KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Algirdas Kazla

MEDICININIŲ DOKUMENTŲ AUTOMATIZUOTOS ANALIZĖS METODIKOS TYRIMAS

Magistro darbas

Darbo vadovas
doc. dr. Eimutis Karčiauskas

KAUNAS, 2005

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

TVIRTINU
Katedros vedėjas
doc. dr. E. Bareiša
2005 05 23

MEDICININIŲ DOKUMENTŲ AUTOMATIZUOTOS ANALIZĖS METODIKOS TYRIMAS

Informatikos magistro baigiamasis darbas

Kalbos konsultantė
Lietuvių k. katedros lektorė
dr. J. Mikelionienė
2005 05 16

Vadovas
doc. dr. E. Karčiauskas
2005 05 23

Recenzentas
prof. habil. dr. H. Pranevičius
2005 05 23

Atliko
IFM 9-1 gr. stud.
Algirdas Kazla
2005 05 23

KAUNAS, 2005

TURINYS

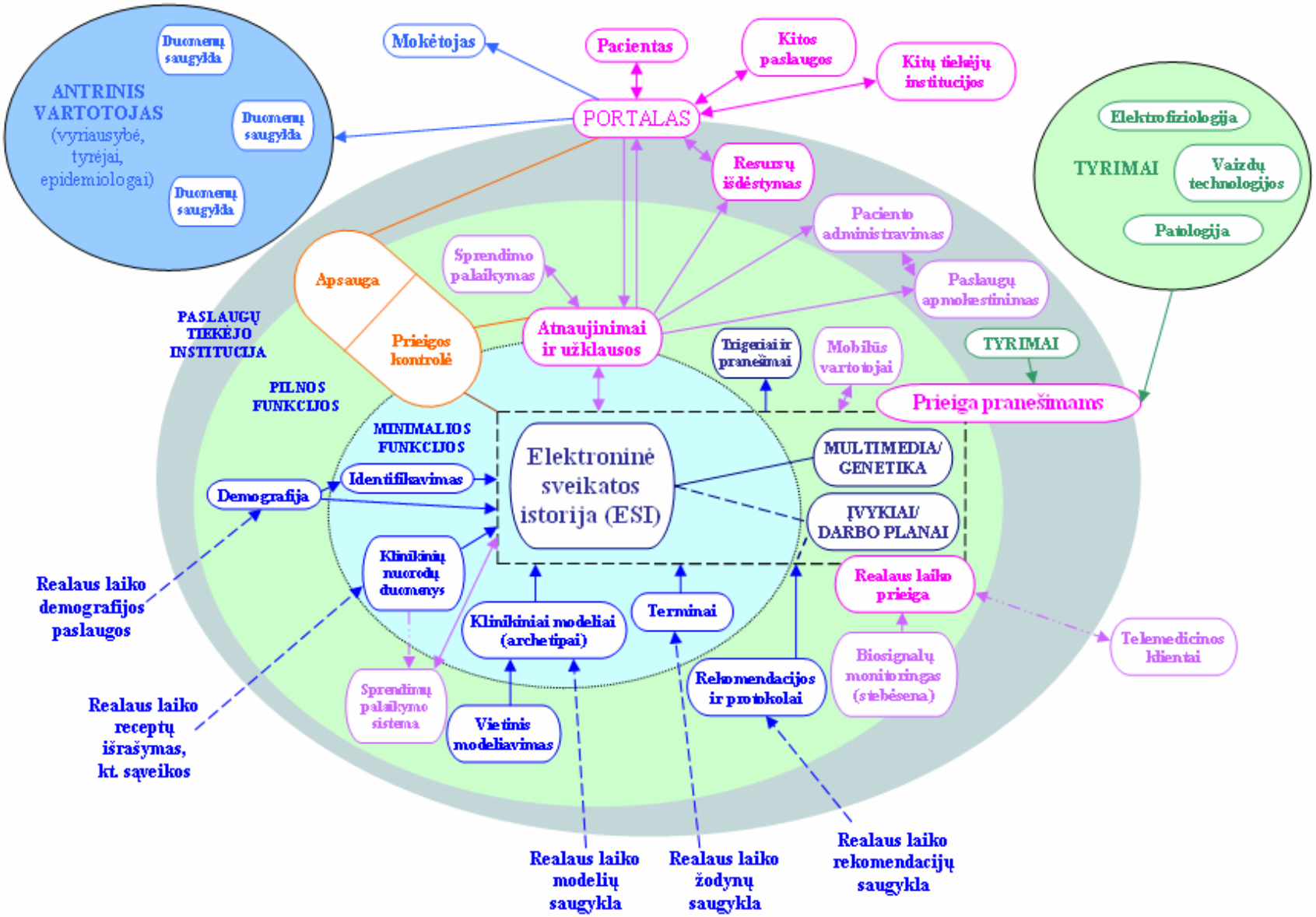
1. ĮVADAS

1.1. Prielaidos ir tikslai

Daugiau nei trys dešimtmečiai prabėgo nuo to laiko, kaip 1973 metais JAV pirmasis bendro naudojimo medicinos tyrimams skirtas kompiuteris buvo prijungtas prie tuometinio ARPAnet tinklo [31]. Šis tinklas dar nebuvo pasiekęs šiuolaikinio interneto lygio, tačiau plačios informacijos apsikeitimo ir informacinių technologijų galimybės padidinti sveikatos apsaugos saugumą, kokybę ir efektyvumą buvo akivaizdžios. Po kompiuterio atsiradimo, tai buvo dar vienas didelis žingsnis medicinos kompiuterizacijoje. Nuo to laiko daug ligoninių, universitetų, ir privačių organizacijų aktyviai medicinoje taiko informacines technologijas.

Esminis sveikatos apsaugos informacinių technologijų infrastruktūros kūrimo elementas yra elektroninė sveikatos istorija (**ESI**) ir jų sistemos [23]. Elektroninė sveikatos istorija yra apibrėžiama kaip [7] bet kokia informacija, susijusi su praeities, dabarties ir ateities fizine ir psichine sveikata ar būsena žmogaus, priklausančio elektroninių sveikatos istorijų sistemai (sistemoms), kuri skirta fiksuoti, perduoti, priimti, kaupti, sujungti, ir manipuliuoti įvairialypės terpės duomenimis, siekiant pagrindinio tikslo – teikti sveikatos apsaugos paslaugas. Jau nuo 1991 metų yra aktyviai bandoma eliminuoti popierines pacientų sveikatos istorijas, pakeičiant jas standartizuotais elektroniniais dokumentais. Pagrindinis tikslas nėra atsisakyti popierinių dokumentų, bet turėti **naudotiną** ir **prieinamą** kritiškai svarbią informaciją tada, kada jos reikia. Vienos iš pagrindinių ESI sistemų funkcijų yra [23]: duomenų apie sveikatos būklę saugojimas, tyrimų rezultatų valdymas, įvairių dokumentų kūrimo valdymas (pvz., receptų išrašymo), sprendimų sistemų palaikymas, administracinių procesų valdymas ir kt. (1 pav.).

Bendros (standartizuotos) ESI nauda yra akivaizdi [13] – visos sveikatos sistemos darbo pagerinimas, griaunant tarp organizacines sienas (tarp įvairių sveikatos apsaugos įstaigų), geresnis sveikatos apsaugos paslaugų valdymas, planavimas, mažėjantis informacijos dubliavimas, geresnis sprendimų priėmimas turint daugiau informacijos, plačių statistinių tyrimų palaikymas, ir kt. Tačiau medicinos kompiuterizacija nevyksta centralizuotai ir vienu metu. Kaip ir kitose šakose, informacinės technologijos taikomos daugiau konkretiems poreikiams, o ne bendrai situacijai keisti. Tokios kompiuterizacijos rezultatas – aibė heterogeninių sistemų, kurios neatitinka elektroninių sveikatos istorijų sistemoms suformuluotų reikalavimų ir daugiausia atlieka tik „kompiuterinio popieriaus“ funkciją.



1 pav. Elektroninės sveikatos istorijos kontekstas [32]

Istoriškai, labiausiai elektroninių medicininių istorijų reikėjo norint atlikti medicininius (statistinius) tyrimus, klinikinį auditą ir sveikatos paslaugų valdymą [30]. Taip pat elektroninės sveikatos istorijos turėjo tapti sprendimų priėmimo sistemų dalimis. Statistiniams tyrimams atlikti yra reikalingos dvi elektroninės sveikatos istorijos dokumento savybės – palyginamumas ir lankstumas. Akivaizdu, kad tiesioginis palyginamumas yra neįmanomas kai duomenys yra mažai struktūrizuoti, ar laisvo teksto formos. Iš kitos pusės, lankstumas, kuris čia yra suprantamas kaip struktūros savybė įvairiausiais būdais agreguoti įvairius duomenis, pabrėžia struktūros laisvumo laipsnį, kuris yra didžiausias esant laisvo teksto formai. Tam, kad elektroniniai dokumentai tiktų sprendimų priėmimo sistemoms, yra labai svarbi medicininės elektroninės istorijos forma – tai yra medicininės istorijos informacinis modelis, palaikantis įvairius paciento ir jo priežiūros proceso aprašymus [30]. Kadangi medicina apima labai daug sąvokų, procesų ir koncepcijų (pvz., SNOMED medicininių terminų nomenklatūra susideda maždaug iš 364 tūkstančių sąvokų ir 1,45 milijono semantinių ryšių [35]), tai bendros (standartizuotos) ESI kūrimas nėra trivialus.

1.2. Pasaulinės tendencijos

Paskutinių dviejų dešimtmečių standartizuotų ir vieningų elektroninių sveikatos istorijų sistemų kūrimo praktika parodė, kad dar nedaug priartėta prie homogeninių ESI sistemų. Tuo labiau, nėra aišku, ar išvis įmanoma pasiekti norimą homogeniškumo lygį [10]. Todėl bendra standartizacija yra labiau orientuota į elektroninių sveikatos istorijų apsikeitimo standarto, o ne į istorijų saugojimo terpės, kūrimą. Buvo sukurta (ir dabar toliau kuriama, tobulinama) eilė įvairių tiek ESI apsikeitimui, tiek jų saugojimui skirtų standartų. Populiariausi iš jų – amerikietiškasis HL7 (*Health Level 7*) [20] standartas, skirtas duomenims keistis; ENV 13606 [14] – Europos preliminarus sveikatos istorijų apsikeitimo standartas; DICOM [6] – standartas skirtas įvairialypės terpės duomenimis keistis; GEHR (*Good European/Electronic Health Record*) [18] – elektroninių sveikatos istorijų standartas, skirtas aprašyti, saugoti ir keistis lanksčiais elektroniniais sveikatos dokumentais (GEHR pagrindu yra kuriamas nauja ESI platforma – openEHR [28]); pradžioje GEHR buvo kurtas Europoje, o jo rezultatus perėmė ir darbus pratęsė Australija.

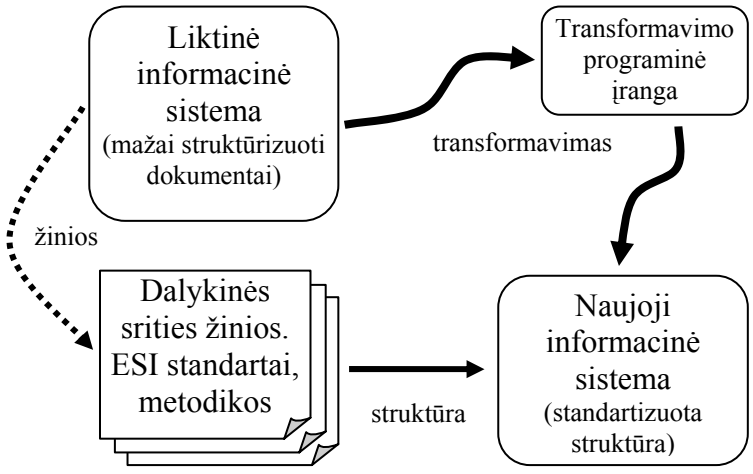
Per paskutiniuosius metus vyksta gan aktyvus globalus standartizavimas. HL7, GEHR, ANSI (*American National Standards Institute*), ISO (*International Organization for Standardization*), CEN (*European Committee for Standardization*) organizacijų pastangos yra nukreiptos į bendradarbiavimą, standartų harmonizaciją, siekiant sukurti nors ir skirtingas, tačiau suderinamas sveikatos apsaugos elektroninės informacijos saugojimo ir apsikeitimo sistemas.

1.3. Darbo tikslas

Informacinių technologijų diegimas, ir konkrečiai sveikatos apsaugos kompiuterizacija, Lietuvoje vyksta gan sparčiai. Tačiau problemos, kylančios kuriant sveikatos apsaugos informacinių technologijų infrastruktūrą yra tokios pačios, kokios sutinkamos ir pasaulinėje praktikoje – atskiros heterogeninės informacinės sistemos, programinės įrangos sprendimų taikymų konkretumas, orientuotas į įvairius siaurus standartus ir kt.

Atsižvelgdama į pasaulinį sveikatos apsaugos informacinių technologijų infrastruktūros standartizavimo procesą, Lietuva taip pat turi prisidėti prie šio proceso, pirmiausia įgyvendindama vieningas sveikatos istorijų sistemas ligoninių skyrių, ligoninių, galų gale ir visos Lietuvos mastu. Sąlygos šiam darbui atlikti dabar yra palankios, kadangi jau yra nusistovėję tam tikri, pasaulyje pripažinti ir suderinami, standartai tiek sveikatos istorijoms saugoti, tiek jomis keistis.

Šio darbo tikslas – sudaryti ir išbandyti automatizuotą medicinos dokumentų analizę, kuri išgautų duomenis iš siauriems taikymams skirtų liktinių informacinių sistemų ir talpintų juos naujoje sistemoje, sudarytoje pagal elektroninių sveikatos istorijų standartuose rekomenduojamas metodikas. Tai yra, bus siekiama transformuoti per eilę metų sukauptus medicininius duomenis iš „kompiuterinio popieriaus“ (nestruktūrizuotų arba mažai struktūrizuotų įrašų) į griežtos struktūros sistemą, paremtą elektroninių sveikatos istorijų standartais. Konceptuali darbo schema pateikta 2 paveiksle.



2 pav. Konceptuali darbo schema.

2. ELEKTRONINIŲ SVEIKATOS DOKUMENTŲ ARCHITEKTŪROS PRINCIPŲ ANALIZĖ

2.1. Preliminarus Europos ESI apsikeitimo standartas ENV 13606

Kaip minėta darbo įvade, populiariausi ir dabar labiausiai vystomi elektroninių sveikatos istorijų saugojimo bei apsikeitimo standartai yra: Jungtinėse Amerikos Valstijose sukurtas ir dabar plačiai pasaulyje taikomas Health Level 7; Europos standartizacijos komiteto kuriamas CEN ENV 13606, bei tarptautinės GEHR/openEHR sistemos. Iš pastarųjų metų darbų, kuriuos vykdo šiuos standartus kuriančios organizacijos, matyti standartų harmonizavimo (tai yra suderinamumo) tendencijos [28]. Todėl kuriant elektroninių istorijų sistemą, pakanka, kad ji būtų orientuota į bent vieną iš šių standartų. Kadangi darbas yra atliekamas vystant Lietuvos sveikatos apsaugos informacinių technologijų infrastruktūrą, bus labiausiai atsižvelgiama į Europos standartizacijos komiteto CEN (*Comité Européen de Normalisation/European Committee for Standardization*) preliminarų elektroninių sveikatos istorijų apsikeitimo standartą ENV 13606.

Sveikatos apsaugos srityje Europos standartizacijos komitete darbus atlieka Sveikatos informatikos techninis komitetas TC 251. Šio komiteto tikslas – informacijos ir komunikacinių technologijų standartizavimas sveikatos srityje užtikrinant skirtingų sistemų suderinamumą ir bendradarbiavimą [25]. Tai yra reikalavimų specifikavimas sveikatos informacijos struktūrai, palaikančiai tiek klinikines, tiek administracines procedūras, techninių metodų kūrimas bendradarbiaujančioms sistemoms palaikyti, reikalavimų, orientuotų į sveikatos sistemų saugumą, apsaugą ir kokybiškumą, specifikavimas.

Pradinis CEN elektroninių sveikatos istorijų standartas „ENV 12265 Elektroninės sveikatos istorijos architektūra“ buvo pamatinis standartas, aprašantis esminius principus, kurių pagrindu turėtų būti kuriamos elektroninės sveikatos istorijos [8]. Remiantis šiuo standartu, 1999 metais buvo paskelbtas keturių dalių preliminarus išplėstinis elektroninių sveikatos istorijų apsikeitimo standartas ENV 13606. 2001 metų gruodį Sveikatos Informatikos komitetas CEN/TC 251 patvirtino naują darbo grupę „EHRcom“, kurios užduotis buvo recenzuoti ir patikslinti preliminarų standartą ENV 13606. „EHRcom“ darbo tikslas – pasiūlyti peržiūrėtą ir pataisytą preliminaraus standarto versiją, kurią Europos standartizacijos komitetas galėtų priimti kaip formalų standartą 2004 metais [28]. Preliminaraus standarto recenzavimas paremtas praktikoje sukaupta patirtimi tiek iš komercinių produktų, tiek iš pilotinių parodomųjų sistemų elektroninių sveiktos istorijų komunikacijos srityje.

Pagrindinė „EHRcom“ vykdomojo komiteto iškelta misija, susijusi su ESI apsikeitimo standartu, yra [8]: sukurti griežtą ir ilgalaikę elektroninių sveikatos istorijų informacijos aprašymo architektūrą, tam, kad palaikyti sistemų ir komponentų bendradarbiavimą, kurie turi sąveikauti su ESI servisais:

- kaip tarpinės programinės įrangos sistemos;
- perdavimui, pridėjimui ar modifikavimui bei prieigai prie sveikatos istorijų;
- per elektroninius pranešimus ar paskirstytuosius objektus;
- išlaikant pradinę klinikinę prasmę, kurią pateikė autorius;
- atspindint duomenų konfidencialumą, numatytą autoriaus ir paciento.

Pagrindinis standarto tikslas yra apibrėžti esminius programinės įrangos ar sveikatos istorijų sistemos elementus, kurių pagalba bus įmanomas bendradarbiavimas tarp sistemų, teikiančių, arba reikalaujančių priėjimo prie elektroninių sveikatos istorijų duomenų. Sveikatos istorijos subjektas yra individualus asmuo, o komunikacija (informacijos perdavimo veiksmas) daugumoje yra susijusi su šio asmens priežiūra.

Standartas apibrėžia labai aukšto lygio rekomendacinę architektūrą, kuri visiškai nepriklauso nuo ją įgyvendinančių priemonių. Nėra įvedami jokie aplinkos, jos konfigūracijos ir įgyvendinimo būdų bei perdavimo priemonių ribojimai.

Vienas iš svarbiausių preliminaraus elektroninių sveikatos istorijų apsikeitimo standarto ENV 13606 akcentų yra medicininių archetipų įtraukimas į standartą [22, 28]. Šis žingsnis labai priartino (suderinamumo prasme) preliminarų standartą prie eilės Europos Sąjungos vykdomų projektų (Synapses [33] ir SynEx [34]), bei GEHR projekto.

2.1.1. Standarto struktūra

Preliminarus sveikatos istorijų apsikeitimo standartas ENV 13606 yra sudarytas iš keturių dalių [8]:

- **Pirma dalis: Išplėstinė architektūra**

  Pirminis Europos preliminarus standartas ENV 12265 apibrėžė būtinus architektūrinius komponentus, tačiau tai buvo tik elementarūs komponentai. Išplėstinė architektūra apibrėžia papildomus komponentus, skirtus aprašyti elektroninės sveikatos istorijos struktūrą ir semantiką – tai suderina elektroninius dokumentus su eile klinikinių, etinių ir teisinių reikalavimų, bei leidžia keistis šiais dokumentais. Vienas iš išplėstinės architektūros tikslų yra apibrėžti architektūrinius komponentus kurie yra būtini sveikatos istorijos turinio konstravimui, palaikymui, naudojimui ir apsikeitimui įgyvendinti:

- Įrašyti įvairių pavidalų informaciją (pvz., tekstinę, koduotą, vaizdinę, garsinę);
- Atspindėti duomenų grupavimą ar struktūrinę organizaciją;
- Įgalinti elektroninių sveikatos istorijų sistemas atlikti procesus, kurie reikalingi klinikinėje praktikoje: priežiūros, audito ir prieigos kontrolė, ir kt.

- **Antra dalis: Srities terminų sąrašas**

  Šioje dalyje aprašoma aibė priemonių, įgalinančių įvairių lygių suderinamumą tarp elektroninių sveikatos istorijų, sukurtų skirtingose sistemos arba skirtingų komandų toje pačioje sistemoje. Priemonės yra orientuotos į elektroninių sveikatos istorijų naudojimą, kad būtų palaikoma:
  - Originali įrašo prasmė gaunančiosios sistemos galutiniam vartotojui;
  - Gautų duomenų suliejimas, leidžiantis ilgalaikį požiūrį į duomenis;
  - Tam tikro lygio automatinis apdorojimas, pvz., informacijos išgavimas.
- **Trečia dalis: Paskirstymo taisyklės**

  Trečioje dalyje aprašoma aibė paskirstytosios prieigos (prie elektroninių istorijų) taisyklių. Šios taisyklės skirtos visos arba dalies ESI paskirstymui valdyti. Taip pat pateikiamos priemonės apsaugos ir audito strategijai, atributams aprašyti ir įgyvendinti.
- **Ketvirta dalis: Pranešimai informacijos apsikeitimui**

  Ketvirtoje dalyje pateikiama aibė pranešimų šablonų, kurie įgalina dalies arba visos ESI apsikeitimą, atsakant į užklausos pranešimą, arba atliekant simetrinės elektroninių sveikatos istorijų saugyklos atnaujinimą. Pranešimai yra specifikuoti abstrakčiai, tai yra pateikti pranešimų informaciniai modeliai. Ketvirtos dalies priede yra pateiktos šių pranešimų XML schemos.

Atsižvelgiant į šio magistrinio darbo tikslą sukurti tam tikrą informacinę elektroninių sveikatos istorijų sistemą, daugumoje bus gilinamasi į pirmąsias dvi preliminaraus standarto dalis – siūlomą architektūrą (pirma dalis), ir dalykinės srities semantinį modelį (antra dalis).

2.1.2. Pirma dalis: Išplėstinė architektūra

2.1.2.1. Įvadas

[10] Pirmoji preliminaraus standarto dalis „Išplėstinė architektūra“ aprašo **atraminę architektūrą** (*reference architecture*) – abstrakčius elektroninių sveikatos istorijų struktūros ir turinio modelius. Tai yra aukšto abstrakcijos lygio šablonas, skirtas ESI sistemų kūrėjams.

Preliminarus standartas prENV12265 buvo pirma **ESIA** (**E**lektroninės **S**veikatos **I**storijos **A**rchitektūra – *Electronic Healthcare Record Architecture*) versija, kurios tikslas buvo pateikti pagrindus saugiam sveikatos istorijos apsikeitimui įvairiomis aplinkybėmis. Išplėstinė architektūra yra atnaujinta ESIA versija. Naujoje versijoje yra tiksliau pateikta architektūrinė organizacija ir ryšiai tarp komponentų. Naujos architektūros tikslas yra *bet kokios sveikatos informacijos pateikimas tokiu būdu, kad ji būtų atpažįstama ir suprantama pagal turinį ir kontekstą net ir tada, kai yra atskirta nuo informacijos sukūrimo šaltinio*. Išplėstinė architektūra akcentuoja „konteksto“ kaip architektūrinio principo svarbą, kadangi dauguma kitų principų, kurie yra būtini efektyviam sveikatos istorijos apsikeitimui, įvairiais lygiais remiasi būtent konteksto principu.

„Kontekstas“ yra apibrėžiamas kaip tekstas, esantis prieš ir po nagrinėjamą ištrauką, suteikiantis platesnę ar labiau suprantamą prasmę ištraukai nei tuo atveju, jei ištrauka būtų skaitoma atskirai. Elektroninių sveikatos istorijų apsikeitimui kuriami standartai yra labiau orientuoti į informacijos apsaugojimą, konteksto išsaugojimą, kad duomenys nebūtų pamesti, ar blogai interpretuoti, tačiau ne į konteksto praturtinimą. Taip daroma netyčinė žala pacientui. ESIA architektūroje, kontekstas gali būti aprašytas ne tik tekstu, bet ir architektūriniais komponentais, kurie po vieną ar aibėmis supa, turi savyje, jungia, apriboja ir/arba apibūdina turinį taip, kad keitimasis juo yra saugus ir vienareikšmiškas. Turinys gali būti tiek įvairialypės terpės duomenys, tiek ir paprasti tekstiniai laukai.

Išplėstinė architektūra nepateikia jokių su elektroninio dokumento komunikacija susijusių vietos, tipo ar laikinių apribojimų. Komunikacija čia yra apibrėžiama kaip informacijos apsikeitimo veiksmas. Preliminarus standartas neapibrėžia jokių taisyklių, kaip šis veiksmas turi būti atliktas. Tokiu būdu išplėstinė architektūra yra nepriklausoma nuo elektroninių dokumentų organizacijos (pvz. antrinės ar pirminės priežiūros istorijos) ir nuo laikinių kategorijų, kurios yra taikomos dokumentams (pvz. epizodiškas ar ilgalaikis požiūris).

2.1.2.2. ESI komunikacijos architektūra

Preliminarus standartas ENV13606 yra susijęs su elektroninių sveikatos istorijų komunikacija, tai yra apsikeitimu elektroninėmis istorijomis, bet ne su šių įrašų saugojimu sistemoje. Į dalį ar visą asmens sveikatos istorijos galima žiūrėti iš komunikacijos pusės, kaip į duomenis, skirtus perduoti, ar keistis – toks požiūris yra pagrindinis standarto akcentas.

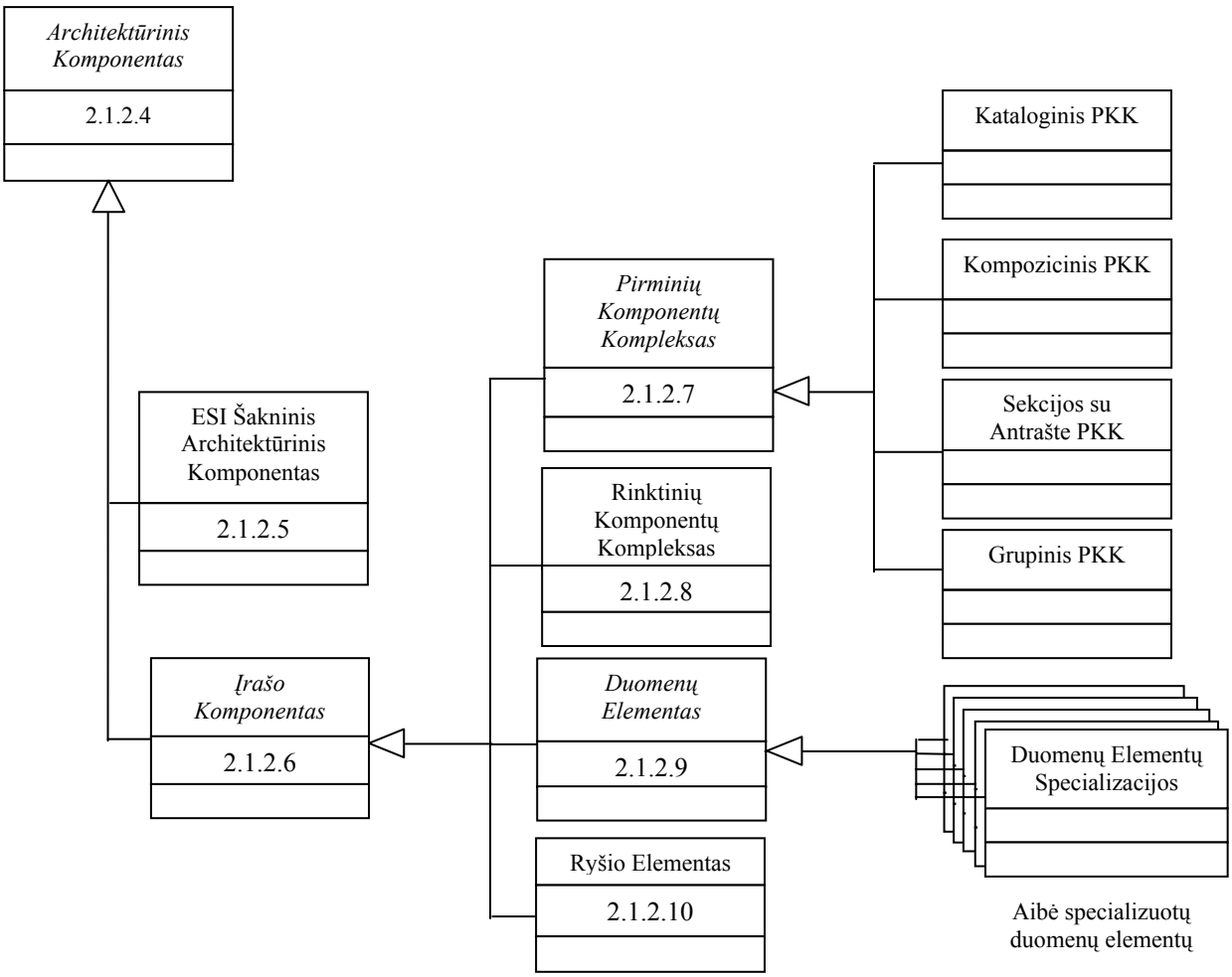
Elektroninių sveikatos istorijų komunikacinis vaizdas yra pakartotinis sukauptų klinikinių duomenų panaudojimas kitame kontekste. Gali būti daug tokių komunikacinių vaizdų, ir visi jie turi atitikti pirmąją standarto dalį, tai yra išplėstinę architektūrą. Išplėstinė

architektūra reikalauja, kad komunikacinis vaizdas atitiktų architektūros apibrėžiamus architektūrinius komponentus. Architektūriniai komponentai gali būti sujungiami įvairiais būdais, kas leidžia elektroninėje formoje dokumentuoti sveikatos priežiūros procesus, įvykius, veiksmus, būsenas. Taip pat gali būti apibrėžti loginiai požiūriai į duomenis, kas leidžia keistis klinikine informacija „į problemą“ orientuotu būdu, chronologine tvarka ar kt. Požiūriai yra sudaromi panaudojant ryšius, kurie savo lankstumu įgalina informacijos vienetų sujungimą taip lanksčiai, kaip to reikalauja sveikatos apsaugos sritis.

2.1.2.3. Architektūriniai komponentai

Šis skyrius aprašo Elektroninių sveikatos istorijų architektūros komponentus. Komponentai yra aprašyti pasinaudojant Unifikuotos modeliavimo kalbos (*Unified Modeling Language*) notacija, orientuota į objektinį modeliavimą. Tačiau standartas nereikalauja, kad sistemos įgyvendinančios šį standartą būtų kuriamos pagal objektinę metodologiją.

Preliminarų standartą atitinkantis komunikacinis vaizdas turi būti sudarytas iš 3 paveiksle pateiktų komponentų. Komponentų aprašymas pateiktas skyreliuose nuo **2.1.2.4** iki **2.1.2.10**.



3 pav. Architektūriniai komponentai.

2.1.2.4. Architektūrinis komponentas

Architektūrinis komponentas (AK) yra išplėstinės architektūros fundamentali sudedamoji dalis. AK yra abstrakti klasė, tai yra jos tipo egzempliorius galima kurti tik iš specializuotų AK klasių. Architektūrinis komponentas aprašo pačius bendriausius ryšius ir atributus, kurie yra būdingi visiems išplėstinės architektūros komponentams. Pagrindinės architektūrinio komponento klasės yra elektroninės sveikatos istorijos šakninis architektūrinis komponentas ir įrašo komponentas.

2.1.2.5. ESI šakninis architektūrinis komponentas

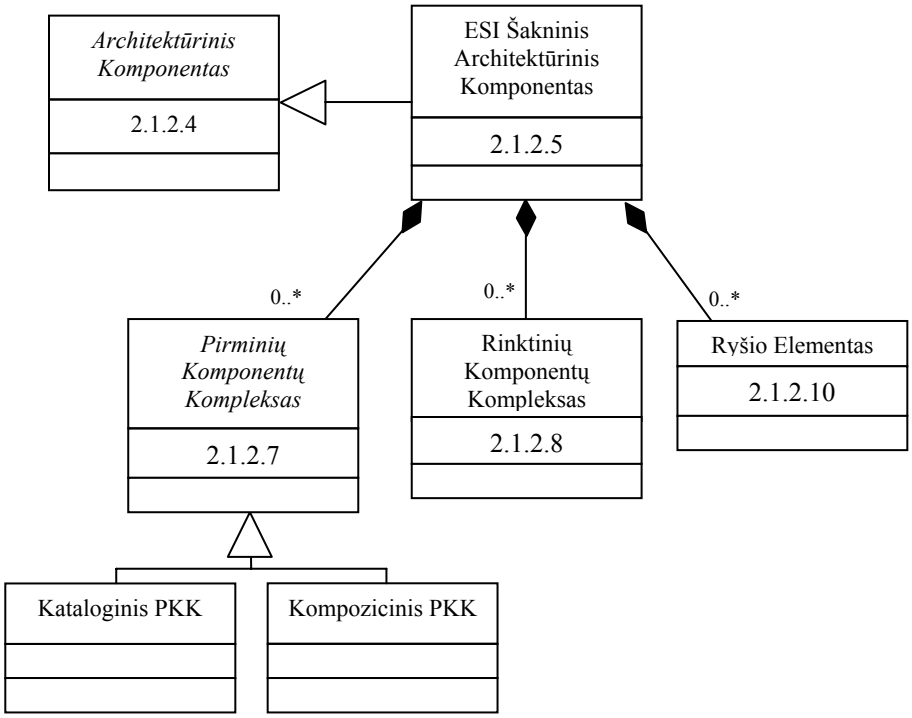
Elektroninės sveikatos istorijos šakninis architektūrinis komponentas (ESI ŠAK) yra komponentas, turintis savyje paciento elektroninę sveikatos istoriją. Idealiu atveju informacinėje sistemoje turėtų būti tik vienas ESI šakninio komponento egzempliorius vienam pacientui. Tačiau praktiškai tam tikrais atvejais yra galimos ESI šakninių komponentų kopijos:

- yra reikalavimas tam tikro skyriaus elektroninius įrašus laikyti atskirai nuo bendrų elektroninių įrašų;
- įvyko paciento identifikavimo klaida, dėl kurios atsirado ESI ŠAK kopija.

ESI šakninis architektūrinis komponentas turi savyje bent pačią paprasčiausią identifikacijos informaciją apie pacientą (paciento identifikacijos numerį). ESI ŠAK turi savybę, kuri teikia pirminį informacijos kontekstą visiems šakninio komponento nariams (visiems jo apimamiems komponentams).

ESI šakninis architektūrinis komponentas pasižymi visomis savybėmis (atributais, ryšiais) kaip ir jo paveldima klasė - architektūrinis komponentas. Be to, ESI ŠAK gali turėti (nuo nulio iki daug) tokių tipų narius (4 pav.):

- Kompozicinis pirminių komponentų kompleksas;
- Kataloginis pirminių komponentų kompleksas;
- Rinktinių komponentų kompleksas;
- Ryšio elementas;



4 pav. ESI Šakninio architektūrinio komponento kontekstas.

2.1.2.6. Įrašo komponentas

Įrašo komponentas (ĮK) yra abstrakti architektūrinio komponento specializacija, kuri gali būti (yra) identifikuojama atskirai (t.y. galimos nuorodos į šį komponentą). Įrašo komponentas yra tėvinė visų komponentų, skirtų ESI duomenims ir struktūroms aprašyti, klasė. ĮK specializacijos yra:

- Pirminių komponentų kompleksas;
- Rinktinių komponentų kompleksas;
- Duomenų elementas;
- Ryšio elementas.

Kiekvienas ĮK egzempliorius turi atributą, kuris yra naudojamas „tėviniam“ įrašo komponentui identifikuoti. Tokiu būdu sudaroma komponentų hierarchija, kurios šakninis komponentas yra ESI šakninis komponentas.

Įrašo komponentas įveda svarbų komponento kategorijos atributą. Šis atributas, apibrėžiamas specialiu terminu iš galimų terminų sąrašo, leidžia iš skirtingų šaltinių gautus ESI komponentus traktuoti homogeniškai (peržiūrint, atliekant paiešką ar aukštesnio lygio agregacijos analizę). Pasirinkta komponento egzemplioriaus kategorija turi atspindėti bendrą komponento apjungiamos informacijos turinį. Preliminarus standartas nurodo tik dalį galimų ĮK kategorijų.

2.1.2.7. Pirminių komponentų kompleksas

Pirminių komponentų kompleksas (PKK) yra įrašo komponento specializacija, paveldinti visus ĮK atributus ir ryšius. Pirminių komponentų kompleksas yra abstrakti klasė, kurios specializacijos yra Kataloginis PKK, Kompozicinis PKK, Sekcijos su antrašte PKK, ir Grupinis PKK (1 lentelė).

Pirminių komponentų komplekso tikslas yra įrašo komponentų grupavimas, atspindintis pirminį informacijos kontekstą. Pirminis informacijos kontekstas – tai topologinis duomenų talpinimas ir pozicionavimas elektroninėje sveikatos istorijoje, tiksliai taip, kaip duomenys buvo talpinami autoriaus (sveikatos priežiūros agento – žmogaus, įrankio, ar programinės įrangos). Skirtingos PKK specializacijos atspindi skirtingus informacijos detalumo lygius – pradedant nuo kompleksų, kurie atspindi paciento ilgalaikį sveikatos įrašą, iki kompleksų kurie sujungia glaudžiai susijusius duomenų elementus. Pagrindinis papildomas atributas, kuriuo svarbi PKK specializacija, yra **anotacijos identifikatorius**. Šiuo identifikatoriumi standartizuotu būdu (reikšme iš terminų sąrašo) yra apibendrinama pagrindinė komplekso turinio kontekstinė informacija. Plačiau anotacijos identifikatoriai yra aprašyti skyriuje **2.1.3.2 Anotacijų identifikatoriai**.

Pirminių komponentų komplekso specializacijos tam tikru lygiu persidengia, ir jų apibrėžimai nėra visiškai tikslūs (formaliai atskiriami). Tačiau praktikoje toks aukšto lygio skaidymas pasitvirtina (projektai GEHR [18], Synapse [33] ir EHCR-SupA [9]).

1 lentelė. Pirminių komponentų komplekso specializacijos

PKK tipo pavadinimas	Apibūdinimas	Komponentų pavadinimų pavyzdžiai
Kataloginis PKK	Aukšto lygio ESI poskyris, dažniausiai grupuojantis per tam tikrą laiką sukauptus įrašus. Įrašų grupavimas gali būti atliktas pagal organizacijos departamentus ar skyrius, komandas, sveikatos problemas ir kt.	– Bendros praktikos gydytojo įrašas – Diabeto gydymo įrašas
Kompozicinis PKK	Aibė įrašų, susijusių su vienu metu suteiktomis sveikatos priežiūros paslaugomis; įrašai grupuojami pagal sveikatos priežiūros veiklą. Dažnai būtent tokio detalumo duomenimis keičiasi įvairūs sveikatos apsaugos dalyviai.	– Konsultacija – Receptas – Pagrindinių organizmo būklės rodiklių lentelė
Sekcijos su antrašte PKK	Poskyris, grupuojantis bendros temos, ar išplaukiančius iš bendro sveikatos priežiūros proceso įrašus. Dažnai būtent tokio detalumo duomenimis keičiasi įvairūs sveikatos apsaugos dalyviai.	– Ligos istorija – Tyrimų išvados – Gydymo planas
Grupinis PKK	Žemo lygio elementarių (duomenų elementų ir kitų grupinių PKK) komponentų agregacija, atvaizduojanti sudėtinę klinikinę koncepciją. Tai gali būti vieno testo keletas matavimų, sveikatos priežiūros veiksmo keletas pasekmių, aibė glaudžiai susijusių simptomų ir t.t.	– Fonokardiograma – Insulino leidimo grafikas – Skirtingas baltųjų ląstelių skaičius – Kraujo spaudimas pacientui stovint, gulint – Kraujo spaudimo matavimas iš sistolinio ir diastolino spaudimų.

2.1.2.8. Rinktinių komponentų kompleksas

Rinktinių komponentų kompleksas (RKK) yra įrašo komponento specializacija. Rinktinių komponentų kompleksas yra panašus į pirminių komponentų kompleksą, tačiau skirtingai nuo PKK, rinktinių komponentų komplekso narių egzemplioriai nėra tiesioginiai jo palikuonys, bet tik nuorodos į komponentų egzempliorius, kurių pirminės informacijos kontekstas yra kitur. Tai yra, **komponentai sudarantys tam tikrą turinį tam tikrame kontekste (agreguoti pirminių komponentų komplekso) gali būti pakartotinai panaudojami *nedubliuojant* kituose kontekstuose**. Tokiu būdu kuriami alternatyvūs „požiūriai“ į tuos pačius duomenis, nepakeičiant jau esamos duomenų išdėstymo struktūros (pirminio informacijos konteksto). RKK nariais gali būti: pirminių komponentų kompleksas (visos jo specializacijos), rinktinių komponentų kompleksas („požiūris“ kitame „požiūryje“), duomenų elementas, ryšio elementas.

Rinktinis komponentų kompleksas nebūtinai turi turėti pastovius narius. RKK gali būti užpildytas duomenimis atsakant į tam tikrą užklausą, tai yra užpildomas dinamiškai. Pvz.: užklausa gali būti sugeneruotas dabar paciento vartojamų vaistų sąrašas (užklausos rezultatas – RKK su nuorodomis į dabar vartojamų vaistų įrašus).

2.1.2.9. Duomenų elementas

Duomenų elementai (DE) yra smulkiausi (nedalomi) struktūriniai vienetai, į kuriuos gali būti skaidomas ESI turinys, neprarandant jo prasmės. Konceptualiai DE yra identifikacijos, atributų informacijos, pavadinimo ir duomenų turinio darinys. Skirtingose sistemose ir aplinkose DE duomenų turinys gali būti labai įvairus. Kad užtikrinti saugų duomenų apsikeitimą, yra apibrėžiami tam tikro lygio duomenų turinio standartai. Tokiu būdų yra sukuriama eilė DE specializacijų, pvz.:

- Kiekybiškai įvertinamo stebėjimo duomenų elementas

  Šis tipas pateikia šabloną, kaip kiekybiškai įvertinami stebėjimai turėtų būti pateikiami komunikacijai. Šablonas pateikia instrukcijas, kaip aprašyti matuojamą savybę, skaitinį rezultatą, komparatorių naudojimą, diapazonus ir t.t.
- Vaistų duomenų elementas

  Šis tipas pateikia šabloną informacijai apie vaistus – išrašytus, rekomenduotus, paruoštus ir t.t., kaip ji turėtų būti pateikta komunikacijai.
- Komentarų duomenų elementas

  Šis duomenų elemento tipas yra skirtas bet kokiu būdu komentuoti norimus komponentus. Komentaras su įrašo komponentu gali būti susiejamas per ryšio elementą.

Aibė duomenų elementų specializacijų yra apibrėžta ir šio preliminaraus standarto ketvirtoje dalyje.

2.1.2.10. Ryšio Elementas

Ryšio elementas (RE) yra įrašo komponento specializacija. Ryšio elementas jungia architektūrinio komponento egzempliorių, vadinamą **ryšio šaltiniu** su kitu AK egzemplioriumi, vadinamu **ryšio adresatu**. Pats RE negali būti nei ryšio šaltinis nei adresatas. Ryšio tarp komponentų pobūdis yra nusakomas ryšio elemento pavadinimo kategorijos atributu, trumpiau vadinamu ryšio elemento pavadinimu.

2.1.3. Antra dalis: Srities terminų sąrašas

2.1.3.1. Įvadas

[18] Paciento sveikatos informacijai keliaujant iš vienos ESI sistemos į kitą yra svarbu, kad dokumentų prasmė, tokia kokia buvo išreikšta juos sukūrusio autoriaus, liktų nepakitusi ir patikima, net jei ir gaunančiosios sistemos vidinė architektūra skiriasi nuo siunčiančiosios.

Elektroninė sveikatos istorija didėja ir sudėtingėja. Tampa nepraktiška ir nepatogu ją peržiūrinėti tiesiogiai, kaip vientisą dokumentą. Turi būti galimybės, leidžiančios išgauti iš istorijos specifines ištraukas, kurios yra svarbios tik tam tikram sveikatos apsaugos sektoriui. Taip pat svarbu, kad tokios sveikatos istorijų ištraukos gaunamos iš kitų sistemų galėtų būti prijungtos prie gaunančiosios sistemos tokiu būdu, kad tiktų ne tik galutiniam vartotojui skaityti, bet ir tam tikru lygiu kompiuteriniam apdorojimui.

Antroji preliminaraus standarto dalis „Srities terminų sąrašas“ pateikia semantinę metodologiją, kuria remiantis įmanomas teisingas klinikinių pranešimų apsikeitimas ir jų turinio interpretavimas.

Visa sveikatos apsauga susideda iš individualių *klinikinių situacijų* (sveikatos priežiūros kontekste galinčių įvykti, ar vykstančių reiškinių – būsenų, procesų, veiklų ir kt.), kurios elektroninėje sveikatos istorijoje autoriaus yra užrašomos kaip *klinikiniai įrašai*. Klinikinis įrašas susideda iš tam tikros aibės klinikinių koncepcijų ir aibės kontekstinės informacijos. Antroje preliminaraus standarto dalyje pateikiamos priemonės leidžia sukurti aukšto lygio „antrinį“ informacijos lygmenį: įrašų elementų anotacijos (kurios yra sudarytos iš standartinių koduočių) apibendrina svarbiausius elemente saugomos informacijos aspektus ir sumažina tikimybę, kad turinys bus interpretuotas klaidingai.

2.1.3.2. Anotacijų identifikatoriai

Anotacija – tai ENV 13606 standarto priemonė, skirta standartizuotu būdu apibendrinti pagrindinę kontekstinę informaciją, kuri siejasi su duomenų elementu arba grupiniu pirminių komponentų kompleksu. Pagrindinis šių priemonių tikslas yra užtikrinti teisingą kontekstinės informacijos turinio interpretaciją.

2.1.3.3. Anotacijų tipai ir baziniai deskriptoriai

Anotacijos yra įgyvendintos aibe *anotacijų tipų* (2 lentelė), atspindinčių pagrindines galimas konteksto savybes, kurios gali būti išsaugotos elektroninės sveikatos istorijos įraše. Kiekvienam anotacijos tipui priklauso tam tikras *bazinių deskriptorių* sąrašas, kurie yra galimos to tipo reikšmės. Aibė bazinių deskriptorių, nustatytų tam tikram įrašo komponentui, yra to komponento *anotacijos*. Anotacijos yra skirtos antriniam informacijos turinio aprašymui. Tai yra, anotacijomis standartizuotu būdu apibendrinamas jau išsaugotas informacijos turinys. Kiekvienas duomenų elemento ar grupinio PKK egzempliorius ESI sistemoje tiesiogiai (arba per nuorodą) turi turėti tinkamas anotacijas.

2 lentelė. Anotacijų tipai

Anotacijos tipas	Apibrėžimas ir pavyzdžiai
Būtinos anotacijos	
Anotacijos patikimumas	Nusako anotacijos patikimumą. Pvz., anotacija peržiūrėta ir patvirtina autoriaus, anotacija sukurta automatinio apdorojimo, anotacija nepatikima ir kt.
Rekomenduojamos anotacijos	
Informacijos subjektas	Nusako elemento saugomos informacijos subjektą (t.y. apie kokį subjektą saugoma informacija). Pvz.: pacientas, paciento giminė, donoras ir kt.
Gyvavimo ciklas	Anotacijos tipas, skirtas sveikatos priežiūros veikloms ir veiksmams. Nusako veiksmo būseną: atliktas, vykdomas, planuojamas, neatliktas (atmestas, nutrauktas ir kt.).
Galimybė	Anotacijos tipas, skirtas klinikinėms būsenoms ir būklėms. Nusako elemento saugomos informacijos galimybę. Pvz.: dabar esantis, tikslas, numanomas, yra pavojus (įvykti).
Proceso būsena	Nusako proceso būseną: naujas, vykstąs, prieš tai vykęs.
Neigimo ženklas	Elemente išreikštos informacijos neigimas: patvirtintas, paneigtas.
Tikrumo ženklas	Elemente išreikštos informacijos užtikrinimas – ar ji patvirtinta prideramais faktais, ar tik spėjama: neabejotina, abejotina.
Pirminis klinikinis kontekstas	
Situacija	Nusako klinikinę situaciją, kurią aprašo elementas.
Žinojimo būdas	Nusako elemento saugomos informacijos objektyvumo įvertį. Pvz.: specialisto stebėjimas ar veiksmas – objektyvus, paciento nusiskundimas – subjektyvus ir kt.
Antrinis klinikinis kontekstas	
Rolė	Nusako elemento rolę: diagnozė, problema, kreipimosi priežastis, įspėjimas.
Aktualumas	Nusako informacijos aktualumą: pirminis, antrinis.
Skubumas	Nusako skubumą: netikėtas atvejis, eilinis (rutininis) atvejis.
Susijusios temos	Nusako susijusias su elemento informacija temas: instrumentus, vaistus, metodus, esamą paciento būseną, kūno dalį.
Kūno sistema	Nusako apie kurią kūno sistemą, ar dalį yra saugoma informacija.

2.1.3.4. Anotacijų patikimumas

Anotacijos tipas „Anotacijos patikimumas“ yra būtinas, visi kiti anotacijų tipai nėra būtini. Gali būti, kad tam tikriems komponentų egzemplioriams anotacijos nebus reikalingos – tokiu atveju komponento anotacija turi susidėti iš vienos anotacijos reikšmės (tam tikro bazinio deskriptoriaus iš „Anotacijos patikimumas“ anotacijos tipo).

2.1.3.5. Ryšys tarp anotacijų ir anotuojamos informacijos

Prasmė, kurią perteikia aibė anotacijos bazinių deskriptorių turi nesikirsti ir neprieštarauti informacijai, kuri yra saugoma anotuojamame elemente. Baziniai deskriptoriai, kuriais anotuojamas grupinis pirminių komponentų kompleksas, neturi kirstis ar prieštarauti anotacijoms, kuriomis yra anotuoti komplekso nariai.

2.1.3.6. Paveldėjimas ir reikšmės pagal nutylėjimą

Anotacijų tipai neturi bazinių deskriptorių reikšmių pagal nutylėjimą. Tai yra, nustatant tam tikrą anotacijos tipą elementui, reikia išrinkti ir priskirti konkrečią bazinio deskriptoriaus reikšmę. Baziniai deskriptoriai nėra paveldimi – daugumoje atvejų bendra kontekstinė informacija, priskiriama įrašui, nebus tinkama kiekvienai atskirai įrašo daliai. Tačiau jei reikia anotacijos gali būti atkartojamos žemesniuose hierarchiniuose lygiuose, kur tinka.

2.1.3.7. Archetipai ir detalūs deskriptoriai

Anotacijomis apibendrinimas klinikinių įrašų turinys. Šis apibendrinimas yra labai „stambus“ (*coarse-grained*), ir negali užtikrinti tikslių turinio detalių. Yra poreikis kontekstinę informaciją aprašyti detalesniu būdu.

Archetipai ir detalūs deskriptoriai yra priemonės, kuriomis galima kurti detalesnes kontekstinės informacijos kodavimo sistemas. Kiekvieno įrašo saugoma informacija gali būti išreikšta per aibę detalių deskriptorių (pavienių žodžių ar trumpų frazių, dažniausiai daiktavardžių, kurių kiekvienas pavadina tam tikrą koncepciją, priklausančią semantinei kategorijai). Detalūs deskriptoriai, sujungiami semantiniais ryšiais (dažniausiai veiksmažodžiais arba sangrąžiniais veiksmažodžiais), sudaro semantinį tinklą. *Kategorinė struktūra*, kuri aprašo kaip semantinės kategorijos ir semantiniai ryšiai yra susiję tam tikroje terminologinėje sistemoje, valdo deskriptorių ir semantinių ryšių kombinacijas.

2.1.3.8. Archetipų aprašymo kalba[1]

ADL (*Archetype Definition Language*) – tai formali archetipų aprašymo kalba. Archetipai savo ruožtu yra formalios apribojimais pagrįstos išraiškos, skirtos tam tikros dalykinės srities žinių aprašymui [19].

ADL naudoja dviejų kalbų sintakses tam tikro duomenų modelio semantiniams apribojimams aprašyti: cADL (*constraint form of ADL*) – apribojimų aprašymo ADL ir dADL (*data definition form of ADL*) – duomenų aprašymo ADL. Naudingiausia, kai patys bendriausi informaciniai modeliai yra panaudojami duomenims aprašyti, pvz.: loginės koncepcijos PACIENTAS, DAKTARAS, ir LIGONINĖ gali būti išreiškiamos naudojantis klasėmis DALYVIS ir ADRESAS. Archetipai yra naudojami užtikrinti (apribojant) teisingus duomenų egzempliorius, sudarytus iš bazinių klasių. Tokiu būdu sudaromos lanksčios ir nesenstančios informacinės sistemos – aprašant pakankamai paprastus informacinius modelius ir duomenų bazių schemas, archetipus panaudojant semantinei srities daliai aprašyti visiškai už programinės įrangos ribų.

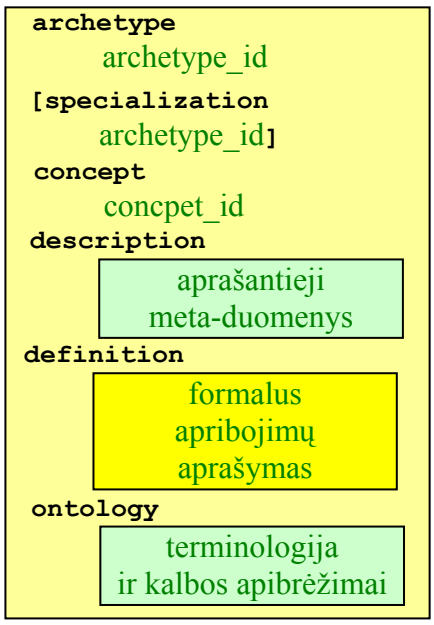
dADL kalba formaliai išreiškiami duomenų egzemplioriai, besiremiantys užduotu informaciniu modeliu. Šis formalus duomenų aprašymas yra skirtas tiek žmogui skaityti, tiek ir interpretuoti kompiuteriu, ir yra visiškai nesusietas su informacinio modelio sudarymo metodologija ar realizacija (pvz.: sąryšine arba objektine metodologijomis).

cADL yra kalba, kuri leidžia aprašyti apribojimus duomenims iš objektine metodologija sudaryto informacinio modelio, tam kad šie apribojimai būtų išreikšti archetipuose ar kituose formaliuose žinių aprašymo modeliuose. cADL yra naudojama tiek archetipų kūrimo metu, tiek ir galutinės sistemos naudojimo metu – sistema pagal archetipuose aprašytus apribojimus užtikrina duomenų teisingumą.

Archetipo, aprašyto ADL kalba, struktūra pavaizduota 5 paveiksle. Pagrindinės šios struktūros dalys:

- **archetipas** (*archetype*). Specialia koduote aprašytas archetipo identifikatorius, kuris identifikuoja archetipą globalioje archetipų erdvėje;
- **specializacija** (*specialization*). Nebūtina dalis – nurodo specializuojamojo (tėvinio), tai yra patikslinamo, archetipo identifikatorių. Archetipas gali turėti tik vieną specializuojamąjį archetipą.
- **koncepcija** (*concept*). Koduotu būdu aprašyta archetipo koncepcija, atspindinti tam tikrą realaus pasaulio koncepciją, pvz.: pacientas, kraujo spaudimas, ir pan.

[1] Archetipų kalbos ADL aprašymas yra paimtas iš naujausios ENV 13606 standarto versijos [19]. Atliekant šį darbą, naujoji standarto versija dar nebuvo patvirtinta.



5 pav. ADL kalba aprašyto archetipo struktūra.

- **aprašymas** (*description*). Archetipą aprašančiosios meta informacijos dalis (autorius, archetipo būsena, kada galima archetipą naudoti, kada negalima ir kt.). Ši dalis yra aprašoma dADL kalba.
- **apibrėžimas** (*definition*). Pagrindinė archetipo dalis – formaliai išreikšti archetipo apribojimai. Ši dalis yra aprašoma cADL kalba.
- **ontologija** (*ontology*). Ontologijos dalis susideda iš penkių sekcijų: terminologijos antraštė (archetipo aprašymo kalbos – anglų, prancūzų ir kt., ir vertimų duomenys), terminų apibrėžimai (aprašomi lokalūs archetipo terminų kodai), apribojimų apibrėžimai (aprašomi lokalūs apribojimai), terminų atvaizdavimas (lokalių terminų atvaizdavimas į globalias terminologijas), apribojimų atvaizdavimas (lokalių apribojimų atvaizdavimas į globalius). Ši dalis yra aprašoma dADL kalba.

Pats paprasčiausias archetipo pavyzdys ADL kalba:

```
archetype
      adl-test-ENTRY.most_minimal.draft
concept
      [at0000]    --  empty definition test
definition
      ENTRY[at0000] matches {*}
ontology
      primary_language = <"en">
      languages_available = <"en", ...>
      term_definitions("en") = <
            items("at0000") = <text = <"most minimal">;
            description = <"most minimal">>
      >
```

2.2. Medicinos ontologija elektroninių sveikatos istorijų standartuose

2.2.1. Įvadas

Norint atlikti medicininių dokumentų automatizuotą ontologinę analizę duomenų išgavimui, pirmiausia reikia turėti dalykinės srities – t.y. medicinos – žinių modelį (ontologiją). Medicininė ontologija yra kuriama sudarinėjant formalias medicinos ontologijas, terminų ir sąvokų žodynus ir kt. Tiek dėl ontologijų mokslo naujumo, tiek dėl medicinos sritis apimties, kol kas nėra bendros viską tiksliai aprašančios medicininės ontologijos. Elektroninių sveikatos istorijų saugojimo ir apsikeitimo standartų kūrimas, *medicininių archetipų* koncepcijos įvedimas ir standartų projektavimas remiantis *dvigubo modelio* koncepcija taip pat prisideda prie medicinos dalykinės srities specifikavimo elektroninėje formoje žinių lygmenyje.

2.2.2. Dvigubas ESI modelis

Klasikinės informacinės sistemos yra kuriamos taip, kad dalykinės srities koncepcijos, kuriomis turi operuoti sistema, yra tiesiogiai susiejamos su programine įranga ir duomenų bazių modeliais. Nors toks viengubo modelio būdas leidžia sąlyginai greitai kurti sistemas, dažniausiai tokių sistemų gyvavimo laikas yra gan trumpas, ir jų modifikavimas bei išplėtimas yra labai brangūs.

Kaip rodo praktika, medicinos srityje toks viengubo modelio būdas yra ypač netinkamas, kadangi pati medicinos sritis yra per plati, kad ją būtų galima aprašyti vienu žemo lygio modeliu. Tuo labiau, medicinos sritis yra labai besikeičianti [29]:

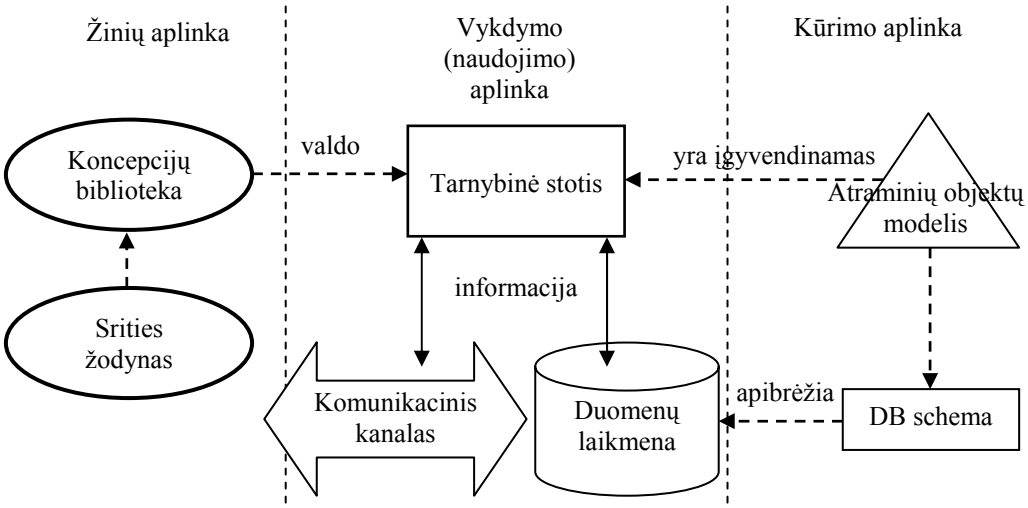
1. Į plotį, nes vis yra atrandama nauja svarbi informacija;
2. Į gylį, nes vis yra atrandamas didesnis detalumas;
3. Į sudėtingumą, nes vis yra atrandami nauji ryšiai.

Kitoks elektroninių sveikatos istorijų sistemų kūrimo būdas yra pasiūlytas GEHR [18] projekto. Griežtai pagrįstas žinių modeliavimu, šis būdas remiasi vienu esminiu principu: *žinių ir informacijos lygmenų atskyrimu informacinėje sistemoje* [2]. Čia terminai „žinios“ ir „informacija“ yra suprantami taip:

- **Informacija** – tai faktai apie specifinę esybę.
- **Žinios** – tai faktai, kurie tinka visoms tam tikros klasės esybėms.

Informacinių sistemų kūrimas, besiremiantis dviejų lygių modeliavimu yra galimybė pasiekti žinių lygmens bendradarbiavimą tarp sistemų. Tokiu būdų skirtingų sistemų kūrėjai turi susitarti tik dėl: (1) daug mažesnių programinių modelių, kurie apibrėžia bendrines dalykinės srities koncepcijas, o ne visas srities žinias; (2) techninių bendradarbiavimo būdų,

tokių kaip CORBA, COM, ir kt. Iš kitos pusės, šiuo būdu būtent sistemos vartotojai turi sutarti ir nustatyti dalykinės srities žinių modelius, kurie susideda iš medicininių archetipų, sudarytų iš standartizuotų srities žodynų ir taisyklių (6 pav.).



6 pav. Dvigubo modelio informacinių sistemų kūrimo metodologija [4].

Remiantis dvigubo modelio metodologija, viengubas, susietas su srities koncepcijomis programinės įrangos modelis tampa mažu *atraminių objektų modeliu* (atramine architektūra), o dalykinės srities koncepcijos yra išreiškiamos atskiromis formaliomis išraiškomis koncepcijų bibliotekoje. Kadangi programinė įranga (6 pav., Tarnybinė stotis) yra atraminio objektų modelio įgyvendinimas, ji visiškai nepriklauso nuo srities koncepcijų, t.y. įvedant į sistemą naujus koncepcinius modelius pačios sistemos keisti nereikia.

2.2.3. Medicinos ontologijos idėjos

Koncepcijų biblioteka, kuri yra formalus dalykinės srities koncepcijų apibrėžimas, yra ne kas kita kaip dalykinės srities – o šiuo atveju medicinos – ontologija. Tokios ontologijos sudarymas yra gan problemiškas, kadangi reikalauja rasti būdus, kaip sudėtingas realaus pasaulio koncepcijas formalizuoti tiek žmogui, tiek kompiuteriui suprantamu būdu, išvengiant viengubo sistemos modelio.

Sudarant medicinos srities ontologiją (formalizuojant medicinos žinias), GEHR projekte, buvo išskirti penki šios ontologijos lygmenys (medicinos dalykinės srities dalys) [2]:

1. **Principai**

   Pirmasis išskirtas lygis yra dalykinės srities kalbos ir principų ontologija. Medicinos srities principai yra anatomija, parazitologija, farmakologija, ir kt. Žinios apie esybes ir procesus yra bendrai priimti dalykinės srities faktai – tai dalykai, kurie yra teisingi visiems esybių egzemplioriams (pvz., žmogaus širdžiai) ar procesams (pvz., embrioniniam vystimuisi). Tokiu būdu pirmojo

lygio žinios yra nepriklausomos nuo jomis besinaudojančių vartotojų. Galima sakyti šis lygis perteikia žinias be jokio konkretaus požiūrio į jas.

2. **Turinys**

   Antrame ontologiniame lygyje žinios tampa labiau specifinės kiekvienam panaudojimo atvejui ir vartotojui. Šį lygį galima išskaidyti į aibę mažesnių lygmenų, priklausomų nuo skirtingų žinių panaudojimo kontekstų. Visi šie lygmenys atitinka tam tikrą srities žodyno elementų kompoziciją (panašiai, kaip molekulės yra sudarytos iš atomų kompozicijos). Taigi antrasis lygmuo yra principų (pirmojo lygmens elementų) kompozicija į tam tikras bazines turinio struktūras. Dažnai šis lygis gali būti padalintas į *visur esantį* ir *panaudojimo atveju paremtą* turinį. Visur esantis turinys – tai koncepcijos, kurias visi srities vartotojai naudoja ir supranta taip pat. Pavydžiai medicinoje gali būti kraujo spaudimas, kūno masė. Kiekviena iš šių koncepcijų aprašo tam tikrą pirmojo ontologinio lygio elementų naudojimą, kurie žodyne šiuo būdu gali būti visai nesusieti. Tai yra, pvz., kraujo spaudimas kaip klinikinis matavimas yra sistolinio ir diastolinio spaudimų kompozicija – dviejų pirmojo ontologinio lygmens elementų kompozicija antrajame lygmenyje. Kita antrojo lygmens dalis – panaudojimo atveju paremtas turinys – yra specifiniai procesai, kurie vyksta tam tikrais (panaudojimo) atvejais. Tokio tipo koncepcijas skirtingų sričių vartotojai supranta skirtingai.

3. **Struktūra**

   Struktūrinės koncepcijos (trečiasis ontologinis lygis) yra kuriamos srities vartotojų, tam kad suteikti prasmę antrojo lygio elementams (šių elementų poaibiams), kurie paprastai gali atrodyti nesusiję. Struktūros dažniausiai yra aukšto lygmens metodologinės idėjos ar procesai. Pavyzdžiui „į problemą orientuotas sveikatos įrašas“ yra plačiai naudojama struktūra, kuri susideda iš: paciento nusiskundimų (subjektyvios informacijos), specialisto tyrimų (objektyvios informacijos), problemos įvertinimo, gydymo plano ir t.t. Procesų pavyzdžiai gali būti bet kokie paciento tyrimai, pvz., širdies ir kraujagyslių tyrimas, akių tyrimas. Įvairių informacijos elementų organizacija į tam tikras struktūras tiek kompiuteriui, tiek žmogui palengvina sveikatos informacijos analizę.

4. **Saugykla**

   Iki šiol aprašyti lygmenys leidžia sukurti „struktūrizuotą turinį, išreikštą baziniais (žodyno ir kitais) elementais“. Ketvirtajame ontologiniame lygyje

reikia nustatyti loginę informacijos apie tam tikrą subjektą organizaciją. Klinikinei informacijai apie pacientą šis lygis atitinka stambią struktūrą, dažniausiai vadinamą „sveikatos istorija“, kurioje yra saugoma visa su pacientu susijusi informacija. Šios informacijos elementai turi būti prasmingi priklausomai nuo subjekto (paciento). Tai yra, elementai turi turėti savyje visą kontekstinę informaciją susijusią su elementų surinkimu ar sukūrimu, įrašiusio asmens identifikatorių, įrašymo datą, laiką ir kt. Medicinoje ketvirtojo lygio koncepcijos yra: šeimos istorija, dabar naudojami vaistai, nusiskundimų sąrašas, receptas, kontaktinė paciento informacija ir kt.

5. **Komunikacija**

   Paskutinis penktasis lygis aprašo koncepcijas, kurios yra susiję su tam tikros informacijos atrinkimu ir pritaikymu apsikeitimui su kitais vartotojais. Tipinės šio lygio koncepcijos yra: dokumentas, ataskaita, išrašas.

Toks ontologijos skaidymas yra patogus analizuojant sritį, tačiau norint praktiškai formaliai įvairiais lygiais aprašyti ontologiją, reika turėti formalizavimo taisykles.

2.2.4. Medicininiai archetipai

Medicinos dalykinės srities formalizacija pradedama [2] įvedant koncepcijos, kaip diskrečios esybės, sąvoką: *koncepcija yra individualiai identifikuojama dalykinės srities esybė*. Tada, ontologija yra suprantama kaip aibė (ar poaibis) visų srities esybių, išreikštų formaliai. Diskrečios koncepcijos yra svarbios tuo, kad operavimas jomis (apibrėžimas, peržiūrėjimas, reikšmės platinimas, naudojimas) yra įmanomas individualiai, tai yra (bendru atveju) nepriklausomai nuo visų likusių ontologijos koncepcijų. Jei ši savybė nebūtų tenkinama – operuoti koncepcijomis iš nebaigtos ontologijos būtų neįmanoma. Iš kitos pusės ši savybė nedraudžia tarp koncepcinių priklausomybių viename ontologiniame lygyje, tačiau gerai apibrėžtoje ontologijoje aukštesniuose ontologiniuose lygiuose (antrame ir kituose), ryšių tarp koncepcijų turėtų būti kuo mažiau. Taigi koncepcija yra aiškus dalykinės srities idėjos aprašymas, kuris yra individualiai identifikuojamas srities vartotojų ir yra naudojamas kaip savaime pakankama informacija informacijos apsikeitimo veiksme.

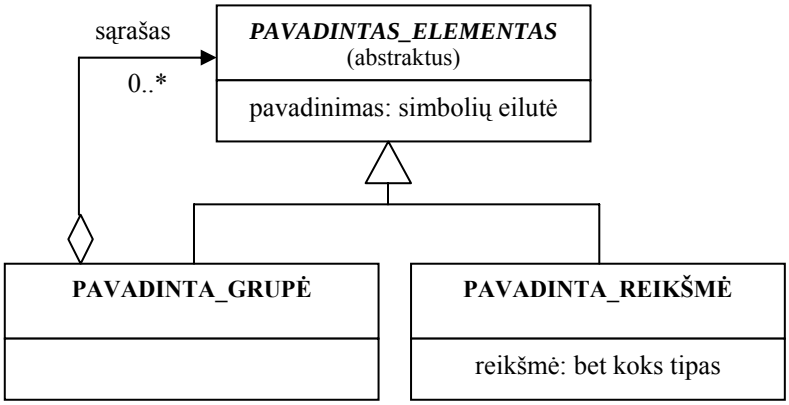
Pačiame pirmame ontologiniame (principų) lygmenyje koncepcijos dažnai yra išreiškiamos semantiniu tinklu (pvz., žodynu ir taisyklėmis). Šis žinių lygmuo gali būti suprantamas kaip aibė susietų faktų, klasifikacijų, apibrėžimų ir kt. Tokia žinių forma yra tinkama, jei šios žinios yra nekintančios, įmanoma prie jų prieiti ir jomis naudotis. Žinios aukštesniuose ontologinius lygiuose yra išreiškiamos per žemesnius lygius.

Aukštesnio lygio koncepcijos yra ne kas kita, kaip įvairios žemesnio lygio koncepcijų variacijos. Pavyzdžiui, koncepcija „asmuo“ atitinka didelį kiekį abstrakčios koncepcijos „žmogiška būtybė“ variantų, įskaitant visus galimus asmens bruožus – išvaizdos, elgesio ir panašiai. Iškyla klausimas, kokios yra ribos tarp koncepcijų, koks kintamumas yra priimtinas, kad viena koncepcija nevirstų kita. Ribas tarp koncepcijų galima nubrėžti įvedant apribojimais pagrįstą koncepcinį modelį. Kiekvieną koncepciją galima aprašyti aibe apribojančių taisyklių, kurios iš begalės koncepcijos variantų išskiria tam tikrą svarbią aibę koncepcijų egzempliorių, kurie atitinka pradinę koncepciją. Kitais žodžiais tariant, yra įmanoma kiekvienai srities koncepcijai sukurti apibrėžimą, sudarytą iš struktūrinių, tipinių, reikšminių ar elgesio apribojimų.

Apribojimais pagrįstas koncepcijos apibrėžimas yra pavadintas **archetipu** [2]. Archetipas – tai prototipinis modelis, apimantis eilę koncepcijos variantų. Tai yra formalus dalykinės srities koncepcijos specifikavimo būdas, kuriuo galima išreikšti dalykinės srities žinias tiek žmogui, tiek kompiuteriui gerai suprantamu būdu. Archetipas atitinka srities tam tikros koncepcijos eilę galimų kombinacijų, sudarytų iš atraminio objektų modelio elementų (6 pav.). Pagrindiniai archetipų naudojimo privalumai yra [8]:

- Archetipus kuria dalykinės srities specialistas, o ne informacinių technologijų inžinierius;
- Informacinės sistemos gali būti sukurtos remiantis tik atraminiu modeliu, o archetipai gali būti kuriami ir pildomi po sistemos įdiegimo. Tokios sistemos gali gyvuoti ilgiau;
- Jei sistemos gali keistis archetipais, yra įmanomas žinių lygmens bendradarbiavimas tarp jų;
- Remiantis archetipais yra įmanomos „protingos“ informacinių sistemų užklausos.

Kaip archetipo pavyzdį, galima pateikti kraujo spaudimo koncepciją. Natūralia kalba šis archetipas yra užrašomas taip: „kraujo spaudimas susideda iš dviejų matavimų – sistolinio ir diastolinio spaudimų“. Tai yra, kraujo spaudimo koncepcija susideda iš dviejų mažesnių koncepcijų, kurios turi savo kiekybinius įverčius. Tam, kad aprašyti archetipą, pirma turi būti apibrėžtas tam tikras visai sistemai bendras atraminis objektų modelis (7 pav.). Pavyzdžio paprastumui pateikiamas minimalus atraminis modelis, kurio užteks norimam archetipui aprašyti.



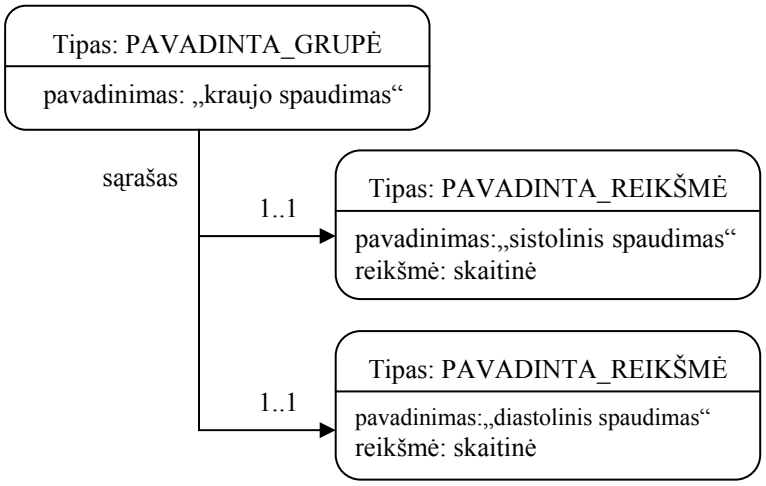
7 pav. Siauras atraminis objektų modelis dvigubo modelio projektavimo metodologijai [2].

Atraminis modelis susideda iš trijų klasių:

- **PAVADINTAS_ELEMENTAS**: tai abstrakti tėvinė klasė, nusakanti aibę srities objektų, kurie gali turėti pavadinimą (pvz., „pacientas“, „vaistas“, „širdis“, „organas“). Pavadinimas klasės egzemplioriams gali būti nustatytas per klasės kintamąjį *pavadinimas*.
- **PAVADINTA_REIKŠMĖ**: tai konkreti klasė, kuri paveldi visas klasės PAVADINTAS_ELEMENTAS savybes (tai yra ji gali turėti pavadinimą). Be to, ji įveda naują lauką – galimą reikšmę. Šios klasės pavydžiai gali būti įvairūs įverčiai, pvz.: iš dalykinės srities išraiškos „x = 30“ gauname: pavadinta_reikšmė(pavadinimas = „x“, reikšmė = „30“) ir t.t.
- **PAVADINTA_GRUPĖ**: tai konkreti klasė, kuri skirta kurti elementų masyvus. Masyvo elementas gali būti bet koks objektas, kurio tėvinė klasė yra PAVADINTAS_ELEMENTAS.

Galima dar pastebėti, kad šis siauras atraminis modelis yra suprojektuotas remiantis plačiai objektinio projektavimo praktikoje naudojamu kompozicijos programavimo šablonu [17]. Taip pat, šis mažas atraminis modelis neturi savyje visiškai jokios dalykinės srities informacijos, todėl gali būti panaudojamas bet kurio srityje.

Remiantis atraminiu modeliu galima kurti pavienius įverčių objektus – su įverčio pavadinimu ir įverčiu, bei objektų grupes su pavadinimu. Remiantis natūralia kalba aprašytu archetipu galima identifikuoti, kad koncepcija „kraujo spaudimas“ yra grupė su pavadinimu, turinti du elementus – dvi pavadintas reikšmes – „sistolinio spaudimas“ ir „diastolinio spaudimas“ (8 pav.).



8 pav. Archetipo „kraujo spaudimas” diagrama.

Pavyzdžio paprastumui buvo panaudoti tik esminiai kraujo spaudimo koncepcijos elementai. Praktikoje net ir toks paprastas archetipas kaip kraujo spaudimas yra išreiškiamas sudėtingiau. Yra įvedami galimi reikšmių rėžiai, matavimų su naujais pavadinimai galimybės, matavimo ypatumų identifikacija ir kt. Visi šie elementai tik dar labiau susiaurina kraujo spaudimo koncepcijos egzempliorių aibę ir įgalina lengviau ir lanksčiau ją valdyti.

3. ŽINIOMIS PAREMTA AUTOMATIZUOTA DUOMENŲ IŠGAVIMO METODIKA

Trumpai apžvelgus žiniomis pagrįstą automatinę teksto analizę (skyrius **3.1**), sudaroma automatizuota duomenų išgavimo iš liktinių sistemų metodika (skyrius **3.2**) bei aprašomas jai įgyvendinti sukurtas programinės įrangos prototipas (skyrius **3.3**).

3.1. Žiniomis pagrįsta automatinė teksto analizė

3.1.1. Duomenų išgavimas ir teksto analizė

Informacinės technologijos, o ypač internetas ir milžiniškos duomenų saugyklos labai paspartino duomenų išgavimo (*data extraction*) paradigmos vystimąsi. Automatinė duomenų analizė akivaizdžiai svarbi visoms mokslo šakoms. Norint panaudoti informaciją, esančią elektroninėje formoje kaip duomenis įvairiems automatiniams taikymams, informacija turi būti pateikta tam tikroje struktūroje. Tačiau nemažai informacijos yra saugoma ir pateikiama laisvo teksto, arba mažai struktūrizuotoje formoje. Norint panaudoti tokią informaciją, tenka atlikinėti reikiamų duomenų išgavimą.

Duomenų išgavimas yra ypač aktualus medicinai. Jau eilę metų elektroninėje formoje kaupti įvairūs medicininės praktikos duomenys ir rezultatai, dėl elektroninių sveikatos istorijų standarto nebuvimo, yra tiesiog „kompiuterinis popierius“ – informacija yra elektroninėje formoje, tačiau automatiniams algoritmams ar apsikeitimui ji yra visiškai netinkama. Tam, kad atlikti medicininius statistinius tyrimus, visus turimus duomenis iš laisvo teksto ar mažai struktūrizuotos formos reikia perkelti į tam tikrą griežtą struktūrą. Duomenų išgavimas medicinoje yra tiriamas (ir iš dalies taikomas praktiškai) tiek analizuojant natūralios kalbos tekstus išgaunant žinių lygmens duomenis [1, 4, 21], tiek ir atliekant mažai struktūrizuotų duomenų perkėlimą į griežtesnės struktūros formą [3, 16] ir kt.

Kaip minėta darbo tiksle, bus išbandoma tam tikros siauros srities (laboratorinių rezultatų – klinikinio kraujo tyrimo) medicininių dokumentų automatizuota analizė, tam kad šiuos dokumentus pervesti iš „kompiuterinio popieriaus“ į griežtesnės struktūros informacinę sistemą. Paprasčiausias automatinės analizės, pervedančios informacinę sistemą iš vienos struktūros į kitą, būdas yra tiesioginis struktūrų laukų atvaizdavimas vienas į kitą. Konkrečiu atveju ir apskritai medicinos srityje, pasižyminčioje labai skirtingomis ir įvairiomis dokumentų struktūromis, tiesioginis laukų atvaizdavimas tinka tik dalinai. Todėl šiame darbe, priedo prie tiesioginio laukų atvaizdavimo, bus atliekama ir sudėtingesnė duomenų analizė, paremta žiniomis.

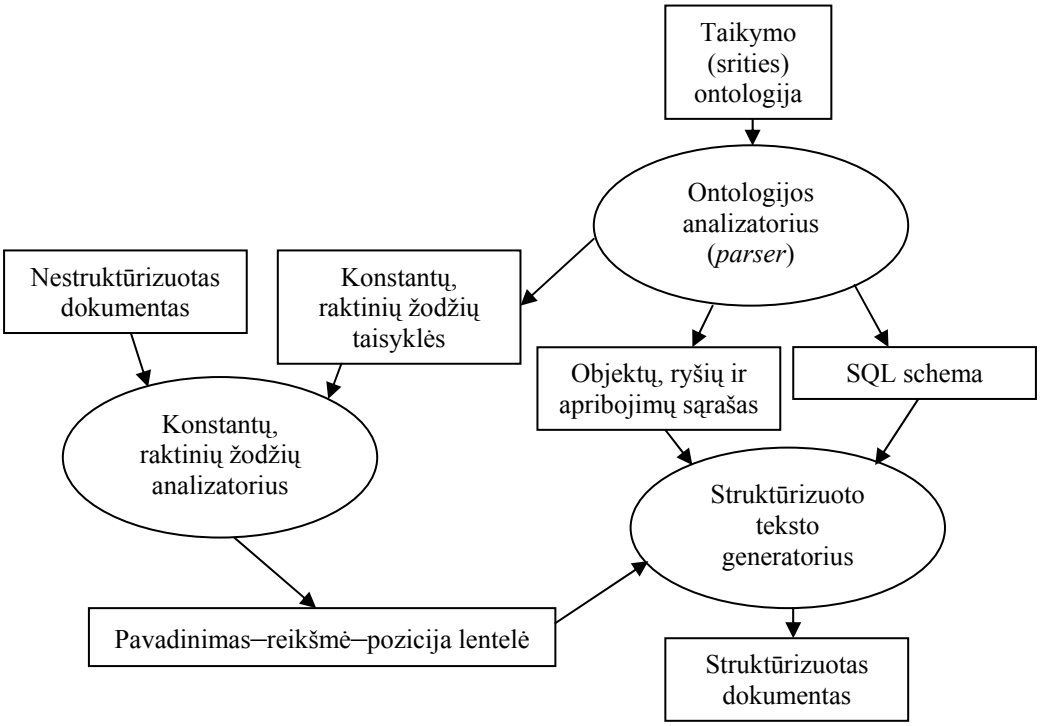
3.1.2. Žiniomis paremta duomenų išgavimo metodika

Duomenų išgavimas iš nestruktūrizuotų ar mažai struktūrizuotų dokumentų gali būti atliktas sukuriant įvairius ryšius tarp duomenų informacinio turinio. Toks ryšių sukūrimas perveda nestruktūrizuotą ar mažai struktūrizuotą dokumentą į griežtesnės struktūros formą. Tam, kad atpažinti dokumentų turinio semantiką reikia turėti aprašytą tiriamos dalykinės srities žinių modelį arba ontologiją. Ontologija formaliai yra apibrėžiama kaip tam tikros srities sąvokų visumos specifikavimas išreikštu pavidalu [26]. Siauresne prasme, tai yra žinių bazė sutartais terminais aprašanti faktus, kurie tiriamos srities kontekste yra laikomi teisingais. Remiantis tokiu semantiniu duomenų modeliu (ontologija), galima kurti srities objektų egzempliorius, susiejant juos įvairiomis srities taisyklėmis, kas ir yra nauja srities duomenų struktūros forma. [12] pateikta apibendrinta ontologinio duomenų išgavimo ir pervedimo į griežtesnę struktūrą metodika, bei eksperimento rezultatai. Eksperimentas atliktas su labai siauros srities nestruktūrizuotais dokumentai, kurie turi savyje daug informacijos ir gali būti aprašyti gan siaura ontologija (mažu ontologiniu modeliu). Gauti aukšti informacijos išgavimo rezultatai (išgauta apie 90% visos buvusios svarbios informacijos). Principinė metodikos schema pateikta 9 paveiksle.

Metodika naudoja praturtintą semantinį duomenų modelį, kuris apibrėžia ontologiją, aprašančią tokį požiūrį į dalykinę sritį (dalykinės srities vaizdą), kokio reikia vartotojui. Semantinis duomenų modelis leidžia sukurti ontologinio modelio egzempliorių, kuris susideda iš aibės objektų, aibės ryšių tarp šių objektų ir aibės apribojimų objektams. Be to, semantinio modelio papildymas leidžia nustatyti duomenų pateikimą ir galimus konteksto raktinius žodžius kiekvienam ontologijos objektui.

Metodikos įėjimo duomenys yra srities ontologija, bei nestruktūrizuotas tekstas, išėjimo – struktūrizuoti dokumentai duomenų bazėje. Vienintelis žingsnis šioje metodikoje reikalaujantis žmogaus įsikišimo yra taikymo ontologijos sudarymas. Kai ontologija yra sudaryta, ji gali būti naudojama bet kokiems nestruktūrizuotiems dokumentams iš įvairių šaltinių, reikalaujant tik kad dokumentai atitiktų duotąją ontologiją.

Pirmas metodikos žingsnis – ontologijos analizatoriaus darbas. Duotajai taikymo ontologijai analizatorius sukuria būsimos struktūrizuotų dokumentų duomenų bazės schemą (SQL sakinių rinkinį). Objektų pavadinimai ontologijoje tampa sugeneruotų SQL lentelių arba lentelių atributų pavadinimais. Analizatorius iš ontologijos taip pat išgauna objektų, ryšių ir apribojimų sąrašą atvaizdavimui tarp ontologijos ir duomenų bazės schemos. Šį sąrašą naudos struktūrizuoto teksto generatorius. Galiausiai ontologijos analizatorius sugeneruoja konstantų ir raktinių žodžių atpažinimo taisykles.



9 pav. Ontologija paremtos duomenų išgavimo ir pervedimo į griežtesnę struktūrą iš nestruktūrizuotų dokumentų metodikos principinė schema [12]. Stačiakampiais pažymėti duomenys, ovalais – procesai.

Ontologijos analizatorius yra kviečiamas tik vieną kartą. Toliau, paeiliui kviečiami konstantų ir raktinių žodžių analizatorius ir struktūrizuoto teksto generatorius. Šie procesai kviečiami kiekvienam nestruktūrizuotam dokumentui. Konstantų ir raktinių žodžių analizatorius pritaiko ontologijos analizatoriaus sugeneruotas taisykles (kurios yra išreikštos reguliariųjų išraiškų forma) ir generuoja „pavadinimas–reikšmė“ poras, kurios yra išsaugomos „Pavadinimas–reikšmė–pozicija“ sąraše (9 pav.). Struktūrizuoto teksto generatorius, panaudodamas objektų, ryšių ir apribojimų sąrašą kaip atvaizdavimą tarp ontologijos ir duomenų bazės schemos, užpildo duomenų bazę atributų reikšmėmis ir ryšiais.

Gauti eksperimento rezultatai yra geri, tačiau tiesioginis šios metodikos taikomumas medicinos srityje yra abejotinas, kadangi, kaip sąlyga eksperimentui ir geriems jo rezultatams, nurodomas dokumentų tipas – dokumentai, turintys savyje daug informacijos ir aprašomi gan siauru ontologiniu modeliu. Nors medicinos dokumentus (konkrečiai – pacientų įrašus, ligos istorijas ir kt.) galima klasifikuoti kaip daug savyje turinčius informacijos dokumentus, tačiau jie apima labai plačią medicininę ontologiją. Tuo labiau, vieninga visą apimanti medicininė ontologija kol kas neegzistuoja.

Norint praktiškai panaudoti šią metodiką medicinos informatikoje pirma turi būti galimybė mažai struktūrizuotus ar nestruktūrizuotus medicinos dokumentus klasifikuoti pagal jų turinį ta prasme, kad tam tikros klasės dokumentai apimtų tam tikrą, sąlyginai siaurą medicininės ontologijos dalį. Tokiu atveju, išskiriant *aktyvią ontologijos dalį*, būtų galima atlikti panašų ontologija paremtą duomenų išgavimą iš mažai struktūrizuotų dokumentų. Iš kitos pusės, turint tam tikrus nestruktūrizuotus dokumentus, galima tik jų apimamai sričiai

aprašyti dalines medicininės ontologijas ir taikyti jas šių dokumentų vertimui į griežtesnę struktūrą. Pastaruoju atveju dalinės medicininės ontologijos turėtų būti aprašomas tam tikru standartiniu būdu – kad jas būtų galima sujungti į vieną didesnę ontologiją ir kad jos būtų suderinamos su kitose sistemose egzistuojančiomis ontologijomis.

3.2. Automatizuota medicininių dokumentų analizės metodika

Remiantis atlikta analize, sudaroma medicininių dokumentų analizės metodika. Metodikos tikslas – automatizuota liktinių medicinos dokumentų migracija į griežtesnės struktūros elektroninių sveikatos istorijų sistemą. Šia metodika esami medicinos dokumentai (atskiri tekstiniai dokumentai, nestruktūrizuoti arba mažai struktūrizuoti duomenų bazių įrašai ir t.t.) analizuojami automatinių teksto analizės algoritmų, kurie išgauna aktualius duomenis remiantis griežtesnės struktūros ESI sistemos saugoma dalykinės srities semantika. Nagrinėjamuoju atveju ši semantika yra užduodama archetipais, bei juose manipuliuojamomis sąvokomis iš žodynų. Kadangi medicinos informatikoje automatinis apdorojimas yra laikomas mažiausiai patikimu, tai ši metodika yra ***automatizuota*** archetipais pagrįsta liktinių duomenų išgavimo metodika, nes visą jos procesą turi prižiūrėti medicinos ekspertas.

Metodika susideda iš dviejų žingsnių:

3.2.1. Analizė

Tai pirmasis metodikos žingsnis, kurio pagrindinis tikslas – pradinių duomenų paruošimas liktinės sistemos migracijos procesui.

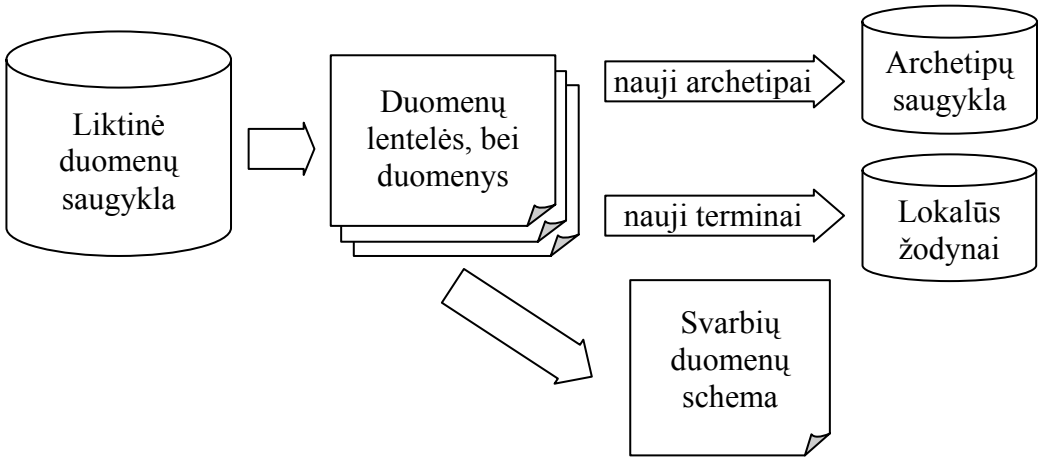
3.2.1.1. Liktinės sistemos analizė

Turima liktinė sistema (duomenų bazė, atskirų bylų sankaupos ir kt.) yra analizuojama informatikos inžinieriaus ir medicinos specialisto. Pirma ištiriamos galimybės liktinę sistemą tiesiogiai perkelti į naująją ESI sistemą. Tai galima atlikti jei liktinė sistema neturi laisvo teksto ar mažai struktūrizuotų duomenų laukų su įvairiais duomenimis. Tokiu atveju migracijos proceso metu duomenų laukai nebūtų (arba būtų minimaliai) analizuojami teksto analizatorių ir perkeliamų duomenų patikra žymiai supaprastėtų.

Liktinės sistemos analizės metu nagrinėjamos turimų duomenų struktūros, išskiriami tie laukai, kurie yra svarbūs ir turi būti perkelti į naująją sistemą (jie formaliai aprašomi **svarbių duomenų schemoje**, 10 pav.). Svarbių duomenų schema yra transformacijos dokumentas tarp liktinių duomenų struktūrų ir tarpinių dokumentų, kuriais manipuliuojama migracijos procese.

Priimant prielaidą, kad liktinėje sistemoje yra tam tikrų duomenų, kurie nėra semantiškai aprašyti naujoje sistemoje (archetipais paremta sistema praktiškai niekada nėra pilna semantine prasme), tiriant duomenis liktinėje sistemoje turi būti identifikuojamos naujos sąvokos, kurių nėra naujos sistemos žodynuose ir sudaromi nauji archetipai naujoms semantinėms struktūroms aprašyti.

Koncepcinė šio žingsnio schema pateikta 10 paveiksle.

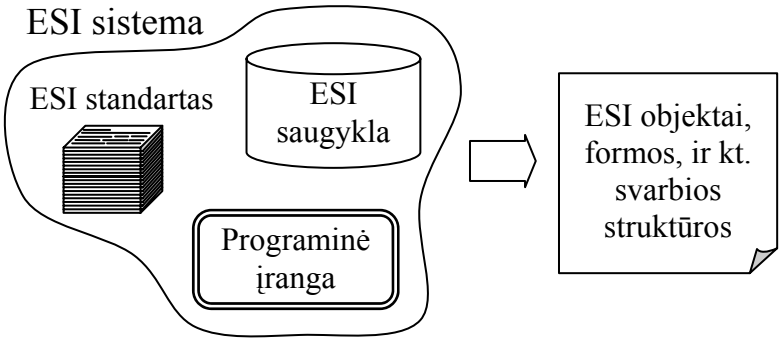


10 pav. Koncepcinė liktinės sistemos analizės schema. Pagrindiniai informacijos srautai ir objektai.

3.2.1.2. ESI sistemos (standarto) analizė

Analizuojama turima ESI sistema (arba standartas), identifikuojant objektus, formas ir kt. svarbias ESI struktūras, į kurias bus rašomi iš liktinės sistemos išgaunami duomenys (11 pav.). Bendruoju (paprasčiausiu) atveju šios struktūros – tai tekstiniai dokumentai, kurių semantinę prasmę aprašo archetipai.

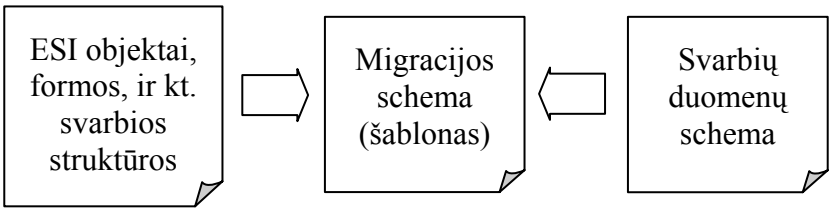
Praktiškai šis žingsnis gali būti atliekamas tik vieną kartą. Jo rezultatas gali būti pakartotinai naudojamas ir su kitomis liktinėmis sistemomis, kurias reikia perkelti į šią ESI sistemą (standartą).



11 pav. Koncepcinė ESI sistemos (standarto) analizės schema.

3.2.1.3. Migracijos schemos sudarymas

Analizės pirmojo ir antrojo žingsnių (**3.2.1.1 Liktinės sistemos analizė**, **3.2.1.2 ESI sistemos (standarto) analizė**) rezultatų sujungimas – **migracijos schemos** (**šablono**) iš liktinės sistemos į naująją ESI sistemą sudarymas. Ši schema aprašo kokie laukai (duomenys) iš liktinės sistemos yra perkeliami į tam tikrus ESI sistemos laukus ir kaip tas perkėlimas yra atliekama – tiesiogiai atvaizduojant ar analizuojant tekstą, jei analizuojant – kokie yra analizės žingsniai, kokių duomenų ieškoma ir t.t. (12 pav.).



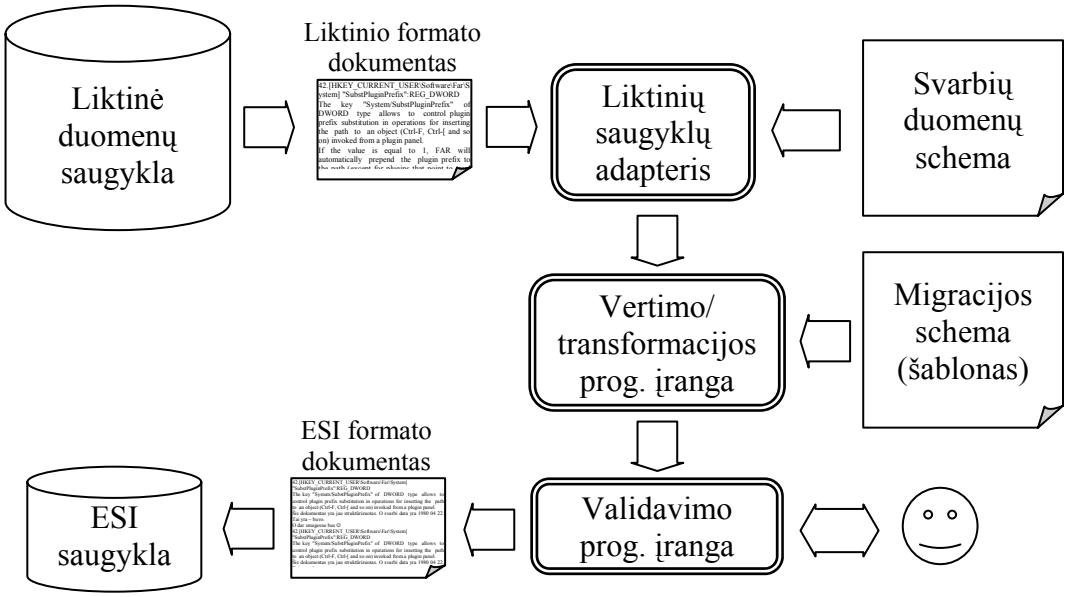
12 pav. Koncepcinė migracijos schemos (šablono) sudarymo schema.

3.2.2. Migracija

Antrasis automatizuotos archetipais pagrįstos liktinių duomenų išgavimo metodikos žingsnis yra pats duomenų išgavimo procesas. Šiuo procesu liktinių duomenų egzemplioriai iš liktinės saugyklos yra perkeliami į ESI sistemą, atliekant tam tikras transformacijas pagal migracijos schemą, gautą analizės žingsnyje. Transformuoti duomenys yra pateikiami medicinos ekspertui, kad šis juos validuotų. Tik validuoti duomenys gali būti perkeliami į naująją ESI sistemą. Taip pat šioje sistemoje turėtų būti saugomi ir tam tikri liktinių duomenų transformacijos protokolai, kad esant reikalui būtų galima sužinoti iš kur ir kaip tam tikri duomenys atsidūrė ESI sistemoje.

Migracijos procesui vykdyti reikalinga speciali programinė įranga (13 pav.):

1. **Liktinių saugyklų adapteris**. Programinė įranga, kuri iš bet kokios duomenų saugyklos (atskirų bylų, duomenų bazių ir kt.) gali per apibendrintą sąsają, perduoti liktinių duomenų (tarpinių dokumentų formate) srautą, gaunamą transformuojant liktinius duomenis pagal *svarbių duomenų schemą*.
2. **Vertimo/transformacijos programinės įranga**, kuri skirta liktiniams duomenims tarpinių dokumentų formate analizuoti ir transformuoti į naujos ESI sistemos duomenų struktūras. Analizė ir transformacija yra atliekamos pagal *migracijos šabloną*. Liktinių duomenų laukai, kurie nėra atvaizduojami tiesiogiai, yra analizuojami automatiniu teksto analizės algoritmu, paremtu žiniomis iš ESI sistemos archetipų ir terminų saugyklų. Ši programinės įrangos dalis yra sudėtingiausia visoje migracijos sistemoje.
3. **Validavimo programinė įranga** skirta medicinos ekspertui, prižiūrinčiam migracijos procesą. Ši įranga turi leisti greitai peržvelgti vieno dokumento liktinius ir išgautuosius duomenis, bei kokiomis taisyklėmis remiantis šie duomenys buvo išgauti (tokia informacija turi būti saugoma liktinių duomenų transformacijos protokole). Programinės įrangos vartotojo sąsaja turi būti kiek galima paprastesnė, kad palengvintų ir pagreitintų eksperto darbą.



13 pav. Koncepcinė migracijos proceso schema. Išskirtos (dvigubi stačiakampiai suapvalintais kampais) pagrindinės migracijos proceso programinės įrangos dalys.

3.3. Sukurta programinė įranga

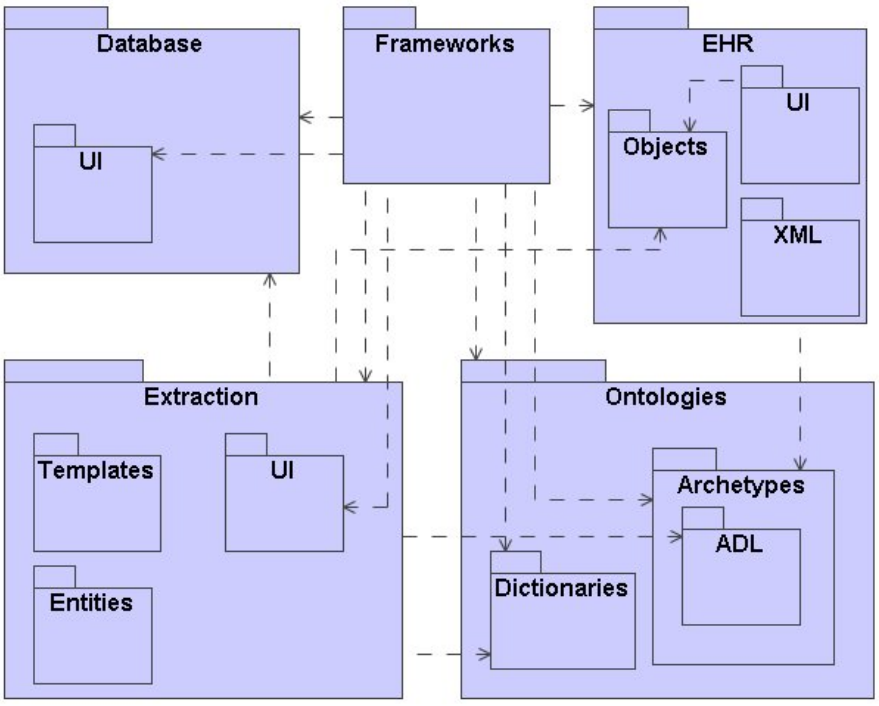
3.3.1. Įvadas

Remiantis sudaryta metodika, eksperimentui atlikti reikalinga programinė įranga, kuri susideda iš trijų dalių – liktinių duomenų adapterio, liktinių duomenų transformatoriaus ir transformavimo validatoriaus. Darbo metu buvo realizuotos prototipinės šios programinės įrangos versijos.

Visa programinė įranga buvo rašoma Microsoft C# programavimo kalba, skirta .NET 1.1 platformai. Visi tarpiniai ir galutiniai dokumentai yra saugomi (bei jais manipuliuojama) pasitelkiant XML [15] meta kalbą.

3.3.2. Sistemos paketai

Visas programinis kodas yra suskirstytas į paketus. Šakninis paketas yra *kut.ibme*, išsišifruojantis kaip „***K****aunas* ***u****niversity of* ***t****echnology,* ***I****nstitute of* ***Bio******m******e****dical* ***E****ngineering*“ (Kauno technologijos universitetas, Biomedicininės inžinerijos institutas).



14 pav. Pagrindiniai sistemos paketai ir priklausomybės tarp jų.

Pagrindinių sistemos paketų diagrama pateikta 14 paveiksle. Trumpi šių paketų aprašymai pateikti 3 lentelėje.

3 lentelė. Pagrindinių prototipinės sistemos paketų aprašymas

Paketo pavadinimas	Paskirtis
Frameworks	Valdančiųjų klasių paketas. Šiame pakete yra realizuotos aukščiausios valdančiosios klasės.
Database	Duomenų saugyklų sąsajų paketas. Šiame pakete realizuotos duomenų adapterio liktinėms sistemoms sąsajos ir klasės.
Database.UI	Grafinės vartotojo sąsajos formos, skirtos darbui su duomenų saugyklomis.
Extraction	Duomenų išgavimo (tiesioginio atvaizdavimo ir teksto analizės) paketas. Šiame pakete aprašyti ir įgyvendinti baziniai (paprasti) teksto analizės algoritmai ir pagalbinės priemonės.
Extraction.UI	Duomenų išgavimo grafinės vartotojo sąsajos klasių paketas.
Extraction.Templates	Duomenų išgavimo procesą aprašančių šablonų paketas.
Extraction.Entities	Duomenų išgavimo proceso metu konstruojamų išgautos informacijos medžių objektų paketas.
Ontologies	Dalykinės srities ontologijos aprašymui skirtas paketas.
Ontologies.Archetypes	Prototipinės archetipų realizacijos paketas.
Ontologies.Archetypes.ADL	Prototipinių ADL kalba pagrįstų archetipų realizacijos paketas.
Ontologies.Dictionaries	Prototipinių žodynų realizacijos paketas.
EHR	Paprasčiausių elektroninių sveikatos istorijų realizacijos paketas (įgyvendintas minimaliai).
EHR.Objects	Paprasčiausių elektroninių sveikatos istorijų bazinių objektų realizacijos paketas (įgyvendintas minimaliai).
EHR.XML	Paprasčiausių elektroninių sveikatos istorijų saugojimo XML formate įrankiai.
EHR.UI	Paprasčiausių elektroninių sveikatos istorijų peržiūros realizacijos paketas (įgyvendintas minimaliai).

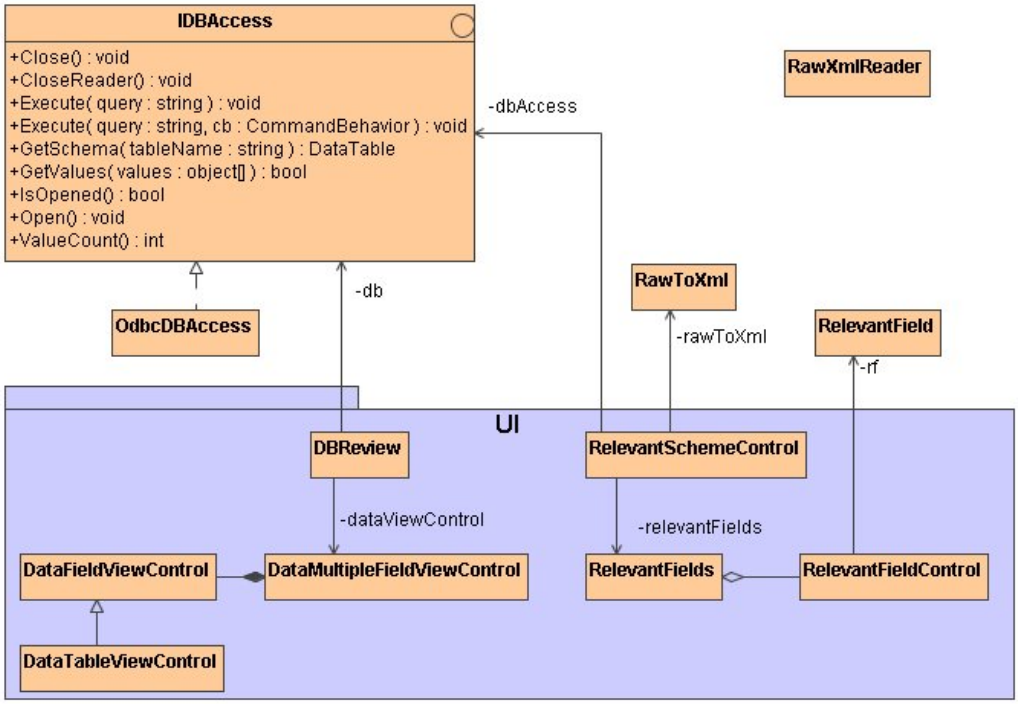
3.3.3. Liktinių duomenų adapteris

Liktinių duomenų adapterio programinė įranga yra įgyvendinta *kut.ibme.Database* pakete. Šio paketo klasių diagrama pateikta 15 paveiksle. Pagrindinės šio paketo klasės liktinių duomenų skaitymui iš liktinės sistemos – tai sąsaja *IDBAccess* ir *OdbcDBAccess*. *IDBAccess* yra apibendrinta liktinių duomenų prieigos sąsaja, kurios užtenka, kad duomenų

migracijos sistema galėtų skaityti liktinius duomenis. Šios sąsajos pagalba liktinių duomenų saugykla gali būti bet kas – nuo įvairiausių DBVS, iki bylų sankaupų ir pan. Kiekvienai skirtingai liktinių duomenų saugyklai tereikia įgyvendinti šią sąsają ir sistema dirbs su tais duomenimis. Duotųjų liktinių duomenų atveju (Microsoft FoxPro 2.6 DBVS) buvo įgyvendinta *OdbcDBAccess* klasė, kuri realizuoja *IDBAccess* ir leidžia duomenis skaityti iš ODBC (*Open Data Base Connectivity*) tipo valdiklių.

Tokiu atveju, jei duomenų bazė yra prieinama tik laikinai (nėra galimybių ja naudotis per visą duomenų migracijos procesą) buvo sukurtos klasės, skirtos liktinių duomenų perkėlimo į XML meta kalba aprašytas bylas – *RawToXml*, bei *RawXmlReader*. Šių klasių pagalba pasirinktos duomenų bazės lentelės visi įrašai su išrinktais laukais ir *svarbių duomenų schema* (**3.2.1.1 Liktinės sistemos analizė**) yra įrašomi į diską XML bylų pavidale. Pavyzdys, kaip atrodė eksperimente *svarbių duomenų schema* XML formate, pateiktas skyriuje **9.2.4.1 Svarbių duomenų schema**. Duomenų bazės lentelės „anketa“ egzemplioriaus pavyzdys XML formate pateiktas skyriuje **9.2.4.2 Lentelės „anketa“ egzempliorius (XML)**.

Kitos *kut.ibme.Database* klasės (*DBReview*, *RelevantSchemeControl* ir kt.) – grafinės vartotojo sąsajos elementai, formos, kuriuose vartotojas mato liktinių duomenų bazės lenteles, jų duomenis, gali išsirinkti tam tikrą lentelę, bei jai sudaryti *svarbių duomenų schemą.*



15 pav. kut.ibme.Database paketo klasės ir ryšiai tarp jų.

3.3.4. Liktinių duomenų transformatorius

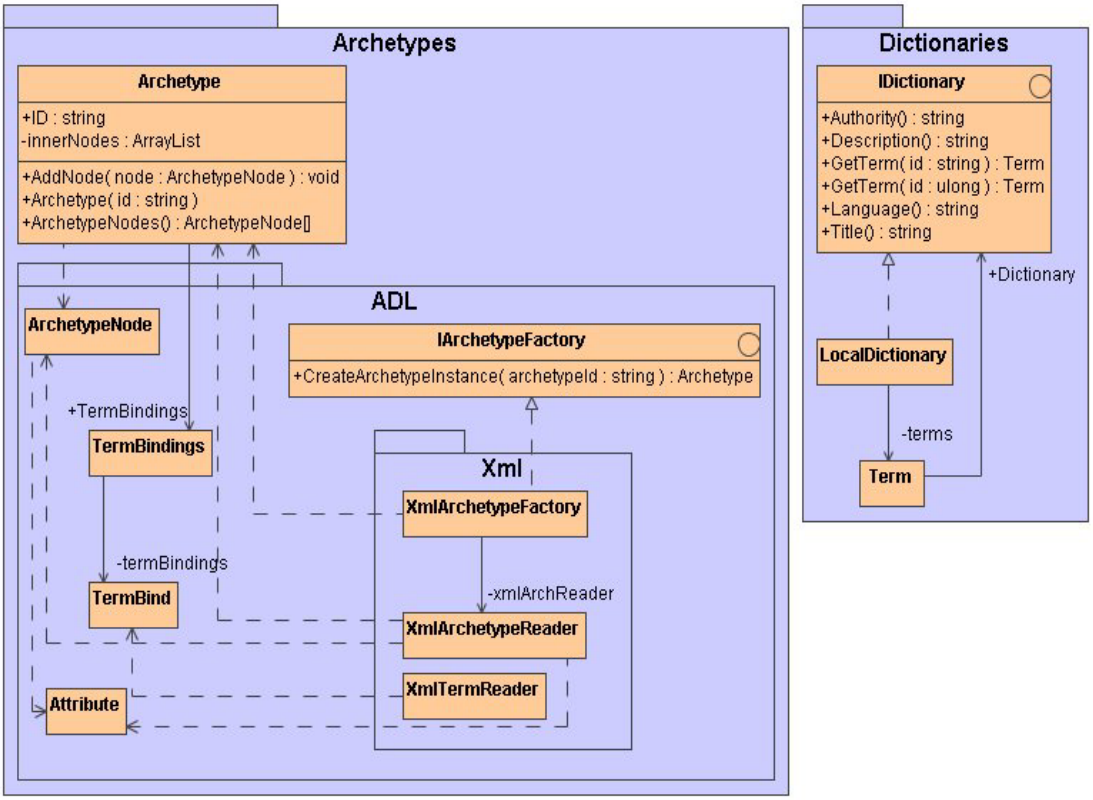
Liktinių duomenų transformatoriaus programinė įranga yra įgyvendinta *kut.ibme.Extraction* pakete. Šio skyriaus poskyriuose trumpai aprašyti šio paketo vidiniai paketai, klasės ir įgyvendinti teksto analizės algoritmai.

Bendru atveju pradinė informacija tekstiniu pavidalu patenka į šio paketo klases (*Converter*, *IStringExtractAnalyzer* realizacijas) ir yra analizuojama, bandant išgauti iš jos tam tikrus duomenis, tam tikru būdu, kaip yra aprašyta išgavimo (transformacijos) šablone (poskyris **3.3.4.3 Informacijos išgavimo šablonai**). Duomenų išgavimas remiasi ESI sistemoje esančiomis žiniomis, aprašytomis medicininiais archetipais, kurie yra parašyti ADL kalba ir saugomi XML tipo bylose. Plačiau apie ADL kalbą rašoma skyriuje **2.1.3.8. Archetipų aprašymo kalba**. Archetipų realizaciją eksperimento metu aprašyta skyriuje **3.3.4.1 Archetipų aprašymas**. Duomenų išgavimo metu yra konstruojamas specialus išgautų duomenų medis, kurio objektai ir struktūra aprašyti **3.3.4.2 Išgaunamos informacijos duomenų struktūra** skyriuje. Šis medis yra duomenų transformacijos protokolas, minimas skyriuje **3.2.2 Migracija**.

3.3.4.1. Archetipų aprašymas

Eksperimento meto archetipai sudaryti remiantis Europos preliminariame ESI standarte EN 13606 pateiktu ADL kalbos aprašymu. (**2.1.3.8. Archetipų aprašymo kalba**), bei saugomi XML tipo bylose. Šie archetipai yra pateikti skyriuje **9.2.3 Sudaryti ir naudoti archetipai**.

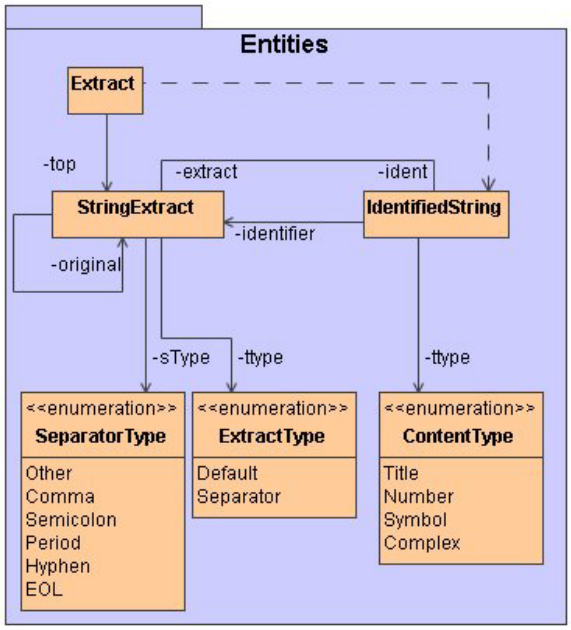
Kadangi ADL kalba yra gan plati, bei kol kas yra kuriama ir nėra galutinio standarto, šiame prototipe nebuvo įgyvendintas pilnas (visos) šios kalbos palaikymas (interpretatorius). Buvo apsiribota tik minimalia šios kalbos interpretatoriaus realizacija ir minimalia struktūra, kuri gali aprašyti ir perskaityti ADL archetipus XML formate, su primityviais atributais ir archetipų kompozicijomis. Archetipus įgyvendinančio paketo diagrama pateikta 16 paveiksle. *Archetype* iš *kut.ibme.Ontologies.Archetypes* paketo yra pagrindinis vieną archetipą nusakantis objektas. *kut.ibme.Ontologies.Archetypes.ADL* paketo *ArchetypeNode* ir *Attribute* yra pagrindiniai archetipo sudedamieji objektai. *kut.ibme.Ontologies.Archetypes.ADL.Xml* paketo klasės skirtos archetipų išsaugojimui, ir užkrovimui iš XML tipo bylų. *kut.ibme.Ontologies.Dictionaries* pakete yra realizuotos klasės, skirtos terminologijų sudarymui. Archetipai per `ontology` dalį (5 pav., **2.1.3.8 Archetipų aprašymo kalba**) atvaizduoja savo lokalius terminus į globalias terminologijas.



16 pav. Prototipe įgyvendinto archetipų interpretatoriaus klasių diagrama (kut.ibme.Ontologies paketas).

3.3.4.2. Išgaunamos informacijos duomenų struktūra

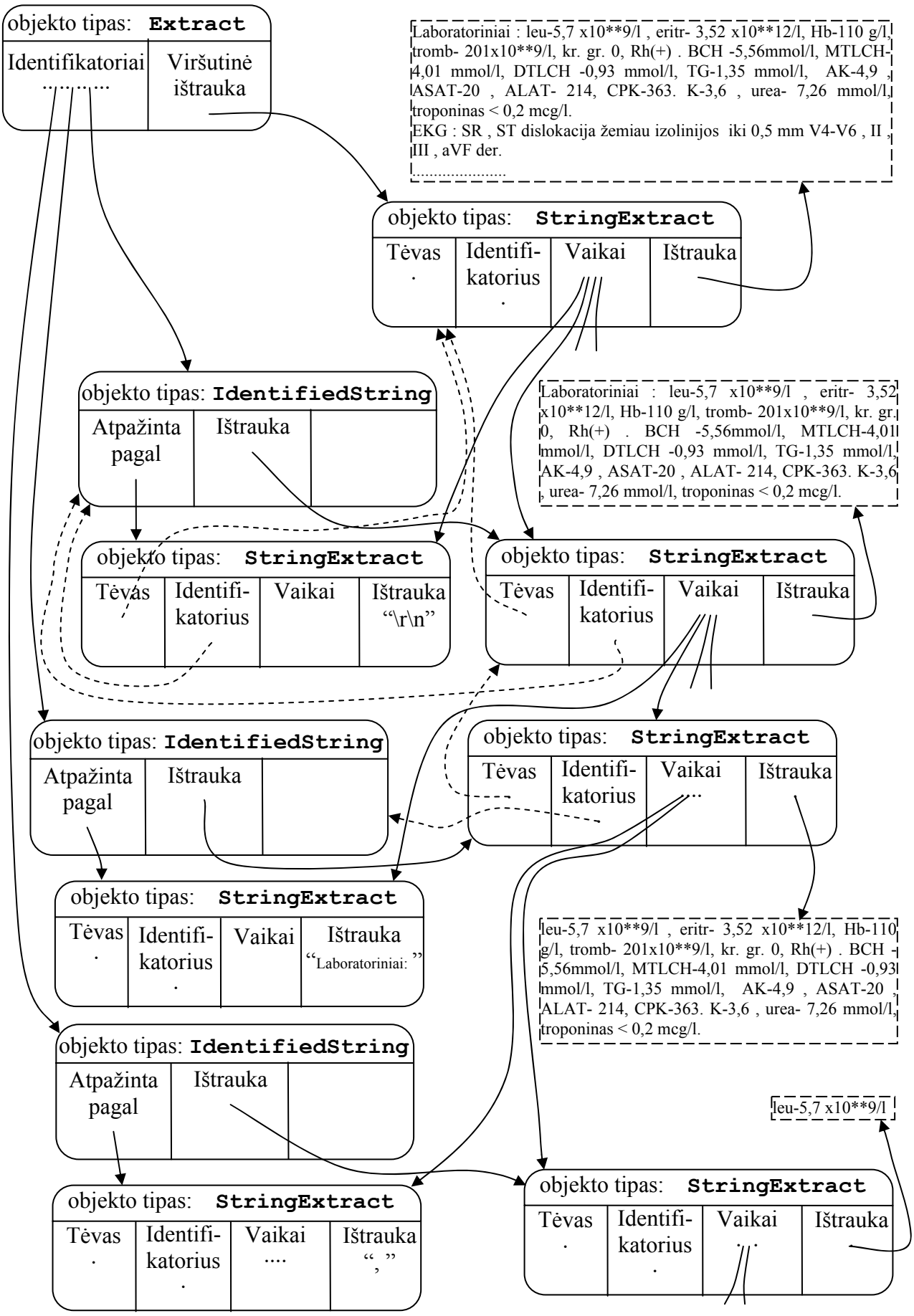
Kaip minima metodikoje, išgaunant informaciją iš tekstinių laukų yra svarbus atsekamumas, t.y. duomenų išgavimo protokolas, kuriame atspindėtų iš kur, kokie ir pagal ką buvo išgauti tam tikri duomenys. Toks protokolas yra įgyvendintas duomenų išgavimo metu konstruojant specialią duomenų struktūrą, kuri yra panaši į dvigubai sujungtą medį. Į šį medį įeina trijų rūšių objektai – *Extract*, *StringExtract*, ir *Identifier* (17 pav.) iš *kut.ibme.Extracion.Entities* paketo.



17 pav. Išgaunamos informacijos atsekamumo duomenų struktūra.

Išgaunamos informacijos duomenų medis turi vieną šakninį objektą – *Extract*. Šis objektas sukuriamas kiekvienam tekstiniam laukui. Toliau analizuojant tekstą ir jį skaidant į dalis, kiekvienai daliai yra sukuriamas *StringExtract* („tekstinė ištrauka“) objektas, bei pagal ką šis objektas buvo identifikuotas – *IdentifiedString* („teksto identifikatorius“). Kiekvienas tekstinės ištraukos objektas (*StringExtract*) turi savo tėvinę tekstinę ištrauką – vidinis kintamasis *original* – (jos neturi tik viršutinė ištrauka, kurios tėvas – *Extract*), bei savo identifikatorių. Vienam laukui suformuotas tekstinių ištraukų medis yra išsaugomas XML formato byloje (*kut.ibme.Extracion.ExtractSerializer* klasės pagalba).

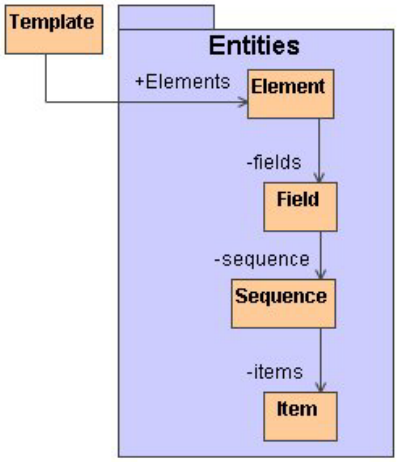
Pavyzdys, kaip išgaunamosios informacijos duomenų medis atrodo realiai, pateiktas 18 paveiksle, kuriame yra pavaizduota vieno iš lentelės „anketa“ lauko „diag_tyr“ turinio atlikta analizė. Išsaugoto XML tipo byloje medžio pavyzdys pateiktas skyriuje **9.2.5 Išgautų duomenų medis (XML)**.



18 pav. Išgaunamos informacijos duomenų medžio pavyzdys. Medžio struktūra išanalizuotam tekstui. Žemesniuose lygiuose dėl diagramos aiškumo nėra pavaizduoti atgaliniai (punktyriniai) ryšiai.

3.3.4.3. Informacijos išgavimo šablonai

Tekstinė informacija duotuose laukuose gali būti labai įvairi. Tačiau įgyvendinus keletą paprastų teksto analizės algoritmų (pvz. skaidymas pagal tam tikrus skyriklius, atpažinimas tam tikrų raktažodžių ir pan.) galima juos efektyviai kombinuoti ir gauti patogius įrankius duomenims išgauti iš turimų duomenų. Tai yra, sudarinėti šių algoritmų valdymo šablonus, pagal kuriuos vieni ar kiti algoritmai tam tikra tvarka yra taikomi tekstui, ar jo dalims. Tokie (pavyzdiniai) informacijos išgavimo šablonai yra įgyvendinti *kut.ibme.Extraction.Templates* pakete (19 pav.). Eksperimento metu naudotas vienas šablonas. Jis su paaiškinimu pateiktas skyriuje **9.1.1.2 Migracijos schemos sudarymas**. Taip pat, kad teksto analizės algoritmai dirbtų efektyviai, tam tikriems laukams galima apriboti žinių aibę – tai yra, apriboti archetipus, pagal kuriuos atliekama teksto analizė. Tokie apribojimai įvedami nurodant norimą archetipą *Element* ir *Item* klasėse.



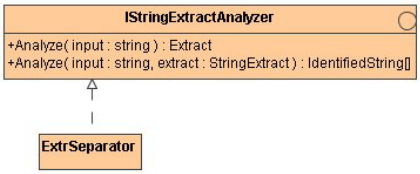
19 pav. Informacijos išgavimo šablonų klasės.

Šablonų objektai ir jų aprašymas:

- *Template* – pagrindinis objektas, kuris aprašomas kiekvienai liktinės duomenų bazės lentelei.
- *Element* – pirminio grupavimo elementas. Jame galima grupuoti vieną ar kelis laukus iš duomenų bazės lentelės. Taip pat, galima apriboti analizę nurodžius norimą archetipą.
- *Field* – objektas, aprašantis vieno lauko analizę.
- *Sequence* – antrinis grupavimo elementas. Jame grupuojami veiksmai, kurie turi būti atliekami su vienu duomenų bazės lauku.
- *Item* – atominis analizės objektas, kuriame gali būti nurodytas archetipas, arba raktažodis, pagal kurį reikia atlikti analizę.

3.3.4.4. Teksto analizės sąsajos

Praktiškai gali egzistuoti eilė teksto analizės algoritmų. Tam kad juos būtų paprasta naudoti, buvo sukurta viena apibendrinta sąsają. Ši sąsaja ir ją realizuojančių klasių diagrama pateikta 20 paveiksle.



20 pav. Informacijos transformatoriaus sąsaja.

3.3.4.5. Teksto analizės algoritmai

Darbe realizuoti pora duomenų analizės ir transformacijos algoritmų. Pirmas iš jų – tai algoritmas, pagal skyriklių reguliariąją išraišką suskaidantis tekstą į atskiras dalis. Skyrikliai gali būti bet kokie: nuo paprastų – pavieniai simboliai, eilutės pabaigos simboliai, iki sudėtingų – kablelis kuris yra tarp žodžių bet ne tarp skaičių ir pan. Šis algoritmas įgyvendintas *kut.ibme.Extraction.ExtrSeparator* klasėje. Jo grąžinamas rezultatas – išgaunamų duomenų medis (**3.3.4.2 Išgaunamos informacijos duomenų struktūra**). Šis algoritmas yra gan paprastas ir universalus. Plačiau apie reguliariųjų išraiškų sudarymą šiame darbe rašoma skyriuje **4.2.4 Migracijos schemų sudarymas**.

Kitas algoritmas – konkretesnis, kuris vykdo išgaunamos informacijos šablone nurodytus veiksmus. Skirtingai nuo *ExtrSeparator*, šis algoritmas nėra universalus, nes buvo taikytas prie esamų duomenų (dėl paprastesnės realizacijos). Kuriant realią, o ne prototipinę sistemą šis algoritmas būtų sudėtingesnis. Esminė algoritmo dalis – pagal nurodytus šablone archetipus, raktažodžius generuojamos reguliariosios išraiškos, pagal kurias yra atliekama medicininių dokumentų analizė. Rezultatai – išanalizuoti ir į XML bylas surašyti bei nuorodomis ir archetipais sujungti išgauti duomenys, sudarantys elementarią elektroninę sveikatos istoriją.

3.3.5. Išgautų duomenų validatorius

Kaip minima metodikos aprašyme (**3.2 Automatizuota medicininių dokumentų analizės metodika**), be adapterio ir transformavimo programinės įrangos, dar turi būti sukurta validavimo programinė įranga, skirta medicinos ekspertui, prižiūrinčiam migracijos procesą. Atliekant eksperimentą buvo sukurta pavyzdinė įranga (klasės *kut.ibme.Extraction.UI* pakete), kuri leidžia vizualiai peržiūrėti išgautų duomenų medį. Realioje sistemoje ši įranga būtų žymiai sudėtingesnė, bet dirbtų panašiu principu – paprasta sąsaja pateikianti medicinos ekspertui išgaunamus duomenis ir nurodanti iš kur jie buvo išgauti, bei ką jie reiškia.

4. AUTOMATIZUOTOS MEDICININIŲ DOKUMENTŲ ANALIZĖS EKSPERIMENTAS

4.1. Įvadas

Turint prototipinę programinę įrangą, metodika buvo išbandoma su Kauno kardiologijos klinikos pacientų duomenų baze (**4.2.1 Liktinė sistema**). Ši bazė yra pildoma maždaug nuo 1999 metų ir dabar joje yra apie vienuolika tūkstančių įrašų su įvairia informacija (bendri duomenys apie pacientą, diagnozės, rekomenduoti vaistai, laboratoriniai tyrimai ir pan.). Laukų užpildymas bazėje yra labai įvairus.

Atliekant eksperimentą visa asmeninė pacientų informacija buvo cenzūruota (paties pirmojo metodikos žingsnio metu).

Toliau šiame skyriuje pateikti visi metodikos žingsniai ir gauti rezultatai. Atitinkami darbo su programine įranga žingsniai, bei programos vaizdai pateikti skyriaus **9. PRIEDAI** poskyryje **9.1 Darbas su prototipine programine įranga**.

4.2. Analizė

Žingsnio tikslas – paruošti liktinius duomenis migracijos procesui.

4.2.1. Liktinė sistema

Eksperimento liktinė sistema – Kauno kardiologijos klinikos pacientų duomenų bazė. Duomenų bazė yra įgyvendinta su „Microsoft FoxPro 2.6a“ duomenų bazių valdymo sistema.

4 lentelė. Duotosios liktinės duomenų bazės lentelių sąrašas

Lentelės pavadinimas	Paskirtis
anketa	Pagrindinė duomenų bazės lentelė. Vienas lentelės kortežas – pilna anketa apie vieną pacientą.
apskr	Lietuvos apskričių sąrašas.
chi_ops	Instrumentiniai invaziniai tyrimai ir gydomosios procedūros.
gydyt	Gydytojų sąrašas.
ista	Siunčiančių įstaigų sąrašas.
istaig	Siunčiančių įstaigų sąrašas.
ligos	Ligų ir sveikatos problemų sąrašas, pagal tarptautinę statistinę ligų ir sveikatos problemų klasifikaciją TLK–10 [36].
operac	Atliktų operacijų sąrašas.
paslauga	Paslaugų sąrašas.
perkel	(nežinoma)
plsky	Klinikų, skyrių sąrašas.
prof	Gydymo profiliai (?)
rajonai	Lietuvos rajonų sąrašas.
simbol	Tam tikrų specialių simbolių (pvz., matavimo vienetų) sąrašas.

Jokios projektavimo ar kūrimo informacijos apie šią duomenų bazė nėra, todėl visa informacija (lentelės, laukai ir kt.) yra gauta atliekant atvirkštinę inžineriją „Microsoft Visio“ paketo pagalba. Lentelių ir laukų paskirtis buvo analizuojama konsultuojantis su medicinos specialistu. Duomenų bazės lentelės ir jų paskirtis pateikti 4 lentelėje.

Eksperimentas buvo atliekamas ne su visa duomenų baze, bet tik su viena pagrindine jos lentele – „anketa“. Viso įrašų šioje lentelėje – 11443. Šios lentelės laukai su juose saugomų duomenų aprašymu pateikti 5 lentelėje.

5 lentelė. Eksperimente naudojamos liktinės duomenų bazės lentelės „*anketa*“ laukų sąrašas. Stulpelis „*Užpildymas*“ reiškia kokia dalis iš visų duomenų bazėje esančių šios lentelės įrašų (11443) turi tam tikrą lauką netuščią (tik teksto tipo laukams).

Lauko Nr.	Lauko pavadinimas	Paskirtis	Užpildymas %
0	eilnr	Anketos eilės numeri (pirminis raktas)	
1	istnr	Tos pačios reikšmės kaip ir *eilnr*, tik tekstiniam pavidale.	99
2	istmet	Istorijos numeris (antrinis raktas?)	
3	ambulat		
4	sdser	Paciento socialinio draudimo serija	11
5	sdnum	Paciento socialinio draudimo numeris	
6	asmkod	Paciento asmens kodas	
7	dokument	Paciento pateiktas dokumentas	38
8	dokser	Paciento pateikto dokumento serija	36
9	doknum	Paciento pateikto dokumento numeris	
10	passer	Paciento paso serija	46
11	pasnum	Paciento paso numeris	
12	kasrup	Kas rūpinasi/rūpinosi pacientu	50
13	lytis	Paciento lytis	
14	atdata	Atvykimo data, valanda, minutės.	
15	atval		
16	atmin		
17	isvdata	Išvykimo data	
18	viso_lovad		
19	lov_kit		
20	lygis		
21	gimdat	Paciento gimimo data	
22	pavard	Paciento pavardė	99
23	vardas	Paciento vardas	99
24	draudrus		

5 lentelės tęsinys

Lauko Nr.	Lauko pavadinimas	Paskirtis	Užpildymas %
25	draudist	Apskrities kodas (į lentelę *apskr*)	
26	rajkodas	Rajono kodas (į lentelę *rajonai*)	
27	socpad		
28	darbov	Paciento darbovietė	86
29	kaimas	Ar pacientas iš kaimo	
30	adresas	Paciento adresas	97
31	tel	Paciento telefonas	
32	inval	Invalidumas	
33	siukodas		
34	perkel		
35	siukit	Siunčiančios įstaigos kodas (į lentelę *istaig*)	
36	butpag		
37	but_kod		24
38	skub		
39	skubs		
40	hos_kas		
41	hos_priez		
42	dgn_skod	Ligos kodas iš siunčiančios įstaigos	80
43	dgn_siunt		54
44	dgn_skod1		48
45	dgn_pkod		82
46	dgn_epik	Sutrumpintas pagrindinis ligos kodas	82
47	dgn_pagr	Pagrindinis (nustatytas) ligos kodas	82
48	dgn_komp		74
49	dgn_gret	Gretutiniai ligos kodai	69
50	dgn_gret1		64
51	dgn_gret2		49
52	dgn_gret3		32
53	dgn_gret4		21
54	dgn_gret5		16
55	ligeig1		
56	ligeig2		
57	ligeig3		
58	kitstac		

5 lentelės tęsinys

Lauko Nr.	Lauko pavadinimas	Paskirtis	Užpildymas %
59	kitstp		
60	pirmkart		
61	perdum		
62	isr_siunt		0
63	pat_ind	Patologinės indikacijos	82
64	lig_anam	Viskas, kas susieję su pagrindine diagnoze, subjektyviai	82
65	lig_anao	Viskas, kas susieję su pagrindine diagnoze, objektyviai (ištyrus)	82
66	**diag_tyr**	**Diagnostiniai tyrimai, konsultacijos**	**83**
67	lig_eiga		81
68	gydymasm	Gydymo vaistai	82
69	gydymasi		37
70	gydymasc		36
71	lig_bukl	Paciento būklė išvykstant	81
72	rek_gyd	Kokie paskirti vaistai išvykstant	77
73	rek_reabt		
74	rek_reab		30
75	rek_amb		75
76	rek_darb		
77	prognoze		71
78	lig_isr		
79	lig_isrk1		63
80	lig_isrk2		50
81	lig_isrk3		39
82	lig_isrk4		34
83	aritm	Ar buvo ritmo sutrikimų	
84	aritm_kod1	Rimto sutrikimų kodai pagal TLK–10 [36] klasifikaciją.	48
85	aritm_kod2		38
86	aritm_kod3		32
87	laid	Ar buvo laidumo sutrikimų	
88	laid_kod1	Laidumo sutrikimų kodai pagal TLK–10 [36] klasifikaciją.	28
89	laid_kod2		25
90	laid_kod3		22
91	lsn	Ar buvo lėtinis širdies nepakankamumas	

5 lentelės tęsinys

Lauko Nr.	Lauko pavadinimas	Paskirtis	Užpildymas %
92	lsn_kod1	Lėtinio širdies nepakankamumo kodai pagal TLK–10 [36] klasifikaciją.	64
93	lsn_kod2		37
94	lsn_kod3		27
95	usn	Ar buvo ūminis širdies nepakankamumas	
96	usn_kod1	Ūminio širdies nepakankamumo kodai pagal TLK–10 [36] klasifikaciją.	21
97	usn_kod2		16
98	usn_kod3		13
99	kita	Kiti sutrikimai	
100	kita_kod1	Kitų sutrikimų kodai pagal TLK–10 [36] klasifikaciją.	25
101	kita_kod2		21
102	kita_kod3		16
103	sutapo		
104	gydyt_hosp		
105	hosp_rub		
106	gydyt		
107	gydyt_pav		85
108	skyr_ved		
109	sekt_vad		
110	nusisk	Ligonio nusiskundimai	40
111	a_morbi	Ligos anamnezė	40
112	a_vitae	Gydymo anamnezė	39
113	a_laboris	Paciento darbo sąlygos	34
114	a_alerg	Paciento alergijos	37
115	st_prae		39
116	st_spec		29
117	diagnoze	Pagrindinė diagnozė	39
118	tyr_pla	Tyrimų planas	39
119	gyd_pla	Gydymo planas	38

Šios lentelės egzemplioriaus pavyzdys yra pateiktas skyriuje **9.2.1**. Kaip matyti iš lentelės užpildymo statistikos, laukai pildomi labai įvairiai. Daugumoje yra pildomi pagrindiniai laukai – vardas, pavardė, ir panaši informacija, taip pat diagnozė, tyrimai, prognozės, paskirti vaistai. Pagrindinis dominantis laukas su kuriuo atliekamas eksperimentas – „*diag_tyr*“, kuriame yra aprašoma diagnostinė informacija – pacientui atliktų tyrimų rezultatai, bei konsultacijos. Keletas šio lauko turinio pavyzdžių:

Laboratoriniai : leu-5,7 x10**9/l , eritr- 3,52 x10**12/l, Hb-110 g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01 mmol/l, DTLCH -0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 , urea- 7,26 mmol/l, troponinas < 0,2 mcg/l.
EKG : SR , ST dislokacija žemiau izolinijos iki 0,5 mm V4-V6 , II , III , aVF der.
2Decho : KSGDD- 47 , sienelės po 11 , MM -187 , MI - 115.II° reguerg,. per MV , TV . Išvada - fibroziniai ir kalcinotinai MŽ pakitimai . Ao ž . pakitimai sklerotiniai . Saiki Ao stenozė . I° regurg. per Ao V . Susilpnėjusi KS sistolinė f-ja , sutrikusi diastolinė f-ja . IF-42% .
KG - duomenys dng.

- -

Laboratoriniai: kraujo - Hb 125 g/l, leuk. 9,2x10**9/l, eritr. 3,9x10**12/l, tromb. 308x10**9/l, ENG 55 mm/val., K 4,6 mmol/l, Na 135 mmol/l, urea 12,86 mmol/l. CRB 20,7 mcg/l. BCH 4,44 mmol/l, MTLCH 2,81 mmol/l, DTLCH 1,08 mmol/l, TG 1,19 mmol/l, AK 3,1. ASAT 28 U/l, ALAT 33 U/l, CPK 790 U/l.
EKG: PV, pilna AV blokada. Įstačius EKS - registruojama efektyvi stimuliatoriaus veikla.
Krūtinės ląstos rö: diafragma be pakitimų, pleuros sinusai laisvi, plaučių oringumas padidėjęs, I° veninė stazė, šaknys strūktūrinės, a. pulmonalis dex. 14 mm. Širdies skersmuo nepadidėjęs, talija paryškėjusi. Aorta vidutiniškai difuziškai, kiek daugiau kylančiojoje dalyje, išsiplėtusi.
2D echo: KSGDD 54,8, ind. 27,68, TSP storis diast. 14, KS US storis 14, KS MM/MI 336/169,7, sant. sienos storis 0,511, IF 45%. KPr 72 x 44, DPr 61 x 45, ao ž. 21,4, ao ties sinotub. jungtimi 31,4, SJI 1,59. Bloga vaizdo kokybė. KSH, II° KPr dilatacija, DPr dilatacija. Sumažėjusi sistolinė KS funkcija. Kontrakcijos sutrikimai gali būti lemiami ir stimuliatoriaus veiklos.
Konsultacijos: Infektologas - Erysipelas acuta pedis sin. Rekomenduotas 7 d. gydymo kursas doksiciklinu 100 mg x 1.

4.2.2. Liktinės sistemos analizė

Išanalizavus liktinę duomenų bazę (pasinaudojant sukurtos programinės įrangos įrankiu „DB naršyklė“, **9.1.1.1 Liktinės sistemos analizė**, 27 pav.), eksperimentui buvo pasirinkta lentelė „*anketa*“. Iš jos pasirinkti penki paprasti laukai: *vardas*, *pavard*, *gimdat*, *atdata*, *lytis* (atitinkamai paciento vardas, pavardė, gimimo data, atvykimo į gydymo įstaigą data, lytis), kurie naujoje sistemoje gali būti atvaizduoti tiesiogiai. Bei vienas sudėtingas laukas *diag_tyr* (diagnostiniai tyrimai, konsultacijos), kuris yra mažai struktūrizuotas formos su įvairia informacija, todėl jį teks analizuoti. Pasirinkti laukai formaliai aprašomi *svarbių duomenų schemoje*.

Svarbių duomenų schema yra sudaroma kitoje eksperimentinės programinės įrangos kortelėje – „Laukų atrinkimas“ (**9.1.1.1 Liktinės sistemos analizė**, 28 pav.). Sudarant svarbių duomenų schemą reikia išskirti raktinius lentelės laukus (pagal kuriuos kiekvienas iš lentelės egzempliorių bus identifikuojamas) bei pasirinkti norimus perkleti laukus. Konkrečiu atveju raktiniai laukai buvo pasirinkti *eilnr* ir *istmet*, bei visi laukai buvo pasirinkti kaip reikalingi. Pagal svarbių duomenų schemą reikiami duomenys iš duomenų bazės pervedami į XML tipo bylas diske. Eksperimente gautoji svarbių duomenų schema XML formate pateikta skyriuje **9.2.4.1 Svarbių duomenų schema (XML)**.

Taip pat, liktinės sistemos analizės žingsnyje duomenyse reikia identifikuoti naujas žinias, kurių dar nėra ESI sistemoje. Kadangi eksperimento atveju ESI sistema yra tuščia, tai visiems išgaunamiems duomenims turėjo būti sudaromi žinių modeliai.

Metodikoje minima naujoji ESI sistema yra paremta archetipais. Kadangi archetipai savyje neturi apibrėžtų terminų, tik nuorodas (atvaizdavimą) į globalių terminologijų terminus, eksperimente sudarytiems archetipams dar buvo sudarytas ir mažos apimties žodynėlis (jo fragmentas pateiktas 6 lentelėje, visas žodynėlis – skyriaus **9.2.2 Sudarytas ir naudotas žodynas**, 14 lentelėje). Žodynas buvo išsaugotas XML tipo byloje (jos fragmentas pateiktas skyriaus **9.2.2** pabaigoje).

Žodyne buvo surašyti visi žinomi (ir liktinėje duomenų bazėje naudojami) terminų trumpinimai, kad iš jų automatinis teksto analizės algoritmas galėtų sudaryti reguliariąsias išraiškas šių terminų paieškai analizuojamame tekste.

6 lentelė. Eksperimento metu sudaryto ir naudoto žodyno fragmentas

Identifikacijos numeris	Pirminis terminas	Galimi trumpinimas
500	eritrocitas	RBC erit eritr eritrocitai
501	hemoglobinas	HGB Hb
502	hematokritas	HCT PCV Ht
503	vidutinis eritrocito tūris	MCV

Pagal norimus išgauti liktinius duomenis sudaryti trys archetipai: žmogaus asmeninai duomenys, laboratorinių tyrimų rezultatai, klinikinis kraujo tyrimas. Sudarytų archetipų struktūros fragmentas (žmogaus asmeninių duomenų archetipas) pateiktas 7 lentelėje. Visų eksperimente sudarytų archetipų struktūros pateiktos skyriaus **9.2.3.2 Archetipų struktūros** 16 lentelėje.

7 lentelė. Eksperimente sudarytų archetipų struktūrų fragmentas

Archetipo pavadinimas	Koncepcijos pavadinimas	Koncepcijos tipas	Kiti apribojimai	Koncepcijos identifikacija
Žmogaus asmeniniai duomenys				800
	vardas	tekstas		801
	pavardė			802
	gimimo data			803
	atvykimo data			804
	lytis			805

Žmogaus asmeninių duomenų archetipo struktūra gauta pagal norimus išgauti asmens duomenis (vardas, pavardė, gimimo data ir kt.). Sudarant laboratorinių tyrimų rezultatų archetipą, buvo panaudota archetipų agregacijos savybė (galimybė archetipams agreguoti vieni kitus). Taip buvo sukurtas vienas mažas archetipas laboratoriniams tyrimams į kurį gali įeiti įvairių archetipų (su tyrimų duomenimis) nuorodos. Eksperimentui buvo pasirinktas vienas toks archetipas – klinikinis kraujo tyrimas. Pusiau formalus šio tyrimo aprašymas buvo paimtas iš [27]. Dalis jo pateikta 8 lentelėje.

8 lentelė. Klinikinio kraujo tyrimo normatyvų fragmentas

Analitė	Norma		Vienetai
RBC Eritrocitai	vyr	4,5-5,9	$*10^{12}$/l
	mot	4,5-5,2	$*10^{12}$/l
HGB Hemoglobinas	vyr	140-180	g/l
	mot	120-160	g/l
HCT (PCV) Hematokritas	vyr	41-53	%
	mot	36-46	%
MCV Vidutinis eritrocito tūris	82-98		Fl

Visa klinikinio kraujo tyrimo normatyvų lentelė yra pateikta skyriaus **9.2.3.1 Klinikinių kraujo tyrimų rezultatų normatyvai** 15 lentelėje.

Archetipai buvo sudaromi ADL kalba ir išsaugomi XML formato bylose. Archetipų XML schemos fragmentas su paaiškinimais pateiktas skyriaus **9.2.3.2 Archetipų struktūros** pabaigoje.

4.2.3. ESI sistemos (standarto) analizė

Antrasis analizės žingsnis metodikoje – esamos ESI sistemos analizė. Kadangi darbe jokios ESI sistemos nėra, tai šiame žingsnyje buvo sudaromas šablonas, kuris atitiks pseudo elektroninę sveikatos istoriją. Ši schema tiesiog turi savyje nuorodas į kitas bylas – išgautų paciento asmeninių duomenų bylą, laboratorinių tyrimų bylas, informacijos išgavimo protokolą ir kt. Tai yra, eksperimento atveju ESI sistema pasirenkama pati paprasčiausia – archetipais apribojamos išgautų duomenų hierarchijos, saugomos XML bylų sistemose. Tokios sistemos pavyzdys su XML bylų fragmentais pateiktas skyriuje **4.4.1 Naujoji informacinė sistema**.

4.2.4. Migracijos schemų sudarymas

Migracijos schema – tai planas, pagal kurį liktiniai duomenys yra analizuojami ir perkeliami į naujosios ESI sistemos struktūras.

Turint išgaunamų duomenų žinių modelį (žodyną ir archetipus) reikia sudaryti taisykles (algoritmus), kaip duomenys bus išgaunami. Struktūrizuoti duomenys yra tiesiogiai atvaizduojami iš liktinės duomenų bazės į naujosios ESI sistemos struktūras. Nestruktūrizuoti ar mažai struktūrizuoti laukai (eksperimento atveju – „diag_tyr“) turi būti apdorojami duomenų išgavimo algoritmais.

Eksperimento programinės įrangos trečioji kortelė „Paruošimas“ yra skirta vienos lentelės egzempliorių to paties lauko peržiūrai ir teksto analizės žingsnių nustatymui – tai yra migracijos šablono sudarymui. Prototipe grafinė vartotojo sąsaja teksto analizės žingsniams parinkti nebuvo kuriama, kadangi ji būtų labai sudėtinga (leidžianti kurti lanksčią schemą, turinčią savyje įvairius skyriklius, analizės algoritmus, archetipus ir kt.). Migracijos šablonas XML formate buvo kuriamas teksto redaktoriumi.

Išanalizavus eilę lentelės „anketa“ laukų „diag_tyr“ matosi, kad kiekvienas tyrimas (viename lauke) yra atskirtas naujos eilutės simbolio, bei prieš kiekvieną tyrimą yra parašytas jo pavadinimas, užbaigiamas dvitaškiu (pvz., „Laboratoriniai: ...“, „EKG: ...“). Todėl pirma reikia atlikti skaidymą pagal naujos eilutės simbolį, o po to kiekvieną iš gautų laukų analizuoti, ieškant raktinio žodžio kurio pabaigoje būtų parašytas dvitaškis. Taip bus įmanoma atskiroms lauko dalims identifikuoti juose saugomas žinias ir šiems laukams taikyti tik tas žinias aprašančius archetipus (išskirti *aktyvią archetipų aibę*).

Klinikinio kraujo tyrimo duomenys yra rašomi po raktažodžio „laboratoriniai“ ir daugumoje juos sudaro „terminas–reikšmė“ poros. Tačiau šios poros yra užrašomos labai įvairiai, pvz., klinikinio kraujo tyrimo parodymas „leukocitai“ gali būti:

- leuk. - 8,5x10**9/l
- leuk. 7,3-6,6x10**9/l
- leuk.6,1x10**9/l
- leu-10,6 x10**9/l
- leu- 16,4x10**9/l

ir panašiai. Tam kad visuose įrašuose būtų teisingai atpažinta įvairiai saugoma informacija, teksto analizės metu iš sudaryto žodyno yra generuojamos reguliariosios išraiškos, pagal kurias analizės algoritmas bando atpažinti terminus. Šias išraiškas generuoja vienas iš eksperimentui sukurtos programinės įrangos įrankių.

Taigi, išanalizavus įvairius duomenų lauko „diag_tyr“ egzempliorius sudaromas migracijos šablonas (paprastumo dėlei čia jis pateiktas pseudo komandomis):

```
su lentele „anketa“ atlikti
    išgavimo elementas (1)
    pagal asmens duomenų archetipą kut-ibme-ehr-test.gpic-person.draft
       tiesiogiai atvaizduoti (iš lentelės į ESI struktūrą)
           lauką vardas į lauką at0001
           lauką pavard į lauką at0002
           lauką gimdat į lauką at0003
           lauką atdata į lauką at0004
           lauką lytis į lauką at0005
```

```
išgavimo elementas (2)
su lauku „diag_tyr“
     (a) suskaidyti lauko turinį pagal eilutės pabaigos simbolį
     (b) analizuoti
          1. jei randamas raktažodis „laboratoriniai“, analizuoti
          pagal klinikinį kraujo tyrimo archetipą kut-ibme-ehr-
          test.laboratory_results.draft, prieš tai lauko turinį
          suskaidant per kablelius esančius ne tarp skaičių.
          2. jei randamas raktažodis „EKG“ ....
```

Tikrasis eksperimento šablonas buvo sudarytas XML formate ir su paaiškinimais yra pateiktas skyriuje **9.1.1.2 Migracijos schemos sudarymas**. Šablono elementų aprašymas yra pateiktas skyriuje **3.3.4.3 Informacijos išgavimo šablonai.**

4.3. Migracija

Šiame žingsnyje analizės paruošti (tačiau neapdoroti) duomenys XML bylų pavidale yra pateikiami automatizuotam procesui, kuris juos išanalizuoja ir perveda į naujos struktūros sistemą.

Kiekviena neapdorota duomenų byla yra perduodama automatiniam algoritmui, kuris pagal migracijos šabloną atlieka analizę. Šios analizės rezultatas – išgautų duomenų medis XML formate ir naujai suformuoti ESI sistemos elementai. Išgautų duomenų medžio pavyzdys pateiktas skyriuje **9.2.5 Išgautų duomenų medis (XML)**.

Pagal šį medį (transformacijos protokolą) vartotojas gali atlikti išgautų duomenų validaciją. Kaip minėta poskyryje **3.3.5 Išgautų duomenų validatorius**, eksperimente buvo įgyvendinta tik pavyzdinė tokio tipo programinė įranga (kortelė „Validavimas“, skyrius **9.1.2.2 Validavimas**), kuria galima vizualiai peržiūrėti išgautų duomenų medį bei klaidas ir perspėjimus, kuriuos sugeneravo automatinis teksto analizės algoritmas.

Atlikus migraciją ir jos validaciją (patvirtinus, kad išgauti duomenys teisingi – eksperimento metu tai nėra daroma) gaunami naujos struktūros įrašai. Šie įrašai yra labai supaprastinti elektroninių sveikatos istorijų variantai (paprastos XML formato bylos su išgautais duomenimis). Vizualiai (medžio pavidale) elektroninės sveikatos istorijos vartotojui yra pateikiamos kortelėje „ESI peržiūra“ (skyrius **9.1.2.3 ESI peržiūra**).

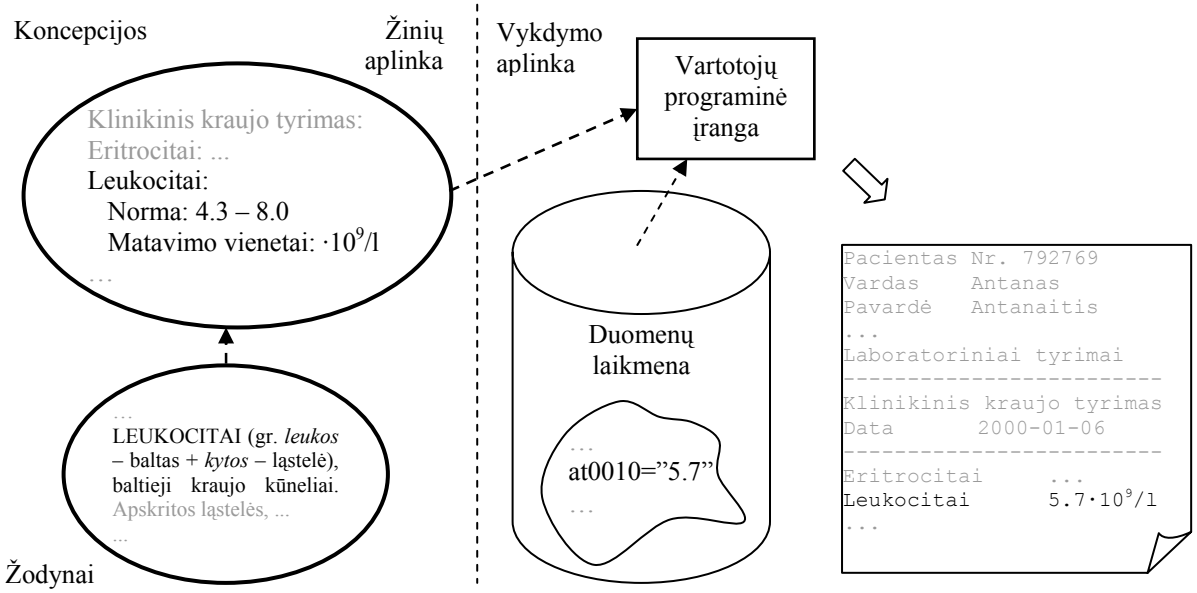
4.4. Rezultatai

Automatizuoto archetipais pagrįsto liktinių duomenų išgavimo proceso pagrindiniai rezultatai:

- Gauta dviejų lygmenų informacinė sistema, sudaryta iš žinių (dalykinės srities faktų) ir informacijos (konkrečių dalykinės srities informacijos egzempliorių).
- Iš liktinės sistemos su mažai struktūrizuotais laukais išgauti duomenys, kuriuos platesnėmis apimtimis ir automatizuotu būdu gali analizuoti ir tirti dalykinės srities specialistai.

4.4.1. Naujoji informacinė sistema

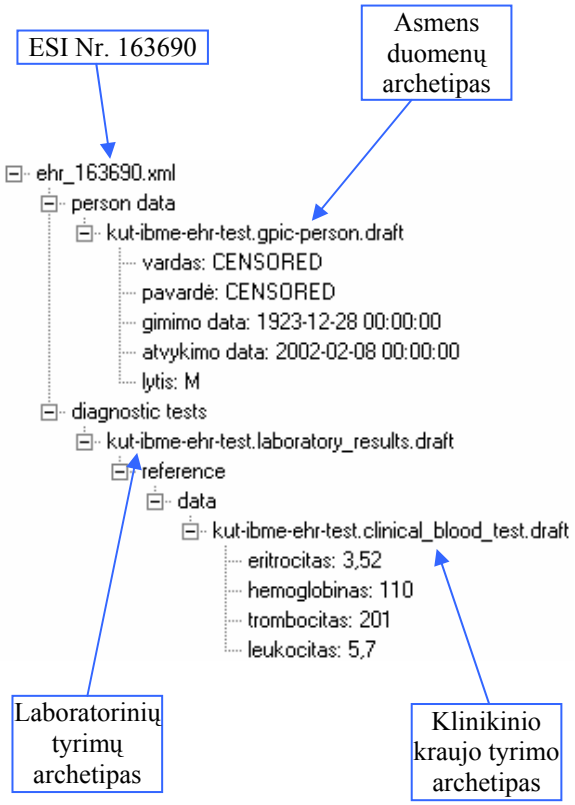
Eksperimente sudarytos ir išbandytos dviejų lygmenų informacinės sistemos schema pateikta 21 paveiksle. Žinių lygmeniui priklauso eksperimentinis žodynėlis, sudarytas iš 39 sąvokų ir viso 60 jų variacijų, bei trys sudėtinės koncepcijos – asmens duomenų, laboratorinių tyrimų ir klinikinio kraujo tyrimo archetipai.



21 pav. Eksperimentine sudaryta dviejų lygmenų informacinė sistema.

Informacijos lygmeniui priklauso duomenų egzemplioriai, automatinės teksto analizės pagalba išgauti iš liktinės sistemos mažai struktūrizuotų laukų, bei nesudėtingos struktūros naujosios duomenų laikmenos (konkrečiai darbe – atskiros specialaus formato XML bylos). Duomenų laikmenos jungiasi į hierarchinę (medžio) struktūrą, taip sudarydamos pseudo elektroninių sveikatos istorijų įrašų dalis. Duomenų laikmenų medžio pavyzdys pateiktas 22 paveiksle.

Pseudo ESI sistemos bylos atspindi vieną iš pagrindinių archetipinių informacinių sistemų privalumų – archetipai (žinios) gali būti bet kokio sudėtingumo, tačiau pati laikmena saugoti duomenis yra labai paprasta. Kaip pavyzdys, žemiau pateikti žmogaus asmeninių duomenų (vardo, pavardės, adreso, gimimo metų ir kt.) ir laboratorinių tyrimų duomenų laikmenos, kurių žinios yra saugomos ne kartu su duomenimis, bet nurodytuose archetipuose.



22 pav. Eksperimentine gauta naujų ESI įrašų hierarchinė struktūra, sudaryta iš atskirų laikmenų, apribotų archetipais.

Žmogaus asmeninių duomenų laikmena (lokalūs terminai at0001, at0002 ir t.t. atvaizduojami į globalius terminus per nurodytą archetipą):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<datainstance id="1;1;16369;0" name="person data">
    <datagroup archetype_id="kut-ibme-ehr-test.gpic-person.draft">
        <datavalue localterm="at0001" data="PACIENTO_VARDAS" />
        <datavalue localterm="at0002" data="PACIENTO_PAVARDĖ" />
        <datavalue localterm="at0003" data="1923.12.28 00:00:00" />
        <datavalue localterm="at0004" data="2002-02-08 00:00:00" />
        <datavalue localterm="at0005" data="True" />
    </datagroup>
</datainstance>
```

Klinikinio kraujo tyrimo laikmena, per nuorodą priklausanti laboratorinių tyrimų laikmenai (lokalūs terminai at0001, at0002 ir t.t. atvaizduojami į globalius terminus per nurodytą archetipą):

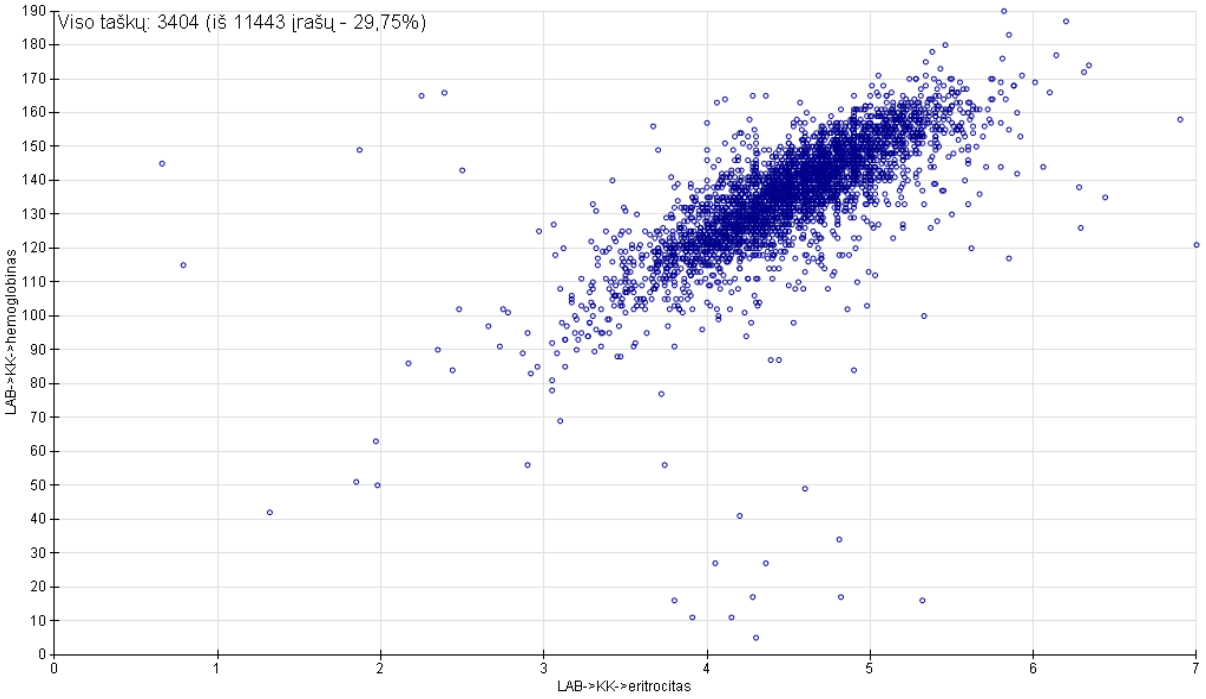
```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<datainstance id="di_2_163690_0_0" name="">
    <datagroup archetype_id="kut-ibme-ehr-test.clinical_blood_test.draft">
        <datavalue localterm="at0001" data="3,52" />
        <datavalue localterm="at0002" data="110" />
        <datavalue localterm="at0008" data="201" />
        <datavalue localterm="at0010" data="5,7" />
    </datagroup>
</datainstance>
```

Kita archetipinių sistemų savybė – iš laikmenų ir jas nusakančių archetipų galima konstruoti dinaminius vartotojo sąsajos dialogus. Tai yra, kiekvienam archetipui nebūtina kurti atskiro įvedimo/išvedimo/peržiūros lango, pakanka tik sukurti gerą tokių langų

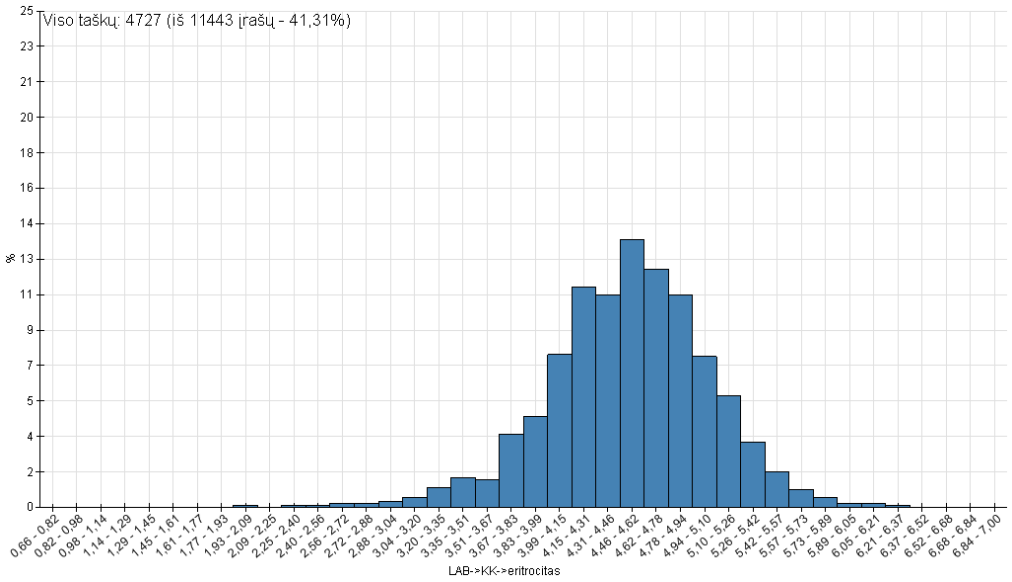
generavimo paprogramę, generuojančią dialogų formas pagal archetipus. Tokie dialogai automatiškai užtikrintų įvedamų duomenų patikrą, pagalbą, paaiškinimus apie įvedamuosius duomenis ir pan. Eksperimento metu sudaryta pavyzdinė grafinės vartotojo sąsajos generavimo paprogramė, kuri pagal pseudo elektroninį sveikatos įrašą generuoja apie pacientą turimos informacijos medį (22 pav.).

Pagal ESI sistemos žinių modelį iš eksperimentui pasirinktos liktinės sistemos išgauti duomenys išsaugomi naujojoje pseudo ESI sistemoje. Darbe įgyvendinti tiek tiesioginiu atvaizdavimu, tiek teksto analize paremti duomenų išgavimo metodai. Bandymai atlikti su žmogaus asmeniniais duomenimis (tiesioginis atvaizdavimas) ir su klinikinio kraujo tyrimo rezultatais (analizė).

Teksto analizės būdu iš mažai struktūrizuotų liktinės duomenų bazės laukų išgauti duomenys specialia forma gali būti pateikti dalykinės srities specialistui. Tai leidžia atlikti plačius statistinius tyrimus su „kompiuteriniame popieriuje“ – mažai struktūrizuotuose ar nestruktūrizuotuose laukuose – išsaugotais duomenimis, kuriuos praktiškai specialistai turėtų skaityti ir nagrinėti po vieną, išrinkdami sau reikiamą informaciją. Eksperimente įgyvendintas pavyzdinis duomenų peržiūros įrankis (23 pav.), kuriuo, pagal ESI sistemos žinių modelį, grafiškai galima peržiūrėti: a) kaip vienas išmatuojamasis dydis priklauso nuo kito dydžio (X/Y grafikas), b) vieno dydžio pasiskirstymą (histograma).



(a)



(b)

23 pav. Eksperimentine išgautų duomenų grafinio pateikimo pavyzdys: priklausomybės, pvz., eritrocitų kiekio priklausymas nuo hemoglobino kiekio kraujyje (a) ir pasiskirstymas, pvz., eritrocitų kiekis kraujyje (b).

4.4.2. Duomenų išgavimo patikimumo ir efektyvumo tyrimas

Norint įsitinkinti automatizuoto liktinių duomenų išgavimo proceso patikimumu buvo atliktas nedidelės duomenų imties perkėlimo tyrimas. Toks tyrimas reikalingas priedo prie perkeliamų duomenų validacijos, kad įsitinkinti duomenų perkėlimo efektyvumu – kiek ir kaip gerai automatinis analizės procesas atpažįsta norimus išgauti duomenis. Buvo atlikta ekspertinė analizė – specialistui vizualiai peržiūrint 100 liktinės duomenų bazės mažai struktūrizuotų laukų ir fiksuojant reikalingus duomenis. Ekspertinė analizė truko maždaug 55 minutes, tai yra apie 0.03 įrašo/s. Ekspertinės bei automatinės (pirmojo sistemos varianto) analizių palyginimas pateiktas 9 lentelėje.

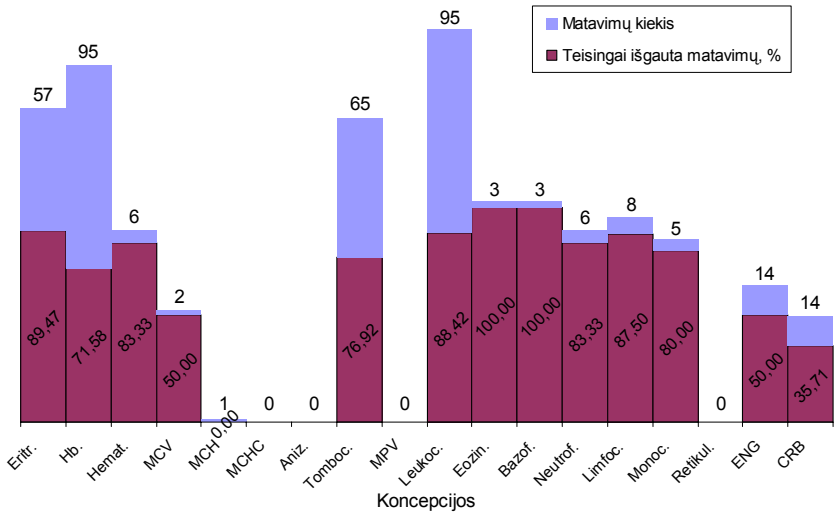
Stulpelyje „Ekspertinė analizė“ pateikiama kiek terminų su matavimais buvo rasta vizualiai peržiūrint tiriamus įrašus, o stulpelyje „Automatinis išgavimas“ pateikiami automatinės analizės išgavimo duomenys. Automatinė analizė pirma atpažįsta terminą (stulpelis „Rasta terminų“ – kiek procentų ekspertinės analizės būdu rastų terminų atpažino automatika), o po to bando išskirti ir konvertuoti galimą skaitinę tuo terminu žymimą reikšmę. Skaitinių reikšmių konvertavimo duomenys pateikti stulpelyje „Išgauti matavimai“. Stulpelis „Viso matavimų“ rodo kiek procentų skaitinių reikšmių buvo išgauta nuo ekspertinės analizės. Stulpelis „Teisingai“ nusako kiek procentų automatinės analizės išgautų skaitinių reikšmių buvo tos, kurios buvo identifikuotos ir ekspertinės analizės metu. Stulpelis „Viso išgauta teisingai“ – stulpelių „Viso matavimų“ ir „Teisingai“ sandauga – procentai, kiek automatinis algoritmas išgavo koncepcijų su matavimais teisingai. Stulpelyje „Kiti duomenys“ pateikta informacija svarbesnė medicinos ekspertams: „Nerasta matavimo

vienetų“ – kiek iš surastų koncepcijų neturėjo parašytų matavimo vienetų, „Už normos ribų“ – kiek automatinio algoritmo konvertuotų skaičių nepateko į klinikinio kraujo tyrimo normatyvų ribas. Viso teisingai išgautų matavimų kiekio grafikas pateiktas 24 paveiksle.

9 lentelė. Automatinio liktinių duomenų išgavimo patikimumo ir efektyvumo tyrimas – palyginimas su ekspertine analize: pirmas sistemos variantas. Duomenų imtis – 100 įrašų

Koncepcija	Ekspertinė analizė (duomenų kiekis N_e)	Automatinis porų „*terminas–matavimas*“ išgavimas						
		Rasta terminų		Išgauta matavimų				
		Teisingai	Klaidingai	Viso	Teisingai	Viso išgauta teisingai (E_e)	Kiti duomenys	
							Nerasta mat. vienetų	Už normos ribų
Eritrocitai	**57**	**94,7%**	1,8%	89,5%	100,0%	**89,5%**	1,9%	54,9%
Hemoglobinas	**95**	**97,9%**	–	71,6%	100,0%	**71,6%**	3,2%	14,7%
Hematokritas	**6**	**100,0%**	–	83,3%	100,0%	**83,3%**	50,0%	0,0%
Vid. eritrocito tūris (MCV)	**2**	**100,0%**	–	50,0%	100,0%	**50,0%**	100,0%	100,0%
Vid. Hb kiekis eritrocite (MCH)	**1**	**100,0%**	–	0,0%	–	–	100,0%	–
Vid. Hb konc. eritrocite (MCHC)	–	–	–	–	–	–	–	–
Anizocitozė	–	–	–	–	–	–	–	–
Trombocitai	**65**	**98,5%**	–	76,9%	100,0%	**76,9%**	0,0%	14,0%
Vid. trombocitų tūris (MPV)	–	–	–	–	–	–	–	–
Leukocitai	**95**	**97,9%**	–	88,4%	100,0%	**88,4%**	1,1%	32,1%
Eozinofilai	**3**	**100,0%**	–	100,0%	100,0%	**100,0%**	33,3%	0,0%
Bazofilai	**3**	**100,0%**	–	100,0%	100,0%	**100,0%**	0,0%	0,0%
Neutrofilai	**6**	**100,0%**	–	83,3%	100,0%	**83,3%**	16,7%	100,0%
Limfocitai	**8**	**100,0%**	–	87,5%	100,0%	**87,5%**	12,5%	85,7%
Monocitai	**5**	**100,0%**	–	80,0%	100,0%	**80,0%**	20,0%	0,0%
Retikuliocitai	–	–	–	–	–	–	–	–
Eritrocitų nusėdimo greitis (ENG)	**14**	**92,9%**	–	50,0%	100,0%	**50,0%**	23,1%	57,1%
C reaktyvinis baltymas (CRB)	**14**	**100,0%**	–	35,7%	100,0%	**35,7%**	50,0%	80,0%
Viso	**374**	**97,6%**		78,3%	100,0%	**78,3%**	6,8%	31,4%

Iš 9 lentelės matyti, kad terminų atpažinimo prasme automatinė analizė dirba gana gerai – daugumoje randama nuo 92% iki 98% visų įrašuose esančių ieškomų terminų. Konvertavimo (matavimo iš tekstinės eilutės į skaičių vertimo) operacijos kokybė yra prastesnė. Vienuose iš labiausiai paplitusių matavimų – hemoglobinas, trombocitai konvertavimas įvykdytas sėkmingai tik atitinkamai 71.6% ir 76.9% nuo visų esamų reikšmių.



24 pav. Automatinio liktinių duomenų išgavimo patikimumo ir efektyvumo tyrimas – pirmojo sistemos varianto rezultatai: iš kiek įvairių koncepcijų matavimų buvo aptikta teisingai.

Išnagrinėjus nepavykusių konversijų atvejus pastebėta, kad daugumoje tekstinės eilutės vertimas į skaičių nepavyksta dėl pernelyg paprastos konversijos operacijos – atmetus terminą, skyriklį tarp termino ir matavimo vienetus bandoma konversija su likusiu tekstu. Tokiu būdu išvengiama dalies neteisingų matavimų išgavimo, atmetant visas eilutes, kurios prieš tai vykdyto analizės žingsnio metu buvo blogai suskaidytos pagal koncepcijas, skyriklius. Todėl visi išgauti matavimai yra teisingi (9 lentelė, „Išgauta matavimų – Teisingai“ visos eilutės – 100%).

Dažniausiai (apie 70% nepavykusių konversijų atvejų) konvertuojamas tekstas dar turi papildomų simbolių, likusių iš prieš tai vykdyto apdorojimo. Tai yra tiek brūkšneliai, tiek taškai, kurių neatpažįsta ir neišskiria kaip skyriklių tekstinę eilutę kableliais skaidantis analizės algoritmas. Pagerinus šį skaidymą, arba konversijos algoritmą ir išvengus apie 70% paprastos konversijos daromų klaidų, teisingai išgaunamų matavimų kiekis (hemoglobino, trombocitų) turėtų pagėrėti apie 20% (maždaug iki 90%).

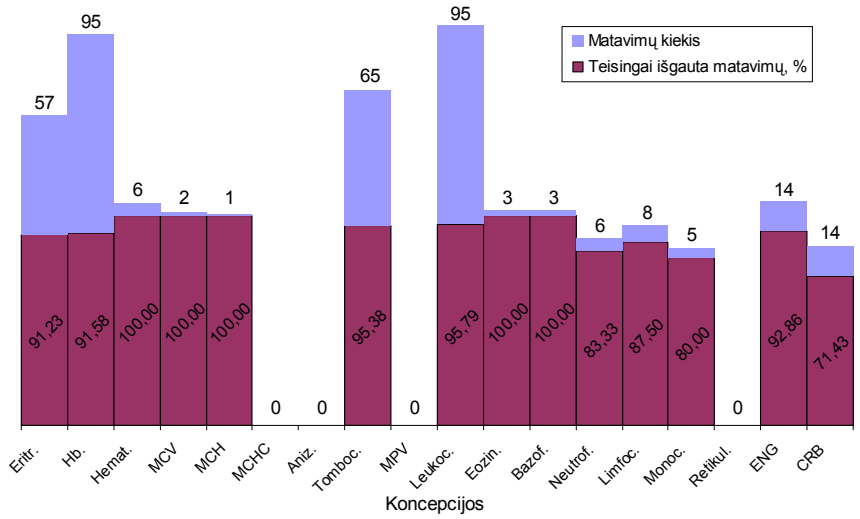
Eksperimento meto pasirinkta gerinti konversijos iš tekstinės eilutės į skaičių algoritmą. Atlikti du pakeitimai: pirma, pakeičiamas tekstinės eilutės „terminas–matavimas“ apdorojimas prieš konversiją – vietoje to, kad iškirpti terminą iš šios eilutės, atmetama visa eilutės pradžia iki paskutinio termino simbolio, antra, konversijos komanda yra atliekama ne tiesiogiai verčiant tekstinę eilutę į skaičių, bet pasinaudojant reguliariosiomis išraiškomis –

ieškant skaičiaus tekstinėje eilutėje. Atlikus tokius pakeitimus programos kode (nes esama analizės šablonų realizacija neleidžia tokio žemo lygmens pakeitimų), gauti nauji duomenų išgavimo rezultatai, kurie pateikti 10 lentelėje.

10 lentelė. Automatinio liktinių duomenų išgavimo patikimumo ir efektyvumo tyrimas – palyginimas su ekspertine analize: antrasis sistemos variantas. Duomenų imtis – 100 įrašų. Stulpelis „Pagerinta“ rodo, kiek procentų padidėjo teisingų matavimų išgavimas lyginant su pirmuoju sistemos variantu

Koncepcija	Ekspertinė analizė (duomenų kiekis N_e)	Automatinis porų „*terminas–matavimas*“ išgavimas						
		Rasta terminų		Išgauta matavimų				
		Teisingai	Klaidingai	Viso	Teisingai	Viso išgauta teisingai (E_e)	Pagerinta	Kiti duom. Už normos ribų
Eritrocitai	**57**	**94,7%**	1,8%	93,0%	98,1%	**91,2%**	**1,75%**	56,6%
Hemoglobinas	**95**	**97,9%**	–	94,7%	96,7%	**91,6%**	**20,00%**	18,9%
Hematokritas	**6**	**100,0%**	–	100,0%	100,0%	**100,0%**	**16,67%**	16,7%
Vid. eritr. tūris (MCV)	**2**	**100,0%**	–	100,0%	100,0%	**100,0%**	**50,00%**	100,0%
Vid. Hb kiekis eritrocite (MCH)	**1**	**100,0%**	–	100,0%	100,0%	**100,0%**	**100,0%**	100,0%
Vid. Hb konc. eritrocite (MCHC)	–	–	–	–	–	–	–	–
Anizocitozė	–	–	–	–	–	–	–	–
Trombocitai	**65**	**98,5%**	–	96,9%	98,4%	**95,4%**	**18,46%**	12,7%
Vid. tromboc. tūris (MPV)	–	–	–	–	–	–	–	–
Leukocitai	**95**	**97,9%**	–	97,9%	97,8%	**95,8%**	**7,37%**	35,5%
Eozinofilai	**3**	**100,0%**	–	100,0%	100,0%	**100,0%**	**0,00%**	0,0%
Bazofilai	**3**	**100,0%**	–	100,0%	100,0%	**100,0%**	**0,00%**	0,0%
Neutrofilai	**6**	**100,0%**	–	100,0%	83,3%	**83,3%**	**0,00%**	100,0%
Limfocitai	**8**	**100,0%**	–	100,0%	87,5%	**87,5%**	**0,00%**	87,5%
Monocitai	**5**	**100,0%**	–	100,0%	80,0%	**80,0%**	**0,00%**	0,0%
Retikuliocitai	–	–	–	–	–	–	–	–
Eritr. nusėdimo greitis (ENG)	**14**	**92,9%**	–	92,9%	100,0%	**92,9%**	**42,86%**	30,8%
C reaktyvinis baltymas (CRB)	**14**	**100,0%**	–	92,9%	76,9%	**71,4%**	**35,71%**	69,2%
Viso	**374**	**97,6%**		96,0%	96,4%	**92,5%**	**14,17%**	32,9%

Kaip matyti iš 10 lentelės, atlikus pakeitimus konvertavimo operacijoje gautas žymus išgaunamųjų duomenų kiekio padidėjimas (stulpelis „Pagerinta“). Su šiuo padidėjimu, sumažėjo išgaunamų duomenų teisingumas (stulpelis „Išgauta matavimų – Teisingai“), tačiau stebint bendrus rezultatus – stulpelis „Viso išgauta teisingai“ – galima teigti, kad daugumoje atvejų toks išgaunamų duomenų teisingumo sumažėjimas nėra svaresnis už bendrą išgautų duomenų patikimumą. Stulpelis „Viso išgauta teisingai“ rodo, kiek bus galima pasitikėti tam tikro matavimo išgavimų skaičiumi atlikus eksperimentą su visa liktine duomenų baze. Antruoju sistemos variantu gautas teisingai išgaunamų matavimų grafikas pateiktas 25 paveiksle.



25 pav. Automatinio liktinių duomenų išgavimo patikimumo ir efektyvumo tyrimas – antrojo sistemos varianto rezultatai: iš kiek įvairių koncepcijų matavimų buvo aptikta teisingai.

Atlikus tyrimą su maža įrašų imtimi ir patikrinus išgaunamos informacijos patikimumą bei efektyvumą, atliekamas visos liktinės duomenų bazės migravimo eksperimentas. Šio eksperimento pagrindinės statistikos:

- Liktinėje sistemoje analizuotinų įrašų yra 11443. Iš jų apie 83% (9490) diagnostinės informacijos laukų „diag_tyr“ yra užpildyti. Pradinis transformavimas iš liktinės duomenų bazės į neapdorotų duomenų XML bylas užtruko 1 min 55 s (99.5 lentelių egzemplioriai/s)[2].
- Atliktas automatinės analizės ir migracijos procesas (be medicinos eksperto), kurio metu analizuoti 9490 laukai, užtruko apie 15 minučių, kas yra 10.5 įrašų/s (apie 350 kartų greičiau nei ekspertinė analizė). Proceso metu sukurta apie 8 × 11443 XML bylų (neapdoroti duomenys, išgavimo protokolas, pseudo ESI ir kt.)

[2] Šie ir kiti spartos duomenys gauti dirbant su vienu kompiuteriu: procesorius AMD Athlon XP 1800+ 1.53 GHz, 1Gb operatyviosios atmintinės.

11 lentelė. Visos liktinės duomenų bazės automatinio duomenų migravimo eksperimento statistiniai duomenys

Koncepcija	Viso rasta terminų (K_v)	Išgauta matavimų			Kiti duomenys			
		I sistemos variantas	II sistemos variantas	Pagerinta	Nerasta mat. vienetų		Už normos ribų	
					I s. v.	II s. v.	I s. v.	II s. v.
Eritrocitai	**5296**	89,56%	98,30%	**8,74%**	4,15%	4,15%	46,45%	48,37%
Hemoglobinas	**7617**	72,13%	96,69%	**24,56%**	2,60%	2,60%	12,67%	13,58%
Hematokritas	**682**	90,62%	99,41%	**8,80%**	63,49%	63,49%	17,96%	20,65%
Vid. eritrocito tūris (MCV)	**117**	92,31%	99,15%	**6,84%**	100,0%	100,0%	25,00%	26,72%
Vid. Hb kiekis eritrocite (MCH)	**100**	92,00%	100,0%	**8,00%**	99,00%	99,00%	20,65%	23,00%
Vid. Hb konc. eritrocite (MCHC)	**38**	73,68%	100,0%	**26,32%**	100,0%	100,0%	100,0%	100,0%
Anizocitozė	**0**	–	–	–	–	–	–	–
Trombocitai	**5616**	76,67%	94,12%	**17,45%**	1,03%	1,03%	12,22%	13,30%
Vid. tromboc. tūris (MPV)	**3**	66,67%	100,0%	**33,33%**	100,0%	100,0%	50,00%	33,33%
Leukocitai	**7667**	86,29%	99,22%	**12,93%**	1,51%	1,53%	34,33%	36,82%
Eozinofilai	**324**	89,81%	99,07%	**9,26%**	33,02%	33,33%	60,82%	60,75%
Bazofilai	**235**	91,06%	98,72%	**7,66%**	7,66%	7,66%	6,07%	7,76%
Neutrofilai	**536**	89,55%	98,51%	**8,96%**	41,98%	41,98%	98,75%	98,48%
Limfocitai	**557**	88,33%	96,59%	**8,26%**	40,22%	40,75%	58,74%	60,59%
Monocitai	**390**	90,51%	99,49%	**8,97%**	37,69%	37,69%	54,11%	54,64%
Retikuliocitai	**34**	67,65%	94,12%	**26,47%**	20,59%	20,59%	26,09%	31,25%
Eritrocitų nusėdimo greitis (ENG)	**1157**	43,22%	96,63%	**53,41%**	25,41%	25,41%	31,60%	30,77%
C reaktyvinis baltymas (CRB)	**1367**	25,09%	91,59%	**66,50%**	61,74%	61,81%	67,06%	52,40%
Viso	**31736**	77,84%	97,08%	**19,24%**	9,92%	9,94%	30,04%	30,95%

– Automatinės analizės metu atlikta 31736 konvertavimo operacijų iš teksto eilučių į skaitines reikšmes. Iš jų (su antruoju sistemos variantu) apie 97.1% (30808 konversijų) buvo sėkmingos. Pirmuoju sistemos variantu

gauta 77.84% (24703) sėkmingų konversijų. Visi su konversijomis susiję statistiniai duomenys pateikti 11 lentelėje.

– Grafinis išgautų duomenų pateikimas dalykinės srities specialistui užtrunka apie 3 min 40 s, įkeliant 11443 ESI (iš dvigubo modelio informacinės sistemos) į kompiuterio atmintinę (apie 52 ESI/s). Atitinkamų duomenų išsaugotų buferyje įkėlimo laikas praktiškai lygus nuliui.

11 lentelėje stulpelis „Viso rasta terminų“ nurodo kiekį terminų, rastų analizuojant 9490 netuščių įrašų. Stulpelyje „Išgauta matavimų“ pateikti pirmosios ir antrosios sistemų variantų išgautų matavimų kiekiai, bei kiek procentų antrasis sistemos variantas pagerino pirmojo varianto išgavimo efektyvumą. Stulpelyje „Kiti duomenys“ pateikiami abiejų sistemų gautos papildomos statistikos – kiek „terminas–reikšmė“ porų neturėjo matavimo vienetų ir kiek matavimų nepateko į klinikinio kraujo tyrimo normatyvų ribas (normatyvai pagal [35] pateikti skyriuje **9.2.3.1**).

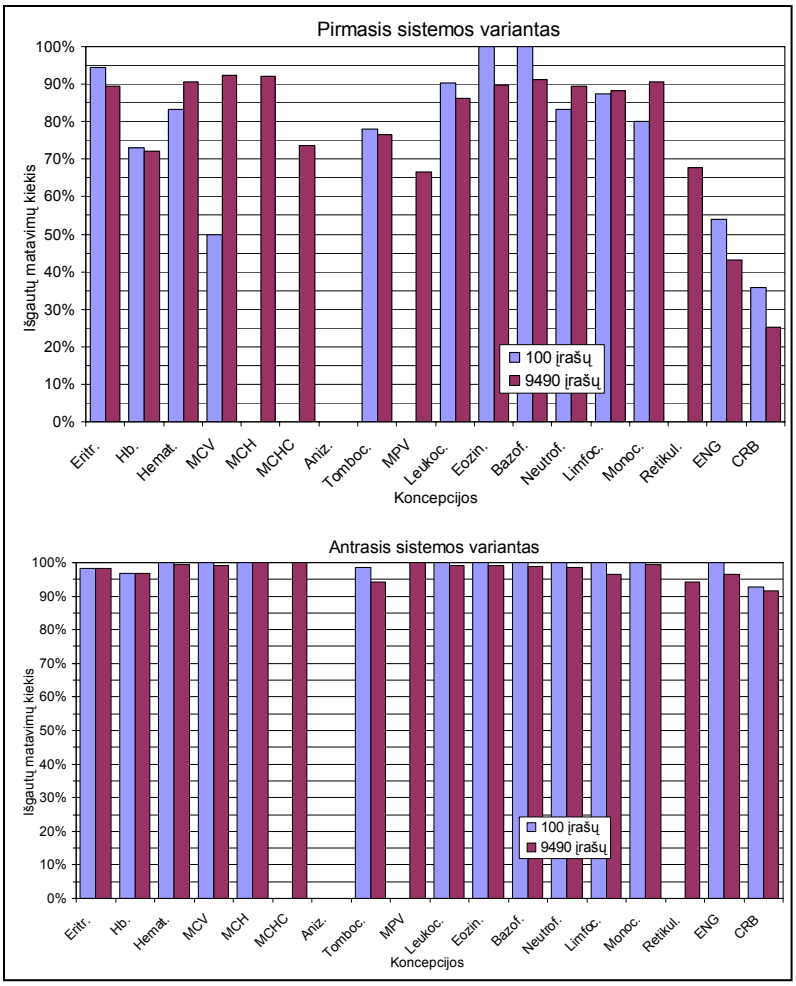
12 lentelė. Automatinio duomenų migravimo eksperimento statistiniai duomenys: išgautų matavimų kiekių palyginimas esant 100 ir 9490 įrašų imtims su skirtingais sistemos variantais.

Koncepcija	I sistemos variantas		II sistemos variantas	
	100 įrašų F_e	**9490 įrašų F_v**	**100 įrašų F_e**	**9490 įrašų F_v**
Eritrocitai	94,44%	89,56%	98,15%	98,30%
Hemoglobinas	73,12%	72,13%	96,77%	96,69%
Hematokritas	83,33%	90,62%	100,00%	99,41%
Vid. eritrocito tūris (MCV)	50,00%	92,31%	100,00%	99,15%
Vidutinis Hb kiekis eritrocite (MCH)	–	92,00%	100,00%	100,00%
Vid. Hb koncentracija eritrocite (MCHC)	–	73,68%	–	100,00%
Anizocitozė	–	–	–	–
Trombocitai	78,13%	76,67%	98,44%	94,12%
Vid. tromboc. tūris (MPV)	–	66,67%	–	100,00%
Leukocitai	90,32%	86,29%	100,00%	99,22%
Eozinofilai	100,00%	89,81%	100,00%	99,07%
Bazofilai	100,00%	91,06%	100,00%	98,72%
Neutrofilai	83,33%	89,55%	100,00%	98,51%
Limfocitai	87,50%	88,33%	100,00%	96,59%
Monocitai	80,00%	90,51%	100,00%	99,49%
Retikuliocitai	–	67,65%	–	94,12%
Eritrocitų nusėdimo greitis (ENG)	53,85%	43,22%	100,00%	96,63%
C reaktyvinis baltymas (CRB)	35,71%	25,09%	92,86%	91,59%
Viso	80,27%	77,84%	98,36%	97,08%

Atlikus ekspertinį 100 įrašų imties tyrimą, kiekvienai koncepcijai gautas įvertis N_e, kuris reiškia identifikuotą koncepcijos kiekį tirtuose laukuose. Atliekant koncepcijų išgavimą automatiniu būdu (vėlgi su 100 įrašų) gaunami įverčiai K_e ir M_e. K_e reiškia automatikos atpažintų terminų kiekį, o M_e – išgautų matavimų kiekį ($N_e \geq K_e \geq M_e$).

Taigi, automatinių algoritmų išgavimo efektyvumą su 100 įrašų imtimi galima skaičiuoti pagal formulę:

$$E_e = M_e / N_e \quad (1)$$



26 pav. Automatinio liktinių duomenų išgavimo su skirtingomis duomenų imtimis rezultatai: išgautų matavimų kiekis nuo išgautų koncepcijų skaičiaus.

Pagal (1) gauti rezultatai pateikti lentelių 9, 10 stulpeliuose „Išgauta matavimų – Viso išgauta teisingai“. Tačiau atliekant eksperimentą su visa liktine sistema, realus koncepcijų kiekis N_v nėra žinomas. Todėl efektyvumo E_v pagal (1) paskaičiuoti negalima. Vietoje to, tam kad įvertinti, ar galima pasitikėti 100 įrašų tyrimo gautais rezultatais ir pagal juos spręsti apie visos liktinės sistemos migracijos proceso rezultatus, skaičiuojamas kitas įvertis, kuris reiškia, kiek matavimų buvo išgauta nuo rastų terminų skaičiaus:

$$F_e = M_e / K_e \quad (2)$$

Pagal (2) apskaičiuoti išgautų matavimų įverčiai abiems sistemos variantams pateikti 12 lentelėje. Iš grafinio šių rezultatų atvaizdavimo (26 pav.) matoma koreliacija tarp F_e ir F_v įverčių abiejų sistemos variantų atvejais. Tai leidžia daryti prielaidą, kad 100 įrašų (tik 1% visos duomenų imties) tyrimas yra korektiškas ir pagal šio tyrimo rezultatus galima spręsti apie visos liktinės sistemos migracijos proceso kokybę.

5. IŠVADOS

1. Ypač sudėtingoms dalykinėms sritims, tokioms kaip medicina, klasikiniai informacinių sistemų kūrimo metodai tinkami tik gana siauriems taikymams. Šiuolaikiniai medicininių informacinių sistemų standartai, tokie kaip HL7, EN 13606, GEHR ir kt., kuriami „dviejų lygių“ metodikos pagrindu, kuri loginį informacinės sistemos lygį išskaido į du lygmenis – žinių ir informacijos. Tokiose sistemose dalykinės srities žinios egzistuoja tik labai aukštam abstrakcijos lygyje, apibrėžiamame globaliomis terminologijomis ir archetipais. Tačiau kartu šios sistemos yra ir labai sudėtingos – minėtų standartų taikymas praktikoje nėra trivialus.

2. Klasikiniais būdais sukurtos ir siauriems taikymams skirtos medicininės informacinės sistemos dažnai neturi galimybių bendradarbiauti su kitomis sistemomis, o jose sukaupti duomenys egzistuoja tik kaip „kompiuterinis popierius“, kuriam informacinių technologijų teikiamos galimybės praktiškai yra netaikytinos.

3. Darbe pasiūlyta universali metodika, kaip iš siauriems taikymams skirtų liktinių informacinių sistemų su įvairia informacija užpildytais laukais automatizuotu būdu išgauti vertingus duomenis, iš jų suformuojant naują, dviejų lygių metodikos paremtą informacinę sistemą. Išgavimo procesas paremtas žiniomis, kurios saugomos naujosios informacinės sistemos žinių lygmenyje.

4. Pagal pasiūlytą metodiką sukurtas programinės įrangos prototipas liktiniams duomenims analizuoti ir išgauti. Prototipą sudaro trys dalys: liktinių duomenų adapteris, duomenų transformatorius ir validatorius. Pagrindinė dalis – duomenų transformatorius – atlieka sudėtingų, mažai struktūrizuotų liktinės sistemos laukų teksto analizę pagal sudarytą išgavimo šabloną, kuriame formaliai užrašomi teksto analizės žingsniai, kaip reikia skaidyt tam tikrą lauką, kokias žinias panaudoti paieškai ir pan. Išgavimo šablonas savyje turi apimti nuo pačių bendriausių iki pačių smulkiausių (tokių kaip teksto į skaičių konvertavimas) operacijų aprašymus.

5. Eksperimentas atliktas su Kauno kardiologijos klinikos duomenų baze (9490 netuščių analizuotinų įrašų iš 11443), analizuojant mažai struktūrizuotus laukus ir išgaunant laboratorinių tyrimų rezultatus. Gauti eksperimento rezultatai rodo, kad atsižvelgiant į išgaunamus duomenis ir jų specifiką, eilės nesudėtingų teksto analizės įrankių kombinavimas leidžia gan tiksliai ir patikimai išgauti daugumą duomenų.

6. Atliktas tyrimas rodo, kad išgaunama praktiškai visa (90%–98%) ieškoma informacija. Neišgauta automatiniu būdu informacija gali būti randama žmogaus atliekamo patikros proceso metu, pagal išgavimo algoritmo registruojamus įvairius klaidų ir perspėjimų panešimus. Automatizuota liktinės sistemos analizė (su pilna patikra) apie 4 – 5 kartus pagreitintų medicinos eksperto darbą išgaunant reikiamus duomenis. Be patikros šis procesas pagreitėja iki 350 kartų.

7. Sudaryta metodika bei programinė įranga yra universalios. Patobulinus prototipinę programinę įrangą (įgyvendinus pilną ADL kalbos interpretatorių ir kt.), ją galima taikyti ne tik medicinai, bet ir kitoms dalykinėms sritims, įvairioms informacinėms sistemoms, duomenims ir pan. Dalykinės srities žinios ir duomenų išgavimo šablonai (kurie yra derinami pagal turimus duomenis) gali keistis/būti keičiami nepriklausomai nuo programinės įrangos.

6. LITERATŪRA

1. Aronson, A. R. *Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program* [interaktyvus]. 2001 – [žiūrėta 2004-05-20]. Prieiga per internetą: http://skr.nlm.nih.gov/papers/references/metamap_01AMIA.pdf

2. Beale, T. *Archetypes: Constraint-based Domain Models for Future-proof Information Systems* [interaktyvus]. 2002 – [žiūrėta 2004-03-25]. Prieiga per internetą: http://www.deepthought.com.au/it/archetypes/archetypes.pdf

3. Bird, L., Goodchild, A., Heard, S. *Importing Clinical Data into Electronic Health Records - Lessons Learnt from the First Australian GEHR Trials* [interaktyvus]. 2001 – [žiūrėta 2004-05-05]. Prieiga per internetą: http://titanium.dstc.edu.au/papers/HIC2002.pdf

4. Chu, St., Cesnik, B. *Knowledge representation and retrieval using conceptual graphs and free text document self-organisation techniques* [interaktyvus]. Iš International Journal of Medical Informatics. 2001 – [žiūrėta 2004-04-20]. Prieiga per internetą: http://linkinghub.elsevier.com/retrieve/pii/S1386505601001563

5. *CEN Strategy: 2010* [interaktyvus]. European Committee for Standardization, 1998 – [žiūrėta 2004-03-31]. Prieiga per internetą: http://www.cenorm.be/cenorm/aboutus/generalities/strategy/strategy.pdf

6. Standartas *DICOM - Digital Imaging and Communications in Medicine* [interaktyvus]. National Electrical Manufacturers Association, [žiūrėta 2004-06-18]. Prieiga per internetą: http://medical.nema.org/

7. *Draft Standard Specification for Continuity of Care Record* [interaktyvus]. European Committee for Standardization, 2000 – [žiūrėta 2004-05-18]. Prieiga per internetą: http://www.centc251.org/WGII/N-03/ASTM_Draft_Standard Specification for Continuity of Care Record_CCR E31.28_DRAFT 2.02_6_25-031.doc

8. *Draft initial report version 0.4 of Task Force EHRCOM – Revision of ENV 13606: Electronics Health Record Communication*. CEN/TC 251 Health Informatics Technical

Committee, Committee European Normalisation, 2002 – [žiūrėta 2004-04-10]. Prieiga per internetą: http://www.centc251.org/tcmeet/doclist/TCdoc02/N02-032rev.pdf

9. Projektas *EHCR-SupA – Electronic Healthcare Record Support Action* [interaktyvus], [žiūrėta 2004-06-15]. Prieiga per internetą: http://www.chime.ucl.ac.uk/work-areas/ehrs/EHCR-SupA/

10. *Electronic healthcare record communication - Part 1: Extended architecture* [interaktyvus]. ENV13606-1, CEN/TC 251 Health Informatics Technical Committee, European Committee for Standardization, 1999. 111 p.

11. *Electronic healthcare record communication - Part 2: Domain Term List* [interaktyvus]. ENV13606-2, CEN/TC 251 Health Informatics Technical Committee, European Committee for Standardization, 1999. 62 p.

12. Embley, D. W., Campbell, D. M., Smith, D. R., Liddle, W. S. *Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Documents* [interaktyvus], 1998 – [žiūrėta 2004-03-10]. Prieiga per internetą: http://osm7.cs.byu.edu/deg/papers/cikm98.pdf

13. *ERDIP EHR Model Options Report: EHR Definition Project – Discussion Document, version 1.1* [interaktyvus]. NHS Information Authority. 2001, [žiūrėta 2004-03-25]. Prieiga per internetą: http://www.nhsia.nhs.uk/erdip/pages/docs_egif/evaluation/technical/ehrmodopt.pdf

14. *European Standardization of Health Informatics - Technical Committee 251* [interaktyvus], [žiūrėta 2004-04-08]. Prieiga per internetą: http://www.centc251.org/

15. *Extensible Markup Language (XML)* [interaktyvus], [žiūrėta 2004-06-18]. Prieiga per internetą: http://www.w3.org/XML/

16. *Final Report: The Oacis-GEHR Transformation Process* [interaktyvus]. 2001 – [žiūrėta 2004-04-26]. Prieiga per internetą: http://www.gpcg.org/publications/docs/projects2001/GPCG_Project2_01.pdf

17. *Design Patterns: Elements of Reusable Object-Oriented Software* / Eric Gamma, Richard Helm, Ralph Johnson, John Vlissides. – Boston: Addison-Wesley Pub Co, 1997. - 395 p.

18. Projektas *Good European Health Record / Good Eletronic Health Record* [interaktyvus], [žiūrėta 2004-05-31]. Prieiga per internetą: Europa – http://www.chime.ucl.ac.uk/work-areas/ehrs/GEHR/; Australija – http://titanium.dstc.edu.au/gehr/.

19. *Health informatics - Electronic healthcare record communication - Part 2: Archetype Interchange Specification (Draft 17/09/2003)* [interaktyvus]. prEN 13606-2, CEN/TC 251 Health Informatics Technical Committee, European Committee for Standardization, 2003.

20. Projektas *Health Level Seven (HL7)* [interaktyvus], [žiūrėta 2004-07-25]. Prieiga per internetą: www.hl7.com

21. Herzig, W. T., Johns, M. *Extraction of medical information from textual sources: a statistical variant of the Boundary Word method* [interaktyvus]. 1997, [žiūrėta 2004-04-20]. Prieiga per internetą: http://www.amia.org/pubs/symposia/D004065.PDF

22. Kalra, D., Lloyd, D., Darlinson, M., Mori, A. R. *Headings for Communicating Information for the Personal Health Record* [interaktyvus]. CHIME, UCL Medical School, 1998, [žiūrėta 2004-04-16]. Prieiga per internetą: http://www.nhsia.nhs.uk/headings/pdf/chime1.pdf

23. *Key Capabilities of an Electronic Health Record System - Letter Report*. Board on Health Care Services, Institute of Medicine of The National Academies, USA [interaktyvus]. 2003 – [žiūrėta 2004-03-25]. Prieiga per internetą: http://books.nap.edu/html/ehr/NI000427.pdf

24. Projektas *Konsultacinė pagalba dėl sveikatos informacijos poreikio, informacinių sistemų / technologijų infrastruktūros visuose sveikatos priežiūros lygiuose įvertinimo, Ataskaita B* [interaktyvus], [žiūrėta 2005-04-21]. Prieiga per internetą: http://www.sam.lt/images/Dokumentai/eSveikata/IKTanalize2003/ataskaitab.pdf

25. *Market, environment and objectives of CEN/TC 251 – Health Informatics as approved by resolution BTC 39/2000* [interaktyvus]. 2000 – [žiūrėta 2004-04-06]. Prieiga per internetą: http://www.cenorm.be/nr/cen/doc/PDF/6232.pdf

26. Maskeliūnas, S. *Ontologijų panaudojimas verslo ir informacinėms sistemoms intelektualizuoti* [interaktyvus]. 2003 – [žiūrėta 2004-03-10]. Prieiga per internetą: http://www.ktu.lt/lt/mokslas/konf03/konf_02/IT2003/Sekcija03.pdf

27. *Medicina ir farmacija Lietuvoje* [interaktyvus], [žiūrėta 2004-09-27]. Prieiga per internetą: http://www.medicine.lt/

28. Projektas *openEHR* [interaktyvus], [žiūrėta 2004-07-20]. Prieiga per internetą: http://www.openehr.org/

29. Rector, A. L. *Clinical Terminology: Why Is It So Hard?* [interaktyvus]. 2001 – [žiūrėta 2004-06-10]. Prieiga per internetą: http://www.med.uni-heidelberg.de/mi/education/mi/mdoc/Q4 Clinical Terminology - Why is it so hard (Rector).pdf

30. Rector, A.L., Nolan, W.A., Kay, S. *Foundations for an Electronic Medical Record* [interaktyvus]. 1991 – [žiūrėta 2004-04-10]. Prieiga per internetą: http://www.cs.man.ac.uk/mig/ftp/pub/papers/alr-foundations.pdf

31. Shortliffe, E. H. *The Evolution of Health-Care Records in the Era of the Internet* [interaktyvus]. Stanford Medical Informatics (SMI), Stanford University, Stanford, California USA. [žiūrėta 2004-04-12]. Prieiga per internetą: http://www-smi.stanford.edu/pubs/SMI_Reports/SMI-98-0740.pdf

32. *Standards Requirements for the Electronic Health Record & Discharge/Referral Plans - Final Report* [interaktyvus]. ISO/TC 215 Ad Hoc Group, 2002 – [2004-03-14]. Prieiga per internetą: http://secure.cihi.ca/cihiweb/en/downloads/infostand_ihisd_isowg1_finalreportJan03_e.pdf

33. Projektas *Synapses* [interaktyvus], [žiūrėta 2004-07-26]. Prieiga per internetą: http://www.cs.tcd.ie/synapses/public/

34. Projektas *SynEx – Synergy on the Extranet* [interaktyvus], [žiūrėta 2004-05-07]. Prieiga per internetą: http://www.chime.ucl.ac.uk/HealthI/SynEx

35. *Systematized Nomenclature of Medicine (SNOMED)* [interaktyvus], College of American Pathologists, [žiūrėta 2005-04-10]. Prieiga per internetą: http://www.snomed.org

36. *Tarptautinė statistinė ligų ir sveikatos problemų klasifikacija (dešimtoji redakcija)* [interaktyvus], [žiūrėta 2004-10-15] Prieiga per internetą: http://www.lsic.lt/tlk/

7. SUMMARY

ANALYSIS OF AUTOMATIC DATA EXTRACTION
FROM MEDICAL DOCUMENTS

Automatic data extraction from medical legacy systems into archetype-based systems is analyzed, developed and tested in this work. Electronic health record system (EHRS) is a must in today's healthcare environment. Lots of up-to-date medical systems are still built with classic development approaches, with semantics hard coded into systems. Modern EHRS standards propose new "two-level" methodology, which is based on separation of knowledge and information levels. This work suggests a methodology for heterogenic medical legacy systems that exist today to be transformed into ones, built with "two-level" methodology. Transformation is based on knowledge, residing in new system. By creating a comprehensive transformation scheme, it is possible to analyze and extract relevant data from semi structured or unstructured text fields with mixed information. Suggested methodology is tested with software prototype by extracting laboratory results of clinical blood test from semi structured fields of cardiology database. Achieved results are about 95% of data successfully transferred from legacy system. This approach preserves medical data accumulated during long years of work and transforms it into more useful form, creating structured data from unstructured text fields. It allows an automatic means of information technologies to be used by medicine expert to analyze and interpret legacy data (draw charts, calculate statistics and so on).

8. TERMINŲ IR SANTRUMPŲ ŽODYNAS

ADL (*Archetype Definition Language*) – archetipų aprašymo kalba. Tai formali kalba aprašyti archetipams, kurie yra apribojimais pagrįsti dalykinės srities modeliai.

ARCHETIPAS (*archetype***) –** tai formali atskiros koncepcijos iš dalykinės srities išraiška, apibrėžta duomenų apribojimais. Archetipais apriboti duomenų egzemplioriai atitinka duotąją atraminę architektūrą.

ATRAMINĖ ARCHITEKTŪRA (*reference architecture***)** – tai nedidelis bendrinių klasių modelis, iš kurių konstruojami dalykinės srities duomenų egzemplioriai (su archetipais nusakoma semantika).

DBVS – duomenų bazių valdymo sistema. Tai dažniausiai kompiuterizuota ir automatinė sistema, skirta valdyti dideliems duomenų kiekiams.

EKG – elektrokardiograma. Žmogaus širdies elektrinių impulsų įrašas, naudojamas širdies ligoms aptikti ir tirti.

ESI – elektroninė sveikatos istorija. Tai bet kokia informacija, susijusi su praeities, dabarties ir ateities fizine ir psichine sveikata ar būsena žmogaus, priklausančio elektroninių sveikatos istorijų sistemai (sistemoms), kuri skirta fiksuoti, perduoti, priimti, kaupti, sujungti, ir manipuliuoti įvairialypės terpės duomenimis, siekiant pagrindinio tikslo – teikti sveikatos apsaugos paslaugas.

„KOMPIUTERINIS POPIERIUS“ – tai duomenys elektroninėje formoje, pritaikyti skaityti ir suprasti žmogui, bet ne kompiuteriui. Tai informacinės sistemos ar jų dalys, kuriose saugomi duomenys nėra lengvai panaudotini taikant automatinius duomenų apdorojimo įrankius. Paprastai tai yra mažai struktūrizuoti ar nestruktūrizuoti tekstiniai laukai su mišria informacija.

LIKTINĖ SISTEMA (*legacy system*) – tai bet kokia informacinė sistema, kurią yra labai sudėtingą, arba praktiškai neįmanoma modifikuoti, taikant ir derinant prie naujų ir vis besikeičiančių srities reikalavimų.

REGULIARIOSIOS IŠRAIŠKOS (*regular expressions*) – tai šablonas, apibrėžiantis aibę tekstinių eilučių, neišvardinant jų.

XML (*Extensible Markup Language*) – universali žymėjimo meta kalba, skirta kurti žymėjimo kalbas konkretiems taikymams.

9. PRIEDAI

9.1. Darbas su prototipine programine įranga

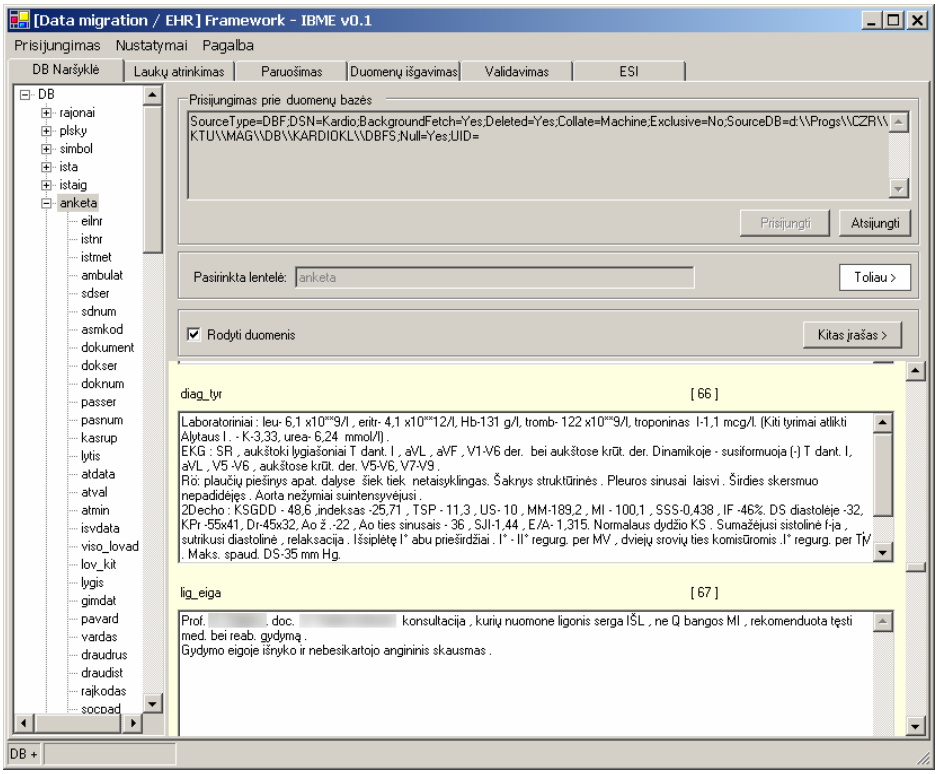
Šiame skyriuje pateikti darbo su prototipine programine įranga žingsniai pagal sudarytą automatizuotą archetipais pagrįstą liktinių duomenų išgavimo metodiką.

9.1.1. Analizė

Tai pirmasis metodikos žingsnis, kurio pagrindinis tikslas – pradinių duomenų paruošimas liktinės sistemos migracijos procesui.

9.1.1.1. Liktinės sistemos analizė

Pirmoji prototipinės programinės įrangos kortelė – „DB naršyklė“ (27 pav.). Šioje kortelėje galima jungtis prie nurodyto duomenų šaltinio (konkrečioje darbo realizacijoje – tam tikros ODBC duomenų bazės). Prisijungus prie duomenų bazės galima analizuoti jos struktūrą ir duomenis.



27 pav. Eksperimentinė prog. įranga. Duomenų bazės peržiūros grafinė vartotojo sąsaja.

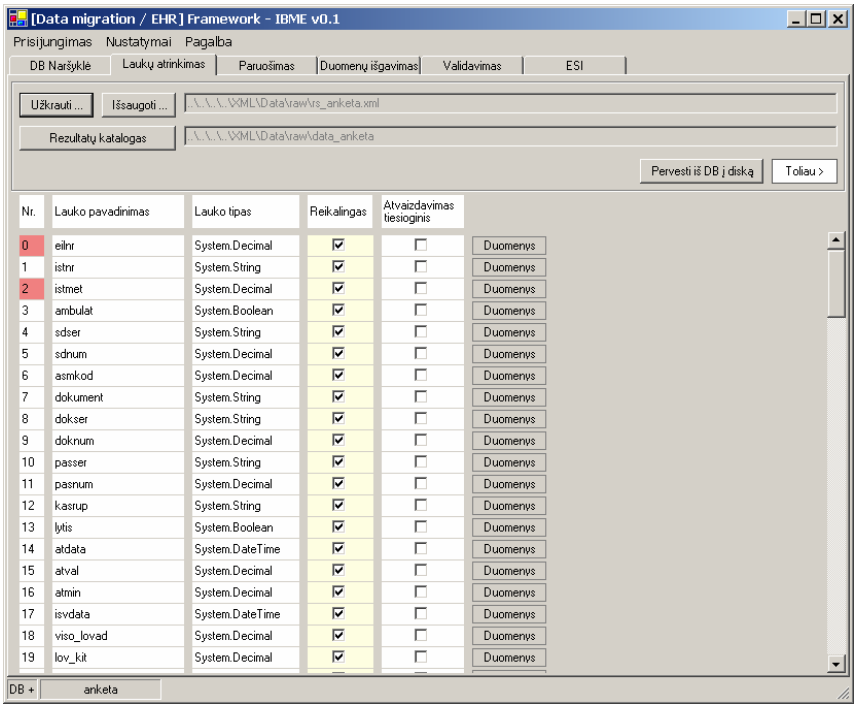
Antroji programinės įrangos kortelė – „Laukų atrinkimas“ (28 pav.). Šioje kortelėje yra išvardinti visi pasirinktos lentelės (šiuo atveju – „*anketa*“) laukai. Raudonai pažymima (pelės spragtelėjimu ant lauko numerio) raktiniai lentelės laukai, bei atitinkamose kolonėlėse „Reikalingas“ ir „Atvaizdavimas tiesioginis“ – ar duomenys reikalingi ir ar jie yra tiesiogiai

atvaizduojami. Sudarius šią schemą ji yra išsaugojama XML formato byloje. Kitą kartą nagrinėjant tą pačią lentelę programa automatiškai ras sudarytąją schemą ir ją užkraus.

Žemiau yra pateikta ištrauka iš svarbių duomenų schemos XML formate (visa ši schema yra pateikta **9.2.4.1 Svarbių duomenų schema (XML)**):

```
1   <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
2   <table name="anketa">
3     <field name="eilnr" type="System.Decimal" nr="0" isRelevant="true" isDirect="false"
    isKey="true" />
4     <field name="istnr" type="System.String" nr="1" isRelevant="true" isDirect="false" />
5     <field name="istmet" type="System.Decimal" nr="2" isRelevant="true" isDirect="false"
    isKey="true" />
6     <field name="ambulat" type="System.Boolean" nr="3" isRelevant="true" isDirect="false" />
7   ……..
```

Antroje eilutėje pradedama aprašyti lentelės „*anketa*“ schema, o toliau išvardinami visi laukai, pvz., penktoje eilutėje yra aprašytas „*istmet*“ laukas, kurio tipas yra sveikas skaičius, jis yra svarbus mums, nėra atvaizduojamas tiesiogiai ir yra raktinis.



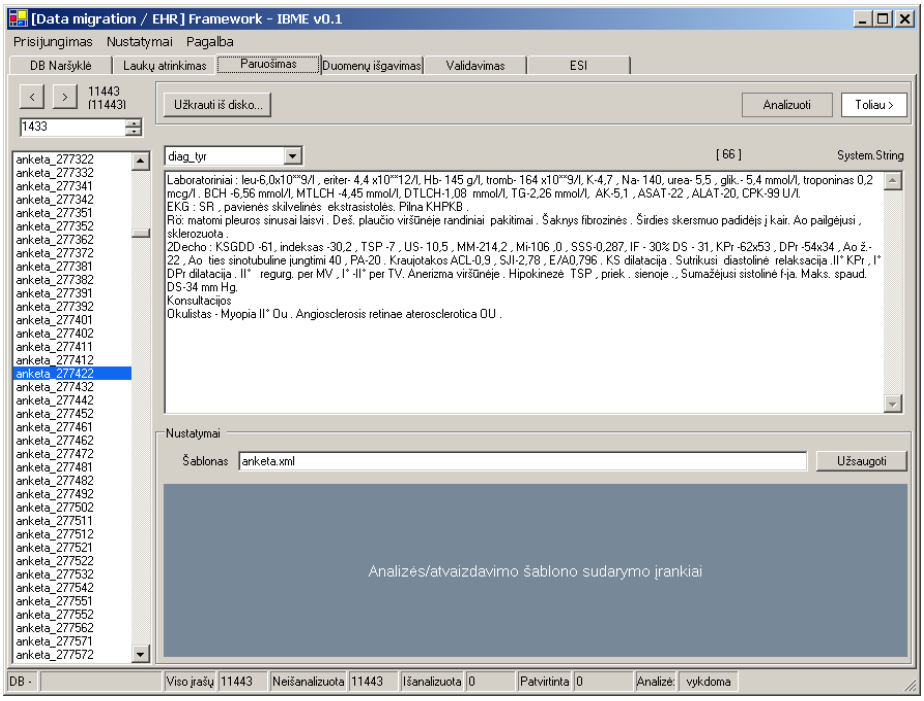
28 pav. Eksperimentinė prog. įranga. Svarbių duomenų schemos sudarymo grafinė vartotojo sąsaja.

Paspaudus mygtuką „Pervesti iš DB į diską“, visi lentelės duomenys pagal svarbių duomenų šabloną yra perkeliami į XML formato bylas, į tam tikrą nurodytą katalogą. Tai leidžia kartą persikėlus liktinius duomenis į *neapdorotas* XML bylas daugiau nenaudoti originalios liktinės saugyklos, kuri pvz., gali būti sunkiai prieinama ar pan. Tai įgalina pernešti liktinius duomenis bet kur. Neapdorotų duomenų XML bylos turinio ištrauka (visa byla yra pateikta **9.2.4.2 Lentelės „anketa“ egzempliorius (XML)**):

```
1   <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
2   <table name="anketa" nr="1747" key="anketa_r_163690">
3       <field name="eilnr" value="16369" />
4       <field name="istnr" value="16369" />
5       <field name="istmet" value="0" />
6       <field name="ambulat" value="False" />
7   .........
```

9.1.1.2. Migracijos schemos sudarymas

Trečioji prototipinės programinės įrangos kortelė „Paruošimas“ (29 pav.) yra skirta analizuoti duomenų laukus ir jiems nustatyti atitinkamus analizės žingsnius, paremtus žiniomis (t.y. sudaryti migracijos schemą). Prototipe grafinė vartotojo sąsaja teksto analizės žingsniams parinkti nebuvo kuriama, kadangi ji būtų labai sudėtinga. Migracijos schema XML formate buvo kuriama teksto redaktoriumi.



29 pav. Eksperimentinė prog. įranga. Migracijos šablono sudarymo grafinė vartotojo sąsaja.

Eksperimente sudarytoji migracijos schema (schemos paaiškinimas pateiktas 13 lentelėje):

```
1   <?xml version="1.0" encoding="utf-8" ?>
2   <template xmlns="http://tempuri.org/anketa1.xsd" id="1" tablename="anketa">
3       <element id="1" name="person data" archetype="kut-ibme-ehr-test.gpic-person.draft">
4           <field name="vardas" direct="at0001" />
5           <field name="pavard" direct="at0002" />
6           <field name="gimdat" direct="at0003" />
7           <field name="atdata" direct="at0004" />
8          <field name="lytis" direct="at0005" />
9       </element>
10      <element id="2" name="diagnostic tests">
11          <field name="diag_tyr">
12              <sequence nr="1" type="Separator|EOL" separator="\r\n|\n" />
13              <sequence nr="2" type="Default">
14                  <item       iid="1"       aid="kut-ibme-ehr-test.laboratory_results.draft"
    keyword="local::100" separator=":" preprocess=",[^0-9]" />
15                  <item iid="2" did="local::101" keyword="local::101" separator=":" />
16                  <item iid="3" did="local::102" keyword="local::102" separator=":" />
17              </sequence>
18          </field>   </element> </template>
```

13 lentelė. Eksperimente naudoto migracijos šablono paaiškinimas

Migracijos šablono eilutės numeris	**Aprašymas**
2	Šablono aprašomoji eilutė – joje nurodoma šablono schema (nebūtinai) ir litinės duomenų lentelės pavadinimas, kuriai bus taikomas šis šablonas. Visos kitos eilutės aprašo duomenų perkėlimo/analizės etapus.
3	Pirmasis lentelės „*anketa*“ analizės elementas. Jame nurodoma, kad analizė atliekama pagal žmogaus asmeninių duomenų archetipą (*kut-ibme-ehr-test.gpic-person.draft*). Tai reiškia, kad visi šiame elemente paminėti lokalūs terminai (at0001, at0002 ir kt.) yra iš šio archetipo.
4 – 8	Pirmojo lentelės „*anketa*“ analizės elemento turinys, susidedantis iš penkių laukų esybių. Viena lauko esybė (*field*) nurodo vieną lentelės lauką, o savo viduje aprašo kaip reikia analizuoti šį lauką. Šiuose penkiuose laukuose išvardinti lentelės laukai – vardas, pavard, gimdat, ir kt., kurie tiesiogiai atvaizduojami (*direct*=....) į tam tikrus 3 eilutėje minėtojo archetipo (žmogaus asm. duomenų) laukus (at0001 ir t.t.). Šis pavyzdys parodo paprastą (tiesioginį) lentelės laukų atvaizdavimą į archetipinę struktūrą.
10	Pradedamas antrasis lentelės „*anketa*“ analizės elementas.
11	Nurodoma, kad analizuojamas laukas „*diag_tyr*“.
12	Pirmasis analizės žingsnis – nurodoma, kad lauką reikia apdoroti tokiu būdu: suskaidyti jį pagal skyriklį (*separator*) „\r\n|\n“ (eilutės pabaiga). Šis apdorojimas reikalingas kadangi iš pradinių duomenų (psl. 52) matome, jog atskiros dalys (Laboratoriniai, EKG ir kt.) yra atskirtos eilutės pabaigos simboliu. Antrajam analizės žingsniui perduodamas suformuotas išgautų duomenų medis (praktiškai, jis būtų sudarytas iš 1–15 eilučių, kaip parodyta išgautų duomenų medžio pavyzdyje, skyriuje **9.2.5**).
13	Antrasis analizės žingsnis, turintis keletą analizės variantų, kurie taikomi visi paeiliui kiekvienai iš ateinančio išgautų duomenų medžio šakai, bandant atpažinti tam tikrą svarbią informaciją.

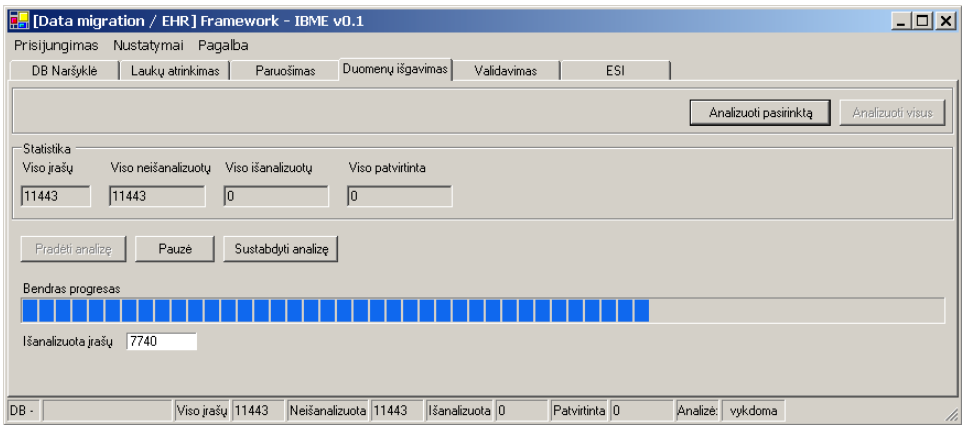
13 lentelės tęsinys

Migracijos šablono eilutės numeris	Aprašymas
14	Pirmasis analizės variantas – analizė atliekama pagal laboratorinių tyrimų archetipą (*kut-ibme-ehr-test.laboratory_results.draft*). Ši analizė atliekama tik tada, jei pradiniuose duomenyse yra randamas raktažodis, kurio identifikacija lokaliame žodyne yra 100 (raktažodis – „laboratoriniai“), atskirtas skyrikliu „:“. Jei šią sąlygą pradiniai duomenys tenkina, tada jie yra apdorojami pagal reguliariąją išraišką *preprocess* – šiuo atveju tai yra lauko suskaidymas per kablelius, paskui kuriuos nėra skaičiaus (pvz. „A – 6,0, B – 7,1“ būtų suskaidytas į „A – 6,0“ ir „B – 7,1“). Galų gale šiems suskaidytiems duomenims yra pritaikoma analizė pagal nurodytą archetipą.
15 – 16	Kiti analizės variantai, tik atpažįstantys tam tikrus duomenis pagal nurodytus raktažodžius.

9.1.2. Migracija

9.1.2.1. Migracijos procesas

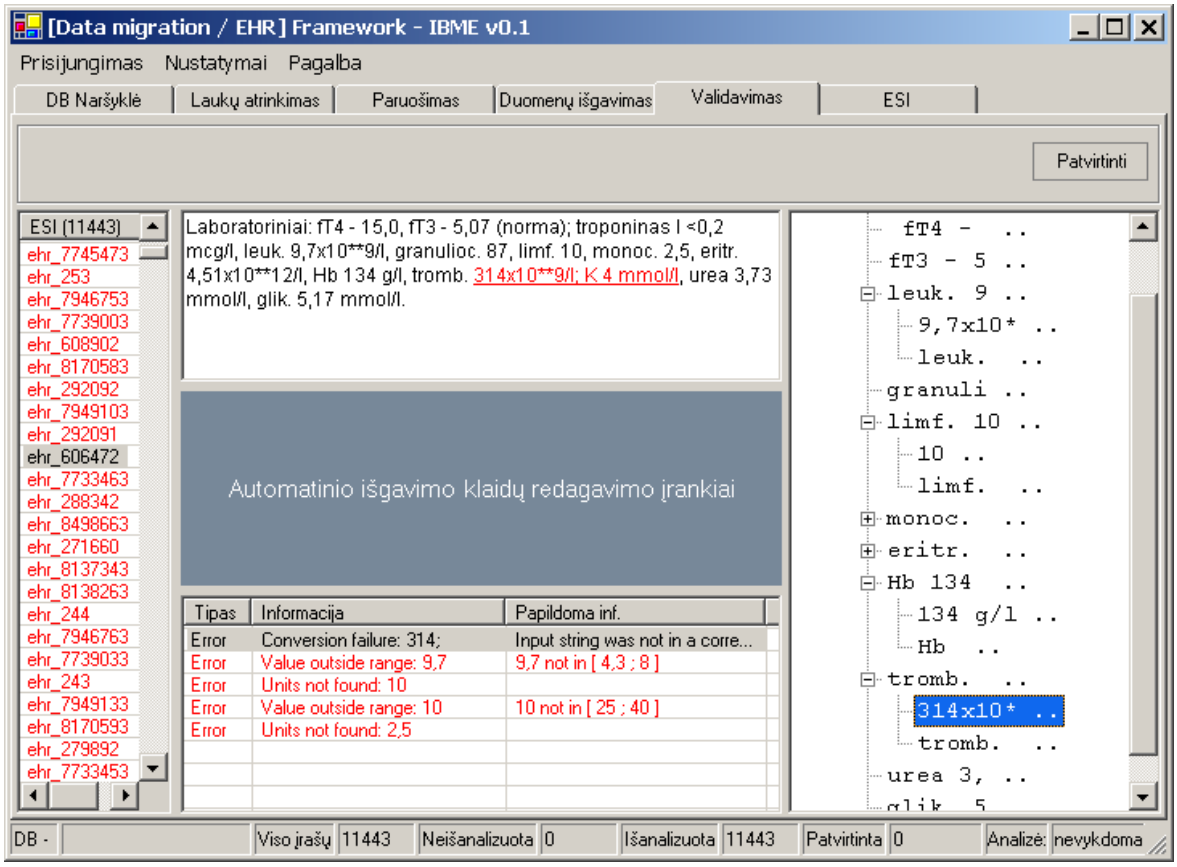
Ketvirtoji programinės įrangos kortelė – „Duomenų išgavimas“ (30 pav.) skirta valdyti bei stebėti liktinės duomenų bazės migravimo procesą. Joje pateikiami migracijos proceso paleidimo, stabdymo ir nutraukimo mygtukai, bei migracijos proceso būsena – kiek viso yra įrašų, kiek įrašų išanalizuota ir kt.



30 pav. Eksperimentinė prog. įranga. Migracijos proceso valdymo bei stebėjimo grafinė vartotojo sąsaja.

9.1.2.2. Validavimas

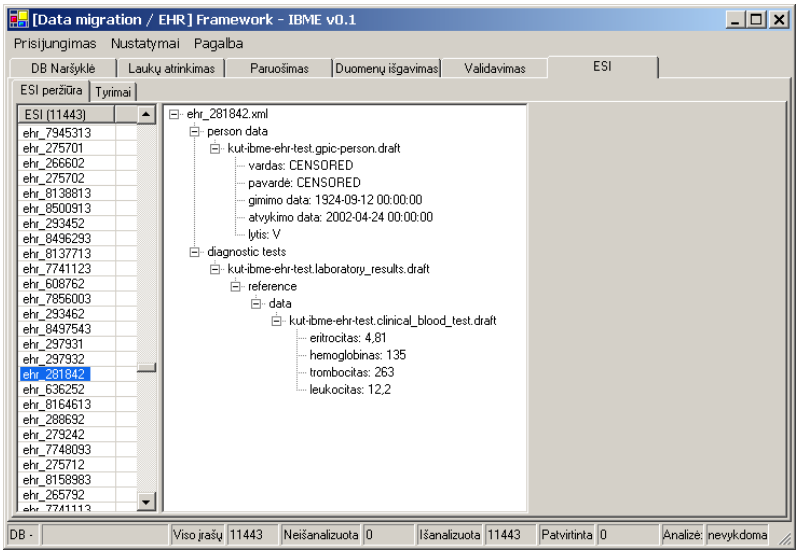
Validavimas yra vienintelis migracijos žingsnis, kuriame prie automatinio duomenų išgavimo turi prisidėti ir žmogus. Šiame žingsnyje išgauti duomenys yra pateikiami vartotojui, kad šis patikrintų, ar automatinis teksto analizės algoritmas viską atpažino teisingai. Eksperimente realizuota validavimo grafinė sąsają yra kortelėje „Validavimas“ (31 pav.). Šioje kortelėje pateikiamas visų išgautų įrašų sąrašas (kairėje), duomenų išgavimo protokolas (dešinėje), klaidų bei perspėjimų pranešimai, užfiksuoti automatinio teksto analizės algoritmo (apačioje) ir laukas iš kurio buvo išgaunami duomenys (viršuje). Spragtelint pele ant tam tikros klaidos dialogo apačioje, išgavimo protokolo medis yra išskleidžiamas ties ta vieta, kur buvo užfiksuota pasirinkta klaida. Taip pat yra paryškinama įrašo teksto dalis, su kuria susijusi ši klaida (teksto spalva pakeičiama į raudoną, o pats tekstas pabraukiamas).



31 pav. Eksperimentinė prog. įranga – validavimas. Suformuotų ESI sąrašo, klaidų ir perspėjimų, bei išgautų duomenų medžio vizualus pateikimas.

9.1.2.3. ESI peržiūra

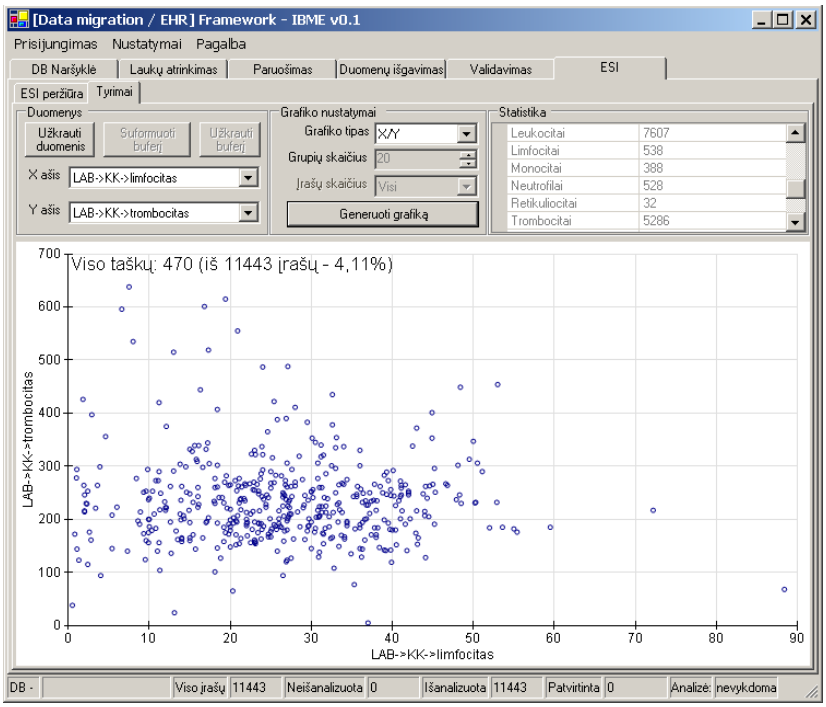
Sukurtiems naujosios ESI sistemos įrašams peržiūrėti yra skirta šeštos programinės įrangos kortelės „ESI“ kortelė „ESI peržiūra“ (32 pav.). Joje pateikiamas visų suformuotų ESI sąrašas, bei pasirinktos sveikatos istorijos duomenys (medžio pavidale). Apie šį medį (ir iš ko jis sudarytas) rašoma skyriuje **4.4.1 Naujoji informacinė sistema**.



32 pav. Eksperimentinė prog. įranga – suformuotų ESI peržiūra.

9.1.2.4. Statistika

Antroji kortelės „ESI“ kortelė „Tyrimai“ (33 pav.) yra skirta pademonstruoti išgautų ir struktūrizuotų duomenų panaudojimo galimybes. Pagal ESI sistemos žinių modelį, šioje kortelėje galima peržiūrėti duomenų vienas nuo kito priklausomybes, bei kiekvieno iš skirtingų matavimų pasiskirstymą (**4.4.1 Naujoji informacinė sistema**). Kortelėje galima pasirinkti užkrauti duomenis iš ESI bylų, o po to išsaugoti jas pagalbiniame buferyje, kadangi užkrovimas iš dviejų lygmenų sistemos yra lėtesnis. Kiekvienai ašiai galima priskirti norimą matavimą. Grafikai yra dviejų rūšių (pasirenkama ties „Grafiko tipas“) – matavimų priklausomybės ir pasiskirstymai. Spaudžiant „Generuoti grafiką“ suformuojamas pasirinktų duomenų ir pasirinkto tipo grafikas.



33 pav. Eksperimentinė prog. įranga: darbo su struktūrizuotais duomenimis galimybės – matavimų priklausomybių tyrimas.

9.2. Eksperimento duomenys

9.2.1. Lentelės „anketa“ egzemplioriaus pavyzdys

Darbe naudota Kauno kardiologijos klinikos ligonių duomenų bazė. Jos aprašymas pateiktas skyriuje **4.2.1 Liktinė sistema**. 14 lentelėje pateiktas vienas darbe naudotos lentelės „anketa“ egzepliorius.

14 lentelė. Eksperimente naudotos liktinės duomenų bazės lentelės „*anketa*“ įrašas

Lauko Nr.	Lauko pavadinimas	Lauko turinys
0	eilnr	16369
1	istnr	16369
2	istmet	0
3	ambulat	False
4	sdser	CENSORED
5	sdnum	CENSORED
6	asmkod	CENSORED
7	dokument	CENSORED
8	dokser	CENSORED
9	doknum	CENSORED
10	passer	CENSORED
11	pasnum	CENSORED
12	kasrup	CENSORED
13	lytis	True
14	atdata	2002.02.08 00:00:00
15	atval	0
16	atmin	0
17	isvdata	2002.02.21 00:00:00
18	viso_lovad	13
19	lov_kit	0
20	lygis	9
21	gimdat	1923.12.28 00:00:00
22	pavard	CENSORED
23	vardas	CENSORED
24	draudrus	CENSORED
25	draudist	CENSORED
26	rajkodas	CENSORED
27	socpad	CENSORED

14 lentelės tęsinys

Lauko Nr.	Lauko pavadinimas	Lauko turinys
28	darbov	CENSORED
29	kaimas	CENSORED
30	adresas	CENSORED
31	tel	CENSORED
32	inval	0
33	siukodas	0
34	perkel	False
35	siukit	
36	butpag	False
37	but_kod	
38	skub	False
39	skubs	0
40	hos_kas	0
41	hos_priez	0
42	dgn_skod	
43	dgn_siunt	
44	dgn_skod1	
45	dgn_pkod	I20.0.2.2
46	dgn_epik	MIC.Cardiosclerosis post infarctum myocardii . Stenosis S1-35% , S2-45%, S6-99%, S9-50%, S12-50% , S13 -75% (02 02 18) . Angina pectoris instabilis IIIB+ 2 (RV) (02 02 08) .
47	dgn_pagr	Nestabili krūtinės angina be persirgto MI. Vidutinė rizika. Troponino kiekis nep
48	dgn_komp	Insuff. cordis. Cl. f. II .
49	dgn_gret	Cataracta senilis .
50	dgn_gret1	E78.0
51	dgn_gret2	
52	dgn_gret3	
53	dgn_gret4	
54	dgn_gret5	
55	ligeig1	2
56	ligeig2	0
57	ligeig3	1

Lauko Nr.	Lauko pavadinimas	Lauko turinys
58	kitstac	0
59	kitstp	False
60	pirmkart	False
61	perdum	0
62	isr_siunt	
63	pat_ind	NKA
64	lig_anam	Gydyta I akių sk . 02 04 iki 02 08 , kur atlikta kataraktos operacija . Dėl NKA perkelta į I kard. sk.Atvykusi skundėsi spaudžiančio pobūdžio skausmais krūtinėje ramybėje bei minimalių krūvių metu , dusuliu didesnio fizinio krūvio metu .
65	lig_anao	Širdies veikla ritmiška. Tonai dusloki, ties aorta II° sistolinis ūžesys . AKS - 120/80 mmHg . Plaučiuose alsavimas vezikulinis, apat. dalyse nedaug stazinių karkalų. .
66	**diag_tyr**	Laboratoriniai : leu-5,7 x10**9/l , eritr- 3,52 x10**12/l, Hb-110 g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01 mmol/l, DTLCH -0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 , urea- 7,26 mmol/l, troponinas < 0,2 mcg/l. EKG : SR , ST dislokacija žemiau izolinijos iki 0,5 mm V4-V6 , II , III , aVF der. 2Decho : KSGDD-47 , sienelės po 11 , MM -187 , MI - 115.II° reguerg,. per MV , TV . Išvada - fibroziniai ir kalcinotinai MŽ pakitimai . Ao ž . pakitimai sklerotiniai . Saiki Ao stenozė . I° regurg. per Ao V . Susilpnėjusi KS sistolinė f-ja , sutrikusi diastolinė f-ja . IF-42% . KG - duomenys dng.
67	lig_eiga	Prof. GYDYTOJO_PV, doc. GYDYTOJO_PV vizitacija : šiuo metu yra tipiška NKA klinika , ligonė sirgusi MI , tikslinga atlikti KG . KG duomenys aptarti dalyvaujant prof. GYDYTOJO_PV , prof. GYDYTOJO_PV , doc. GYDYTOJO_PV , yra trijų koronarų liga,

Lauko Nr.	Lauko pavadinimas	Lauko turinys
		pakitimai ženklūs , priešangininis gydymas pilno efekto neduoda , tikslinga AKJO . Gydymo eigije ligonės savijauta nežymiai pagerėjo , skausmai kartojasi rečiau .
68	gydymasm	ISDN 60 mg x1 , fozinoprilis 5 mg x1 , metoprololis 100 mg x1 , simvastatinas 20 mg x1 , acidi acetylsalicylici 100 mg x1 .
69	gydymasi	
70	gydymasc	
71	lig_bukl	Stabili . Išrašoma į namus . Ligonė dėl operacinio gydymo nėra apsisprendusi .
72	rek_gyd	ISMN 50 mg x1 , fozinoprilis 10 mg x1 , metoprololis 100 mg x1 , simvastatinas (vasilip) 20 mg x1 Rp 0089735 .
73	rek_reabt	3
74	rek_reab	
75	rek_amb	Stebėti BPG .
76	rek_darb	0
77	prognoze	Gyvenimo atžvilgiu- patenkinama , pilno pasveikimo nesitikima , reikalinga AKJO .
78	lig_isr	True
79	lig_isrk1	I25.2
80	lig_isrk2	Z03.5.1.3
81	lig_isrk3	
82	lig_isrk4	
83	aritm	False
84	aritm_kod1	
85	aritm_kod2	
86	aritm_kod3	
87	laid	False
88	laid_kod1	
89	laid_kod2	
90	laid_kod3	
91	lsn	True
92	lsn_kod1	I50.2
93	lsn_kod2	

14 lentelės tęsinys

Lauko Nr.	Lauko pavadinimas	Lauko turinys
94	lsn_kod3	
95	usn	False
96	usn_kod1	
97	usn_kod2	
98	usn_kod3	
99	kita	False
100	kita_kod1	
101	kita_kod2	
102	kita_kod3	
103	sutapo	False
104	gydyt_hosp	CENSORED
105	hosp_rub	False
106	gydyt	CENSORED
107	gydyt_pav	CENSORED
108	skyr_ved	CENSORED
109	sekt_vad	CENSORED
110	nusisk	
111	a_morbi	
112	a_vitae	
113	a_laboris	
114	a_alerg	
115	st_prae	
116	st_spec	
117	diagnoze	
118	tyr_pla	
119	gyd_pla	

9.2.2. Sudarytas ir naudotas žodynas

Darbe kurta naujoji ESI sistema yra paremta dviejų lygių metodologija, kurioje griežtai atskiriami žinių ir informacijos lygmenys. Darbe sudarytas ir naudotas žodynas yra žinių lygmens dalis. Žodyną (15 lentelė) sudaro 39 sąvokos. Iš jų 33 sąvokos turi viso 60 variacijų (sinonimų, trumpinimų)

15 lentelė. Eksperimento metu sudarytas ir naudotas žodynas

Identifikacijos numeris	Pirminis terminas	Galimi trumpinimai
500	eritrocitas	RBC erit eritr eritrocitai
501	hemoglobinas	HGB, Hb
502	hematokritas	HCT PCV Ht
503	vidutinis eritrocito tūris	MCV
504	vidutinis hemoglobino kiekis eritrocite	MCH
505	vidutinė hemoglobino koncentracija eritrocite	MCHC
506	anizocitozė	RDW
507	trombocitas	PLT trombocitai tromb
508	vidutinis trombocitų tūris	MPV
509	leukocitas	WBC leuk leu
510	eozinofilas	eozinofilai eozinof eozin eoz
511	bazofilas	bazofilai, bazof, baz
512	neutrofilas	Neutrofilai, neutrof

15 lentelės tęsinys

Identifikacijos numeris	Pirminis terminas	Galimi trumpinimai
513	limfocitas	limfocitai limfoc limf
514	monocitas	monocitai monoc mon
515	retikuliocitas	RETIC retikuliocitai retikulioc
516	eritrocitų nusėdimo greitis	ENG ESR
517	C reaktyvinis baltymas	CRB
100	laboratoriniai tyrimai	lab labor laboratoriniai lab. tyrimai
101	EKG	EKG
102	2Decho	2Decho 2D echo
103	Ro	Rö Ro
104	KT	KT
105	VPAE	VPAE
106	KG	KG
107	VEM	VEM
108	EFT	EFT
109	GFDS	GFDS
110	GFS	GFS
111	elektroencefalograma	EEG
112	EFGDS	EFGDS
113	klinikinis kraujo tyrimas	kraujo tyr.
114	biocheminis kraujo tyrimas	biochem. tyr.

Identifikacijos numeris	Pirminis terminas	Galimi trumpinimai
800	asmens duomenys	
801	vardas	
802	pavardė	
803	gimimo data	
804	atvykimo data	
805	lytis	

Eksperimente sudarytas žodynas saugomas specialaus formato XML byloje. Šios bylos fragmentas, atitinkantis skyriaus **4.2.2 Liktinės sistemos analizė** 6 lentelėje išvardintus klinikinio kraujo tyrimo elementus:

```
1   <?xml version="1.0" encoding="utf-8" ?>
2   <dictionary  xmlns="http://tempuri.org/dictionary.xsd"  authority="KTU.IF"  title="local"
    type="standard" language="lt">
3   ...
4   <concept id="500" type="title">
5       <value>eritrocitas</value>
6       <description>eritrocitų apibrėžimas</description>
7       <synonym type="abbervation" aid="1" value="RBC" />
8       <synonym type="abbervation" aid="2" value="erit" />
9       <synonym type="abbervation" aid="3" value="eritr" />
10      <synonym type="variation" aid="4" value="eritrocitai" />
11  </concept>
12  <concept id="501" type="title">
13      <value>hemoglobinas</value>
14      <description>hemoglobino apibrėžimas</description>
15      <synonym type="abbervation" aid="1" value="HGB" />
16      <synonym type="abbervation" aid="2" value="Hb" />
17  </concept>
18  <concept id="502" type="title">
19      <value>hematokritas</value>
20      <description>hematokrito apibrėžimas</description>
21      <synonym type="abbervation" aid="1" value="HCT" />
22      <synonym type="abbervation" aid="2" value="PCV" />
23      <synonym type="abbervation" aid="3" value="Ht" />
24  </concept>
25  <concept id="503" type="title">
26      <value>vidutinis eritrocito tūris</value>
27      <description>vidutinis eritrocito tūris</description>
28      <synonym type="abbervation" aid="1" value="MCV" />
29  </concept>
30  ...
```

9.2.3. Sudaryti ir naudoti archetipai

9.2.3.1. Klinikinių kraujo tyrimų rezultatų normatyvai

Klinikinio kraujo tyrimo archetipas, panaudotas eksperimente, yra sudarytas pagal [35] klinikinio kraujo tyrimo normatyvus. Šie normatyvai pateikti 16 lentelėje.

16 lentelė. Eksperimente naudoto klinikinio kraujo tyrimo rezultatų normatyvai

SUAUGUSIŲJŲ			
ANALITĖ	NORMA		VIENETAI
RBC Eritrocitai	vyr	4,5-5,9	$*10^{12}$/l
	mot	4,5-5,2	$*10^{12}$/l
HGB Hemoglobinas	vyr	140-180	g/l
	mot	120-160	g/l
HCT (PCV) Hematokritas	vyr	41-53	%
	mot	36-46	%
Vidutinis eritrocito tūris (MCV)	82-98		fl
MCH Vidutinis hemoglobino kiekis eritrocite	27,6-33,3		Pg
MCHC Vidutinė hemoglobino koncentracija eritrocite	82-98		fl
RDW Makrocitų ir mikrocitų santykis (anizocitozė)	11,6-13,7		%
Trombocitai (PLT)	150-350		$*10^{9}$/l
Vidutinis trombocitų tūris (MPV)	7,4-10,4		fl
Leukocitai (WBC)	4.3-8.0		$*10^{9}$/l
Leukocitų formulės diferencinis skaičiavimas	eozinofilai	2-4	%
	bazofilai	0-1	%
	lazdeliniai neutrofilai	3-5	%
	segmentuoti neutrofilai	50-70	%
	limfocitai	25-40	%
	monocitai	2-8	%
RETIC Retikuliocitai	vyr	0,8-2,5	%
	mot	0,8-4,1	%
ENG (ESR) Eritrocitų nusėdimo greitis iki 50 m	vyr	1-15	mm/h
	mot	2-20	mm/h
ENG (ESR) Eritrocitų nusėdimo greitis virš 50 m	vyr	1-20	mm/h
	mot	2-30	mm/h
CRB C reaktyvinis baltymas	iki 5		mg/l

9.2.3.2. Archetipų struktūros

Eksperimente sudaryti trys archetipai: žmogaus asmeninai duomenys, laboratorinių tyrimų rezultatai, klinikinis kraujo tyrimas. Šių archetipų struktūros pateiktos 17 lentelėje. Archetipai buvo sudaromi ADL kalba ir išsaugomi XML formato bylose. Archetipų XML schemos fragmentas su paaiškinimais pateiktas po 17 lentele.

17 lentelė. Eksperimente sudarytų archetipų struktūros

Archetipo pavadinimas	**Koncepcijos pavadinimas**	**Koncepcijos tipas**	**Kiti apribojimai**	**Koncepcijos identifikacija**
Žmogaus asmeniniai duomenys				800
	vardas	tekstas		801
	pavardė			802
	gimimo data			803
	atvykimo data			804
	lytis			805
Laboratorinių tyrimų rezultatai	archetipas	nuoroda	nuoroda į klinikinio kraujo tyrimo archetipą	
Klinikinis kraujo tyrimas				103
	eritrocitai	išmatuojama reikšmė	matavimo vienetai „10**12/l“ reikšmių rėžiai: [4,5; 5,9]	500
	hemoglobinas		matavimo vienetai „g/l“ reikšmių rėžiai: [120; 180]	501
	hematokritas		matavimo vienetai „%“ reikšmių rėžiai: [36; 53]	502
	vid. eritrocito tūris		matavimo vienetai „fl“ reikšmių rėžiai: [82; 98]	503
	vid. hemoglobino kiekis eritrocite		matavimo vienetai „Pg“ reikšmių rėžiai: [27,6; 33,3]	504
	vid. hemoglobino koncentracija eritrocite		matavimo vienetai „fl“ reikšmių rėžiai: [82; 98]	505
	anizocitozė		matavimo vienetai „%“ reikšmių rėžiai: [11,6; 13,7]	506
	trombocitai		matavimo vienetai „10**9/l“ reikšmių rėžiai: [150; 350]	507

17 lentelės tęsinys

Archetipo pavadinimas	Koncepcijos pavadinimas	Koncepcijos tipas	Kiti apribojimai	Koncepcijos identifikacija
	vid. trombocitų tūris		matavimo vienetai „fl“ reikšmių rėžiai: [7,4; 10,4]	508
	leukocitai		matavimo vienetai „10**9/l“ reikšmių rėžiai:[4,3; 8]	509
	eozinofilai		matavimo vienetai „%“ reikšmių rėžiai: [2; 4]	510
	bazofilai		matavimo vienetai „%“. reikšmių rėžiai: [0; 1]	511
	neutrofilai		matavimo vienetai „%“ reikšmių rėžiai: [3; 5]	512
	limfocitai		matavimo vienetai „%“ reikšmių rėžiai: [25; 40]	513
	monocitai		matavimo vienetai „%“ reikšmių rėžiai:[2; 8]	514
	retikuliocitai		matavimo vienetai „%“ reikšmių rėžiai: [0,8; 4,1]	515
	eritrocitų nusėdimo greitis		matavimo vienetai „mm/h“ reikšmių rėžiai: [1; 20]	516
	C reaktyvinis baltymas		matavimo vienetai „mm/h“ reikšmių rėžiai: [0; 5]	517

Žmogaus asmeninių duomenų archetipas, aprašytas ADL kalbos struktūromis XML formate pateiktas žemiau. Šio archetipo paaiškinimas yra pateiktas 18 lentelėje. Paprastumo dėlei dalys **concept** ir **description** nėra rodomos.

```
1   <?xml version="1.0" encoding="utf-8" ?>
2   <achetype archetype_id="kut-ibme-ehr-test.gpic-person.draft">
3   <concept>
6   <description>
18  <definition>
19      <node type="complex object constraint" occurrences="1" id="at0000" rm_class="PERSON">
20          <attribute id="at0001" type="TEXT" />
21          <attribute id="at0002" type="TEXT" />
22          <attribute id="at0003" type="TEXT" />
23          <attribute id="at0004" type="TEXT" />
24          <attribute id="at0005" type="TEXT" />
25      </node>
26  </definition>
27  <ontology>
28      <primary_language>lt</primary_language>
29      <languages_available>lt, ...</languages_available>
30      <terminologies_available>lt, ...</terminologies_available>
31      <term_definitions>
32          <item id="lt">
33             <items>
34                  <item id="at0000">
35                      <description>asmens duomenys</description>
36                      <text>asmens duomenys</text>
37                  </item>
38              </items>
39          </item>
40      </term_definitions>
41      <constraint_definitions>
42          <item id="lt">
43              <items />
44          </item>
45      </constraint_definitions>
46      <term_binding>
47          <item id="local">
48              <items>
49                  <item id="at0000">[local::800]</item>
50                  <item id="at0001">[local::801]</item>
51                  <item id="at0002">[local::802]</item>
52                  <item id="at0003">[local::803]</item>
53                  <item id="at0004">[local::804]</item>
54                  <item id="at0005">[local::805]</item>
55              </items>
56          </item>
57      </term_binding>
58  </ontology>
59  </achetype>
```

18 lentelė. Pagrindinių asmens duomenų archetipo dalių paaiškinimas

Archetipo eilutės numeris	Aprašymas
2	Čia nurodomas unikalus archetipo identifikatorius: **kut-ibme-ehr-test.gpic-person.draft**. Archetipo identifikatorius – tai jo vardas, kuris sudaromas pagal tam tikras taisykles.
20–24	Šiose eilutėse nurodoma, kad archetipas turi penkis paprastus teksto tipo atributus, kurie lokaliai turi identifikatorius at0001, at0002, at0003, at0004 ir at0005.
49–54	Tai terminų atvaizdavimo iš lokalios erdvės į globalią, dalis. Čia lokalūs terminai (at0000, at0001 ir t.t.) yra surišami su globalia terminologija. Šiuo atveju globali terminologija yra minėtasis eksperimentui sudarytas žodynas.

8 lentelėje (skyrius **4.2.2**) aprašytų klinikinio kraujo tyrimo normatyvų fragmento išreiškimas archetipe XML kalba:

```
1   <attribute id="at0001" type="QUANTITY" units="10**12/l" min="4,5" max="5,9" />
2   <attribute id="at0002" type="QUANTITY" units="g/l" min="120" max="180" />
3   <attribute id="at0003" type="QUANTITY" units="%" min="36" max="53" />
4   <attribute id="at0004" type="QUANTITY" units="fl" min="82" max="98" />
5   .........
```

Šių atributų lokalių identifikatorių surišimas su globaliu žodynu:

```
1  <item id="at0001">[local::500]</item>
2  <item id="at0002">[local::501]</item>
3  <item id="at0003">[local::502]</item>
4  <item id="at0004">[local::503]</item>
5  .........
```

Lokaliam žodyne (skyrius 9.2.2, 16 lentelė) identifikatorius „500“ reiškia terminą „eritrocitai“, identifikatorius „501“ – „hemoglobinas“ ir t.t. Tokiu būdu archetipuose nėra išsaugomos konkrečios terminų reikšmės. Tai yra viena iš archetipo savybių: nepriklausymas nuo konkrečios terminologijos, ar kalbos.

9.2.4. Duomenys iš liktinių duomenų adapterio

9.2.4.1. Svarbių duomenų schema (XML)

Pirmuoju metodologijos žingsniu, analizuojant liktinės sistemos turinį, norimos išgauti lentelės duomenys yra aprašomi svarbių duomenų schemoje. Šis schema yra skirta duomenims iš liktinės duomenų bazės pervesti į tarpines XML formato bylas diske. Eksperimente sudaryta ir naudota svarbių duomenų schema:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<table name="anketa">
<field name="eilnr" type="System.Decimal" nr="0" isRelevant="true" isDirect="false" isKey="true" />
<field name="istnr" type="System.String" nr="1" isRelevant="true" isDirect="false" />
<field name="istmet" type="System.Decimal" nr="2" isRelevant="true" isDirect="false" isKey="true" />
<field name="ambulat" type="System.Boolean" nr="3" isRelevant="true" isDirect="false" />
<field name="sdser" type="System.String" nr="4" isRelevant="true" isDirect="false" />
<field name="sdnum" type="System.Decimal" nr="5" isRelevant="true" isDirect="false" />
<field name="asmkod" type="System.Decimal" nr="6" isRelevant="true" isDirect="false" />
<field name="dokument" type="System.String" nr="7" isRelevant="true" isDirect="false" />
<field name="dokser" type="System.String" nr="8" isRelevant="true" isDirect="false" />
<field name="doknum" type="System.Decimal" nr="9" isRelevant="true" isDirect="false" />
<field name="passer" type="System.String" nr="10" isRelevant="true" isDirect="false" />
<field name="pasnum" type="System.Decimal" nr="11" isRelevant="true" isDirect="false" />
<field name="kasrup" type="System.String" nr="12" isRelevant="true" isDirect="false" />
<field name="lytis" type="System.Boolean" nr="13" isRelevant="true" isDirect="false" />
<field name="atdata" type="System.DateTime" nr="14" isRelevant="true" isDirect="false" />
<field name="atval" type="System.Decimal" nr="15" isRelevant="true" isDirect="false" />
<field name="atmin" type="System.Decimal" nr="16" isRelevant="true" isDirect="false" />
<field name="isvdata" type="System.DateTime" nr="17" isRelevant="true" isDirect="false" />
<field name="viso_lovad" type="System.Decimal" nr="18" isRelevant="true" isDirect="false" />
<field name="lov_kit" type="System.Decimal" nr="19" isRelevant="true" isDirect="false" />
<field name="lygis" type="System.Decimal" nr="20" isRelevant="true" isDirect="false" />
<field name="gimdat" type="System.DateTime" nr="21" isRelevant="true" isDirect="false" />
<field name="pavard" type="System.String" nr="22" isRelevant="true" isDirect="false" />
<field name="vardas" type="System.String" nr="23" isRelevant="true" isDirect="false" />
<field name="draudrus" type="System.Decimal" nr="24" isRelevant="true" isDirect="false" />
<field name="draudist" type="System.Decimal" nr="25" isRelevant="true" isDirect="false" />
<field name="rajkodas" type="System.Decimal" nr="26" isRelevant="true" isDirect="false" />
<field name="socpad" type="System.Decimal" nr="27" isRelevant="true" isDirect="false" />
<field name="darbov" type="System.String" nr="28" isRelevant="true" isDirect="false" />
<field name="kaimas" type="System.Boolean" nr="29" isRelevant="true" isDirect="false" />
<field name="adresas" type="System.String" nr="30" isRelevant="true" isDirect="false" />
<field name="tel" type="System.Decimal" nr="31" isRelevant="true" isDirect="false" />
<field name="inval" type="System.Decimal" nr="32" isRelevant="true" isDirect="false" />
<field name="siukodas" type="System.Decimal" nr="33" isRelevant="true" isDirect="false" />
<field name="perkel" type="System.Boolean" nr="34" isRelevant="true" isDirect="false" />
<field name="siukit" type="System.String" nr="35" isRelevant="true" isDirect="false" />
<field name="butpag" type="System.Boolean" nr="36" isRelevant="true" isDirect="false" />
<field name="but_kod" type="System.String" nr="37" isRelevant="true" isDirect="false" />
```

```
41  <field name="skub" type="System.Boolean" nr="38" isRelevant="true" isDirect="false" />
42  <field name="skubs" type="System.Decimal" nr="39" isRelevant="true" isDirect="false" />
43  <field name="hos_kas" type="System.Decimal" nr="40" isRelevant="true" isDirect="false" />
44  <field name="hos_priez" type="System.Decimal" nr="41" isRelevant="true" isDirect="false"
    />
45  <field name="dgn_skod" type="System.String" nr="42" isRelevant="true" isDirect="false" />
46  <field name="dgn_siunt" type="System.String" nr="43" isRelevant="true" isDirect="false" />
47  <field name="dgn_skod1" type="System.String" nr="44" isRelevant="true" isDirect="false" />
48  <field name="dgn_pkod" type="System.String" nr="45" isRelevant="true" isDirect="false" />
49  <field name="dgn_epik" type="System.String" nr="46" isRelevant="true" isDirect="false" />
50  <field name="dgn_pagr" type="System.String" nr="47" isRelevant="true" isDirect="false" />
51  <field name="dgn_komp" type="System.String" nr="48" isRelevant="true" isDirect="false" />
52  <field name="dgn_gret" type="System.String" nr="49" isRelevant="true" isDirect="false" />
53  <field name="dgn_gret1" type="System.String" nr="50" isRelevant="true" isDirect="false" />
54  <field name="dgn_gret2" type="System.String" nr="51" isRelevant="true" isDirect="false" />
55  <field name="dgn_gret3" type="System.String" nr="52" isRelevant="true" isDirect="false" />
56  <field name="dgn_gret4" type="System.String" nr="53" isRelevant="true" isDirect="false" />
57  <field name="dgn_gret5" type="System.String" nr="54" isRelevant="true" isDirect="false" />
58  <field name="ligeig1" type="System.Decimal" nr="55" isRelevant="true" isDirect="false" />
59  <field name="ligeig2" type="System.Decimal" nr="56" isRelevant="true" isDirect="false" />
60  <field name="ligeig3" type="System.Decimal" nr="57" isRelevant="true" isDirect="false" />
61  <field name="kitstac" type="System.Decimal" nr="58" isRelevant="true" isDirect="false" />
62  <field name="kitstp" type="System.Boolean" nr="59" isRelevant="true" isDirect="false" />
63  <field name="pirmkart" type="System.Boolean" nr="60" isRelevant="true" isDirect="false" />
64  <field name="perdum" type="System.Decimal" nr="61" isRelevant="true" isDirect="false" />
65  <field name="isr_siunt" type="System.String" nr="62" isRelevant="true" isDirect="false" />
66  <field name="pat_ind" type="System.String" nr="63" isRelevant="true" isDirect="false" />
67  <field name="lig_anam" type="System.String" nr="64" isRelevant="true" isDirect="false" />
68  <field name="lig_anao" type="System.String" nr="65" isRelevant="true" isDirect="false" />
69  <field name="diag_tyr" type="System.String" nr="66" isRelevant="true" isDirect="false" />
70  <field name="lig_eiga" type="System.String" nr="67" isRelevant="true" isDirect="false" />
71  <field name="gydymasm" type="System.String" nr="68" isRelevant="true" isDirect="false" />
72  <field name="gydymasi" type="System.String" nr="69" isRelevant="true" isDirect="false" />
73  <field name="gydymasc" type="System.String" nr="70" isRelevant="true" isDirect="false" />
74  <field name="lig_bukl" type="System.String" nr="71" isRelevant="true" isDirect="false" />
75  <field name="rek_gyd" type="System.String" nr="72" isRelevant="true" isDirect="false" />
76  <field name="rek_reabt" type="System.Decimal" nr="73" isRelevant="true" isDirect="false"
    />
77  <field name="rek_reab" type="System.String" nr="74" isRelevant="true" isDirect="false" />
78  <field name="rek_amb" type="System.String" nr="75" isRelevant="true" isDirect="false" />
79  <field name="rek_darb" type="System.Decimal" nr="76" isRelevant="true" isDirect="false" />
80  <field name="prognoze" type="System.String" nr="77" isRelevant="true" isDirect="false" />
81  <field name="lig_isr" type="System.Boolean" nr="78" isRelevant="true" isDirect="false" />
82  <field name="lig_isrk1" type="System.String" nr="79" isRelevant="true" isDirect="false" />
83  <field name="lig_isrk2" type="System.String" nr="80" isRelevant="true" isDirect="false" />
84  <field name="lig_isrk3" type="System.String" nr="81" isRelevant="true" isDirect="false" />
85  <field name="lig_isrk4" type="System.String" nr="82" isRelevant="true" isDirect="false" />
86  <field name="aritm" type="System.Boolean" nr="83" isRelevant="true" isDirect="false" />
87  <field name="aritm_kod1" type="System.String" nr="84" isRelevant="true" isDirect="false"
    />
88  <field name="aritm_kod2" type="System.String" nr="85" isRelevant="true" isDirect="false"
    />
```

```
89  <field name="aritm_kod3" type="System.String" nr="86" isRelevant="true" isDirect="false"
    />
90  <field name="laid" type="System.Boolean" nr="87" isRelevant="true" isDirect="false" />
91  <field name="laid_kod1" type="System.String" nr="88" isRelevant="true" isDirect="false" />
92  <field name="laid_kod2" type="System.String" nr="89" isRelevant="true" isDirect="false" />
93  <field name="laid_kod3" type="System.String" nr="90" isRelevant="true" isDirect="false" />
94  <field name="lsn" type="System.Boolean" nr="91" isRelevant="true" isDirect="false" />
95  <field name="lsn_kod1" type="System.String" nr="92" isRelevant="true" isDirect="false" />
96  <field name="lsn_kod2" type="System.String" nr="93" isRelevant="true" isDirect="false" />
97  <field name="lsn_kod3" type="System.String" nr="94" isRelevant="true" isDirect="false" />
98  <field name="usn" type="System.Boolean" nr="95" isRelevant="true" isDirect="false" />
99  <field name="usn_kod1" type="System.String" nr="96" isRelevant="true" isDirect="false" />
100 <field name="usn_kod2" type="System.String" nr="97" isRelevant="true" isDirect="false" />
101 <field name="usn_kod3" type="System.String" nr="98" isRelevant="true" isDirect="false" />
102 <field name="kita" type="System.Boolean" nr="99" isRelevant="true" isDirect="false" />
103 <field name="kita_kod1" type="System.String" nr="100" isRelevant="true" isDirect="false"
    />
104 <field name="kita_kod2" type="System.String" nr="101" isRelevant="true" isDirect="false"
    />
105 <field name="kita_kod3" type="System.String" nr="102" isRelevant="true" isDirect="false"
    />
106 <field name="sutapo" type="System.Boolean" nr="103" isRelevant="true" isDirect="false" />
107 <field name="gydyt_hosp" type="System.Decimal" nr="104" isRelevant="true" isDirect="false"
    />
108 <field name="hosp_rub" type="System.Boolean" nr="105" isRelevant="true" isDirect="false"
    />
109 <field name="gydyt" type="System.Decimal" nr="106" isRelevant="true" isDirect="false" />
110 <field name="gydyt_pav" type="System.String" nr="107" isRelevant="true" isDirect="false"
    />
111 <field name="skyr_ved" type="System.Decimal" nr="108" isRelevant="true" isDirect="false"
    />
112 <field name="sekt_vad" type="System.Decimal" nr="109" isRelevant="true" isDirect="false"
    />
113 <field name="nusisk" type="System.String" nr="110" isRelevant="true" isDirect="false" />
114 <field name="a_morbi" type="System.String" nr="111" isRelevant="true" isDirect="false" />
115 <field name="a_vitae" type="System.String" nr="112" isRelevant="true" isDirect="false" />
116 <field name="a_laboris" type="System.String" nr="113" isRelevant="true" isDirect="false"
    />
117 <field name="a_alerg" type="System.String" nr="114" isRelevant="true" isDirect="false" />
118 <field name="st_prae" type="System.String" nr="115" isRelevant="true" isDirect="false" />
119 <field name="st_spec" type="System.String" nr="116" isRelevant="true" isDirect="false" />
120 <field name="diagnoze" type="System.String" nr="117" isRelevant="true" isDirect="false" />
121 <field name="tyr_pla" type="System.String" nr="118" isRelevant="true" isDirect="false" />
122 <field name="gyd_pla" type="System.String" nr="119" isRelevant="true" isDirect="false" />
123 </table>
```

9.2.4.2. Lentelės „anketa“ egzempliorius (XML)

Išanalizavus liktinę sistemą ir formaliai aprašius norimus išgauti duomenis gaunama svarbių duomenų schema. Pagal ją, visi liktinės duomenų bazės pasirinktos lentelės egzemplioriai yra pervedami į XML tipo bylas, kurios yra duomenų šaltinis likusiems metodikos žingsniams. Eksperimente analizuotos lentelės „anketa“ egzempliorius XML tipo byloje:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<table name="anketa" nr="1747" key="anketa_r_163690">
  <field name="eilnr" value="16369" />
  <field name="istnr" value="16369" />
  <field name="istmet" value="0" />
  <field name="ambulat" value="False" />
  <field name="sdser" value="CENSORED" />
  <field name="sdnum" value="CENSORED" />
  <field name="asmkod" value="CENSORED" />
  <field name="dokument" value="CENSORED" />
  <field name="dokser" value="CENSORED" />
  <field name="doknum" value="CENSORED" />
  <field name="passer" value="CENSORED" />
  <field name="pasnum" value="CENSORED" />
  <field name="kasrup" value="CENSORED" />
  <field name="lytis" value="True" />
  <field name="atdata" value="2002.02.08 00:00:00" />
  <field name="atval" value="0" />
  <field name="atmin" value="0" />
  <field name="isvdata" value="2002.02.21 00:00:00" />
  <field name="viso_lovad" value="13" />
  <field name="lov_kit" value="0" />
  <field name="lygis" value="9" />
  <field name="gimdat" value="1923.12.28 00:00:00" />
  <field name="pavard" value="CENSORED" />
  <field name="vardas" value="CENSORED" />
  <field name="draudrus" value="CENSORED" />
  <field name="draudist" value="CENSORED" />
  <field name="rajkodas" value="CENSORED" />
  <field name="socpad" value="CENSORED" />
  <field name="darbov" value="CENSORED" />
  <field name="kaimas" value="CENSORED" />
  <field name="adresas" value="CENSORED" />
  <field name="tel" value="CENSORED" />
  <field name="inval" value="0" />
  <field name="siukodas" value="0" />
  <field name="perkel" value="False" />
  <field name="siukit" value="" />
  <field name="butpag" value="False" />
  <field name="but_kod" value="" />
  <field name="skub" value="False" />
  <field name="skubs" value="0" />
  <field name="hos_kas" value="0" />
  <field name="hos_priez" value="0" />
```

```
45    <field name="dgn_skod" value="" />
46    <field name="dgn_siunt" value="" />
47    <field name="dgn_skod1" value="" />
48    <field name="dgn_pkod" value="I20.0.2.2" />
49    <field name="dgn_epik" value="MIC.Cardiosclerosis post infarctum myocardii . Stenosis S1-35% , S2-45%, S6-99%, S9-50%, S12-50% , S13 -75% ( 02 02 18) . Angina pectoris instabilis IIIB+ 2 ( RV) ( 02 02 08) ." />
50    <field name="dgn_pagr" value="Nestabili krūtinės angina be persirgto MI. Vidutinė rizika. Troponino kiekis nep" />
51    <field name="dgn_komp" value="Insuff. cordis. Cl. f. II ." />
52    <field name="dgn_gret" value="Cataracta senilis ." />
53    <field name="dgn_gret1" value="E78.0" />
54    <field name="dgn_gret2" value="" />
55    <field name="dgn_gret3" value="" />
56    <field name="dgn_gret4" value="" />
57    <field name="dgn_gret5" value="" />
58    <field name="ligeig1" value="2" />
59    <field name="ligeig2" value="0" />
60    <field name="ligeig3" value="1" />
61    <field name="kitstac" value="0" />
62    <field name="kitstp" value="False" />
63    <field name="pirmkart" value="False" />
64    <field name="perdum" value="0" />
65    <field name="isr_siunt" value="" />
66    <field name="pat_ind" value="NKA" />
67    <field name="lig_anam" value="Gydyta I akių sk . 02 04 iki 02 08 , kur atlikta kataraktos operacija . Dėl NKA perkelta į I kard. sk.Atvykusi skundėsi spaudžiančio pobūdžio skausmais krūtinėje ramybėje bei minimalių krūvių metu , dusuliu didesnio fizinio krūvio metu ." />
68    <field name="lig_anao" value="Širdies veikla ritmiška. Tonai dusloki, ties aorta II° sistolinis ūžesys . AKS - 120/80 mmHg . Plaučiuose alsavimas vezikulinis, apat. dalyse nedaug stazinių karkalų. ." />
69    <field name="diag_tyr" value="Laboratoriniai : leu-5,7 x10**9/l , eritr- 3,52 x10**12/l, Hb-110 g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01 mmol/l, DTLCH -0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 , urea-7,26 mmol/l, troponinas < 0,2 mcg/l. EKG : SR , ST dislokacija žemiau izolinijos iki 0,5 mm V4-V6 , II , III , aVF der. 2Decho : KSGDD- 47 , sienelės po 11 , MM -187 , MI -115.II° reguerg,. per MV , TV . Išvada - fibroziniai ir kalcinotinai MŽ pakitimai . Ao ž . pakitimai sklerotiniai . Saiki Ao stenozė . I° regurg. per Ao V . Susilpnėjusi KS sistolinė f-ja , sutrikusi diastolinė f-ja . IF-42% . KG - duomenys dng." />
70    <field name="lig_eiga" value="Prof. GYDYTOJO_PV, doc. GYDYTOJO_PV vizitacija : šiuo metu yra tipiška NKA klinika , ligonė sirgusi MI , tikslinga atlikti KG . KG duomenys aptarti dalyvaujant prof. GYDYTOJO_PV , prof. GYDYTOJO_PV , doc. GYDYTOJO_PV , yra trijų koronarų liga, pakitimai ženklūs , priešangininis gydymas pilno efekto neduoda , tikslinga AKJO . Gydymo eigije ligonės savijauta nežymiai pagerėjo , skausmai kartojasi rečiau ." />
71    <field name="gydymasm" value="ISDN 60 mg x1 , fozinoprilis 5 mg x1 , metoprololis 100 mg x1 , simvastatinas 20 mg x1 , acidi acetylsalicylici 100 mg x1 ." />
72    <field name="gydymasi" value="" />
73    <field name="gydymasc" value="" />
74    <field name="lig_bukl" value="Stabili . Išrašoma į namus . Ligonė dėl operacinio gydymo nėra apsisprendusi ." />
75    <field name="rek_gyd" value="ISMN 50 mg x1 , fozinoprilis 10 mg x1 , metoprololis 100 mg x1 , simvastatinas ( vasilip) 20 mg x1 Rp 0089735 ." />
```

```
76    <field name="rek_reabt" value="3" />
77    <field name="rek_reab" value="" />
78    <field name="rek_amb" value="Stebėti BPG ." />
79    <field name="rek_darb" value="0" />
80    <field  name="prognoze"  value="Gyvenimo  atžvilgiu-  patenkinama  ,  pilno  pasveikimo
    nesitikima , reikalinga AKJO ." />
81    <field name="lig_isr" value="True" />
82    <field name="lig_isrk1" value="I25.2" />
83    <field name="lig_isrk2" value="Z03.5.1.3" />
84    <field name="lig_isrk3" value="" />
85    <field name="lig_isrk4" value="" />
86    <field name="aritm" value="False" />
87    <field name="aritm_kod1" value="" />
88    <field name="aritm_kod2" value="" />
89    <field name="aritm_kod3" value="" />
90    <field name="laid" value="False" />
91    <field name="laid_kod1" value="" />
92    <field name="laid_kod2" value="" />
93    <field name="laid_kod3" value="" />
94    <field name="lsn" value="True" />
95    <field name="lsn_kod1" value="I50.2" />
96    <field name="lsn_kod2" value="" />
97    <field name="lsn_kod3" value="" />
98    <field name="usn" value="False" />
99    <field name="usn_kod1" value="" />
100   <field name="usn_kod2" value="" />
101   <field name="usn_kod3" value="" />
102   <field name="kita" value="False" />
103   <field name="kita_kod1" value="" />
104   <field name="kita_kod2" value="" />
105   <field name="kita_kod3" value="" />
106   <field name="sutapo" value="False" />
107   <field name="gydyt_hosp" value="CENSORED" />
108   <field name="hosp_rub" value="False" />
109   <field name="gydyt" value="CENSORED" />
110   <field name="gydyt_pav" value="CENSORED" />
111   <field name="skyr_ved" value="CENSORED" />
112   <field name="sekt_vad" value="CENSORED" />
113   <field name="nusisk" value="" />
114   <field name="a_morbi" value="" />
115   <field name="a_vitae" value="" />
116   <field name="a_laboris" value="" />
117   <field name="a_alerg" value="" />
118   <field name="st_prae" value="" />
119   <field name="st_spec" value="" />
120   <field name="diagnoze" value="" />
121   <field name="tyr_pla" value="" />
122   <field name="gyd_pla" value="" />
123 </table>
```

9.2.5. Išgautų duomenų medis (XML)

Paskutiniame automatizuotos medicininių dokumentų analizės žingsnyje automatinis analizės algoritmas, išgaudamas duomenis iš nestruktūrizuotų ar mažai struktūrizuotų laukų, sudaro transformacijos (išgavimo) protokolą – tam tikrų išgavimo objektų medį (**3.3.4.2 Išgaunamos informacijos duomenų struktūra**). Eksperimento metu šis medis yra išsaugomas XML formato byloje. Tokios bylos pavyzdys:

```
1   <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
2   <rawextract rawId="16369">
3   <extract localId="1" originalId="0" startIndex="0" length="664" etype="Default"
    stype="Other" noInOriginal="0" value="Laboratoriniai : leu-5,7 x10**9/l , eritr- 3,52
    x10**12/l, Hb-110 g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01
    mmol/l, DTLCH -0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 ,
    urea- 7,26 mmol/l, troponinas < 0,2 mcg/l. EKG : SR , ST dislokacija žemiau izolinijos iki
    0,5 mm V4-V6 , II , III , aVF der. 2Decho : KSGDD- 47 , sienelės po 11 , MM -187 , MI -
    115.II° reguerg,. per MV , TV . Išvada - fibroziniai ir kalcinotinai MŽ pakitimai . Ao ž .
    pakitimai sklerotiniai . Saiki Ao stenozė . I° regurg. per Ao V . Susilpnėjusi KS
    sistolinė f-ja , sutrikusi diastolinė f-ja . IF-42% . KG - duomenys dng." />
4   <extract localId="2" originalId="1" startIndex="0" length="273" etype="Default"
    stype="Other" noInOriginal="0" value="Laboratoriniai : leu-5,7 x10**9/l , eritr- 3,52
    x10**12/l, Hb-110 g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01
    mmol/l, DTLCH -0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 ,
    urea- 7,26 mmol/l, troponinas < 0,2 mcg/l." />
5   <extract localId="3" originalId="1" startIndex="273" length="2" etype="Separator"
    stype="EOL" noInOriginal="0" value="" />
6   <ident sequence="0" exId="2" idId="3" rootKid="1" subKid="-1" type="Complex" />
7   <extract localId="4" originalId="1" startIndex="275" length="84" etype="Default"
    stype="Other" noInOriginal="1" value="EKG : SR , ST dislokacija žemiau izolinijos iki 0,5
    mm V4-V6 , II , III , aVF der." />
8   <extract localId="5" originalId="1" startIndex="359" length="2" etype="Separator"
    stype="EOL" noInOriginal="1" value="" />
9   <ident sequence="0" exId="4" idId="5" rootKid="1" subKid="-1" type="Complex" />
10  <extract localId="6" originalId="1" startIndex="361" length="282" etype="Default"
    stype="Other" noInOriginal="2" value="2Decho : KSGDD- 47 , sienelės po 11 , MM -187 , MI -
    115.II° reguerg,. per MV , TV . Išvada - fibroziniai ir kalcinotinai MŽ pakitimai . Ao ž .
    pakitimai sklerotiniai . Saiki Ao stenozė . I° regurg. per Ao V . Susilpnėjusi KS
    sistolinė f-ja , sutrikusi diastolinė f-ja . IF-42% ." />
11  <extract localId="7" originalId="1" startIndex="643" length="2" etype="Separator"
    stype="EOL" noInOriginal="2" value="" />
12  <ident sequence="0" exId="6" idId="7" rootKid="1" subKid="-1" type="Complex" />
13  <extract localId="8" originalId="1" startIndex="645" length="19" etype="Default"
    stype="Other" noInOriginal="3" value="KG - duomenys dng." />
14  <extract localId="9" originalId="1" startIndex="664" length="2" etype="Separator"
    stype="EOL" noInOriginal="3" value="" />
15  <ident sequence="0" exId="8" idId="9" rootKid="1" subKid="-1" type="Complex" />
16  <extract localId="10" originalId="2" startIndex="16" length="257" etype="Default"
    stype="Other" noInOriginal="1" value="leu-5,7 x10**9/l , eritr- 3,52 x10**12/l, Hb-110
    g/l, tromb- 201x10**9/l, kr. gr. 0, Rh(+) . BCH -5,56mmol/l, MTLCH-4,01 mmol/l, DTLCH -
    0,93 mmol/l, TG-1,35 mmol/l, AK-4,9 , ASAT-20 , ALAT- 214, CPK-363. K-3,6 , urea- 7,26
    mmol/l, troponinas < 0,2 mcg/l." />
```

```
<extract localId="11" originalId="2" startIndex="273" length="16" etype="Separator" stype="Other" noInOriginal="1" value="Laboratoriniai :" />
<ident sequence="1" exId="10" idId="11" rootKid="100" subKid="-1" type="Complex" />
<extract localId="12" originalId="10" startIndex="0" length="18" etype="Default" stype="Other" noInOriginal="0" value="leu-5,7 x10**9/l" />
<extract localId="13" originalId="10" startIndex="18" length="2" etype="Separator" stype="Other" noInOriginal="0" value="," />
<ident sequence="2" exId="12" idId="13" rootKid="0" subKid="-1" type="Complex" />
<extract localId="14" originalId="10" startIndex="20" length="21" etype="Default" stype="Other" noInOriginal="1" value="eritr- 3,52 x10**12/l" />
<extract localId="15" originalId="10" startIndex="41" length="2" etype="Separator" stype="Other" noInOriginal="1" value="," />
<ident sequence="2" exId="14" idId="15" rootKid="0" subKid="-1" type="Complex" />
<extract localId="16" originalId="10" startIndex="43" length="10" etype="Default" stype="Other" noInOriginal="2" value="Hb-110 g/l" />
<extract localId="17" originalId="10" startIndex="53" length="2" etype="Separator" stype="Other" noInOriginal="2" value="," />
<ident sequence="2" exId="16" idId="17" rootKid="0" subKid="-1" type="Complex" />
<extract localId="18" originalId="10" startIndex="55" length="18" etype="Default" stype="Other" noInOriginal="3" value="tromb- 201x10**9/l" />
<extract localId="19" originalId="10" startIndex="73" length="2" etype="Separator" stype="Other" noInOriginal="3" value="," />
<ident sequence="2" exId="18" idId="19" rootKid="0" subKid="-1" type="Complex" />
<extract localId="20" originalId="10" startIndex="75" length="9" etype="Default" stype="Other" noInOriginal="4" value="kr. gr. 0" />
<extract localId="21" originalId="10" startIndex="84" length="2" etype="Separator" stype="Other" noInOriginal="4" value="," />
<ident sequence="2" exId="20" idId="21" rootKid="0" subKid="-1" type="Complex" />
<extract localId="22" originalId="10" startIndex="86" length="23" etype="Default" stype="Other" noInOriginal="5" value="Rh(+) . BCH -5,56mmol/l" />
<extract localId="23" originalId="10" startIndex="109" length="2" etype="Separator" stype="Other" noInOriginal="5" value="," />
<ident sequence="2" exId="22" idId="23" rootKid="0" subKid="-1" type="Complex" />
<extract localId="24" originalId="10" startIndex="111" length="17" etype="Default" stype="Other" noInOriginal="6" value="MTLCH-4,01 mmol/l" />
<extract localId="25" originalId="10" startIndex="128" length="2" etype="Separator" stype="Other" noInOriginal="6" value="," />
<ident sequence="2" exId="24" idId="25" rootKid="0" subKid="-1" type="Complex" />
<extract localId="26" originalId="10" startIndex="130" length="18" etype="Default" stype="Other" noInOriginal="7" value="DTLCH -0,93 mmol/l" />
<extract localId="27" originalId="10" startIndex="148" length="2" etype="Separator" stype="Other" noInOriginal="7" value="," />
<ident sequence="2" exId="26" idId="27" rootKid="0" subKid="-1" type="Complex" />
<extract localId="28" originalId="10" startIndex="150" length="14" etype="Default" stype="Other" noInOriginal="8" value="TG-1,35 mmol/l" />
<extract localId="29" originalId="10" startIndex="164" length="2" etype="Separator" stype="Other" noInOriginal="8" value="," />
<ident sequence="2" exId="28" idId="29" rootKid="0" subKid="-1" type="Complex" />
<extract localId="30" originalId="10" startIndex="166" length="8" etype="Default" stype="Other" noInOriginal="9" value="AK-4,9" />
<extract localId="31" originalId="10" startIndex="174" length="2" etype="Separator" stype="Other" noInOriginal="9" value="," />
<ident sequence="2" exId="30" idId="31" rootKid="0" subKid="-1" type="Complex" />
```

```
<extract localId="32" originalId="10" startIndex="176" length="8" etype="Default" stype="Other" noInOriginal="10" value="ASAT-20" />
<extract localId="33" originalId="10" startIndex="184" length="2" etype="Separator" stype="Other" noInOriginal="10" value="," />
<ident sequence="2" exId="32" idId="33" rootKid="0" subKid="-1" type="Complex" />
<extract localId="34" originalId="10" startIndex="186" length="9" etype="Default" stype="Other" noInOriginal="11" value="ALAT- 214" />
<extract localId="35" originalId="10" startIndex="195" length="2" etype="Separator" stype="Other" noInOriginal="11" value="," />
<ident sequence="2" exId="34" idId="35" rootKid="0" subKid="-1" type="Complex" />
<extract localId="36" originalId="10" startIndex="197" length="15" etype="Default" stype="Other" noInOriginal="12" value="CPK-363. K-3,6" />
<extract localId="37" originalId="10" startIndex="212" length="2" etype="Separator" stype="Other" noInOriginal="12" value="," />
<ident sequence="2" exId="36" idId="37" rootKid="0" subKid="-1" type="Complex" />
<extract localId="38" originalId="10" startIndex="214" length="17" etype="Default" stype="Other" noInOriginal="13" value="urea- 7,26 mmol/l" />
<extract localId="39" originalId="10" startIndex="231" length="2" etype="Separator" stype="Other" noInOriginal="13" value="," />
<ident sequence="2" exId="38" idId="39" rootKid="0" subKid="-1" type="Complex" />
<extract localId="40" originalId="10" startIndex="233" length="24" etype="Default" stype="Other" noInOriginal="14" value="troponinas < 0,2 mcg/l." />
<extract localId="41" originalId="10" startIndex="257" length="2" etype="Separator" stype="Other" noInOriginal="14" value="," />
<ident sequence="2" exId="40" idId="41" rootKid="0" subKid="-1" type="Complex" />
<extract localId="42" originalId="14" startIndex="7" length="14" etype="Default" stype="Other" noInOriginal="0" value="3,52 x10**12/l" />
<extract localId="43" originalId="14" startIndex="0" length="7" etype="Default" stype="Other" noInOriginal="0" value="eritr-" />
<ident sequence="3" exId="42" idId="43" rootKid="500" subKid="-1" type="Title" />
<extract localId="44" originalId="16" startIndex="3" length="7" etype="Default" stype="Other" noInOriginal="0" value="110 g/l" />
<extract localId="45" originalId="16" startIndex="0" length="3" etype="Default" stype="Other" noInOriginal="0" value="Hb-" />
<ident sequence="4" exId="44" idId="45" rootKid="501" subKid="-1" type="Title" />
<extract localId="46" originalId="18" startIndex="7" length="11" etype="Default" stype="Other" noInOriginal="0" value="201x10**9/l" />
<extract localId="47" originalId="18" startIndex="0" length="7" etype="Default" stype="Other" noInOriginal="0" value="tromb-" />
<ident sequence="5" exId="46" idId="47" rootKid="507" subKid="-1" type="Title" />
<extract localId="48" originalId="12" startIndex="4" length="12" etype="Default" stype="Other" noInOriginal="0" value="5,7 x10**9/l" />
<extract localId="49" originalId="12" startIndex="0" length="4" etype="Default" stype="Other" noInOriginal="0" value="leu-" />
<ident sequence="6" exId="48" idId="49" rootKid="509" subKid="-1" type="Title" />
</rawextract>
```

9.3. Straipsnis

Darbo tema spausdinta publikacija:

Kazla, A. *Modern world of electronic health records. Migration from legacy systems*// Biomedical engineering – 2004: tarptautinės konferencijos pranešimų medžiaga [Kaunas, 2004 m. spalio 28, 29 d.]. Kaunas, Technologija, 2004, p. 23-27.

Modern world of Electronic Health Records. Migration from legacy systems

A. Kazla

Institute of Biomedical Engineering, Kaunas University of Technology, Lithuania

Introduction

Computerization in the field of medicine goes on for more than thirty years. It is now widely accepted, that a modern Electronic Health Record System (EHRS) is a must in a modern hospital. EHRS is a computerized replacement of paper-based medical records. Modern EHRS should be not only extremely flexible - supporting thousands of different medical entities and procedures, providing input, validation and browsing tools, but also be standardized in a way to facilitate sharing of information within hospital and beyond. Medical systems that exist today and don't support such flexibility should be transformed into new ones, preserving all medical data accumulated during long years of work.

EHR standards

The main concept of the EHR is that it should be a patient-centered, long-term view of all health-related information (observations, opiniors, etc) [1].

Overview

Currently developed EHR standards define most important parts of EHR systems: information exchange models, security measures, electronic document architectures, etc. They also define principles on how different medical systems should be linked, in order to form a single distributed electronic health record.

There are three groups in the world that make the biggest impact on the future of EHR standard: International Standards Organization (ISO TC215), European Committee for Standardization (CEN TC251), and Health Level 7. EHR standards are developed by these groups for some time now, but are not final yet. In past several years collaboration between these groups has lead to harmonization works of these three standards, but still the question what standard to adapt must be answered by the user. Moreover, these standards are not standards for EHR systems - they only define some parts of these systems, and not how the systems should be implemented. The development of the fully functional EHR System is an important and non-trivial task for local (country) authority, in its own unique environment.

Methodology for complex domain systems

Primarily the standardization of EHR is hard because of the specific domain – medicine. Up till now there are more then 360,000 concepts with over 975,000 descriptions and approximately 1.47 million relationships (SNOMED [6]). As the domain changes [3] rather quickly, it is far too dynamic to apply conventional software development approach – hard-coded semantics of the domain into applications and databases.

The key concept that allows overcoming such complex domains is *the separation of knowledge and information levels* [2]. Knowledge here is understood as a set of statements that are common for all entities of the domain (e.g., blood pressure consists of two measurements: systolic and diastolic); information – as a set of statements that are specific to some individual entity of the domain (e.g. Johns' blood pressure is 120/80).

The first part of this two-level model is *reference architecture* – generalized building blocks, which lack knowledge of the domain. These blocks can be linked with each other in various ways, thus forming very complex structures. Moreover, reference architecture is built in software development phase, and deployed only once.

The second part of the model is actual knowledge or the *ontology* of the domain. It defines domains' vocabulary and concepts (rules, relationships, etc.), which constraint *how* building blocks from reference architecture can be linked. These constraints are called *archetypes*. Ontology of the domain is never complete - it can be augmented at any time, without changing reference model.

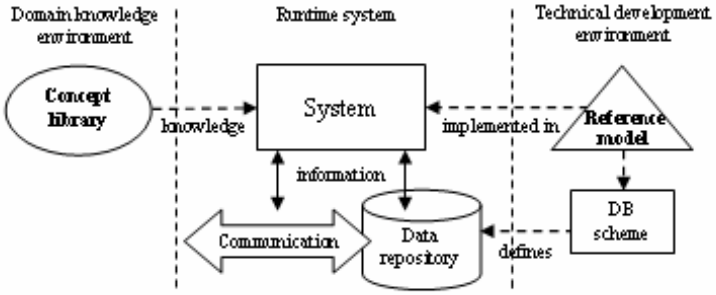


Fig. 1. Two-level methodology system: separation of knowledge & information [2].

Information systems, based on this methodology (Fig. 1), are very flexible. The reference model is defined, built and deployed only once, by software engineers. The ontology, on the other hand, can be altered through entire lifetime of the system, and these changes are done only by the domain specialists.

Legacy data migration

There are a lot of heterogenic healthcare-related systems and a lot of data in them. EHR concept implies that electronic health record as electronic document should be a *view* of all patients' data, gathered from all health-related systems. That is, some legacy systems may be parts – subsystems – of EHRS and act as source for specific data.

Possibility of health-related legacy system to be used as subsystem of EHRS totally depends on the legacy system itself. There are several popular ways to integrate different data sources into EHRS: a direct mapping, or natural language processing (NLP) and tagging. While NLP and tagging works with the rich free text documents (parts of these texts can be identified to be relevant to specific parts of EHR), direct mapping approach works with strictly defined databases. Also, methodologies have been developed for direct mapping to work with archetype-based information systems [5].

In the worse case scenario, when legacy system has both structured and unstructured (e.g. short free text) fields with mixed information, direct mapping cannot be applied, and more sophisticated methodology must be employed.

Semiautomatic archetype-based extraction of legacy data

When legacy data is too complex to apply direct mapping on-the-fly, relevant data needs to be extracted and placed inside new EHRS. This can be done by applying enriched-archetype based extraction. As the small ontology-based data extraction from unstructured documents has been proven quite successful [4], same idea may be used in medicine. Archetypes and terminology *from* EHR standard can be considered as a basic medical ontology. This ontology may be augmented, if it doesn't have some definitions that are present in legacy data. Moreover, specific terms that are relevant to local environment (or legacy data) should be introduced, thus enriching archetype definitions. Finally, employing special tools, based on EHR standard, an extraction of the legacy data can be preformed. Since automatic processing in medicine has the lowest trust, human expert must supervise such extraction process. Steps of this methodology should be:

1. Analysis of legacy application by an IT engineer and domain expert, to determine whether application/database can/should be used directly in EHRS.
2. If legacy application/database cannot be used directly, legacy data should be analyzed by domain expert, identifying relevant data, choosing set of needed archetypes, and augmenting the ontology by enriching archetype vocabularies, defining new archetypes.
3. Start of semiautomatic extraction process. As extraction tool analyzes, extracts and stores legacy data in some temporary storage, domain expert, with help of specific tools, supervises this extraction. Documents which were extracted successfully should be permanently stored in EHRS. The ones, that have errors, should be manually edited, or marked for re-extraction.

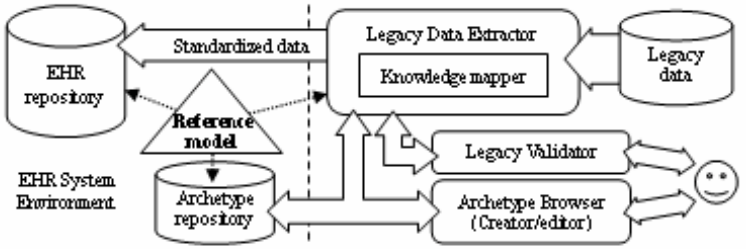


Fig. 2. Coarse-grained framework of semiautomatic legacy data extraction.

There should be a set of tools to aid both the IT engineer and the domain expert in such complex process: a flexible EHR-ontology based legacy data extraction tool, configurable for different data sources; an archetype editor; a comparer utility for quick identification of recognized/not recognized legacy content. Simplified framework, based on this methodology, is shown in Fig. 2.

Conclusions

o It is essential to preserve medical data accumulated in legacy systems, which represent years of work in healthcare. Even thought, suggested semiautomatic archetype-based extraction methodology should be tested, to see how feasible it is, and how useful it is in terms of human labour, it promises to save time for both IT engineers and medicine experts, while transforming legacy systems (and legacy data) to useful ones.

o New EHRS should be based on two-level methodology, separating knowledge and information, in order to be future-proof. Usage of this methodology will save up a lot of EHRS development and maintenance time in future.

o Authorities should start analysis and implementation of EHRS based on EHR standards, even though these standards are not final yet. The foundation of EHR won't change that dramatically to outdate these new EHRS.

References

1. CEN. *Draft Standard Specification for Continuity of Care Record*. 2000. URL: http://www.centc251.org/WGII/N-03/ASTM_Draft_Standard Specification for Continuity of Care Record_CCR E31.28_DRAFT 2.02_6_25-03 1.doc
2. Beale T. *Archetypes: Constraint-based Domain Models for Future-proof Information Systems*. 2002. URL: http://www.deepthought.com.au/it/archetypes/archetypes_new.pdf
3. Rector A. L. *Clinical Terminology: Why Is It So Hard?* 2001. URL: http://www.med.umi-heidelberg.de/mi/education/mi/mdoc/Q4 Clinical Terminology - Why is it so hard (Rector).pdf
4. David W. Embley, Douglas M. Campbell, Randy D. Smith, Stephen W. Liddle. *Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Documents*.1998.URL: http://osm7.cs.byu.edu/deg/papers/ckim98.pdf
5. Bird L.J., Goodchild A., Beale T. *Integrating Health Care Information Using XML-Based Metadata* URL: http://titanium.dstc.edu.au/papers/HIC2000.pdf
6. College of American Pathologists. Systematized Nomenclature of Medicine (SNOMED). URL: http://www.snomed.org

Modern world of Electronic Health Records. Migration from legacy systems

A. Kazla

Institute of Biomedical Engineering, Kaunas University of Technology, Lithuania

Electronic health record system is a must in today's healthcare. This paper gives a quick look at EHR world. The standardization of EHR, conventional software development approaches (especially in information systems), and legacy data are the main problems encountered in a modern EHR world. Emerging EHR standards are based on two-level methodology, where knowledge and information levels are separated. This approach should be used in all medicine-related systems, in order to make them future-proof. The paper also suggests a semiautomatic archetype-based methodology for extraction of legacy data, which should be used to transfer data from legacy systems to the new ones, build with the two-level methodology.