimused. (Junic 2006, 86)

1.3.2 Kimballi alt-üles meetoodika

Kimballi meetoodika puhul on tegemist 1990ndatel avaldatud, peamiselt Kimball Groupi poolt kokku pandud andmeaitade ning ärianalüüsi projektide teostamiseks suunatud meetoodikaga. Vastukaaluks Inmoni meetoodikale, on Kimballi puhul tegemist alt-üles lähenemisega, mille puhul toimub süsteemi arendamine tegevusvaldkondade põhised andmevakkade kaupa. Andmevakkade arendusel lähtutakse dimensionaalsest modelleerimisest ning organisatsiooni hõlmav andmeait moodustub loodud andmevakkade kooslusest (vt joonis 2). (Ross 2009)



Joonis 2. Andmeida ülevaade Kimballi meetoodikal

Allikas: (Junic 2006, 86)

1.3.3 Meetoodika valik

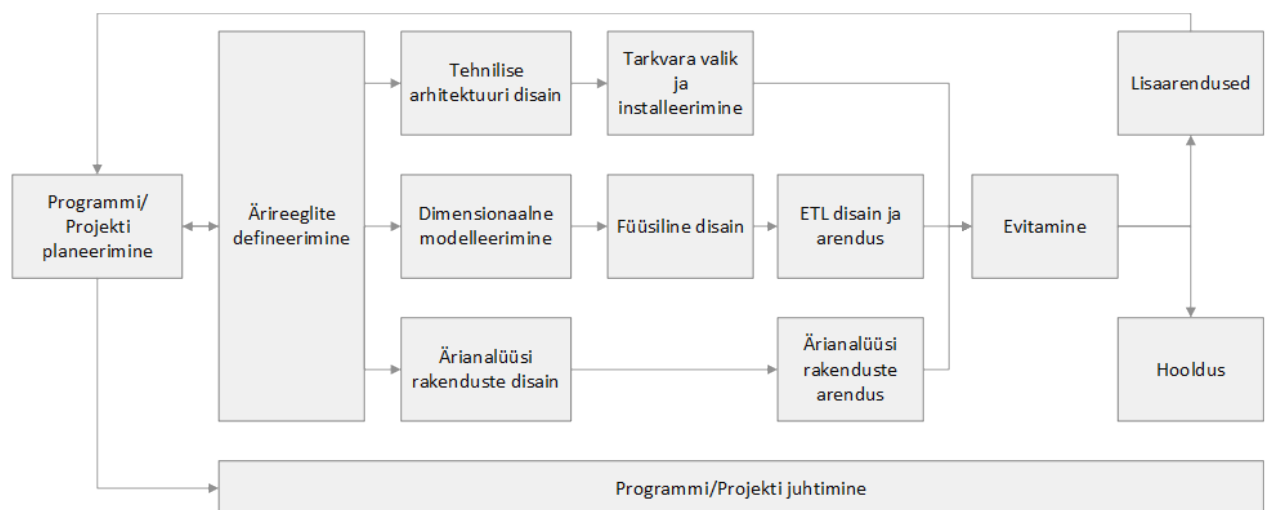
Meetoodika valiku puhul lähtutakse eelkõige konkreetsest olukorrast, et leida parim viis probleemi lahendamiseks. Kimballi alt-üles lähenemise puhul võib negatiivseks aspektiks pidada seda, et loodav dimensionaalne süsteem on eelkõige kasutatav vaid OLAP analüüsiks. Seevastu on Inmani ülevalt-alla lähenemise eeliseks see, et süsteemi keskseks olevat relatsioonilist andmeaita on võimalik kasutada nii OLAP analüüsiks, kui alternatiivseks andmete analüüsiks. Seevastu on süsteemi keskse andmeida loomine eelduseks, et arendada valmis andmevakad, mille põhjal lõppkasutajad andmeid pärivad ning analüüsivad. Kimballi

lähenemise puhul sellist eeldust ei ole ning selle võrra on Kimballi lähenemine lihtsustatud nii ajaliselt kui keerukuselt. Seega on kirjeldatud meetodikate peamiseks erinevuseks see, et Inmani puhul võetakse ette üldine vaade organisatsioonile ning probleemi lahendamisel lähtutakse üldisest organisatsioonist, kui Kimballi puhul lähtutakse konkreetselt ühest protsessist ning organisatsiooni üldisele vaatele ei keskenduta. (Junic 2006, 87)

Lähtuvalt käesoleva magistritöö eesmärgist, milleks on luua andmete analüüsimiseks mõeldud andmeait- ja kuup meediahaldussüsteemile, keskendudes meediaarhiiviga seotud tellimustele, valitakse projekti arendamise meetodikaks Kimballi lähenemine. Seda seetõttu, et eesmärgiks ei ole mitte kogu organisatsiooni hõlmava andmeaida loomine vaid keskendutakse ühele konkreetsele süsteemile, milleks on meediahaldus. Kogu organisatsioonile keskendumine oleks antud magistritöö mõistes liialt mahukas.

1.3.4 Kimballi elutsükkel

Kimballi meetodika arendusprotsessi kirjeldab lähemalt Kimballi elutsükkel, mida rakendades on edukalt läbi viidud tuhandeid andmeaida ja ärianalüüsi projekte. Selle puhul asetseb põhirõhk ettevõtte väärtuse suurendamisel, andmete dimensionaalsel esitamisel ning iteratiivsel arendusel. Kimballi elutsükkel koosneb mitmetest tegevustest saavutamaks hästi toimivat andmeaida disaini, arendust ning evitamist (ing. k. *deployment*) (vt joonis 3). (Ross 2009)



Joonis 3. Kimballi elutsükkel

Allikas: (Ross 2009)

Antud elutsükkel on teataval määral sarnane iteratiivsele tarkvara arenduse protsessile. Kimballi arendustsükkel algab projekti planeerimisega, mille käigus pannakse paika projekti eesmärgid, skoop, ärilised kokkulepped, ressursid, tehtavad tööd ning projekti ajakava. Projekti planeerimine lähtub osaliselt süsteemile esitatud ärilistest nõuetest ehk ärireeglitest, mille defineerimine on elutsükli järgmiseks sammuks. See, kui edukas on andmeaida projekt sõltub paljuski sellest, kui hästi mõistetakse, milliseid ärilisi vajadusi peab andmeait täitma. Seetõttu on arendajatel andmeaitasid arendades olulisem organisatsiooni ärilise poolega kursis olla, kui tavapäraselt tarkvara luues. Ärireeglitest lähtuvad järgmised kolm tegevuste voogu: tehnilise lahenduse voog, andmetega seonduv voog ning lõppkasutajale mõeldud ärianalüüsi (ing. k. *BI* ehk *Business Intelligence*) rakenduste voog, mida täidetakse paralleelselt. (Kimball *et al* 2008, 4)

Tehnilise lahenduse väljatöötamisel on vajalik eelkõige lähtuda süsteemile seatud ärilistest reeglitest, ehk sellest mida loodav süsteem peab tegema. Lisaks tuleb arvesse võtta olemasolev olukord ja selle tehnoloogiline lahendus ning sealt edasi on võimalik koostada tulevase süsteemi arhitektuurne lahendus. Tehnilise voo lõpetab loodud arhitektuuri elluviimine läbi valitud tarkvara installeerimise ning töökorda seadmise. (Kimball *et al* 2008, 5-6)

Ärireeglitest tulenevalt pannakse paika loodava süsteemi andmevajadused, millele järgneb dimensionaalse andmemudeli loomine, faktidel ja dimensioonidel põhinev disain. Lähemalt vaadeldakse dimensionaalset mudelit süsteemi analüüsi peatükis. Andmemudeli loomisele järgneb füüsiline disain, mille raames luuakse andmebaas, andmebaasi tabelite struktuur, turvasätted ning võimalusel OLAP andmebaasi disain. Füüsilisele disainile järgneb andmete laadimise protsessi koostamine ning andmete laadimine andmeaita. Andmete laadimise ja teisendamise ehk ETL protsessi koostamine on andmeaida arendusportsessi mahust ligi 70% ning seetõttu omab selle arukas ja läbimõeldult koostamine koguprotsessis suur osa. Ärianalüüsi rakenduste valikul tuleb eelkõige ettevõtte äri poolega läbi arutada, kuidas ja milliseid vahendeid andmetele ligipääsemiseks kasutusele võetakse. Rakendusteks peaksid olema vahendid, mis võimaldavad mugavalt päringuid koostada ja andmeid vaadata. (Kimball *et al* 2008, 7)

Eelnevalt välja toodud kolm tegevuste voogu võtab kokku süsteemi evitamine, mille käigus antakse valminud süsteem kasutusse. Süsteemi evitamisele järgneb süsteemi hooldus, mille puhul jälgitakse süsteemi kasutamist, andmete laadimise protsessi toimimist ja sellele

kuluvaid ressursse, lisatakse vajadusel indekseid ning tehakse süsteemist varukoopiaid. Lisaks süsteemi hooldamisele võib tulevikus tulla ette olukordi, kus süsteemi tuleb täiendada. Selle puhul on oluline silmas pidada, et lisaarenduste käigus ei tehtaks muudatusi juba olemasolevates andmetes. Lisaks eeltoodule viiakse iteratsioonide kestel läbi projekti juhtimine, mille eesmärgiks on hoida projekti planeeritud kursil. Projektijuhtimise raames kindlustatakse, et projekt jääks skoobi piiresse ning tagatakse sujuv kommunikatsioon äri ja arenduse poolte vahel. (Kimball *et al* 2008, 8)

1.4 Ülevaade tööst

Antud magistritöö puhul on tegemist projektiga, mille käigus antakse ülevaade Eesti Rahvusringhäälingus kasutusel olevast meediahaldussüsteemist ning analüüsitakse ja arendatakse nimetatud infosüsteemi tarbeks andmeid ja -kuup. Töö tulemuseks on toimiv süsteem, mis on arendatud lähtudes eelkõige organisatsiooni huvidest, valminud analüüsist ning parimast andmeida arendamise praktikast.

Kogu töö on jaotatud viite osasse: sissejuhatus, olemasolev süsteem ja alternatiivid, realisatsiooni analüüs, realisatsiooni teostamine ning kokkuvõte. Sissejuhatavas osas kirjeldatakse töö tausta ja sellega haakuvat probleemi, töö eesmärgid ning eesmärkide täitmiseks kasutatavat meetodikat. Olemasoleva süsteemi ja alternatiivide peatükis kirjeldatakse täpsemalt vaatluse all oleva meediahaldussüsteemi rolli ning asetsemist teiste ERR-is olevate süsteemide suhtes. Kirjeldatakse meediahaldussüsteemiga seonduva arhiivi kasutuse põhimõtteid ja vajalikkust organisatsioonile. Uuritakse lähemalt, millisel viisil on varem sarnaseid süsteeme loodud ning leitakse sobivaim lahendus. Realisatsiooni analüüsi käigus antakse ülevaade loodavast süsteemist, viiakse läbi nõuete analüüs, uuritakse põhjalikumalt kasutajate infovajadusi, koostatakse dimensionaalne andmemudel ning uuritakse andmete kvaliteeti. Realisatsiooni teostuse peatükis antakse ülevaade tehnilisest lahendusest ning põhjendatakse tarkvara valikut. Seejärel kirjeldatakse täpsemalt ETL protsesside koostamist ning toimimist; andmekuubi olemust ning arendusprotsessi; süsteemide evitamist ja andmete laadimise automatiseerimist. Lisaks viiakse läbi loodud süsteemile esitatud nõuete testimine ning arutletakse selle üle, millised on võimalused loodud süsteemi edasiseks arendamiseks. Töö viimaseks osaks on kokkuvõte.

1.5 Töö sihtrühm

Käesolev magistritöö võiks pakkuda huvi nii ettevõtetele kui andmeaitade süsteemide analüütikutele ja arendajatele. Organisatsioonidele võib olla huvipakkuvaks see, kuidas saada oma äritegevuse käigus tekkinud andmetest rohkem kasu ning käesolev töö annabki ülevaade, kuidas see ühe konkreetse näite puhul realiseeritud on. Süsteemi analüütikutele võib pakkuda huvi andmemudelitega seonduv informatsioon. Tarkvara arendajatele võib huvi pakkuda töö realisatsiooni käsitlev osa, milles kirjeldatakse süsteemi arendust ning selleks kasutusele võetud komponente.

2. OLEMASOLEV SÜSTEEM

Järgnevas peatükis antakse ülevaade hetkel olemasolevast süsteemist, selle komponentidest ja arhitektuurist. Lisaks kirjeldatakse arhiivmaterjali tellimise protsessi, esitatakse nõuded loodavale süsteemile ning analüüsitakse, millised on olemasoleva olukorra eelised ja puudused. Lisaks eeltoodule hinnatakse andmeaida rajamise otstarbekust.

2.1 Meediahaldussüsteem

Käesoleva magistritöö aluseks olev meediahaldussüsteem ehk MAM (ing. k. *Media Asset Management*) moodustab ühe osa Eesti Rahvusringhäälingu infosüsteemide kogumist. Kõnealune meediahaldussüsteem on kasutusel ERR-i telemajas ning seega toimub selles süsteemis peamiselt telesaadetega seonduvate videofailide haldamine, arhiveerimine ning arhiivist tellimine. Antud magistritöös keskendutakse meedia arhiiviga seonduvale arhiivmaterjali tellimise funktsionaalsusele, mis seisneb varasemalt läbi sama süsteemi arhiveeritud andmete uuesti küsimisel.

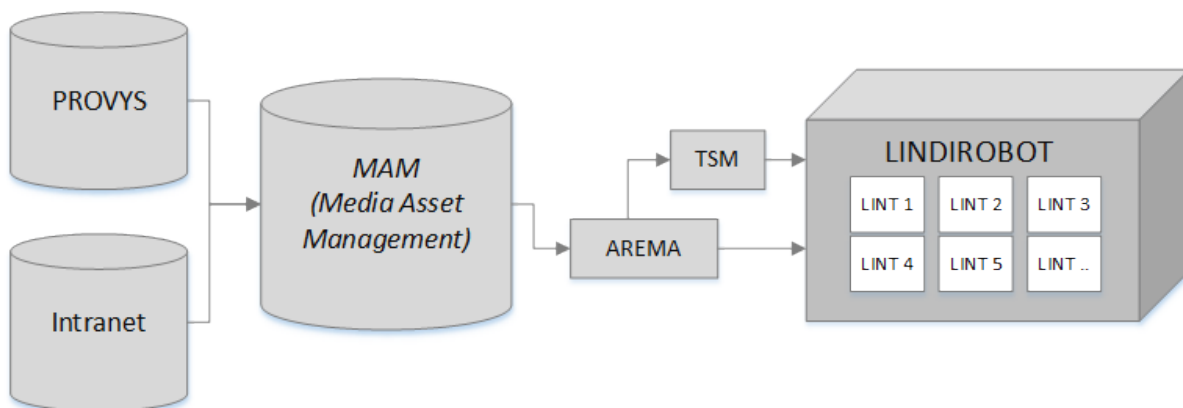
Arhiivi tellimusi tehakse enamjaolt järgnevate tegevuste tarbeks:

- 1) katteplaanid saadetele – kõige tihedamini kasutatakse süsteemi katteplaanide arhiivist tellimiseks. Selle tegevuse raames tellitakse mõnest arhiivis olevast videofailist sobiva pikkusega lõik või lõigud, mida kasutatakse uue saate tegemisel;
- 2) saadete eetrisse panemiseks – varem salvestatud ja monteeritud saated võivad minna eetrisse mõne aja möödudes pärast valmimist ja seetõttu on nad juba arhiivi salvestatud ning eetrisse minekul tellitakse nad sealt uuesti;
- 3) uudislugude tarbeks – arhiivist tellitakse uudiste koostamiseks videofaile, lõike ja pilte sinna varasemalt salvestatud materjalist;
- 4) veebilehekülje arhiiv.err.ee haldamiseks – veebileheküljele uute saadete ülesse panemiseks tehakse tellimus meediahaldussüsteemi, mis edastab seepeale tellitud videofaili koos metaandmetega ning süsteemi haldaja saab uue saate järelvaatamiseks avalikuks teha.

2.1.1 Olemasoleva süsteemi arhitektuur

Järgnevalt kirjeldatakse olemasoleva süsteemi arhitektuuri ning antakse ülevaade sealsetest komponentidest (vt joonis 4).

Saadetega seonduvat informatsiooni (nt info saate osalejate kohta) hoitakse süsteemis nimega PROVYS. See on süsteem, milles saavad alguse saadetega seonduvad videofailid, neile määratakse nimetus ning lisatakse külge metaandmed (nt kirjeldus). Failide nimetused on üle kogu süsteemi unikaalsed ning selle moodustamist kirjeldatakse alapeatükis 2.1.3. Lisaks on organisatsioonis kasutusel Intranet süsteem, mis täidab organisatsiooni siseveebi rolli, kuid lisaks on sinna juurde arendatud videofailide tellimise funktsionaalsus. Meediafailide haldamine toimub süsteemis nimega MAM, see on süsteem, mille abil toimub failide arhiveerimine, vajadusel arhiivist küsimine ning arhiiv.err.ee süsteemis avalikustamise ette valmistamine. AREMA (ing. k. *Archive and Essence Manager*) ning TSM (ing. k. *Tivoli Storage Manager*) on tarkvarad, mille kaasabil on võimalik suhelda lindirobotiga.



Joonis 4. Olemasoleva süsteemi arhitektuur

Lindirobot on füüsiline, lindikassettidel põhinev andmete varundamise seadeldis. Tegemist on IBM'i poolt pakutava lahendusega. Lindirobotit võib ette kujutada kui suurt kappi, mille riiulitel asetseb hulk lindikassette. Pilt lindirobotist on esitatud lisades (vt lisa 1). Lindikassettide mahutavus on varieeruv, vanemad kassetid mahutavad 800MB andmeid, uuemad mahutavad kuni 2,5TB. Seadeldises on võimalik lindikassette välja vahetada ning juurde lisada. Arvestades failide mahu pidevat kasvu, lisatakse juurde vaid uuemaid 2,5TB mahutavusega lindikassette. Juhul kui kõik riiulite nn vabad kohad täis saavad on võimalik soetada lisakapp ning see olemasolevasse süsteemi juurde liita. Kassettide lisamiseks, eemaldamiseks ning küsimiseks toimetab riiulite vahel robot, lähtuvalt AREMA või TSM'i

poolt antud käsklustest. Need süsteemid hoiavad endas infot selle kohta, kuidas lindikassetid seadmes paiknevad ning millised on lindikassetidele salvestatud failid. Kassetide identifitseerimiseks on igal kassetil triipkood, mida robot vajadusel skaneerib.

2.1.2 Arhiivmaterjali tellimise protsess

Arhiivist tellimuste tegemiseks on võimalikud kaks viisi: tellida materjali otse läbi meediahaldussüsteemi või kasutada Intranet süsteemi. Intranet süsteem on kasutusel eelkõige siseveebina, kuid sinna on juurde lisatud arhiivist tellimuste tegemise võimalus.

Mõlemal viisil kasutatakse tellimuste tegemiseks mugavat veebipõhist rakendust. Tellimuse tegemist alustatakse tavapäraselt märksõna otsinguga, mille abil leitakse üles soovitud fail. Juhul, kui leitud failist soovitakse saada mõni lõik, siis märgitakse süsteemis ära sobivad kaadrid ning seejärel kinnitatakse tellimus. Lõigu tellimise puhul määratakse tellimuse juures ära lõigu alguse ja lõpu ajakoodid, kujul "tt:mm:ss:ff", milles "ff" määrab ära kaadrid. Ajakoodide järgi toimub edasises protsessis faili ette valmistamine. Võimalik on tellida ka kogu saade ning seejuures ajakoode ei märgita. Ühe tellimuse raames on võimalik küsida infot mitme videofaili küljest. Kui tellimus on kasutaja poolt sisestatud, asub süsteem tellimust täitma.

Juhul, kui tellimus tehti läbi Intranet süsteemi siis salvestatakse tellimuse info Intraneti andmebaasi ning saadetakse selle põhjal päring meediahaldussüsteemi, kes omakorda hakkab tellimust täitma. Juhul, kui tellimust alustati MAM'ist siis algab tellimus sellest etapist. MAM edastab tellimuse AREMA süsteemile, mis omakorda viib läbi suhtluse lindirobotiga. Olenevalt tellitud videofaili arhiveerimise aastast suhtleb AREMA, kas otse lindirobotiga (failid alates aastast 2014) või kasutab AREMA lindirobotiga suhtluse vahelüli TSM-i (failid kuni aastani 2014). Protsess toimub nii seetõttu, et varasemalt ei olnud AREMA tarkvaral oskust otse lindirobotiga suhelda. Nende süsteemide tööks on kommunikeerida tellimuste päring lindirobotile, sealhulgas päringus oleva ajakoodi tõlkimine. Faili küsimisel otsib robot õige lindikasseti ning pistab selle lugejasse. Lugejas valmistatakse ette küsitud fail, vajadusel lõigatakse failist vastavalt ajakoodile sobiv osa ning edasi saadetakse see eelnevalt määratud kausta, mille kaudu on võimalik kasutajal tellitud fail kätte saada.

2.1.3 Failide nimetamine

Failide nimesid käsitletakse süsteemis failide identifikaatorina. Eristatakse tavalisi faile ning uudistega seonduvaid faile. Unikaalne nimetamine on näiteks oluline saate versioonide

eristamiseks. Samuti toimub failide arhiivist leidmine nende nimede põhjal. Nimedest on võimalik välja lugeda faili loomise aasta, projekt (ehk saade), mille raames on antud fail loodud ning lisaks inimloetavale tekstile on nimedes kirjas mitmed tehnilised tähised, mida süsteemid faili kohta info saamiseks kasutavad.

Nimetamise näited:

- 1) tavaline faili nimetamise muster: {indeks}-{jrk. nr}_{tehniline tähis}_{saate nimetus}.{faili laiend}. Milles muutuja {indeks} jaotub omakorda järgmiselt: {aasta}-{projekti nr}{osa nr}. Näiteks ühe “Puutepunkti” saate faili nimi on selline: 2013-002021-0222_0001_D10_PUUTEPUNKT.MXF;
- 2) uudise faili nimetamise muster: {kpv kujul aaaakkpp}_{tähis}_{nimetus}.{faili laiend}. Näiteks eesti keelse “Aktuaalse kaamera” tarvis kasutatud Hilma af Klinti näituse videolõigu faili nimetus on järgmine: 20150313_AKEST_HILMA-AF-KLINT.MXF.

2.2 Nõuded loodavale süsteemile

Loodavalt süsteemilt oodatakse eelkõige võimalust saada lisainformatsiooni meedia arhiiviga seonduvate tellimuste osas. Seda seetõttu, et antud andmeid ei ole varasemalt põhjalikult uuritud, kuid võimalikud tulemused on huvipakkuvad ning kasulikud. Arhiivis olevate tellimustega seonduvalt pakuvad organisatsioonile huvi järgmised aspektid:

- 1) kui kaua võtab aega tellimuse täitmine alates tellimuse alustamisest – võimaldab saada ülevaate sellest, kuidas ajaliselt tellimusi täidetakse;
- 2) leida, kuidas jaotub tellimuste esitamise aktiivsus kasutajate lõikes – võimaldab saada ülevaate sellest, millised kasutajad esitavad enim tellimusi;
- 3) milline on tellimuste esitamise aktiivsus nädalapäevade ning tundide lõikes – annab ülevaate süsteemi kasutatavusest nädalapäevade ning kellaaegade raames;
- 4) millistel aegadel tellitakse kõige enam faile – annab ülevaate sellest, millistel perioodidel on süsteemi töö kõige tihedam;

- 5) kuidas jaotuvad tehtud tellimused staatuste lõikes – annab ülevaate sellest, kui palju on süsteemis õnnestunud, ebaõnnestunud ning lõpetamata tellimusi;
- 6) kui palju on korduvaid tellimusi samadest saadetest ja nende lõikudest – võimaldab saada ülevaate sellest, milliseid saateid kõige enam videolõikude tellimiseks kasutatakse. Selline info aitaks vältida olukorda, kus näiteks kütuse hinna muutustest rääkivas uudises kasutatakse pidevalt ühest ja samast tanklast tehtud katteplaani.

Edasises töös tehtav süsteemi analüüs ning funktsionaalsete nõuete koostamine võtavad aluseks loetletud nõuded, et kindlustada süsteemi vastavust organisatsiooni huvidele.

2.3 Olemasoleva olukorra eelised ja puudused

Järgnevalt analüüsitakse, millised on olemasoleva olukorra eelised ja puudused, et anda ülevaade andmeaida lisandumisega seonduvatest ohtudest ja võimalustest.

Hetkel on meediahaldussüsteem oma komponentide poolest küllaltki keerukas ning uute süsteemide ja võimaluste lisandumisega muutub kasutamine komplitseeritumaks. See annab omakorda väljakutse süsteemi kasutajatele. Juhul, kui organisatsioonis võetakse kasutusele selline tarkvara, millega varasem kokkupuude puudub, võib osutada vajalikuks teha töötajatele sellekohane koolitus ning see muudab süsteemi juurutamise protsessi teataval määral keerulisemaks. Jäädes olemasoleva lahenduse juurde, ei tuleks süsteemi omanikel koolitustegevustele ressursse kulutada.

Teisalt näeb organisatsioon, et süsteemis olevate andmete täielikku potentsiaali hetkel ei kasutata ja nende andmete informatsiooniks muutmine loob olulist lisaväärtust. Andmeaida ja –kuubi loomine võimaldab arhiivi tellimustega seonduvaid andmeid analüüsida ning nende põhjal päringuid koostada. Selline funktsionaalsus on hetkel olemasolevas süsteemist puudu, kuid võimaldaks saada ülevaate tellimustega seonduvast.

Praeguse süsteemi tarbeks on organisatsioonil vajalik riist- ja tarkvara ning töötajad olemas. Uue süsteemi kasutusele võtmisel on võimalik, et vajatakse lisaressursse, näiteks andmebaasiserveri tarkvara või kettamahu näol. Lisaks lasub organisatsioonil kohustus uut süsteemi töökorras hoida ning hallata, mis omakorda tähendab, et tuleb hoolt kanda selle eest, et antud tegevuse tarvis jätkuks süsteemi administraatoritel piisavalt aega. Hetkel on Eesti Rahvusringhäälingus olemas andmeaida süsteemi realiseerimiseks ning haldamiseks vajalikud

ressursid ning seetõttu ei ole vaja lisakulutusi teha. Juhul, kui tulevikus planeeritakse süsteemi edasi arendada, tuleb antud küsimuse juurde tagasi pöörduda.

2.4 Andmeaida otstarbekuse hindamine

Andmeaida projekti läbiviimise ning süsteemi otstarbekuse edukus sõltub eelkõige organisatsiooni ärilise poole ja tehnoloogilise poole üksteise mõistmisest ning koostööst. Juhul, kui süsteemi rajavad arendajad ei võta arvesse ettevõtte ärilisi huve, ei pruugi loodav süsteem oma eesmärki täita. Seetõttu on süsteemi rajamise üheks suurimaks riskiks see, et luuakse süsteem, mis ei ole võimeline osalema ärilise poole otsustusprotsessis. Sellise olukorra vältimiseks antud magistritöö raames rajava andmeaida puhul, korraldatakse koosolekuid ning selgitatakse välja organisatsiooni vajadused ja nõuded loodavale süsteemile. Järgnevalt hinnatakse andmeaida rajamise otstarbekust lähtuvalt Ralph Kimballi poolt selle tarbeks välja toodud aspektidest (2008, 16).

Üheks põhiliseks teguriks on ärilise poole esindaja olemasolu. See on oluline, kuna äriiline pool omab visiooni loodava süsteemi potentsiaalsest kasust organisatsiooni tegevusele. Projekti läbiviimine on edukam, kui ärilise poole esindaja puhul on tegemist tugeva isiksusega, kes on ettevõtte protsessidest hästi informeeritud. Lisaks on võimalik, et projekti kaasatakse mitu äri poole esindajat, eesmärgiga hoida süsteemi rajamine võimalikult objektiivne. (Kimball *et al* 2008, 16)

Andmeaida süsteemi rajamine on otstarbekas, kui organisatsioonil on motivatsiooni see kasutusele võtta. Tihti peale on motivatsiooni allikaks ärilise edukuse poole püüdlemine. Motivatsioon võib olla tingitud välistest teguritest, näiteks nähakse, et süsteemi kaasamine võib anda edu konkurentide ees. Teisalt on reaalne, et andmeaida süsteemi abil on võimalik ettevõtte siseseid protsesse parendada ning produktiivsust tõsta. Seega on läbiviidav projekt mõistlik, kui andmeaida omajad on motiveeritud süsteemi kasutama ning näevad, et see võib reaalselt kasu tuua. (Kimball *et al* 2008, 17)

Lisaks tuleb andmeaida rajamisel hinnata seda, kas süsteem on teostatav. Selleks tuleb arvesse võtta nii tehnilist poolt, kui nõuete täitmiseks vajaminevate andmete olemasolu. Juhul, kui salvestatud andmed on vigased, osaliselt puudulikud või puuduvad täiesti on süsteemi teostatavus kahtluse all. Puudulike või vigaste andmetega ei ole võimalik adekvaatselt ärilise poole otsustusprotsessi toetada. (Kimball *et al* 2008, 17)

Käesoleva magistritöö raames rajatava andmeaida puhul on Eesti Rahvusringhäälingu poolt olemas kaks esindajat, kes omavad loodavale süsteemile esitavatest nõuetest selget visiooni ning seega on süsteemi rajamisel olemas tugev äriiline pool. Lisaks on süsteemi kasutajate vaatest olemas motivatsioon süsteem kasutusele võtta, seda seetõttu, et uurimise all olevaid andmeid ei ole varem analüüsitud ning tulemused on huvipakkuvad. Meediahaldussüsteemis ning Intranetis olevate tellimustega seonduvad andmed on täielikul kujul olemas ning seetõttu võib pidada süsteemi teostatavaks. Lähtuvalt eelnevatest aspektidest võib hinnata, et andmeaida rajamine on otstarbekas.

3. SÜSTEEMI ANALÜÜS

Järgnevas peatükis tuuakse välja loodava süsteemi skoop, kirjeldus ning komponendid. Lisaks antakse ülevaade süsteemile esitatud äriolistest nõuetest, viiakse läbi nõuete analüüs ja pannakse kirja süsteemi kasutaja infovajadused. Lisaks eelnevale koostatakse andmemudel, mis on aluseks süsteemi rajamisele.

3.1 Loodava süsteemi skoop ja kirjeldus

Planeeritav süsteem on oma olemuselt andmeait. Loodava süsteemi skoopi võetakse Intranet süsteemis oleva arhiivmaterjali tellimustega seonduva informatsiooni hoiustamine ning nende andmete põhjal päringute koostamise võimaldamine. Selleks luuakse andmekuup, mille abil on võimalik andmeaidas sisalduvat informatsiooni mõistlikul viisil lõppkasutajatele esitada ning infot analüüsida. Lähtudes antud töö mahust piiritletakse skoopi Intranet süsteemist tulenevate tellimustega. Siiski peetakse antud süsteemi luues silmas seda, et edasiste arenduste sisseviimine oleks võimalikult lihtne.

Andmeaida ja andmekuubi olemust ja arendust käsitletakse pikemalt realisatsiooni peatükis. Seega on loodaval süsteemil kaks põhilist komponenti: andmeait ja –kuup. Lähtuvalt organisatsiooni huvitavatest aspektidest (peatükk 2.2), tuleb andmeaidas hoiustada tellimuste andmed, mis sisaldavad järgmist informatsiooni:

- 1) mis ajal tellimust alustati ning millal see lõpetati (täideti);
- 2) millisest failist tellimus tehti;
- 3) milliseid löike failist telliti;
- 4) milline süsteemi kasutaja tellimuse esitas;
- 5) milline on tellimuse staatus.

Nende andmete põhjal on võimalik koostada päringuid mitmel viisil, neid omavahel kombineerides. Seda, kuidas tellimustega seonduvaid andmeid vaadelda, otsustab lõppkasutaja. Selliselt loodud süsteem on dünaamiline ning võimaldab lõppkasutajal andmeid erinevatest aspektidest analüüsida. Näiteks on faili tellimuse aja järgi võimalik uurida, kui

kaua võtab aega keskmine/kõige pikem/kõige lühem tellimus ning seda, millistel aegadel tehti kõige enam tellimusi.

Süsteemi loomisel peetakse eelkõige silmas Eesti Rahvusringhäälingu poolt süsteemile esitatud nõudeid, et luua lõpptulemus, mis organisatsioonile reaalselt lisaväärtust looks. Teisalt luuakse süsteem selliselt, et laienduste ning muudatuste sisseviimine oleks võimalikult lihtne. Seda seetõttu, et organisatsioonis kasutusel olevad infosüsteemid on pidevas arengus.

3.2 Nõuete analüüs

Järgnevalt kirjeldatakse süsteemile esitatavad nõuded, mis on sisendiks andmemudeli disainimisele, edasisele süsteemi arendusele ning üldisele süsteemi haldamisele. Samuti pannakse kirja andmeaidale seatavad mittefunktsionaalsed nõuded. Nõuetele lisatakse juurde identifikaatorid (F – funktsionaalne nõue, MF – mittefunktsionaalne nõue), et lihtsustada neile viitamist.

3.2.1 Funktsionaalsed nõuded

Välja toodud funktsionaalsed nõuded on kirja pandud võttes arvesse Eesti Rahvusringhäälingu poolt esitatud nõudeid loodavale süsteemile (peatükk 2.2). Seega peab andmeaida kasutajatel olema võimalik teostada järgmisi tegevusi:

- 1) leida, millal algas ning millal lõppes tellimuse täitmine (sh leida kestvus) **(F01)**;
- 2) leida, milline süsteemi kasutaja oli tellimuse esitajaks (sh leida kasutaja poolt esitatud tellimuste kogus) **(F02)**;
- 3) leida, kui palju on tehtud tellimusi ühest failist **(F03)**;
- 4) leida, milliseid lõike on tellitud antud failist **(F04)**;
- 5) leida tellitavate failide kogus määratud ajahetkel **(F05)**;
- 6) leida tellimuste arv tellimuse staatuste lõikes **(F06)**.

3.2.2 Infovajadused

Funktsionaalsete nõuete täitmiseks on süsteemi kasutaja seisukohast järgmised teabevajadused: tellimuste loetelu, failide loetelu, kasutajate loetelu, tellimuse staatuste

loetelu ning failist tellitud videolõikude loetelu. Eeltoodud andmete põhjal on võimalik koostada nii nõuetena määratletud päringuid, kui kombineerida uusi väljundeid.

3.2.3 Mittefunktsionaalsed nõuded

Järgnevalt vaadatakse süsteemi omadusi ehk mittefunktsionaalseid nõudeid lähtuvalt ISO/IEC 9126 standardi kvaliteediatribuutidest. Tegemist on laialdaselt kasutatud tarkvara kvaliteedi standardiga, mis määratleb kvaliteedi hindamiseks kasutatava mudeli. See on defineeritud läbi tarkvarale omaste karakteristikute (nt hooldatavus) ning alam-karakteristikute (nt analüüsitavus, muudetavus, stabiilsus) (vt tabel 1). Mittefunktsionaalsete nõuete defineerimisel lähtutakse antud kvaliteedi atribuutidest, et vältida oluliste nõuete kõrvale jätmist. (Tepandi 2015, 100)

Tabel 1. Standardi ISO/IEC 9126 kvaliteedi mudel

Funktsionaalsus	Töökindlus	Kasutus-kõlblikkus	Tõhusus	Hooldatavus	Porditavus
sobivus	küpsus	arusaadavus	ajaline käitumine	analüüsitavus	sobitavus
õigsus	tõrketaluvus	õpitavus	ressursikasutus	muudetavus	installeeritavus
koostalitlusvõime	taastuvus	käsitsetavus	tõhususe vastavus	stabiilsus	koosoluvõime
turvalisus	töökindluse vastavus	meeldivus		testitavus	vahetatavus
funktsionaalsuse vastavus		kasutus-kõlblikkuse vastavus		hooldatavuse vastavus	porditavuse vastavus

Allikas: (Tepandi 2015, 100)

3.2.3.1 Funktsionaalsus

Loodav süsteem peab täitma süsteemi tellija poolt ette nähtud nõudeid ning funktsionaalsuseid, et täita antud süsteemile seatud eesmärgid (nõuded **F01 – F06**).

3.2.3.2 Töökindlus

Süsteemist tuleb teha regulaarselt varukoopiaid (kord päevas), et võimalike tõrgete puhul oleks võimalik süsteem taastada (**MF01**). Andmeaita andmete laadimise protsess peab

töötama öösiti, et vältida süsteemile lisakoormuse seadmist, kuna päeval ajal pöörduvad süsteemi poole ka lõppkasutajad (MF02).

3.2.3.3 Kasutuskõlblikkus

Loodavas süsteemis tuleb arvesse võtta lokaliseerimist: kasutusel peab olema ajaformaad kujul “pp.kk.aaaa tt:mm:ss” (MF03). Lähtudes olemasolevate süsteemide keelest, peab olema ka loodav süsteem inglise keelne (MF04). Päringute vastuste koostamisel kasutajatele kuvatavad muutujate nimed peavad olema sõbralikud elik kasutajatele mõistetavad (MF05).

3.2.3.4 Tõhusus

Süsteemis loodavaid andmebaasipäringuid tuleb optimeerida, et vältida pikki reaktsiooniaegaid. Kasutaja päringule tuleb vastus edastada 15 sekundi jooksul (MF06).

3.2.3.5 Hooldatavus

Süsteemi peab olema võimalik lihtsasti hooldada, see tähendab, et paranduste ja täienduste sisseviimine peab olema hõlbus. Selle tagamiseks tuleb süsteemi arendades funktsionaalsused implementeerida selliselt, et hilisem täienduste lisamine oleks mõistlikul viisil teostatav (MF07). Realisatsiooni raames tuleb koostada süsteemi täienduste tegemiseks ning süsteemi evitamiseks juhend (MF08).

3.2.3.6 Porditavus

Loodav süsteem peab toimima andmebaasiserveris, mis sisaldab ka teisi andmebaase (MF09). Süsteemi peab olema võimalik ilma lisaarendusteta ülesse seada uues sobivas keskkonnas, seda nii olukorras, kus süsteem evitatakse või juhul, kui süsteemi server välja vahetatakse (MF10).

3.3 Andmemudeli disainimine

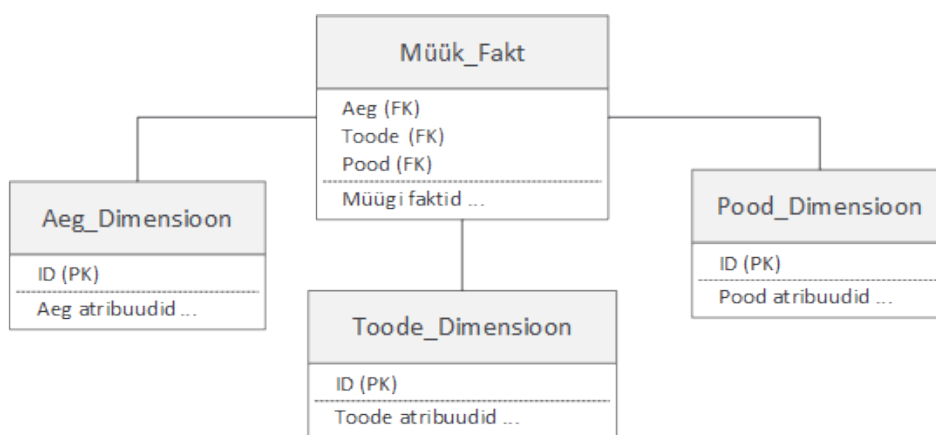
Järgnevas töö osas kirjeldatakse andmemudeli disaini, esitatakse kaks enamlevinud dimensionaalset mudelit (täht- ja lumehelbeskeem) ning viiakse läbi disainimustri valik. Lisaks analüüsitakse ja esitatakse loodava süsteemi andmemudel, mis võetakse edaspidi andmeaida arendamise aluseks.

3.3.1 Tähtskeem ja lumehelbe skeem

Andmehoidlate disainimisel kasutatakse enamjaolt dimensionaalset lähenemist, mille eesmärgiks on muuta andmete struktuur lihtsaks ning kergesti mõistetavaks. Dimensionaalse mudeli olulisteks komponentideks on faktid (ing. k. *facts*) ning dimensioonid (ing. k. *dimensions*). Fakti puhul on tegemist olemit iseloomustava numbrilise väärtusega, näiteks toote hind ja kogus. Dimensiooni puhul on aga tegemist olemit iseloomustavate tekstiliste väärtustega, näiteks toote nimetus ja bränd. Kaks selle valdkonna enamlevinud mudelit on tähtskeem (ing. k. *star schema*) ja lumehelbe skeem (ing. k. *snowflake schema*). (Kimball, Ross 2002, 17)

Tähtskeemi puhul on tegemist dimensionaalse mudeliga, mille keskmeks on ühtne faktidest koosnev tabel (ing. k. *fact table*) ning teda ümbritsevad talle viitavad dimensioonitabelid (ing. k. *dimension table*). Faktitabelisse koondatakse faktid ning välisvõtmed, mis viitavad skeemi dimensioonidele. Selleks, et ära tunda, kas olemit kirjeldav numbriline väärtus on fakt või dimensioon, tuleb mõelda sellele, kas antud väärtus on pidevalt muutuv äriline mõõdik või pigem olemit iseloomustav konstantne number. Näiteks toote hind võib mõnel juhul tunduda konstantse dimensioonina, kuid kui see pidevalt muutub, võib seda käsitleda faktina. Toote kaalu sobib aga käsitleda dimensioonina, kuna enamjaolt on tegemist muutumatu väärtusega. Dimensioonitabelitesse koondatakse eelpool kirjeldatud dimensioonid ehk olemit kirjeldavad diskreetsed tekstilised väärtused. Neisse tabelitesse lisatav info, peaks olema lõppkasutajale võimalikult arusaadaval ja loetaval kujul, näiteks tuleks andmetes sisalduvad koodid asendada arusaadavate väärtustega. Selline lähenemine muudab andmeaida süsteemi kasutajasõbralikuks. Dimensioonitabelisse tuleks koondada võimalikult palju olemile lisaväärtust andvaid kirjeldusi, et hoida süsteemi lõppkasutuse võimalusi piisavalt avarana. Tähtskeemi puhul on loodavad dimensioonitabelid tugevalt denormaliseeritud, et hõlbustada info esitamise arusaadavust. Tähtskeemi dimensioonitabelites määratakse primaarvõti, kuid sinna ei looda seoseid teiste dimensioonitabelitega ning seetõttu välisvõtmeid ei lisandu. Dimensioonitabelites olevad atribuudid on olulised, kuna neid kasutatakse edaspidi päringute

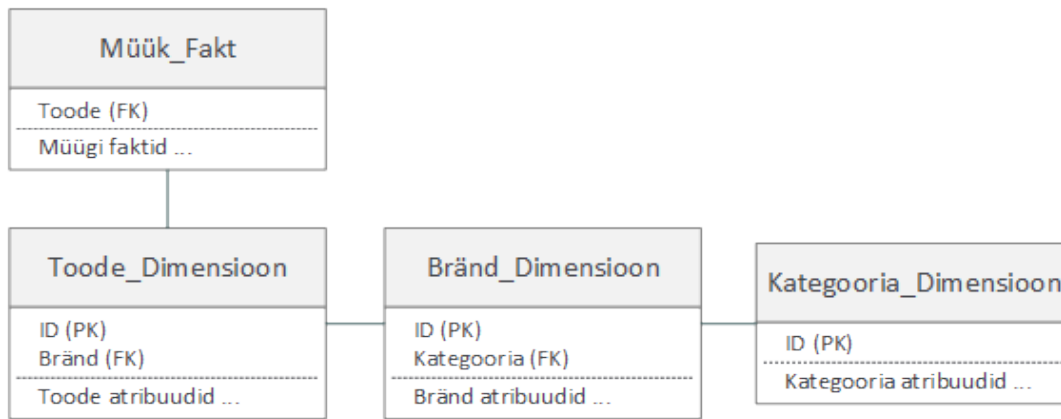
ning raportite veergudena. Andmeaida võimekus on tugevalt seotud sellega, kui kvaliteetsed, kasutatavad, hästi arusaadavad ning informatiivsed on dimensioonid. Kokkuvõtvalt võib öelda, et mida rohkem aega on panustatud dimensioonide läbitöötamisele ning sealsete andmete õigsusele, seda sisukam ja parem on loodud andmeait. Näide tähtskeemist on toodud välja joonisel 5, mille puhul faktitabeliks on tabel nimega “Müük_Fakt” ning dimensioonitabeliteks on tabelid “Aeg_Dimensioon”, “Pood_Dimensioon” ja “Toode_Dimensioon”. Loodud näide iseloomustab hästi tähtskeemi lihtsust ning arusaadavust lõppkasutajale. Lisaks on sellise struktuuri eeliseks parem jõudlus päringute tegemisel, kuna infot ei tule küsida mitmete tabelite ühendamise teel. (Kimball, Ross 2002, 19 - 21)



Joonis 5. Fakti- ning dimensioonitabelite asetsemine tähtskeemil

Allikas: (Kimball, Ross 2002, 22)

Lumehelbe skeem on olemuselt sarnane tähtskeemile, omades nii dimensioonitabeleid kui faktitabeleid, kuid teda on rohkem normaliseeritud kui tähtskeemi. Lumehelbe skeemi puhul on võimalik, et üks dimensioonitabel omab viidet või viiteid teistesse dimensioonitabelisse. Näiteks dimensioon toode omab viidet tabelisse bränd ja viimane omakorda tabelisse kategooria. Siiski muudab tabelite normaliseerimine andmeaida struktuuri komplitseeritumaks, optimeerimise keerulisemaks ning aeglustab päringute tegemist. Kuna andmeaitades võivad tabelite mahud kasvada gigabaitidesse, ei anna dimensioonitabelite normaliseerimine andmeaida kogumahule märkimisväärset võitu. Lisaks muudab lumehelbe esitus päringute tegemise lõppkasutajale keerukamaks. Kokkuvõttes tuleks andmeaida modelleerimisel lumehelbe disain kasutusele võtta vaid siis kui tema eelised on piisavalt põhjendatud. Joonis 6 kirjeldab lumehelbe skeemi väljanägemist. (Kimball, Ross 2002, 57)



Joonis 6. Lumehelbe skeem

Allikas: (Kimball, Ross 2002, 55)

3.3.2 Disainimustri valik

Lähtuvalt eelnevas peatükis välja toodud disainimustritele võetakse loodava süsteemi andmemudelil disainimisel kasutusele tähtskeem. Seda seetõttu, et loodav andmemudel oleks võimalikult lihtsa struktuuriga ning süsteemi jõudlus oleks heal tasemel. Analüüsisid hetkeolukorda ning andmemahte, ei kaalu tähtskeemi eeliseid üle lumehelbe skeemi eelis, milleks on saada võitu kogumahtudest.

3.3.3 Süsteemi andmemudel

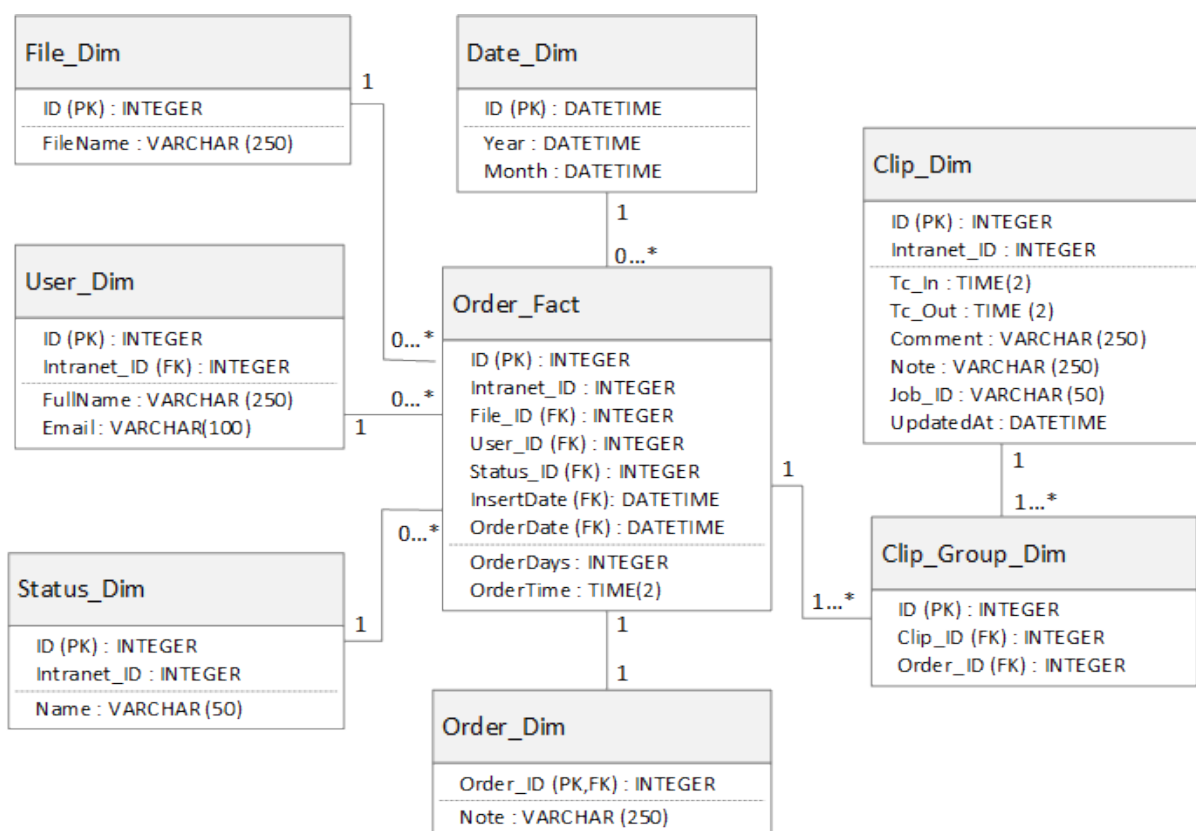
Järgnevalt esitatakse andmeaida loomise aluseks olev andmemudel, mille koostamisel on eelkõige silmas peetud ärivajadusi ning süsteemile esitatud nõudeid. Selleks, et vältida piirangute seadmist süsteemist info küsimisele pööratakse tähelepanu heale modelleerimisele.

Tulenevalt mittefunktsionaalsest nõudest nr 4 (MF04), milleks on, et loodav süsteem tuleb rajada inglise keelsena, luuakse ka andmemudel inglise keeles. Seda seetõttu, et juba olemasolevate süsteemide arendamisel on antud keelt kasutatud ning selliselt säilib ühtsus.

Joonisel 7 on esitatud loodava süsteemi andmemudel, mille moodustavad kaheksa tabelit:

- 1) kuus dimensioonitabelit: Order_Dim, Clip_Dim, Date_Dim, File_Dim, User_Dim, Status_Dim;
- 2) üks faktitabel: Order_Fact;
- 3) üks vahetabel: Clip_Group_Dim.

Järgnevalt on esitatud loodava süsteemi andmemudel:



Joonis 7. Loodava süsteemi andmemudel

3.3.4 Tabelite kirjeldused

Dimensiooni-, vahe-, ning faktitabelite täpsemad kirjeldused tuuakse välja tabelites 2 kuni 9. Primaarvõtmed ning välisvõtmed on märgitud vastavalt PK (ing. k. *primary key*) ja FK (ing. k. *foreign key*).

Tabel 2. Tellimuste faktitabeli kirjeldus

Order_Fact				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti. Automaatselt edenev, sammuga 1.
Intranet_ID	Täisarv	-	Jah	Identifikaator Intranet süsteemis.
File_ID	Täisarv (FK)	-	Jah	Viide File_Dim tabeli ID veerule.
User_ID	Täisarv (FK)	-	Jah	Viide User_Dim tabeli ID veerule.
Status_ID	Täisarv (FK)	-	Jah	Viide Status_Dim tabeli ID veerule.

Order_Fact				
InsertDate	Kuupäev	-	Jah	Tellimuse lisamise kuupäev.
OrderDate	Kuupäev	-	Ei	Tellimuse täideviimise kuupäev.
OrderDays	Täisarv	-	Ei	Tellimuse täitmisele kulunud päevade arv.
OrderTime	Kellaaeg	-	Ei	Tellimusele kulunud aeg minutites, kujul tt:mm:ss.

Tabel 3. Tellimuste dimensioonitabeli kirjeldus

Order_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
Order_ID	Täisarv (PK,FK)	-	Jah	Primaarvõti, sh omab viidet Order_Fact tabelisse.
Note	Tekst	250	Jah	Märkused.

Tabel 4. Failide dimensioonitabeli kirjeldus

File_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti. Automaatselt edenev, sammuga 1.
FileName	Tekst	250	Jah	Faili nimetus (vt peatükk 2.1.3 Failide nimetamine).

Tabel 5. Kasutajate dimensioonitabeli kirjeldus

User_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti. Automaatselt edenev, sammuga 1.
FullName	Tekst	250	Jah	Kasutaja täisnimi.
Email	Tekst	100	Ei	Kasutaja e-posti aadress.

Tabel 6. Klippide vahetabeli kirjeldus

Clip_Group_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti. Automaatselt edenev, sammuga 1.
Clip_ID	Täisarv (FK)	-	Jah	Viide tabeli Clip_Dim ID veerule.
Order_ID	Täisarv (FK)	-	Jah	Viide tabeli Order_Fact ID veerule.

Tabel 7. Klippide dimensioonitabeli kirjeldus

Clip_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti.
Intranet_ID	Täisarv	-	Jah	Identifikaator Intranet süsteemis.
Tc_In	Kellaaeg	-	Jah	Klipi alguse ajakood, kujul tt:mm:ss.
Tc_Out	Kellaaeg	-	Jah	Klipi lõpu ajakood, kujul tt:mm:ss.
Comment	Tekst	250	Jah	Kommentaar.
Note	Tekst	250	Jah	Märkused.
Job_ID	Tekst	50	Jah	Tellitud klipi identifikaator MAM süsteemis.
UpdatedAt	Kuupäev	-	Jah	Antud rea viimase uuendamise kuupäev. Kujul pp.kk.aaaa tt:mm:ss.

Tabel 8. Kuupäeva dimensioonitabeli kirjeldus

Date_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Kuupäev (PK)	-	Jah	Primaarvõti. Unikaalne kuupäev kujul pp.kk.aaaa tt:mm:ss.
Year	Kuupäev	-	Jah	Aasta.
Month	Kuupäev	-	Jah	Kuu.

Tabel 9. Staatuste dimensioonitabeli kirjeldus

Status_Dim				
Veeru nimi	Andmetüüp	Pikkus	Kohustuslik	Kommentaar
ID	Täisarv (PK)	-	Jah	Primaarvõti. Unikaalne, edeneb sammuga 1.
Intranet_ID	Täisarv	-	Jah	Identifikaator Intranet süsteemis.
Name	Tekst	50	Jah	Staatuse nimetus.

3.4 Andmete kvaliteet

Andmeaida üheks eesmärgiks on ettevõtte siseselt otsustusprotsessi toetamine, seeläbi on andmete õigsus ning kvaliteet andmeidas äärmiselt olulised. Kvaliteetsete andmete põhjal otsuseid tehes on võimalik firma poolt pakutavaid teenuseid parendada, kliente põhjalikumalt analüüsida, kulusid juhtida jm. Selleks, et nimetatud tegevuste juhtimist olemasoleva infoga toetada on tähtis vältida valeinfo esitamist ning selle aluseks võtmist otsuste tegemisel. Juhul, kui ettevõtte operatsioonilistes andmebaasides on info vigane, tuleb peatada nende vigade edasi kandumine andmeaita. Väärade andmete sisaldumine andmeidas on üks peamisi põhjuseid, miks kasutajad süsteemi vastu usalduse kaotavad ning läbi selle on võimalik, et tehtud arendus ebaõnnestub täielikult. (Ponniah 2001, 292)

Kvaliteetsed on need andmed, mis on kooskõlas oma sisult ja mahult neile seatud piirangutega. Näiteks soo väljal on võimalikud väärtused “mees” või “naine”. Lisaks tuleb kindlustada vastavus seatud andmetüübile, kui väli eeldab tekstilist sisestust, peab see ka nii olema. Andmed ei tohiks läbi süsteemi korduda, kohustuslikud ja vajalikud väljad peavad olema alati täidetud ning andmed peavad olema vastavuses süsteemis kehtivate ärireeglitega. Lisaks eelnevale ei tohi ära unustada, et süsteemides tuleks säilitada vaid kasutajale lisaväärtust loovaid andmeid, ehk seda informatsiooni, millest ettevõttele ka realselt kasu on. (Ponniah 2001, 294-295)

Andmeait, milles sisalduvad andmed on oma sisult ja kujult õiged, loob mitmeid häid eeliseid. Igal konkreetsel ettevõttel võivad need eelised erineda. Näiteks müügiga tegeleva ettevõtte võib tänu mitmest operatiivsüsteemist kokku toodava info põhjal olla lihtsam omada ülevaadet oma klientidest ja seeläbi pakkuda paremat klienditeenindust. Lisaks võib olla võimalik parandada turundust ja kampaaniate läbiviimist, kui saada kompaktne ülevaade klientide poolt otsetavatest kaupadest ja neist toodetest, mida klient kunagi ei osta. Seda

näiteks piirkonna ning aja lõikes. Analüüsidest möödunud perioodide müügitulemusi on võimalik teha järeldusi ning hinnata seda, mida tulevik võib tuua. Kõige tähtsam on see, et andmeaida põhjal loodud päringud ja analüüsid kajastavad reaalselt olukorda ning ei esita meile väärinfot. Ilmselgelt on see oluline, kuna ebaadekvaatsetel andmetel põhinev andmeait ei saa mingil juhul äri ning otsustusprotsessi toetada, pigem vastupidi – see mõjub halvasti. Seega võib pidada andmete ebaõigsust ja halba kvaliteeti andmeaidas üheks suurimaks riskiks. (Ponniat 2001, 295-296)

Loodava süsteemi arendamise aluseks olevate andmete kvaliteeti võib hinnata heaks. Tabelite veergudel olevad andmed on kooskõlas neile määratud andmetüüpidega. Tekstilistel väljadele on salvestatud tekstilised väärtused ning sama kehtib nii numbriliste kui kuupäev tüüpi väljade kohta. Lisaks on andmed kooskõlas veergudele seatud mahu piirangutega. Andmebaasides olevad omavahel seonduvad tabelid on seotud välisvõtmetega. Andmete kvaliteedi miinusena võib välja tuua selle, et esineb veergusid, mille väljadele on salvestatud tühjad tekstilised väärtused. Selliste veergude puhul tuleks hinnata, kas need ikka annavad süsteemile lisaväärtust.

Selleks, et kindlustada kvaliteetsete andmete laadimist andmeaita, kontrollitakse laadimise protsessis sisendandmete olemasolu ning vastavust andmetüüpidele. Juhul, kui sisendandmetes esineb vigaseid ridu, proovitakse need võimalusel parandada, näiteks andmetüüpe teisendades.

4. REALISATSIIONI TEOSTAMINE

Neljas peatükk annab ülevaate süsteemi tehnilisest teostusest. Esmajärgus kirjeldatakse tehnilist lahendust ning põhjendatakse tarkvara valikut. Seejärel kirjeldatakse täpsemalt andmeida ja –kuubi loomist, ärianalüüsi rakenduste valikut, arutletakse süsteemi optimeerimise üle ning kirjeldatakse andmete laadimise protsessi automatiseerimist. Viimaks analüüsitakse loodud lahenduse järgnevaid arendusvõimalusi ja võimalikke parendusi.

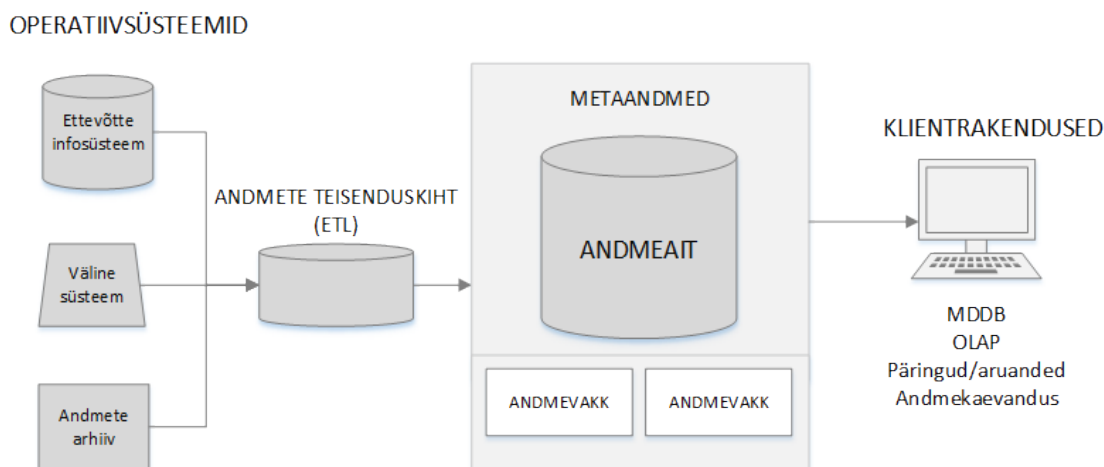
4.1 Loodava tehnilise lahenduse ülevaade

Järgnevalt kirjeldatakse süsteemi tehnilist lahendust, antakse ülevaade kogu süsteemi arhitektuurist, komponentidest, arendamiseks kasutatavast tarkvarast ning selle valikust.

4.1.1 Andmeida üldine arhitektuur

Süsteemi arhitektuuri loomise eesmärgiks on panna paika tema kõigi komponentide täpne asukoht ülejäänute suhtes. Lihtsustatult on andmeida süsteemi tüüpilisteks komponentideks operatiivsüsteemidest tulenevad andmed, andmete teisenduskiht, andmeait ning klientrakendused (vt joonis 8). (Ponniah 2001, 28)

Operatiivsüsteemide andmete all peetakse silmas relatsioonilisi rakenduse andmebaase, mille abil toimub igapäevane äriprotsess, ettevõttes sisemiselt olevaid andmebaase, andmete arhiive ning kasutusel olevaid väliseid andmete allikaid. (Ponniah 2001, 29-30)



Joonis 8. Andmeida tüüpiline arhitektuur

Allikas: (Ponniah, 2001, 133)

Andmete teisenduskihis viiakse läbi teisendusprotsess ETL (ing. k. *Extract Transform Load*), mille käigus eelkirjeldatud allikatest esmalt andmed laetakse, seejärel nad konverteeritakse ja teisendatakse sobivale kujule ning viimaks ladustatakse nad andmeaita. Andmete sobivasse formaati viimine on oluline, et andmeidast päringute ning raportite tegemisel oleksid andmed loetaval kujul ja üheselt mõistetavad. (Ponniah 2001, 31-33)

Andmeaida süsteemi keskmes on andmeait, mille puhul on tegemist operatsioonilistest andmebaasidest eraldiseisva denormaliseeritud kujul oleva andmebaasiga, kuhu koondatakse kokku kõik ETL protsessi läbinud andmed. Samuti võib andmeaida juurde kuuluda multidimensionaalne andmebaas, mille abil on võimalik andmeid dimensionaalselt esitada. (Ponniah 2001, 33)

Andmeaidaga seonduvalt käsitletakse ka metaandmeid, mille puhul on tegemist lisainformatsiooniga, mis käivad andmeaida, kui terviku kohta ning lisaks ETL protsessi läbinud andmete kohta (Ponniah 2001, 35).

Ponniah jaotab metaandmed kolme kategooriasse (2001, 36):

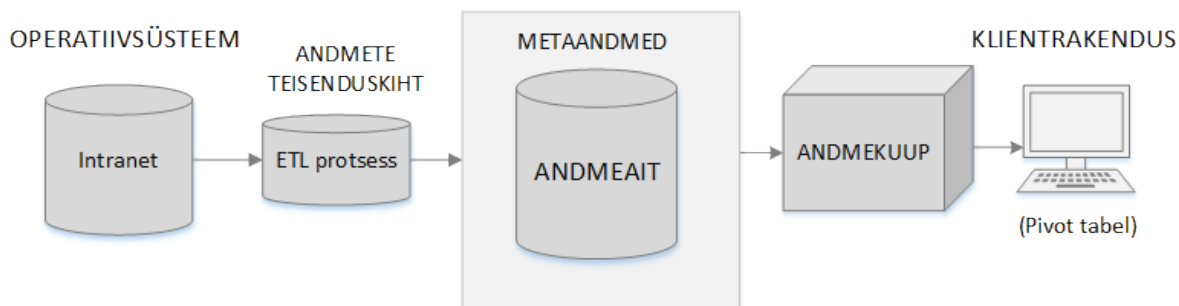
- 1) operatiivsüsteemide metaandmed – sisaldab kogu informatsiooni operatiivsüsteemide andmebaaside kohta, sealhulgas andmestruktuure ning infot tabelite väljade andmetüüpide ja pikkuste kohta;
- 2) andmete teisendusega seotud metaandmed – sisaldab kogu infot andmete teisendusprotsessi kohta;
- 3) lõppkasutaja metaandmed – lisainfo, mis täiendab andmeaita lõppkasutajale tarviliku ärilise terminoloogiaga ning hõlbustab andmeaidas navigeerimist.

Selline info on vajalik andmeaida ühtseks toimimiseks, arendajale ülevaate saamiseks ning lisaks on see oluline kasutajasõbralikkuse aspektist. (Ponniah 2001, 36)

Klientrakenduste all peetakse silmas tarkvara, vahendeid ja meetodeid, mille abil süsteemi lõppkasutajad andmeaidas olevale infole ligi pääsevad, seda analüüsivad ning päringuid ja raporteid koostavad. Klientrakenduste hulka võivad kuuluda ka multidimensionaalsed andmebaasid (MDDBs, ing. k. *multidimensional databases*), mis esitavad andmeaidas olevat informatsiooni mitmemõõtmeliselt, moodustades andmekuube (ing. k. *data cube*), nad pakuvad analüütikutele võimaluse vaadelda andmeid mitmest perspektiivist. Selliselt koostatud süsteemil on seega nelja-kihiline arhitektuur. (Ponniah 2001, 34)

4.1.2 Loodava süsteemi arhitektuur

Eelkirjeldatud üldine andmeida arhitektuur võeti loodava süsteemi arhitektuuri koostamise aluseks. Loodava süsteemi komponentideks on operatiivsüsteem Intranet; andmete teisenduskiht (ETL); andmeait; andmekuup (multidimensionaalne andmebaas) ning klientrakendus, mille abil andmeid lõppkasutajale sobivalt esitada. Loodava süsteemi arhitektuur on esitatud joonisel 9.



Joonis 9. Loodava süsteemi arhitektuur

Operatiivsüsteemi Intranet'i ülevaade on välja toodud punktis 2.1.1. Loodavate komponentide (andmete teisenduskihi, andmeida, andmekuubi ning klientrakenduse) täpsem ülevaade tuuakse välja järgnevatel peatükkides.

4.1.3 Arendamiseks valitud vahendid

Antud süsteemi arendamiseks võetakse kasutusele andmete teisenduskihi (ETL) koostamist võimaldav tarkvara, andmeida loomiseks ning hoiustamiseks vajalik tarkvara ning andmekuubi koostamist võimaldav vahend.

Andmeida hoiustamiseks võetakse kasutusele Microsoft SQL Server, seda eelkõige seetõttu, et antud süsteem on juba Eesti Rahvusringhäälingus kasutusel ning seega ei ole vajalik uute süsteemide ülesse seadmine. Andmeida arendamiseks võetakse kasutusele andmebaasi haldustarkvara Microsoft SQL Server Management Studio 2012, mis võimaldab läbi graafilise kasutajaliidese andmebaase arendada ning neist andmeid pärida.

Andmete laadimiseks operatiivsüsteemidest, teisendamiseks ning andmeaita salvestamiseks võetakse kasutusele Microsoft'i poolt välja töötatud SSIS (ing. k. *SQL Server Integration Services*) komponent, mis võimaldab läbi graafilise kasutajaliidese koostada sisse ehitatud moodulite abil nii andmete laadimise kui teisendamise protsesse, ilma programmikoodi kirjutamata. Lisaks olemas olevatele moodulitele on vajadusel võimalik mooduleid juurde

programmeerida. SSIS komponendi toimimiseks on vajalik Microsoft SQL Server, mis on loodava süsteemi omanikul olemas. Antud tarkvara valiti eelkõige seetõttu, et see kohandub andmeaida andmebaasiserveriga. Lisaks on tegemist rohkete võimalustega mugava ning kasutajasõbraliku vahendiga, mille abil andmete laadimise- ning teisendusprotsesse koostada. Komponent on hästi dokumenteeritud. Samuti on töö autoril antud tarkvara kasutamise positiivne kogemus. (SQL Server Integration Services, 2015)

Andmete teisenduskihi loomiseks kasutades SSIS komponenti on vajalik Visual Studio arendusvahend. Tegemist on samuti Microsofti poolt välja arendatud tarkvara arendajatele mõeldud töövahendiga. Selleks, et Visual Studio't ETL protsessi loomiseks kasutada, on vajalik paigaldada lisapakett, nimega SSDT (ing. k. *SQL Server Data Tools*). Antud pakett võimaldab Visual Studio't kasutada, kui andmebaasi arendusvahendit. See lisab võimaluse andmebaase täiendada, optimeerida ning refaktoreerida. Lisaks annab SSDT pakett võimaluse läbi Visual Studio koostada integratsiooni (ing. k. *Integration Services*) tüüpi projekte, mis koostöös SSIS komponendiga võimaldavad välja arendada loodava süsteemi andmete teisenduskihi. Samuti omab töö autor varasemat kogemust Visual Studio arendusvahendiga. (SQL Server Data Tools (SSDT), 2015)

Andmeaidal põhineva andmekuubi koostamiseks võetakse kasutusele SSAS (ing. k. *SQL Server Analysis Services*) komponent, mis on samuti Microsofti poolt välja töötatud komponent. SSAS toimib samuti koos Microsoft SQL Serveriga, mis on süsteemi omanikul olemas. Antud tarkvara abil on võimalik luua multidimensionaalne mudel ning see andmebaasiserverisse evitada, et lõppkasutajatel oleks võimalik informatsioonile ligi pääseda. SSAS projektide koostamine toimib sarnaselt SSIS komponendiga, mille tarbeks on vaja Visual Studio arendusvahendit ning SSDT paketti. Antud tarkvara valiti seetõttu, et ta kohandub andmeaida hoiustamiseks valitud Microsoft SQL Serveriga. Lisaks on tegemist kasutajasõbraliku graafilise töövahendiga, mis on hästi dokumenteeritud. Samuti on käesoleva magistritöö autoril on olemas varasem kokkupuude antud komponendiga. (Analysis Services, 2015)

Loodava süsteemi arendamiseks installeeritakse olemasolevasse Microsoft SQL Serverisse juurde SSIS ning SSAS komponendid, mis võimaldavad andmete teisendusprotsessi ning andmekuubi tarvis loodud projekte evitada.

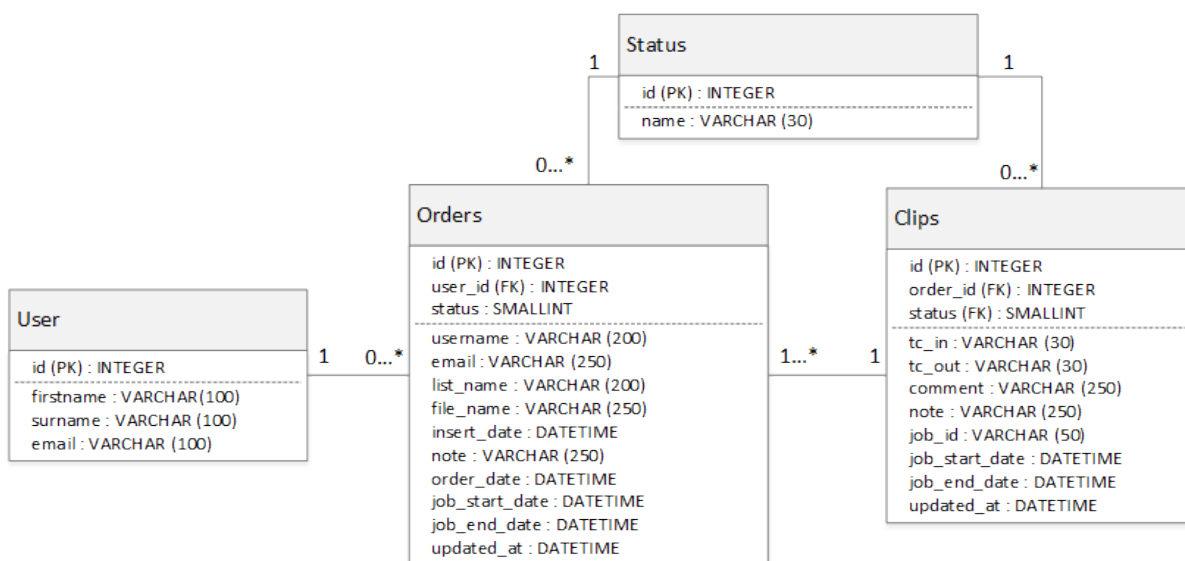
4.2 Andmeait

Andmeaida rajamise protsessi alguses luuakse andmebaasiserverisse uus andmebaas. Seejärel pannakse paika tabelite struktuur, võttes aluseks süsteemi tarvis loodud dimensionaalse andmemudeli (vt peatükk 3.3.3) ning luuakse vajalikud primaar- ja välisvõtmed. Andmebaasi ning tabelite loomise SQL laused on toodud lisa 2. Samuti pannakse paika loodud andmebaasi turvaseaded. Luuakse administraatori õigustega kasutaja, kellele antakse õigused andmeid luua, lugeda, uuendada ning kustutada. Administraatori õigustes hakkab süsteemi kasutama andmete laadimise protsess. Ülejäänud kasutajatele jäetakse õigus andmeid lugeda, et välistada andmete muudatused väljaspool ETL protsessi.

4.2.1 Andmete laadimise protsess (ETL)

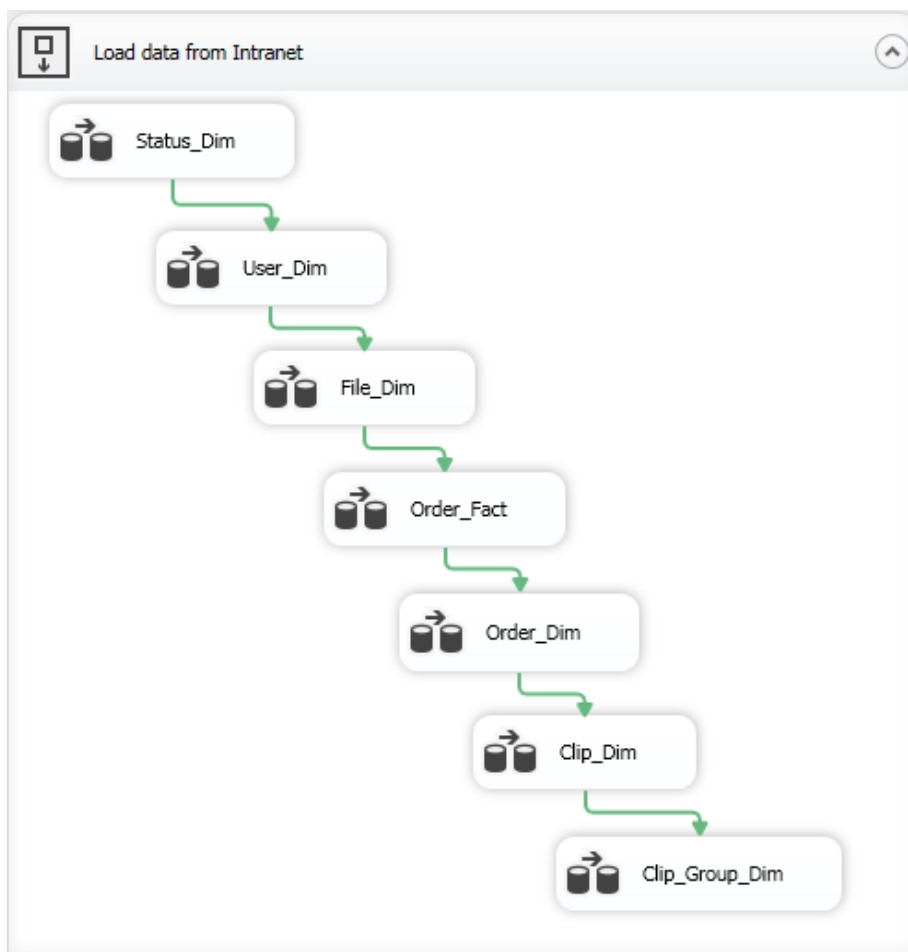
Andmebaasi tabelite struktuuri paika panemise järgselt alustatakse andmete laadimise protsessi loomisega. Antud arendust tehes, peetakse eelkõige silmas seda, et võimalikke lisaarendusi oleks hõlbus süsteemi sisse viia, kuna kasutusel olevad operatiivsüsteemid on pidevas arengus. ETL protsessi koostamiseks kasutatakse eelpool kirjeldatud arendusvahendit Visual Studio 2013, milles luuakse Integration Services tüüpi projekt. Seejärel luuakse ning salvestatakse projekti juurde andmebaasiühendused Intraneti ning andmeaida kohta.

Intranet süsteemist on nõuete täitmiseks vajalik laadida andmeid tellimuste, klippide, kasutajate ning staatuste tabelitest (vt joonis 10).



Joonis 10. Intranet süsteemist kasutatavad andmed

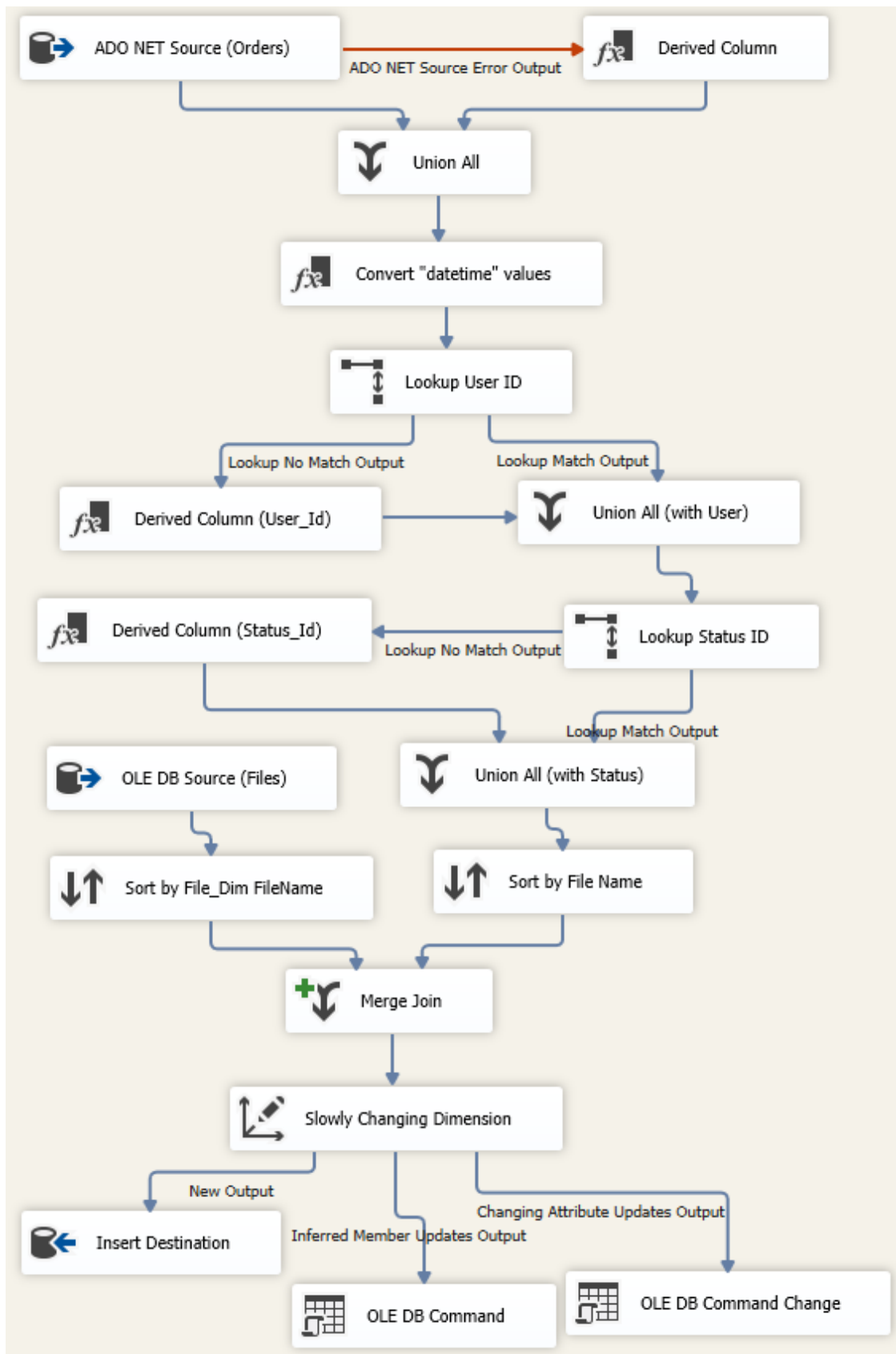
Andmete teisendamise ning laadimise protsessi tarvis luuakse projekti koosseisu pakett, mis hakkab endas koondama kõiki tarvilikke mooduleid. Antud andmeida jaoks luuakse “Load data from Intranet” pakett, milles sisalduvad nii dimensiooni- kui faktitabelitesse andmete laadimised. Paketi andmevoo vaade on toodud välja joonisel 11. Laadimist alustatakse kasutajate andmetest, seejärel laetakse failidega seonduvad andmed ning seejärel tellimustega ja klippidega seonduvad andmed. Iga andmete laadimise voo lüli laeb andmeid ühte konkreetsesse tabelisse, näiteks element User_Dim laeb andmeid kasutajate dimensioonitabelisse.



Joonis 11. ETL protsessi andmete laadimise voog

Andmete laadimise voo joonisel (joonis 11) olevad elemendid sisaldavad endas täpsemaid protsesse, mis on samuti realiseeritud olemasolevate modulite abil. Järgnevalt kirjeldatakse antud protsesside illustreerimiseks lahti tellimuste faktitabelisse andmete laadimine, ehk Order_Fact element (vt joonis 12).

Järgnevalt on esitatud tellimuste faktitabelisse andmete laadimise protsess:



Joonis 12. Tellimuste faktitabeli ETL protsess

Protsess algab elemendiga ADO NET Source (Orders), mille raames päritakse andmeid Intranet süsteemi Orders tabelist. Selleks, et vältida andmete laadimise protsessi ülekoormamist mittevajalike andmetega, küsitakse orders tabelist vaid need väljad, mida tellimuste faktitabeli tarvis vaja läheb. Nendeks väljadeks on id, file_name, user_id, insert_date ning order_date.

Juhul, kui sisendandmete laadimisel mõne rea või välja lugemine ebaõnnestub, suunatakse vigane rida elementi Derived Column, mille abil vigaseid andmeid parandada proovitakse. Näiteks antud süsteemis on võimalik, et Intranetist tuleneva tellimuse täitmise kuupäeva (order_date) veerul olevat kuupäeva ei eksisteeri (nt võib olla tegemist veaga lõppenud tellimusega), siis asendatakse antud väli vastava andmetüübi null elemendiga. Seejärel ühendatakse korrektsed andmed ning parandatud andmed üheks (element Union All).

Järgnevalt konverteeritakse tellimuse sisestamise kuupäev (insert_date) ning tellimuse täitmise kuupäev (order_date) sobivale kujule.

Selleks, et andmeaita salvestatud kasutajate ning tellimuste vahel seosed luua, kasutatakse elementi Lookup User ID. Selle raames leitakse andmeaita User_Dim tabelist kasutaja ID ja Intranet_ID. Omavahel ühendatakse tellimuse küljes olev kasutaja identifikaator user_id ning Users_Dim tabelis olev Intranet_ID, kuid tellimuse faktitabelisse (Order_Fact) salvestatakse User_Dim tabeli ID, kuna see väärtus on primaarvõti. Juhul, kui seonduvat kasutajat ei leitud, asendatakse kasutaja identifikaator väärtusega 0 (element Derived Column (User_Id)). Süsteemi mõistes tähendab see, et seonduv kasutaja puudub. Seejärel ühendatakse mõlema voo väljundid elemendiga Union All (with User).

Staatustega seoste loomiseks kasutatakse elementi Lookup Status ID. Element on üles ehitatud sarnaselt Lookup User ID elemendiga. Esmajärgus ühendatakse Status_Dim tabeli Intranet_ID väärtus ning tellimuse küljes olev staatuse ID, kuid Order_Fact tabelisse salvestatakse Status_Dim ID väärtus, kuna tegemist on antud tabeli primaarvõtme ning unikaalse väärtusega. Juhul, kui elemendi töö käigus tellimuse staatusele vastavat staatust ei leitud, määratakse, et tellimus on ebaõnnestunud (element Derivec Column (Statud Id)). Seejärel ühendatakse mõlema voo väljundid Union All (with Status) elemendiga.

Iga tehtud tellimusega on seotud süsteemis olev meediafail, millest andmeid küsitakse. Andmete laadimise voo järgmiseks sammuks on tellimuste ning nendega seonduvate failide ühendamine. Selleks seotakse Intranet süsteemi orders tabelist päritud faili nimetus

andmeaidas oleva File_Dim dimensioonitabeliga. Elemendi OLE DB Source (Files) vahendusel küsitakse andmeaida File_Dim tabelist faili nimi ning ID. Järgnevalt viiakse läbi omavahel ühendatavate andmete sorteerimised. Intranetist päritud andmed sorteeritakse faili nime põhjal (element Sort by FileName) ning samuti sorteeritakse faili nimetuse põhjal andmeaidast päritud andmed (element Sort by File_Dim FileName). Faili nimetus sobib antud andmete seostamiseks seetõttu, et tegemist on üle süsteemide unikaalse tekstilise väärtusega. Siiski ühendatakse andmeaidas Order_Fact ning File_Dim tabelid kordumatu numbrilise identifikaatoriga. Element Merge Join viib läbi andmete ühendamise.

Järgmise sammuga (Slowly Change Dimension) määratakse ära, millistele andmeaida veergudele varasemalt ette valmistatud väärtused kirjutada. Lisaks on võimalik antud elemendi puhul kindlaks määrata veerud, mille puhul andmeid ei uuendata ning samuti on võimalik määrata veerge, mille puhul andmed uuendatakse, näiteks staatused. Antud elemendiga seonduvad andmete lisamise element (Insert Destination), seonduvate andmete uuendamise element (OLE DB Command) ning andmete uuendamise element (OLE DB Command Change).

Antud protsessis olevad elemendid käivituvad üksteise järel ning seetõttu on mõistlik tähelepanu pöörata nende järjestusele. Juhul, kui Order_Fact tabelisse andmete laadimine on lõpetatud, liigutakse edasi järgneva elemendi täitmise juurde, milleks on Order_Dim. Järjestus on välja toodud joonisel 11.

Andmeaida haldamise lihtsustamiseks saadetakse süsteemist tulenevad veateated süsteemi haldajale SSIS komponendis sisalduva SMTP mooduli abil e-mailile. Selline lahendus on eelkõige hea seetõttu, et pärast süsteemi evitamist automatiseeritakse andmete laadimise protsess ning see hakkab toimima ilma hooldaja pideva kontrollita. Selliselt ei tule süsteemi hooldajal igapäevaselt andmete laadimise protsessi kontrollimas käia ning süsteemi hooldamine muutub mugavamaks.

4.3 OLAP

Järgnevalt antakse ülevaade OLAP'ist, selle erinevatest lähenemise viisidest ning nimetatud eitisuviiside vahel viiakse läbi valik antud süsteemi kontekstis.

Andmete hulk ettevõtetes järjest suureneb ning üha enam nähtakse andmeid olulise varana, mis omakorda on suurendanud huvi nii OLAP-i (ing. k. *Online Analytical Processing*) kui

andmeanalüütika vastu. OLAP kujutab endast reaalajalist analüüsi teostamist, mille põhieesmärgiks on andmeaitade või andmevakkade vahendusel saadaval olevat informatsiooni töödelda, näiteks koostada trendianalüüsi. OLAP pakub iseenesest andmetele ajaloolist vaadet ning annab palju olulist informatsiooni, kuid tulemused võivad olla veelgi huvipakkumamad, kui need kombineerida andmekaevandamise tulemusel tehtud ennustustega. (Mailvaganam, 2007)

OLAP süsteemile on võimalik kolmel erineval moel läheneda (Hammergren, Simon 2009, 137):

- 1) MOLAP (ing. k. *Multidimensional Online Analytical Processing*) puhul on tegemist analüüsi teostamisega mitmemõõtmelise andmebaasi põhjal, milleks võib olla näiteks andmekuup;
- 2) ROLAP (ing. k. *Relational Online Analytical Processing*) erineb eelnevast käsitlusest selle poolest, et tema puhul on analüüsi teostamise aluseks tüüpiline relatsiooniline andmebaas. Siiski nõuab selline käsitlus tabelite ühendamisel suhteliselt suurt jõudlust;
- 3) HOLAP (ing. k. *Hybrid Online Analytical Processing*) on hübriid kahest eelnevast kirjeldusest, tema puhul on proovitud leida kesktee, mis toimiks kõige efektiivsemal viisil. Sellise tehnoloogia pakkujad on tihtipeale tulnud välja oma implementatsiooniga, näiteks kasutada süsteemi *front-end* osana MOLAP lahendust ning *back-end* poolel ROLAP-it.

Käesoleva töö ning arendatava lahenduse kontekstis on kõige mõistlikum minna edasi MOLAP lähenemisega. Seda eelkõige seetõttu, et analüüsitavaid andmeid hoiustatakse andmeaidas, mis on disainitud pidades silmas dimensionaalset lähenemist ning sobib seega andmekuubi aluseks. ROLAP lahendus oleks mõistlik sellisel juhul, kui antud süsteemi mõistes ei oleks dimensionaalse lähenemisega loodud andmeallikat (andmeaita). Lisaks ei oleks ROLAP lahendus jõudluse mõistes kõige efektiivsem. Kolmandana käsitletud HOLAP lahenduse realiseerimine oleks mõistlik näiteks juhul, kui üks osa lähteandmetest paikneks relatsioonilises andmebaasis ning teine osa dimensionaalses andmebaasis, kuid antud süsteemi raames sellist olukorda ei ole ning seega minnakse edasi MOLAP lähenemisega.

4.4 Andmekuup

Järgnevalt kirjeldatakse täpsemalt andmekuubi olemust ning selle arenduse protsessi.

Andmekuup on andmete esituse viis, mille puhul vaadeldakse andmeid multidimensionaalsel kujul. Andmekuubis hoitakse eelkõige neid andmeid, mida analüüsid on võimalik ettevõttele lisaväärtust luua. Andmekuup koosneb dimensioonidest ning mõõdikutest ehk faktidest. Dimensioonidesse koondatud andmed võimaldavad kuupi “tükeldada” ning mõõdikud hoiavad endas numbrilisi väärtuseid, milleks võib olla näiteks summa. Andmekuubi “rakk” ehk üks tükeldatud osake kirjeldab dimensiooni atribuudi ning vastava määratletud mõõdiku suhte (vt joonis 13). (Acharya 2013, 5)

AEG	ASUKOHT				TOOTED			
	Chicago	New York	Toronto	Vancouver	kodutehnika	arvutid	telefonid	turvaseadmed
1. kvartal	605	825	14	400	682	925	789	698
2. kvartal	680	952	31	512	728	1002	870	
3. kvartal	812	1023	30	501	784	984	870	
4. kvartal	927	1038	38	580				

Joonis 13. Kolmemõõtmeline andmekuup

Allikas: (Han, Kamber 2006, 112)

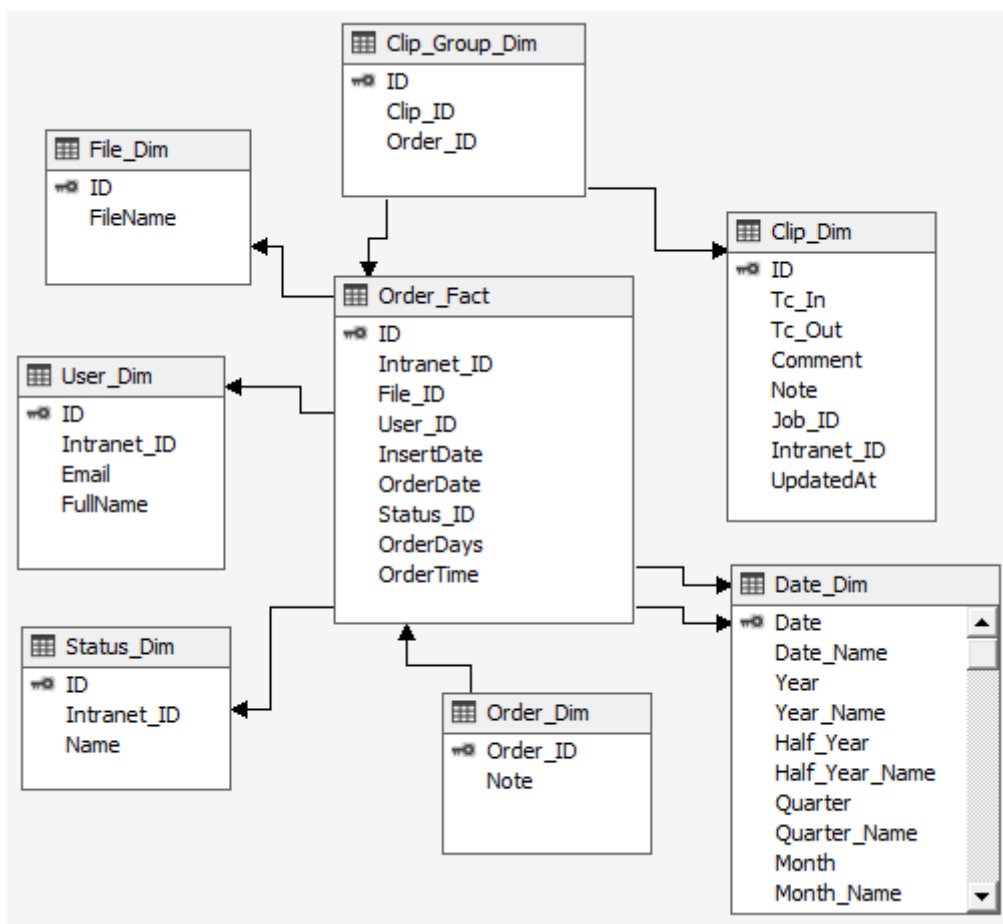
Joonisel 13 on välja toodud kolmemõõtmeline andmekuup. Kolm mõõdet moodustavad järgnevad dimensioonid: asukoht (linn), aeg (kvartal) ning toode (toote tüüp). Mõõdetavaks väärtuseks on antud kontekstis müügitulu dollarites (tuhandetes). (Han, Kamber 2006, 111)

Seega võimaldab antud andmete esituse puhul saada infot selle kohta, kuidas ettevõtte müügitulu jaotub asukohtade, toodete ning aja suhtes. Näiteks võib ühe andmekuubi “rakukese” puhul välja lugeda, et neljandas kvartalis müüdi Vancouveris arvuteid 1038 tuhande dollari eest. Lisaks on antud näite puhul võimalik analüüsida toodete müügi edenemist piirkondade suhtes. Näiteks esimeses kvartalis müüdi Vancouveris telefone oluliselt vähem (14 tuhande dollari eest), kui Chicagos (89 tuhat dollarit).

4.4.1 Andmekuubi arendusprotsess

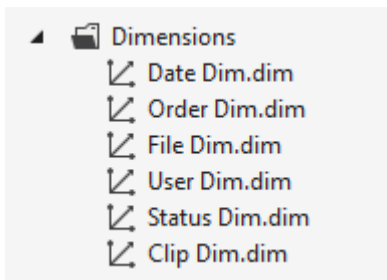
Käesoleva magistritöö raames arendatakse välja andmekuup, mis põhineb loodud andmeidal. Andmekuubi dimensioonideks võetakse andmeidas olevad dimensioonitabelid ning mõõdikuteks faktitabelis olevad väärtused. Lisaks täiendatakse mõõdikuid arvutuslike väärtustega, näiteks failide arvuga.

Andmekuubi arendamise protsess algab vastava projekti loomisega arendusvahendis Visual Studio 2013. Loodava projekti tüübiks on Analysis Services Multidimensional and Data Mining Project. Kõige esimese sammuna luuakse ning salvestatakse ühendus andmeallikasse ehk andmeaita, millest tulenevaid andmeid hakatakse edaspidi andmekuubi abil pärima. Järgmise sammuna pannakse kokku andmeidast kasutatavate tabelite vaade. Antud projekti puhul lisatakse sinna kõiki andmeidas olevad tabelid. Andmevaade on toodud välja joonisel 14.



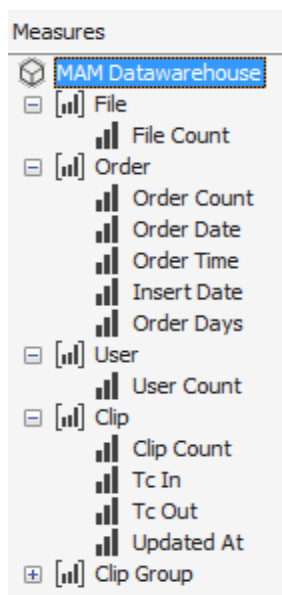
Joonis 14. Andmekuubi lähteandmete vaade

Seejärel liigutakse edasi andmekuubi dimensioonide defineerimise juurde, mille aluseks võetakse andmemudeli dimensioonitabelid. Seega moodustavad andmekuubi dimensioonid andmeidas sisalduvad dimensioonitabelid. Kuupäeva dimensiooni (Date Dim.dim) loomisel kasutatakse arendusvahendi võimalust genereerida automaatselt kuupäeva informatsiooni sisaldav tabel, mis lisatakse andmeallikasse ehk andmeaita. Kuupäeva dimensioon võimaldab teostada detailseid päringuid kuupäevade lõikes. Luuakse kuus dimensiooni (vt joonis 15).



Joonis 15. Andmekuubi dimensioonid

Edasi liigutakse andmekuubi loomise juurde. Selleks luuakse .cube tüüpi fail ning määratakse süsteemi mõõdikud. Mõõdikutena defineeritakse faktitabeli Order_Fact väärtused ning arvutatavad väärtused dimensioonitabelitest, milleks on failide arv (File Count), tellimuste arv (Order Count), kasutajate arv (User Count) ja klippide arv (Clip Count). Lisaks määratakse mõõdikutena klippidega seonduvad ajakoodid (Tc In, Tc Out) ning uuendamise aja (Updated At) väärtused. Andmekuubis defineeritud mõõdikute grupid (ing. k. *measure group*), näiteks File ning atribuudid, näiteks File Count on esitatud joonisel 16. Luuakse 5 mõõdikute gruppi.



Joonis 16. Andmekuubi mõõdikud

Viimase sammuna pannakse paika süsteemi mõõdikute gruppide seosed loodud dimensioonidega. Antud etapp on andmekuubi arendamise üks tähtsamaid osasid, kuna valede seoste olemasolul annab süsteem ebatäpseid vastuseid. Mõõdikute gruppide ning dimensioonide vahel võib eksisteerida mitut tüüpi seoseid. Antud andmekuubi puhul on neist kasutatud nelja tüüpi (Dimension Relationships, 2015):

- 1) fact ehk seos faktiga – sellist tüüpi seos defineeritakse dimensiooni ning mõõdikute grupi vahel, kui antud väärtused asuvad ühes tabelis;
- 2) regular ehk tavaline seos – sellist tüüpi seose võib luua juhul, kui dimensioonitabeli ja faktitabeli vahel on otsene primaarvõti – välisvõti tüüpi seos;
- 3) referenced ehk viidatud seos – antud tüüpi puhul luuakse mõõdikute grupi ja dimensiooni vahele seos läbi vahe-dimensiooni, mis omab viiteid nii antud dimensioonile kui mõõdikute tabelile;
- 4) many-to-many ehk mitu-mitu seos – antud tüüpi seos defineeritakse juhul, kui mõõdiku ning dimensiooni vahel esineb mitu-mitu seos. Sellise seose defineerimiseks on vajalik vahetabel, mis andmete vahelisi seoseid hoiab.

Järgnevalt esitatud maatriks illustreerib loodud seoseid mõõdikute gruppide ning dimensioonide vahel. Maatriksi veergude päistes on välja toodud defineeritud mõõdikute grupid ning iga rida määratleb ühe dimensiooni (vt joonis 17).

Dimensions	[nl] File	[nl] Order	[nl] User	[nl] Clip	[nl] Clip Group
[nl] Date Dim (Order D...	Date				
[nl] Date Dim (Insert D...	Date				
[nl] User Dim	ID	[nl] ID			
[nl] File Dim	[nl] ID	ID			[nl] Order
[nl] Clip Dim				[nl] ID	ID
[nl] Order Dim		Order Number		[nl] Clip Group	Order Number
[nl] Status Dim		ID			
[nl] Clip Group Dim				[nl] Clip Group	[nl] ID
[nl] Clip Group Dim (Or...					[nl] ID

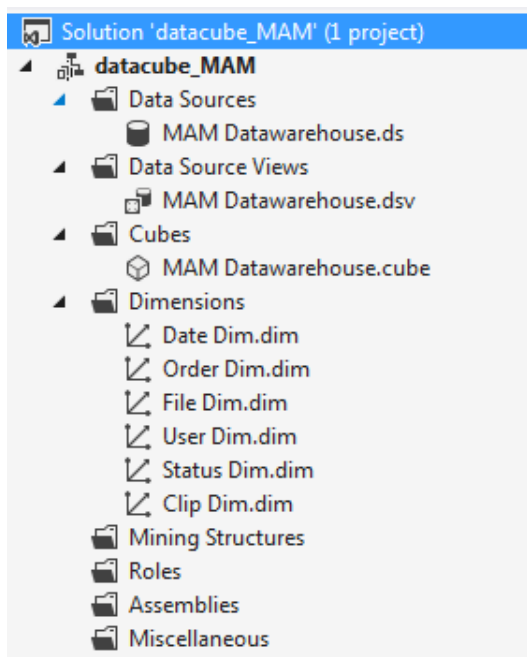
Joonis 17. Andmekuubi dimensioonide ja mõõdikute vahelised seosed

Jooniselt 17 on näha, et mõõdikute grupp tellimus (Order) on süsteemi keskne objekt, kuna tegemist on kõigi dimensioonidega seotud faktiga. Mõõdikute grupi Order puhul on kasutatud tavalist (regular) tüüpi ning mitu-mitu tüüpi seoseid. Seosed kasutajate (User Dim), tellimuste

(Order Dim), failide (File Dim), kuupäeva (Date Dim) ning staatuste dimensiooniga (Status Dim) on tavalist tüüpi, kuna antud faktitabel omab otseseid välisvõtme viiteid neisse dimensioonitabelitesse. Seos videolõikude dimensiooniga (Clip Dim) on mitu-mitu tüüpi ning selleks kasutatakse vahetabelit Clip Group Dim.

Lisaks on arvutatavate mõõdikute leidmiseks loodud seosed vastavate dimensioonidega, näiteks kasutajate arvu leidmiseks on loodud fakt tüüpi seos mõõdikute grupi User ning dimensiooni User Dim vahel läbi kasutaja identifikaatori User ID. Sarnast loogikat kasutatakse ka failide (File), tellimuste ja videolõikude (Clip) puhul.

Loodud projekti üldine struktuur on toodud välja joonisel 18. Andmeaida andmebaasiühendus on salvestatud failis MAM Datawarehouse.ds (kataloog Data Sources) ning andmeaida mudelit on esitatud failis MAM Datawarehouse.dsv (kataloog Data Source View) abil. Kõik dimensioonid on välja toodud kaustas Dimensions. Andmekuup on salvestatud failina MAM Datawarehouse.cube (kataloogis Cubes).



Joonis 18. Andmekuubi projekti struktuur

4.5 Ärianalüüsi rakendus

Lähtuvalt antud süsteemi realiseerimiseks valitud metoodikast, on üheks andmeaida arenduse protsessi osaks ärianalüüsi rakenduse disain ning realisatsioon. Selle tegevuse raames

analüüsitakse, milline saab olema päringute loomiseks kasutatav tarkvara. (Kimball *et al.* 2008, 7)

Antud süsteemi tarvis arendati andmekuup, mille abil on võimalik andmeid dimensionaalselt esitada ning seetõttu keskendutakse ärianalüüsi rakenduse puhul eelkõige andmekuubist informatsiooni pärimisele. Järgnevalt vaadeldakse, millised on võimalused andmekuubi poole pöördumiseks ning sealt informatsiooni küsimiseks.

Üheks võimalikuks viisiks on kasutada tabelarvutusprogrammi Microsoft Excel lisamoodulit Microsoft Power Pivot. Tegemist on mitmete võimalustega andmete analüüsimiseks loodud lisamooduliga, mille abil saab lõppkasutaja ärianalüüsi teostada. Süsteemi loojad peavad antud tarkvara põhiliseks eeliseks suurte hulkade andmete töötamise kiirust. Komponent võimaldab mitmest erinevast allikast andmeid importida ning neid seejärel sobivalt töödelda. Importimise allikaks sobivad relatsioonilised ja multidimensionaalsed andmebaasid, pilveteenused, Exceli failid, tekstifailid ning veebipõhised andmed. Lisaks on võimalus valemite abil andmete põhjal arvutusi teha ja seeläbi uusi mõõdikuid välja töötada. Exceli olemasolul on antud komponendi kasutamine tasuta. (Power Pivot Add-In, 2015)

Lisaks on võimalik andmekuubist infot pärida ka veebipõhiste rakenduste kaudu. Üheks selliseks komponendiks on Kyubiti poolt arendatav AnalysisPortal. Tegemist on OLAP andmekuupide tarvis loodud klientrakendusega, mis põhineb veebiliidesel. Komponent võimaldab lõppkasutajatel mugavalt veebilehitseja kaudu kuubi külge ühenduda, päringuid koostada, andmeid analüüsida, visualiseerida ning lõpptulemusi jagada. Antud tarkvara peamiseks eeliseks on see, et tegemist on lihtsa ning mugava veebirakendusega. Tootja toob rakenduse peamiste positiivsete külgedena välja lihtsad OLAP operatsioonid, hulgaliselt võimalusi andmete visualiseerimiseks ning hea võimalus tulemusi teiste kasutajatega jagada. Lisaks tuuakse välja, et rakenduse seadistamine ettevõtte sisevõrku on lihtne. Antud komponendi maksumus on sõltuvalt kasutajate ning andmekuupide hulgast vahemikus 495 – 1895 USA dollarit. (SSAS Analysis Services Client – AnalysisPortal, 2015)

Lisaks eelnevatele lähenemistele, mille puhul päritakse andmeid rakenduste abil, on võimalik kuubi poole pöördumiseks kasutada päringukeelt. Relatsiooniliste andmebaaside puhul kasutatakse päringute kirjutamiseks SQL keelt kuid andmekuubist andmete pärimiseks tuleb kasutada MDX (Multidimensional expressions) päringukeelt. Tegemist on OLAP andmekuupidest info küsimiseks loodud keelega, mille on välja töötanud Microsoft. MDX

põhineb XMLA (XML for Analysis) spetsifikatsioonil. XMLA puhul on tegemist standardiseeritud SOAP'il põhineva universaalse XML protokolliga, mille abil on võimalik saata päringuid ning saada vastuseid andmekuubist üle HTTP ühenduse. (Multidimensional Model Data Access..., 2015)

Peamiselt on MDX mõeldud andmete küsimiseks, kuid võimaldab ka andmeid lisada ja muuta. MDX päringuid on võimalik koostada näiteks SSDT abil või kasutades andmebaasi haldustarkvara SQL Server Management Studiot. MDXi kasutatakse SSAS puhul peamiselt klientrakendusele andmete tagastamiseks, päringu tulemuste vormistamiseks, andmekuubi võimaluste realiseerimiseks, nt KPI ning arvutuslike väärtuste kirjeldamiseks ning administratiivtoiminguteks, nt dimensioonide turvalisuse tagamiseks. Selleks, et MDXi kasutada on mõistlik omada teadmisi dimensionaalsest modelleerimisest ning antud päringukeele süntaksist, operaatoritest, käskudest ning funktsioonidest. Kokkuvõtvalt on näiteks võimalik antud päringukeelt kasutades andmekuubist küsitav info mõnda ettevõttes kasutatavasse veebirakendusse välja kuvada. (Querying Multidimensional Data..., 2015)

Eeltoodud võimalustest on näha, et andmekuubist andmete pärimiseks on mitmeid erinevaid viise ning ilmselt ei ole sobiva lahenduse leidmine keerukas. Antud töö kontekstis võetakse kasutusele päringute loomine Microsoft Exceli vahendusel. Seda põhjusel, et antud programm on Eesti Rahvusringhäälingus juba tarkvarana kasutusel ja ei nõua seetõttu lisakulutusi uutele programmidele ega suurematele lisaarendustele. Samas peetakse seda vahendit piisavalt heaks, et täita süsteemile esitatud nõudeid. Veebirakenduse põhine süsteem on küll ilmselt mugavam, kuid jäeti valikust välja oma kõrge hinna tõttu. Lisaks on mõeldav pikemas perspektiivis andmekuubi põhjal tehtavate päringute tulemuste integreerimine MAMi süsteemi, kuid pidades silmas antud lahenduse välja töötamise mahtu ei võeta sellise lahenduse loomist antud töö skoopi.

4.6 Süsteemi testimine

Järgnevalt viiakse läbi loodud süsteemi testimine. Selle käigus kontrollitakse süsteemi vastavust seatud funktsionaalsetele nõuetele ning mittefunktsionaalsetele nõuetele, mis on kirjeldatud nõuete analüüsi peatükis 3.2. Süsteemi funktsionaalsusi testitakse kasutades eelneva peatüki analüüsi tulemusena välja valitud klientrakendust Microsoft Excel. Selleks luuakse nimetatud tabelarvutusprogrammis ühendus andmekuubi külge ning päringute koostamiseks kasutatakse PivotTabeli lisamoodulit.

Järgnevalt esitatakse kõik süsteemile esitatud funktsionaalsed nõuded, nende testimise kokkuvõtted koos kommentaaridega ning see, kas nõue on süsteemis korrektselt realiseeritud või mitte.

F01

Nõude kirjeldus: leida, millal algas ning millal lõppes tellimuse täitmine (sh leida kestvus).

Testi kokkuvõte: süsteemis on võimalik koostada päring, mis näitab kuupäevade lõikes, millal konkreetset tellimust (Order Number) alustati (Insert Date), lõpetati (Order Date) ning kui kaua tellimuse täitmine aega võttis (päevades (Order Days), minutites (Order Time)) (vt joonis 19).

Order Number	Insert Date	Order Date	Order Days	Order Time
45836	13.03.2013	13.03.2013	0	01:42:41
45837	28.10.2014	28.10.2014	0	00:45:17
45838	29.04.2013	29.04.2013	0	03:39:22
45839	29.10.2012	29.10.2012	0	00:24:35
45840	28.09.2010	30.09.2010	2	08:15:38
45841	13.01.2012	13.01.2012	0	00:34:24
45842	13.01.2012	29.01.2012	0	00:24:54
45843	13.01.2012	29.01.2012	0	00:20:03

Joonis 19. Nõude F01 väljundandmete näide

Läbis/Ei läbinud: Läbis.

F02

Nõude kirjeldus: leida, milline süsteemi kasutaja oli tellimuse esitajaks (sh leida kasutaja poolt esitatud tellimuste kogus).

Testi kokkuvõte: süsteemis on võimalik koostada päring, mis leiab tellimuse esitanud kasutaja. Samuti on võimalik leida kasutaja poolt esitatud tellimuste kogus (vt joonis 20) ning seeläbi leida süsteemi aktiivseimad kasutajad.

Order Count							
User Full Name	2010	2011	2012	2013	2014	2015	Grand Total
Margit Ossipova	39	367	850	962	928	329	3475
Rahel Selge	76	251	392	622	693	168	2202
Jaak Tammearu		192	342	312	507	100	1453
Ando-Siim Kuldkepp		33	50	223	794	201	1301
Tiina Annus	118	273	353	384	39	14	1181
Maarika Lauri		21	109	328	445	107	1010

Joonis 20. Nõude F02 väljundandmete näide

Läbis/Ei läbinud: Läbis.

F03

Nõude kirjeldus: leida, kui palju on tehtud tellimusi ühest failist.

Testi kokkuvõte: süsteemist on võimalik pärida infot selle kohta, kui palju on tehtud tellimusi konkreetsest failist ning millal need tehtud on. Antud testimisel kasutatud andmete põhjal võib öelda, et kõige enam on tellitud videofaili nimega 2007-019723-0001_0002_D10_KEVADE, kokku läbi aastate 77 korda. Päringu väljundandmete näidis on toodud välja joonisel 21.

Order Count								
File Name	2010	2011	2012	2013	2014	2015	Grand Total	
2007-019723-0001_0002_D10_KEVADE		23	42	4	7	1	77	
2002-011820-0001_0004_D10_VIIMNE-RELIKVIA		21	28	8	4	2	63	
2011-000004-0190_0001_D10_EESTI-LAUL-2011_FINAAL		13	9	13	6	5	46	
2009-002669-0021_0001_D10_EESTIMAA-KUULSAD-INIMES_TAMMSAARE	2	30	7	2	1		42	
2002-011899-0001_0003_D10_NIPERNAADI		13	20	7	1		41	
2010-000004-0172_0006_D10_EESTI-LAUL-2010	1	7	10	14	8		40	

Joonis 21. Nõude F03 väljundandmete näidis

Läbis/Ei läbinud: Läbis.

F04

Nõude kirjeldus: leida, milliseid lõike on tellitud antud failist.

Testi kokkuvõte: süsteemis on võimalik koostada päring, et leida, milliseid videoklippe failist on tellitud. Sealjuures näha alguse- ja lõpu ajakoode (vastavalt Tc In ja Tc Out). Väljundandmete näidis on toodud joonisel 22.

File Name -> Clip Number	Tc In	Tc Out	Clip Count
2014-000934-0528_0001_D10_PEALTNAGIJA	00:23:50	00:25:03	5
2014-000934-0532_0001_D10_PEALTNAGIJA	00:01:19	00:18:35	3
123433	00:19:21	00:36:28	1
125688	00:19:20	00:35:32	1
135565	00:01:19	00:18:35	1
Grand Total	00:01:19	00:18:35	8

Joonis 22. Nõude F04 väljundandmete näidis

Läbis/Ei läbinud: Läbis.

F05

Nõude kirjeldus: leida, tellitavate failide kogus määratud ajahetkel.

Testi kokkuvõte: süsteemist on võimalik leida tellitud failide arv määratud ajahetkel. Kasutades aja dimensiooni on võimalik vaadata tellimuste arvu aastate, kuude ning päevade lõikes, alates süsteemi algusest 2010. aastal. Päringu väljundandmete näidis on toodud joonisel 23.

Order Count							
Day	2010	2011	2012	2013	2014	2015	Grand Total
Monday	89	557	1012	1113	1370	315	4456
Tuesday	151	626	865	1161	1355	374	4532
Wednesday	137	738	830	952	1162	417	4236
Thursday	38	633	766	736	926	420	3519
Friday	69	588	614	918	1148	315	3652
Saturday	2	49	151	172	288	66	728
Sunday		96	267	337	522	79	1301
Grand Total	486	3287	4505	5389	6771	1986	22424

Joonis 23. Nõude F05 väljundandmete näidis

Läbis/Ei läbinud: Läbis.

F06

Nõude kirjeldus: leida tellimuste arv tellimuste staatuste lõikes.

Testi kokkuvõte: süsteemis on võimalik koostada päring, mis näitab tellimuste arvu antud staatuses. Samuti on võimalik lisada tellimuse esitamise aeg ning seeläbi andmeid ajaliselt vaadelda. Väljundandmete näidis on toodud joonisel 24.

Order Count							
Status Name	2010	2011	2012	2013	2014	2015	Grand Total
Created	42	177	110	102	113	27	571
Completed	432	3091	4374	5280	6653	1952	21782
Canceled by administrator	12	19	21	7	5	4	68
Failed						3	3
Grand Total	486	3287	4505	5389	6771	1986	22424

Joonis 24. Nõude F06 väljundandmete näidis

Väljundandmetest on näha, et alates 2012 aastast on loodud (Created) ehk lõpetamata tellimuste arv jäänud stabiilseks. 2015 aastast on andmed esimesest kolmest kuust.

Läbis/Ei läbinud: Läbis.

Süsteemile esitatud mittefunktsionaalsed nõuded ning nende täidetavus süsteemis on esitatud tabelis 10.

Tabel 10. Mittefunktsionaalsete nõuete ülevaatus kokkuvõte

ID	Nõude kirjeldus	Märkus
MF01	Süsteemist tuleb teha regulaarselt varukoopiaid (kord päevas), et võimalike tõrgete puhul oleks võimalik süsteem taastada.	Süsteemi evitamisel seatakse sisse varukoopiate tegemine, kord päevas.
MF02	Andmete laadimise protsess (ETL) peab töötama öösiti, et vältida süsteemile lisakoormuse tekitamist.	Süsteemi uuendamiseks lisatakse andmebaasiserverisse skript, mis töötab igal öösel kell 5:00 ning käivitab andmete laadimise protsessi ja uuendab andmekuubi.
MF03	Tuleb arvestada lokaliseerimist. Kasutusel peab olema ajaformaad kujul "pp.kk.aaaa tt:mm:ss".	Päringute tegemisel kasutatavad kuupäevad ja ajad on kooskõlas vastava formaadiga.
MF04	Loodav süsteem peab olema inglise keeles.	Süsteem on inglise keeles.
MF05	Päringute vastuste koostamisel kasutajatele kuvatavad muutujate nimed peavad olema sõbralikud elik kasutajatele mõistetavad.	Kõik päringute tegemisel kasutatavad nimed on mõistetavad. Puuduvad nimed stiilis „x“ ja „y“.
MF06	Kasutaja päringule tuleb vastus edastada 15 sekundi jooksul.	Funktsionaalsete nõuete (F01-F06) raames tehtavad päringud annavad vastuseid 15 sekundi jooksul.
MF07	Süsteem tuleb rajada selliselt, et hilisem muudatuste sisseviimine oleks mõistlikul viisil teostatav.	Süsteemi arendades peeti silmas, et lisaarenduste tegemine oleks hõlbus.
MF08	Realisatsiooni raames tuleb koostada süsteemi täienduste tegemiseks ning süsteemi evitamiseks juhend.	Juhendid uuenduste sisseviimiseks ning evitamiseks koostatakse antud töö raames ja lisatakse lisadesse (vt pt 4.7).
MF09	Loodav süsteem peab toimima andmebaasiserveris, mis sisaldab ka teisi andmebaase.	Süsteemi arendus viidi läbi andmebaasiserveris, mis sisaldas ka teisi andmebaase. Süsteem toomis veatult.
MF10	Süsteemi peab olema võimalik lisaarendusi tegemata ülesse seada uues keskkonnas.	Süsteemi prooviti ülesse seada uues keskkonnas ning see toimis veatult.

4.7 Andmeaida ja -kuubi evitamine

Antud projekti arendamiseks valitud metoodika (Kimballi elutsükli) järgmiseks sammuks on süsteemi evitamine. Süsteemi evitamise samm võtab kokku kõik eelpool tehtud tegevused, mille raames andmeait valmis. Evitamise lõpptulemusena antakse süsteem üle lõppkasutajatele. Siiski tuleb süsteemi arendajatel enne lõppkasutajatele üle andmist veenduda, et süsteemile esitatud nõuded on täidetud, süsteemis sisalduvad andmed on kvaliteetsed ning loodavad päringud annavad õigeid vastuseid. (Kimball *et al.* 2008, 7-8)

Lähtuvalt süsteemile esitatud mittefunktsionaalsest nõudest nr 8 (MF08), koostatakse süsteemi uuenduse sisseviimise ning süsteemi evitamise juhend. Antud juhendi koostamise eeliseks on peamiselt see, et teadmine, kuidas uuendusi sisse viia ning süsteemi evitada saab dokumenteeritud ning arendajate vahetudes on võimalik teadmine lihtsalt edasi anda. Andmeaida ning -kuubi uuendamise ning evitamise juhendid on esitatud lisades (vt lisa 3 ja 4).

Tulenevalt süsteemi testimise tulemustest on töö autor veendunud, et süsteem on valmis lõppkasutajatele üle andmiseks. Selle tegevuse raames paigaldatakse süsteem lõppkasutajatele ligipääsetavasse serverisse, mis asub ERR-is. Evitamise raames tehakse järgnevad tegevused:

- 1) luuakse andmebaasiserverisse andmeaida tarvis uus andmebaas ning sellega seonduvad objektid, kasutades antud töö lisades olevaid SQL lauseid (vt lisa 2);
- 2) kasutades arendusvahendit Visual Studio 2013 valmistatakse ette andmeaida projekti pakett ning kopeeritakse saadud failid sihtserverisse (tegevuse täpsem kirjeldus on esitatud juhendis, lisa 3);
- 3) sihtserverisse kopeeritud failide hulgast avatakse .SSISDeploymentManifest laiendiga fail ning avanenud abimehe (ing. k. *wizard*) abil paigaldatakse andmeaida pakett sihtserverisse (tegevuse täpsem kirjeldus on esitatud juhendis, lisa 3);
- 4) järgnevalt avatakse arendusvahendis andmekuubi projekt ning see käivitatakse (ing. k. *build*) ja evitatakse (ing. k. *deploy*), seejärel avatakse kasutades käsuriida programm Microsoft.AnalysisServices.Deployment.exe, mille abil valmistatakse ette .XMLA laiendiga fail (täpsem kirjeldus on esitatud juhendis, lisa 4);

- 5) saadud .XMLA fail kopeeritakse sihtserverisse, fail avatakse SSAS andmebaasis kasutades Microsoft SQL Management Studiot ning käivitatakse (täpsem info on esitatud juhendis, lisa 4).

Lisaks defineeritakse andmebaasiserverisse automaatne andmete laadimise ning andmekuubi uuendamise käsklus (SQL Job), kasutades andmebaasi arendusvahendit Microsoft SQL Management Studio. Käsklus hakkab lähtuvalt mittefunktsionaalsest nõudest nr 2 (MF02) igal öösel kell 5.00 andmete laadimise protsessi käivitama ning andmekuupi uuendama.

4.8 Edasised arendusvõimalused

Käesoleva töö raames realiseeritud andmeait ja –kuup said loodud selliselt, et edasiste täienduste sisseviimine oleks hõlbus ning täienduste lisandumisel ei tuleks olemasolevat süsteemi ümber teha. Selliselt on uuenduste lisamine lihtsustatud.

Lähtuvalt sellest, et antud töö raames realiseeritud süsteemi skooopi võeti Intranetist tellimuse andmete andmeaita ladustamine võib ühe edasise lisaarendusena mõelda ka MAMi süsteemist tellimuste andmete üle toomisele antud andmeaita. Täiendavalt tasub mõelda ka tellitud faili saate tüüpide ning tellimuse teinud kasutajagruppide lisamisele antud süsteemi, mis omakorda annaksid kindlasti süsteemile lisaväärtust. Lisaks tasub tähelepanu pöörata süsteemi reaktsiooniaegadele. Samuti on üheks võimalikuks täienduseks andmekuubis tehtavate päringute tulemuste integreerimine MAMi süsteemi, mis muudaks lõppkasutajale süsteemi kasutamist mugavamaks.

Lisaks tuleks loodud süsteemi täiendada, kui lähteandmebaasides muudatusi või lisaarendusi tehakse, et kindlustada uute andmete edasikandumine andmeaita. Lisaks tuleb kindlustada, et muudatuste tegemisel andmeaita andmete laadimine jätkuvalt toimiks. Täienduste sisseviimisel tuleks kindlasti silmas pidada seda, et juba andmeaita laetud andmed kahjustada ei saaks.

5. KOKKUVÕTE

Käesoleva magistritöö eesmärgiks oli esmajärgus analüüsida ning seejärel valmis arendada toimiv ning reaalselt kasu toov andmeait ja -kuup Eesti Rahvusringhäälingu meediahaldussüsteemis tehtavate tellimuste andmete analüüsimiseks. Antud organisatsiooni infosüsteemide koosseisus ei olnud varasemalt nimetatud andmete analüüsimiseks tarvilikke vahendeid, kuid huvi ja vajadus andmete uurimiseks oli olemas.

Antud töö põhitulemusi on kolm:

- 1) loodud süsteemi analüüs, mille raames analüüsiti süsteemile esitatud nõudeid ning koostati multidimensionaalne andmemudel, mis võeti edasiste süsteemide arendamise aluseks;
- 2) valmis arendatud andmeait, milles hoiustatakse arhiivmaterjali tellimusega seonduvaid andmeid;
- 3) valmis arendatud andmekuup, mille abil on võimalik andmeaidas sisalduvate andmete põhjal päringuid koostada ning neid andmeid analüüsida.

Loodud andmeait ning andmekuup moodustavad tervikliku süsteemi. Töö tegemise käigus võeti eelkõige arvesse organisatsiooni huve ning infovajadusi. Töö tulemusena valminud süsteem võimaldab süsteemi omanikul meedia arhiivitellimustega seonduvaid andmeid analüüsida ning seeläbi oma tööd efektiivsemaks muuta.

Seatud eesmärkide saavutamiseks viidi läbi metoodika valik, mille abil projekt läbi viidi. Seejärel uuriti olemasoleva meediahaldussüsteemi olemust, vajalikkust organisatsioonis ning arhiivmaterjali tellimuste protsessi. Samuti uuriti lõppkasutajate infovajadusi, nõudeid loodavale süsteemile ning seda, kas andmeaidade rajamine on võimalik. Järgnevalt määratleti loodava süsteemi skoop, analüüsiti loodavale süsteemile seatud nõudeid, hinnati olemasoleva süsteemi andmete kvaliteeti ning koostati multidimensionaalne andmemudel. Lähtuvalt loodud andmemudelist viidi läbi andmeaidade ning andmekuubi arendustööd. Viimaks teostati loodud süsteemi testimine, töötati välja süsteemi evitamise protsess ning pakuti välja viise, millisel moel on võimalik loodud süsteemi edasi arendada

Käesoleva töö raames loodud süsteemi arendusel peeti silmas, et edasiste täienduste sisseviimine oleks võimalikult lihtne ja selleks ei tuleks süsteemi teisi osi ringi teha. Sellise

lähenemise puhul on süsteemi võimalik hõlpsalt edasi arendada, näiteks lisada tellitud failide saadete tüübid, tellija kasutajagruppi kuulumine ning importida andmeaita ka MAM süsteemist tehtavate tellimuste andmed.

Tehtud magistritööst võib järeldada, et valitud meetoodika, loodud analüüsi (sh multidimensionaalse andmemudeli) ning kasutatud arendusvahendite abil on võimalik valmis arendada andmeait ja –kuup, mille abil on võimalik mitmekülgseid päringuid koostada ning seeläbi andmetele lisaväärtust anda.

Kokkuvõtteks võib öelda, et loodud süsteemi abil on võimalik meediahaldussüsteemi tellimustega seonduvaid andmeid analüüsida, sealjuures nende väärtust organisatsioonile tõsta ning antud magistritöö sissejuhatavas peatükis seatud eesmärgid said töö raames täidetud.

SUMMARY

The purpose of the thesis was to analyse and develop a data warehouse and a data cube for media asset management system, which is used by Estonian Public Broadcasting. The aim of the data warehouse and data cube is to enable end-user to create various queries and analyse the data which is related to media archive orders. There was an interest and need for analysing that kind of information and Estonian Public Broadcasting haven't had a similar system for analysing the data before and that made the result appealing.

The thesis has three main outcomes:

- 1) an analysis of the created system, which contains the requirements analysis and a multidimensional data model, that was used in system development;
- 2) a developed data warehouse, which is used for storing the data related to media archive orders;
- 3) a developed data cube, which is used for accessing and analysing the information stored in the created data warehouse.

Two developed systems are integrated as a one solution that fulfills the system requirements. System owner's needs and interest were taken into account throughout the analysing and development process. The built system allows end-users to create complex inquiries and analyse the data related to media archive orders and thereby increase the effectiveness of their everyday working process.

The system which was developed during this thesis, was made in a way that future developments are easy to add and there would be no need to redevelop the whole system. That makes easier to add additional functionalities.

From this thesis, it can be inferred, that it was possible to develop a data warehouse and a data cube, with a purpose to provide analysing functionalities and increase the value of media asset management system's archive orders data, based on the chosen methodology, created analysis and used development tools.

To conclude – with the help of the built system, it is possible to analyse the data related to media asset management system's archive orders and by that increase the data's value for Estonian Public Broadcasting. All the aims stated for this thesis are accomplished.

KASUTATUD KIRJANDUS

Acharya, A. (2013). Instant Microsoft SQL Server Analysis Services 2012 Dimensions and Cube. Birmingham: Packt Publishing Ltd.

Analysis Services. (2015). Microsoft.
<https://technet.microsoft.com/en-us/library/bb522607.aspx> (05.04.2015)

Dimension Relationships. (2015). Microsoft.
<https://msdn.microsoft.com/en-us/library/ms175669.aspx> (28.04.2015)

Hammergren, T. C., Simon, A. R. (2009). Data Warehousing for Dummies. 2nd ed. Indianapolis: Wiley Publishing, Inc.

Han, J. Kamber, M. (2006). Data Mining: Concepts and Techniques. 2nd ed. s.l: Elsevier Science & Technology.

Junic, N. (2006). Modeling Strategies and Alternatives for Data Warehouse Projects.
http://www.researchgate.net/profile/N_Jukic/publication/220421806_Modeling_strategies_and_alternatives_for_data_warehousing_projects/links/00b7d533329047168f00000.pdf (12.03.2015)

Kimball, R., Ross, M. (2002). The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. 2nd ed. s.l.: John Wiley and Sons, Inc.

Kimball, R., Ross, M. Thornthwaite, W. Mundy, J. Becker, B. (2008). The Data Warehouse Lifecycle Toolkit: Practical Techniques for Building Data Warehouse and Business Intelligence Systems. 2nd ed. Indianapolis: John Wiley and Sons, Inc.

Mailvaganam, H. (2007). Introduction to OLAP.
http://dwreview.com/OLAP/Introduction_OLAP.html (02.03.2015)

Multidimensional Model Data Access (Analysis Services – Multidimensional Data). (2015). Microsoft. <https://msdn.microsoft.com/en-us/library/ms126037.aspx> (11.04.2015)

Ponniah, P. (2001). Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals. s.l.: John Wiley and Sons, Inc.
<https://anuradhasrinivas.files.wordpress.com/2013/03/data-warehousing-fundamentals-by-paulraj-ponniah.pdf> (1.03.2015)

Power Pivot Add-In. (2015). Microsoft. <https://support.office.com/en-nz/article/Power-Pivot-Add-in-a9c2c6e2-cc49-4976-a7d7-40896795d045> (09.04.2015)

Querying Multidimensional Data with MDX. (2015). Microsoft.
<https://msdn.microsoft.com/en-us/library/bb500184.aspx> (11.04.2015)

- Ross, M. (2009). Design Tip #115 Kimball Lifecycle in a Nutshell.
<http://www.kimballgroup.com/2009/08/design-tip-115-kimball-lifecycle-in-a-nutshell/>
(8.03.2015)
- SQL Server Data Tools (SSDT). (2015). Microsoft. <https://msdn.microsoft.com/en-us/library/hh272686%28v=vs.103%29.aspx> (5.04.2015)
- SQL Server Integration Services. (2015). Microsoft. <https://technet.microsoft.com/en-us/library/ms141026.aspx> (5.04.2015)
- SSAS Analysis Services Client – AnalysisPortal. (2015). Kyubit.
<http://www.kyubit.com/SSAS-Analysis-Services-Client> (9.04.2015)
- Tepandi, J. (2015). Tarkvara kvaliteet ja standardid (IDX5721, IDX5722).
<http://deephought.ttu.ee/users/tepandi/pdf/tns-loeng.pdf> (23.03.2015)
- Vallaste, H. (2015). e-Teatmik.
<http://www.vallaste.ee/> (02.03.2015)
- What is OLAP? Definition of OLAP, Advantages and Uses. (2015). OLAP.com.
<http://olap.com/olap-definition/> (19.04.2015)

LISAD

Lisa 1. Illustratsioon IBM lindirobotist



Lisa 2. Andmeaida andmebaasi ja -objektide loomise SQL laused

Andmebaasi loomine

```
CREATE DATABASE MAM_datawarehouse;
```

Tabelite loomine

```
CREATE TABLE [dbo].[File_Dim](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [FileName] [nvarchar](250) NOT NULL,
    CONSTRAINT [PK_File_Dim] PRIMARY KEY CLUSTERED ([ID] ASC))
```

```
CREATE TABLE [dbo].[User_Dim](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Intranet_ID] [int] NOT NULL,
    [FullName] [nvarchar](250) NOT NULL,
    [Email] [nvarchar](100) NULL,
    CONSTRAINT [PK_User_Dim] PRIMARY KEY CLUSTERED ([ID] ASC))
```

```
CREATE TABLE [dbo].[Status_Dim](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Intranet_ID] [smallint] NOT NULL,
    [Name] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Status_Dim] PRIMARY KEY CLUSTERED ([ID] ASC))
```

```
CREATE TABLE [dbo].[Clip_Dim](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Intranet_ID] [int] NOT NULL,
    [Tc_In] [time](2) NOT NULL,
    [Tc_Out] [time](2) NOT NULL,
    [Comment] [nvarchar](250) NOT NULL,
    [Note] [nvarchar](250) NOT NULL,
    [Job_ID] [nvarchar](50) NOT NULL,
    [UpdatedAt] [datetime] NOT NULL,
    CONSTRAINT [PK_Clip_Dim] PRIMARY KEY CLUSTERED ([ID] ASC))
```

```
CREATE TABLE [dbo].[Order_Fact](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Intranet_ID] [int] NOT NULL,
    [File_ID] [int] NOT NULL,
    [User_ID] [int] NOT NULL,
    [Status_ID] [int] NOT NULL,
    [InsertDate] [datetime] NULL,
    [OrderDate] [datetime] NULL,
    [OrderDays] [int] NULL,
    [OrderTime] [time](2) NULL,
    CONSTRAINT [PK_Order_Fact] PRIMARY KEY CLUSTERED ([ID] ASC))
```

```

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_Date_Dim] FOREIGN KEY([InsertDate])
REFERENCES [dbo].[Date_Dim] ([ID])

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_Date_Dim_2] FOREIGN KEY([OrderDate])
REFERENCES [dbo].[Date_Dim] ([ID])

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_File_Dim] FOREIGN KEY([File_ID])
REFERENCES [dbo].[File_Dim] ([ID])

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_Order_Fact] FOREIGN KEY([ID])
REFERENCES [dbo].[Order_Fact] ([ID])

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_Status_Dim] FOREIGN KEY([Status_ID])
REFERENCES [dbo].[Status_Dim] ([ID])

ALTER TABLE [dbo].[Order_Fact] ADD CONSTRAINT
[FK_Order_Fact_User_Dim] FOREIGN KEY([User_ID])
REFERENCES [dbo].[User_Dim] ([ID])

CREATE TABLE [dbo].[Clip_Group_Dim](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Clip_ID] [int] NOT NULL,
    [Order_ID] [int] NOT NULL,
CONSTRAINT [PK_Clip_Group_Dim] PRIMARY KEY CLUSTERED ([ID] ASC))

ALTER TABLE [dbo].[Clip_Group_Dim] ADD CONSTRAINT
[FK_Clip_Fact_Order_Fact] FOREIGN KEY([Order_ID])
REFERENCES [dbo].[Order_Fact] ([ID])

ALTER TABLE [dbo].[Clip_Group_Dim] ADD CONSTRAINT
[FK_Clip_Group_Dim_Clip_Dim] FOREIGN KEY([Clip_ID])
REFERENCES [dbo].[Clip_Dim] ([ID])

CREATE TABLE [dbo].[Order_Dim](
    [Order_ID] [int] NOT NULL,
    [Note] [nvarchar](250) NOT NULL,
CONSTRAINT [PK_Order_Dim] PRIMARY KEY CLUSTERED ([Order_ID] ASC))

ALTER TABLE [dbo].[Order_Dim] ADD CONSTRAINT
[FK_Order_Dim_Order_Fact] FOREIGN KEY([Order_ID])
REFERENCES [dbo].[Order_Fact] ([ID])

```

Lisa 3. Andmeaida projekti uuendamise juhend

Andmeaida projekti uuendamise ning evitamise sammud on järgnevad:

- 1) andmeaida projekti vaatamiseks ning uuendamiseks on vaja programme:
 - a. Visual Studio 2012 või 2013,
 - b. Microsoft SQL Server Data Tools;
- 2) avada projekt (.sln laiendiga fail) kasutades VS-t;
- 3) teha soovitud muudatused/arendused;
- 4) käivitada (build) projekt;
- 5) kopeerida bin\Deployment kaustast failid sihtserverisse:
 - a. Package.dtsx,
 - b. MAM_Datawarehouse.SSISDeploymentManifest;
- 6) kopeeritud failide asukohaks sihtserveril võiks olla {projekti asukoht}\Build;
- 7) ava 'Build' kaustast fail 'MAM_Datawarehouse.SSISDeploymentManifest' ja alusta evitamisega (deploy). Avanenu abimehes (wizard):
 - a. valida "File system deployment",
 - b. deploy asukoht failiserveris: {projekti asukoht}\Deploy,
 - c. edasi (next),
 - d. edasi (next),
 - e. edasi (next),
 - f. (!)kontrollida üle konfiguratsiooni väärtused, et andmeid laetaks õigest operatiivsüsteemist;
- 8) SQL Server Agent job teeb automaatselt igal öösel kell 5.00 andmeaidale ja kuubile uuenduse. Juhul, kui andmete laadimine on vaja kohe käivitada on võimalik SQL Job käsitsi käivitada:
 - a. avada SQL Management Studio,
 - b. logida sisse relatsioonilisse andmebaasi,
 - c. navigeerida "SQL Server Agent" -> "Jobs",
 - d. käivitada SQL Job "MAM_laadimine" (tehakse uuendus ka andmekuubile).

Lisa 4. Andmekuubi projekti uuendamise juhend

Andmekuubi projekti uuendamise ning evitamise sammud on järgnevad:

- 1) andmekuubi projekti vaatamiseks ning uuendamiseks on vaja programme:
 - a. Visual Studio 2012 või 2013,
 - b. Microsoft SQL Server Data Tools;
- 2) avada projekt (.sln laiendiga fail) kasutades VS-t;
- 3) teha soovitud arendused/muudatused;
- 4) käivitada (build) projekt;
- 5) evitada (deploy) projekt;
- 6) avada käsurealt programm Microsoft.AnalysisServices.Deployment.exe;
- 7) avanenud abimehe abil luua .XMLA fail, andmebaasi faili asukoht:
 - a. bin\datacube_MAM.asdatabase;
- 8) kopeerida loodud .XMLA fail sihtserverisse;
- 9) avada sihtserveris Microsoft SQL Server Management Studio ning logida sisse SSAS andmebaasi (Server Type: Analysis Services);
- 10) avada Microsoft SQL Server Management Studios .XMLA fail ning muuta vajadusel andmebaasi ühenduse parameeterid (“ConnectionString”) vastavalt sihtserveri andmeaidale;
- 11) käivitada .XMLA fail, selleks navigeerida Query -> Execute;
- 12) evitamise õnnestumisel kuvatakse sellekohast teksti (Query executed successfully);
- 13) navigeerida andmebaaside nimekirjas vastava kuubi andmebaasile ning see käivitada – andmebaasil paremkliki -> process. Tegevuse õnnestumisel kuvatakse sellekohast teksti (Process succeeded).