

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA

Andrius Lauraitis

**Turinio valdymo sistemos modernizavimo metodika
taikant Topic Maps technologiją**

Magistro darbas

Darbo vadovas

doc. Rita Butkienė

Kaunas, 2011

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA

Andrius Lauraitis

**Turinio valdymo sistemos modernizavimo metodika
taikant Topic Maps technologiją**

Magistro darbas

Recenzentas
doc. dr. Jonas Čėponis

2011-05-24

Darbo vadovas
doc. Rita Butkienė

2011-05-24

Atliko
IFM-9/4 gr. Stud.
Andrius Lauraitis

2011-05-24

Kaunas, 2011

Methodic of applying Topic Maps technology to modernization of a content management system

Summary

The usage of a classic content management system (CMS) has spread in the market of informational technologies because information is edited, authorized and published much easier in a multi-user environment. CMS is an ideal choice for large-scale organizations willing to control big amount of data from various resources. Usually this kind of information is presented in a web portal.

However some weak points are found in a CMS as well. Particularly in organization of information, metadata support areas. Associations between various CMS objects such as blogs, wiki pages, information display elements aren't supported enough. Similar to a file system many typical CMS store specific content objects in hierarchical catalogs. It is quite difficult to use this tree structure as information doesn't fit there. Furthermore one association "*type-instance*" type does not define the purpose of information in the catalogs clearly.

So this article describes how Topic Maps technology helps to avoid the limitations in organization of information used in a content management system. All the strengths of a classic CMS remain. Technology is built on top of CMS and lets the user describe the desired content in a topic map. Methodic explains Ontopia and Liferay integration, necessary configuration, ontology making and displaying Topic Maps information with portlets. Nowadays this open source approach is claimed to be one of the easiest and cheapest in building Topic Maps based web portals.

Turinys

Terminų ir santrumpų žodynas	6
1 Įvadas	8
2 Teminių tinklų ir turinio valdymo sistemų analizė.....	10
2.1 Analizės tikslas ir metodai.....	10
2.2 Tyrimo sritis, objektas ir problema	10
2.2.1 <i>Topic Maps</i> technologija	11
2.2.2 Turinio valdymo sistema (TVS)	28
2.2.3 <i>Liferay TVS</i>	29
2.2.4 Teminiai tinklai ir turinio valymo sistemos	31
2.3 Panašių sistemų Lietuvoje ir tarptautiniu mastu analizė	32
2.3.1 Italų operos teminis tinklas.....	32
2.3.2 Norvegijos kultūrinių įvykių apžvalgos internetinė svetainė.....	33
2.3.3 Projektas „Inotema“	33
2.3.4 Teminiai tinklai programavimo vaizdinių pamokų sistemoje.....	34
2.4 Siūlomas sprendimas	35
2.5 Darbo tikslas ir uždaviniai	35
2.6 Analizės išvados	36
3 Reikalavimų specifikacija	37
3.1 Reikalavimai TVS modernizavimo metodikai	37
3.2 Reikalavimai programavimo mokymo demonstracinei sistemai.....	38
3.2.1 Kompiuterizuojami panaudos atvejai	38
3.2.2 Reikalavimai sistemos vartotojams	43
3.3 Nefunkciniai reikalavimai ir apribojimai	44
3.3.1 Reikalavimai sistemos apimčiai ir sudėtingumui	44
3.3.2 Kiti nefunkciniai reikalavimai	45
3.4 Reikalavimai teminio tinklo duomenims.....	46
3.5 Reikalavimų apibendrinimas	47
4 <i>Ontopia</i> ir <i>Liferay</i> integracijos projektas	48
4.1 Reikalavimų analizė	48
4.2 Sistemos loginė architektūra.....	49
4.2.1 <i>Liferay</i> posistemis.....	50
4.2.2 <i>Ontopia</i> ir <i>Liferay</i> integracijos posistemis.....	52
4.3 Vartotojo sąsajos modelis.....	54
4.4 Sistemos elgsenos modelis	55
4.5 Duomenų bazės schema teminiam tinklui saugoti	55
4.6 Realizacijos modelis	58
5 <i>Ontopia</i> ir <i>Liferay</i> integracijos metodikos realizacija	60
5.1 Konfigūracija	61
5.1.1 Pasiruošimas konfigūracijos darbams.....	62
5.1.2 Integracijai reikalingų portletų failų peržiūra <i>SDK</i> kataloge	62
5.1.3 Log failo analizė	62

5.2	Teminio tinklo ruošimas ir pildymas turiniu	63
5.2.1	<i>Liferay</i> portalo pradinio puslapio paleidimas	63
5.2.2	Turinio atvaizdavimo objektų struktūrų kūrimas temų tipams	63
5.2.3	Temų egzempliorių ruošimas portale	64
5.2.4	Ontologijos pildymas teminių tinklų redaktoriumi <i>Ontopoly</i>	65
5.3	Portalo prototipo kūrimas	67
5.3.1	Puslapių skaičiaus, struktūrų ir temų tipų šablonų nustatymas	67
5.3.2	Portletų konfigūravimas <i>tolog</i> užklausomis	69
5.3.3	Portletų papildomų nustatymų konfigūravimas	71
5.4	Teminio tinklo saugojimas reliacinėje duomenų bazėje.....	72
5.4.1	JDBC tvarkyklės nustatymas	72
5.4.2	Savybių failo parinkimas	72
5.4.3	Duomenų bazės schemos kūrimas	72
5.4.4	Teminio tinklo importavimas į duomenų bazę	73
5.4.5	Teminio tinklo duomenų bazės ryšio užmezgimas su <i>Liferay TVS</i>	73
6	Eksperimentinis demonstracinės sistemos tyrimas	74
6.1	Sukurto teminio tinklo apimties įvertinimas	74
6.2	Demonstracinės sistemos apimties ir sudėtingumo įvertinimas	77
6.3	Sukurtos demonstracinės sistemos įvertinimas	80
7	Išvados.....	83
8	Literatūra	84

Terminų ir santrumpų žodynas

- ArticleList** - *Ontopia* ir *Liferay* integracijos specialus portletas, atvaizduojantis straipsnių sąrašą.
- Association** – *Topic Maps* duomenų modelio (*TMDM*) elementas, specifikuojantis asociaciją.
- DAML** (angl. *DARPA Agent Markup Language*) + **OIL** (angl. *Ontology Interface Layer*) - žinių valdymo technologijos.
- DDL** (angl. *Data Definition Language*) – duomenų struktūrų aprašymo kalba, yra *SQL* kalbos dalis.
- DFD** (angl. *Data Flow Diagrams*) - duomenų srautų diagramos.
- DynamicContent** - *Ontopia* ir *Liferay* integracijos specialus portletas, atvaizduojantis turinį dinamiškai.
- Hook** – specifiniai moduliai, galintys paimti *Liferay* sistemos įvykių informaciją.
- Instance** - *Topic Maps* duomenų modelio (*TMDM*) elementas, specifikuojantis temos egzempliorių.
- YahooTree** – *Ontopia* ir *Liferay* integracijos specialus portletas, stulpeliais atvaizduojantis norimų temų ir jų egzempliorių sąrašą.
- JSP** (angl. *Java Server Pages*) – dinamiškų *web* puslapių kūrimo technologija.
- Liferay** – *Java* priemonėmis sukurta atviro kodo turinio valdymo sistema.
- Log failas** – specialus failas, fiksuojantis sistemos įvykius: klaidas, pranešimus ir t.t
- LTM** (angl. *Linear Topic Maps*) - tiesinis teminio tinklo kūrimo formatas.
- Merge** – *Topic Maps* temų suliejimo operacija.
- Occurrence** - *Topic Maps* duomenų modelio (*TMDM*) elementas, aprašantis pasireiškimą.
- Omnigator** – *Ontopia* modulis, leidžiantis naršyti po kiekvieną sukurta teminį tinklą.
- Ontopia** – teminių tinklų programų kūrimo ir palaikymo įrankis.
- Ontopia Navigator Framework** – *Ontopia Topic Maps* interneto programų kūrimo modulis.
- Ontopoly** – *Ontopia Topic Maps* redaktorius.
- OSL** (angl. *Ontopia Scheme Language*) – *Ontopia* teminio tinklo kūrimo formatas.
- OWL** (angl. *Ontology Web Language*) - internetinių sistemų ontologijų kūrimo kalba.
- Portletas** - grafinės sąsajos programinės įrangos komponentas tvarkomas interneto portaluose.
- RDBMS** (angl. *Relation Database Management System*) – reliacinių duomenų bazių valdymo sistema.
- RDF** (angl. *Resource Description Framework*) – žinių ir informacijos valdymo standartas, aprašantis interneto išteklius.
- RelatedTopics** - *Ontopia* ir *Liferay* integracijos specialus portletas, atvaizduojantis susijusias temas.
- Role** - *Topic Maps* duomenų modelio (*TMDM*) elementas, specifikuojantis asociacijos vaidmenį.

Scope - *Topic Maps* duomenų modelio (TMDM) elementas, specifikuojantis kontekstą.

Subject identifier - *Topic Maps* duomenų modelio (TMDM) elementas, specifikuojantis temos identifikatorių.

Svn – atviro kodo produkto versijų kontroliavimo sistema.

TAO – pagrindiniai teminio tinklo elementų tipai (angl. *Topic, Asociacion, Occurrence*).

TMAPI (angl. *Topic Maps Application Programming Interface*) - API, skirta kurti teminio tinklo pagrindu veikiančias programas.

TMRAP (angl. *Topic Maps Remote Access protocol*) - protokolas, padedantis komunikuoti su nutolusiais *Topic Maps* serveriai.

Tolog – loginė teminio tinklo užklausų kalba.

TologList - *Ontopia* ir *Liferay* integracijos specialus portletas, atvaizduojantis temos pasireiškimų sąrašą.

Topic Maps – žinių atvaizdavimo modelis (liet. Teminis Tinklas)

TopicName - *Ontopia* ir *Liferay* integracijos specialus portletas, atvaizduojantis temos vardą(-us).

TVS – Turinio valdymo sistema

UML (angl. *Unified Modelling Language*) - modeliavimo kalba.

URL (angl. *Uniform Resource Locator*) - specifikuoja internete esančius resursus pagal identifikatorių.

Variant - *Topic Maps* duomenų modelio (TMDM) elementas, specifikuojantis temos vardo variantus.

XSLT (angl. *eXtensible StyleSheet Language Transformation*) - XML failų transformavimo kalba.

XTM (angl. *XML Topic Maps*) - į XML panašus teminio tinklo kūrimo formatas.

ZTM (angl. *Zope Topic Maps*) - Zope karkasu paremta *Topic Maps* pagrindu veikianti turinio valdymo sistema.

1 Įvadas

Informacinių technologijų rinkoje turinio valdymo sistemų (TVS) taikymas yra itin paplitęs dėl paprasto turinio tvarkymo, redagavimo, autorizacijos, publikavimo kelių vartotojų aplinkoje. TVS – idealus pasirinkimas stambiom įmonėm, norinčiom valdyti didelius informacijos kiekius iš įvairių šaltinių, pristatyti save interneto portale.

Šalia stipriųjų turinio valdymo sistemų savybių, aptinkamos ir silpnybės, ypač informacijos klasifikavimo, navigacijos, paieškos, metaduomenų palaikymo srityse. Ryšiai tarp įvairių TVS objektų, tokių kaip internetiniai dienoraščiai, „wiki“ puslapiai ar informacijos atvaizdavimo elementai, dažniausiai palaikomi nepakankamai. Panašiai kaip ir failų sistemoje, daugelyje tipinių TVS specifiniai turinio objektai saugomi hierarchinės struktūros kataloguose [13]. Naudotis tokia struktūra greitai tampa sunku, nes sudėtingesnė informacija nebetelpa į medžio tipo struktūrą, o tik vienos rūšies tarp katalogų egzistuojantis tipo - potipio ryšys nenusako aiškios jų paskirties.

Norint pagerinti navigaciją po „didžiulę turinio žemę“*, reikiamus ryšius tarp TVS objektų reikia programuoti. Alternatyva – taikyti teminių tinklų (angl. *Topic Maps*) technologiją, kurios standartizuotas duomenų modelis [14] užtikrina norimą ontologiją ir semantiką. Tinkamai pritaikius *Topic Maps* turinio valdymo sistemoje, vartotojui pakanka suprojektuoti puslapių struktūrą, sudėti atvaizdavimo komponentus grafinės sąsajos pagalba. Teminis tinklas ne tik leidžia susieti įvairius informacinius objektus (temas) įvairiais ryšiais, bet ir suteikia galimybę neprarasti ryšio semantikos, išreiškiant viską nors ir ribotos, bet natūralios kalbos konstrukcijomis. Pavyzdžiui, ryšys tarp naujienos ir asmens gali būti specifikuotas kaip ryšys su tokia semantika: asmuo paskelbė naujieną (asmuo skelbimo autorius), o naujiena buvo paskelbta asmens (naujiena skelbimo rezultatas). Tokių išsamiai aprašytų ryšių tarp objektų panaudojimas TVS suteikia dilesnes galimybes valdyti ir ieškoti informacijos.

Visgi teminiai tinklai dar nėra pakankamai išplėtoti, o tinkamų sprendimų, kaip integruoti technologiją su tam pritaikyta TVS trūksta. Šiame darbe analizuojami *Topic Maps* ir TVS tiriamieji objektai bei kuriama turinio valdymo sistemos modernizavimo metodika, kuri išaiškintų, kaip sujungti teminio tinklo teikiamas lanksčias informacijos organizavimo ir naudingas TVS turinio pateikimo galimybes į vieną efektyvų derinį. Tokio pobūdžio metodika suteiktų TVS kūrėjams ir taikomųjų programų vystytojams daugiau praktinių žinių apie teminių tinklų naudojimą turinio valdymo sistemose ir taip populiarintų tokias sistemas Lietuvoje.

* Turima omenyje didelis informacijos kiekis turinio valdymo sistemoje, kai susigaudyti joje tampa sunku

Darbo struktūra:

Antrame skyriuje „Teminių tinklų ir turinio valdymo sistemų analizė“ apžvelgiama teminių tinklų paradigma, taikymo sritys, panašaus pobūdžio sukurtos sistemos. Analizuojamos TVS, palaikančios teminių tinklų standartą ir, atsižvelgus į pateiktus kriterijus, pasirenkamas palankiausias variantas.

Trečiame skyriuje „Reikalavimų specifikacija“ keliami reikalavimai kuriamai metodikai ir demonstracinei sistemai. Įvertinami įgūdžiai, kurių reikia vartotojams, norintiems išbandyti metodiką. Demonstracinės sistemos reikalavimai skaidomi į funkcinis ir nefunkcinius. Funkciniai reikalavimai apima panaudojimo atvejų modelį, sistemos funkcijų specifikacijas ir jų skirstymą į posistemius. Nefunkciniai reikalavimai aprašo teminio tinklo standartui, formatams, sistemos apimčiai bei sudėtingumui keliamus reikalavimus. Skyriaus pabaigoje pateikiami reikalavimai teminio tinklo duomenims.

Ketvirtame skyriuje „Ontopia ir Liferay integracijos projektas“ aprašoma *Ontopia* ir *Liferay* integracijos projekcinė dalis. Pateikiama loginė sistemos architektūra (nuo aukščiausio iki žemiausio lygio), specialios klasės ir interfeisai, reikalingi *Ontopia* ir *Liferay* integracijos posistemiumi. Pavaizduojami demonstracinės sistemos vartotojos sąsajos ir elgsenos (norimos vaizdinės medžiagos radimo procesas) modeliai. Analizuojamos teminio tinklo saugojimo reliacinėje duomenų bazėje galimybės.

Penktame skyriuje „*Ontopia* ir *Liferay* integracijos metodikos realizacija“ apžvelgiami integravimo proceso etapai ir žingsniai: konfigūravimo darbai, teminio tinklo ruošimas ir pildymas, puslapių skaičiaus, struktūros parinkimas, *tolog* užklausų programavimas ir teminio tinklo saugojimas duomenų bazėje.

Šeštame skyriuje „Eksperimentinis demonstracinės sistemos tyrimas“ tiriama parengtos metodikos kokybė, kuri nusakoma per tai, kaip išpildyti metodikos ir demonstracinės sistemos reikalavimai. Taip pat tiriamos teminio tinklo galimybės, sistemos apimtis ir sudėtingumas.

2 Teminių tinklų ir turinio valdymo sistemų analizė

Išskyrus darbo taikymo sritį, objektą ir keliamas problemas, analitinėje dalyje pagrindinis dėmesys akcentuojamas į Topic Maps technologiją, svarbiausius elementus, taikymu, vaidmeniu ontologijų kūrimo ir semantiniame tinkle. Teminių tinklų standartas yra pakankamai daug apimantis, o kūrimo įrankių pasirinkimas – platus: nuo *Topic Maps* schemų braižymo, paties tinklo kūrimo, redagavimo ir naršymo iki pritaikymo turinio valdymo sistemoje. Detaliau aptariamas teminių tinklų varikliukas (angl. engine) *Ontopia*, pagrindinės funkcijos.

Apžvelgus teminius tinklus, supažindinama su turinio valdymo sistema (TVS), pagrindinėmis savybėmis, trūkumais. Darbe siekiama suintegruoti teminius tinklus su tam pritaikyta TVS, todėl pateikiamas turinio valdymo sistemų, palaikančių *Topic Maps* standartą, sąrašas. Analizuojami tiek komerciniai, tiek atviro kodo sprendimai ir pasirenkamas priimtinausias – *Ontopia* su *Liferay*.

2.1 Analizės tikslas ir metodai

Pagrindinis analizės tikslas – išsiaiškinti teminių tinklų taikomųjų programų veikimo principus ir praktinį taikymą interneto sistemose, nustatant tam tinkamiausius įrankius bei išnagrinėjant jų galimybes. Tikslas išpildomas pasitelkus šiuos metodus:

- Mokslinės literatūros analizė (įvairios publikacijos, knygos, straipsniai, internetiniai seminarai, vaizdinė medžiaga);
- Panašaus pobūdžio *Topic Maps* pagrindu veikiančių esamų internetinių sistemų ir semantinio tinklo technologijų (*RDF*, *OWL*, *DAML+OIL*) analizė;
- Objektiniai sistemų analizės metodai.

Iškeltas tikslas padeda susidaryti bendrą siekiamo sprendimo viziją, kuri analitinėje dalyje išryškinama, apžvelgus teminių tinklų technologijos pagrindinius konceptus bei sąsajas su turinio valdymo sistema.

2.2 Tyrimo sritis, objektas ir problema

Tyrimo sritis – teminių tinklų (angl. Topic Maps) technologijos integravimo procesas tam pritaikytoje turinio valdymo sistemoje (TVS).

Tiriamasis objektas yra dvilypis, o jo dalys yra šios:

- Teminių tinklų technologija, jos duomenų atvaizdavimo modelis, taikymo sritys;
- Turinio valdymo sistema bei jos galimybės. Analizuojami teminių tinklų paradigmą palaikančių TVS kriterijai, pagal kuriuos išsirenkama tinkamiausia TVS.

Daugelis šiuolaikinių įmonių nori pristatyti savo teikiamas paslaugas internete. Tam naudojamos klasikinės turinio valdymo sistemos, kurios patogios dėl paprastų informacijos kūrimo,

redagavimo, publikavimo galimybių. Visgi stambiose organizacijose informacijos kiekis būna labai didelis, o rasti tinkamą informacijos vienetą tampa sudėtinga. Norint išspręsti šią problemą reikiamus ryšius tarp TVS objektų reikia programuoti patiems. Darbe nagrinėjama alternatyva – teminių tinklų technologijos pritaikymas turinio valdymo sistemoje.

Toliau skyriuje akcentuojamos pagrindinės tiriamojo objekto dedamosios: teminiai tinklai, turinio valdymo sistema bei jų integracija.

2.2.1 *Topic Maps* technologija

Norint išsiaiškinti teminių tinklų svarbiausius dalykus ir galimus taikymus, technologijos analizė pateikiama šiais baziniais aspektais:

- Susipažinimas su *Topic Maps* paradigma (technologijos prigimtis ir sprendžiamos problemos);
- Vaidmuo žinių struktūrose ir informacijos valdyme;
- Teminio tinklo koncepcinis modelis (metamodelis), akcentuojant pagrindinius elementus (angl. *TAO*);
- Vaidmuo ontologijų kūrime;
- Panašių, semantinių tinklų pobūdžio, technologijų (*RDF*, *DAML+OIL*) analizė;
- Vaidmuo internetinių sistemų architektūroje;
- Sąsaja su reliacinėmis duomenų bazėmis;
- Kūrimo įrankiai.

2.2.1.1 *Topic Maps* paradigmos apžvalga

Milžiniškas pasaulinio tinklo (*www*) augimas reikalauja naujų technologijų informaciniams srautams, duomenims ir žinioms valdyti. Informacijos tiek daug, kad dažna problema tampa reikalingų duomenų paieška, klasifikacija. *Topic Maps* (teminiai tinklai) – naujas ir galingas būdas kontroliuoti „plačius“ interneto pasaulio vandenį, standartizuotai skirstyti informaciją. Yra dvi dimensijos ieškant prieigos prie informacijos: kokie informacijos šaltiniai ir kaip ji interpretuojama. *Topic Maps* apima abi šias dimensijas terminų sąrašo nuorodomis, kurios įtraukiamos į dalykinės srities ontologiją [5].

Pradžioje teminiai tinklai buvo suprojektuoti tvarkyti indeksus, turinius, dalykines rodykles, tačiau šiomis dienomis technologija tarnauja kaip įrankis, padedantis apibrėžti analizuojamos problemos sprendimų erdvę, kaupiti žinių bazę.

Topic Maps technologija gali būti interpretuojama, kaip pasirinktos dalykinės srities žemėlapis, bet ne pati teritorija. Teminiai tinklai tampa prasmingi tada, kai identifikuoja tos teritorijos objektus, jų ryšius ir išteklius [1]. Technologija siejama su semantiniu tinklu, nes šioms

paradigmoms keliamas reikalavimas pristatyti žinias prasmingai - užtikrinti žmonių ir kompiuterių bendravimą.

2.2.1.2 Topic Maps žinių struktūrose ir informacijos valdyme

Šiame skyriuje aptariama, kaip teminiai tinklai gali būti panaudojami organizuojant žinias. Prieš gilinantis į patį *Topic Maps* mechanizmą ir sudedamąsias dalis, išsiaiškinami informacijos navigacijos būdai: indeksai, žodynai su paaiškinimais, tezaurai. Analizuojamas semantinio tinklo ir *Topic Maps* ryšys, pateikiamas teminių tinklų vaidmuo ontologijos kūrime [1].

2.2.1.2.1 Indeksai

Žodis „indeksas“ reiškia daug dalykų, turinčių nuorodos elementus. Tradicinis indeksas yra knygos žinių žemėlapis: pateikiamas apžvelgiamų temų sąrašas, kurių paieška domina skaitytoją, bei nuorodos į temas. Pagrindinės tradicinio turinio dalys:

- (Surikiuotas) temų vardų sąrašas;
- Nuorodos į temų egzempliorius.

Pateikiamas Italijos operos solistų ir jų atliekamų kurinių indeksas:

La Bohème, 10, 70, **197-198**, 326
Cavalleria Rusticana, 71, **203-204**
The Girl of the Golden West, see *La fanciulla del West*
Leoncavallo, Ruggiero
 I Pagliacci, 71-72, 122, **247-249**, 326
Madama Butterfly, 70-71, **234-236**, 326
Manon Lescaut, **294**
Mascagni, Pietro
 Cavalleria Rusticana, 71, **203-204**
Puccini, Giacomo, 69-71
 La Bohème, 10, 70, **197-198**, 326
 La fanciulla del West, **291**
 Madama Butterfly, 70-71, **234-236**, 326
 Manon Lescaut, **294**
 Tosca, 26, 70, **274-276**, 326
 Turandot, 70, **282-284**, 326
Rustic Chivalry, see *Cavalleria Rusticana*
singers, 39-52,
 See also individual names
 baritone, 46
 bass, 46-47
 soprano, 41-42, 337
 tenor, 44-45
soprano, 41-42, 337
tenors, 44-45
Tosca, 26, 70, **274-276**, 326
Turandot, 70, **282-284**, 326

Šiame indekse solistai aprašomi reguliariu šriftu, o jų atliekami kūriniai – pasviru. Paryškinti skaičiai nusako nuorodas į temų egzempliorius. „See also“ nuorodos apibrėžia galimus ryšius tarp temų.

Taigi tradicinis indeksas iš esmės yra pasirinkto objekto (knygos, sistemos) žinių žemėlapis, o tai labai glaudžiai susiję su teminio tinklo koncepcija.

2.2.1.2.2 Žodynai su paaiškinimais (angl. *glossaries*)

Žodynai su paaiškinimais – terminų sąrašas, kuris gali būti traktuojamas kaip indeksas su vienu išreikštu egzemplioriumi. Balsų žodynas su paaiškinimais:

bosas: Žemiausias vyro balso tipas. Operose bosai dažniausiai atlieka šventikų vaidmenis.

pirmoji dama: žr. **prima donna**.

Leitmotif (Vokiškai “LIGHT-mo-teef”): pagrindiniam operos solistui paskirta muzikos tema.

prima donna (“PREE-mah DOAN-na”): Italų “pirmoji dama”. Dainininkė, kuri atlieka herojės vaidmenį.

sopranas: Aukščiausias moters balso tipas.

Kaip ir indeksas žodynas su terminais gali turėti nuorodas „žr.“ į susijusias temas. Taip pat žodyne gali būti papildomos informacijos apie terminą [1]. Pagrindiniai akcentai – temų aprašai. Šie išvardinti dalykai susieti su *Topic Maps* technologijos savybėmis.

2.2.1.2.3 Tezaurai

Tezaurai pabrėžia kitus indekso aspektus. Tai tarpusavyje susietų dalykinės srities terminų tinklas, todėl informacinėse sistemose jie gali būti realizuoti taikant *Topic Maps* technologiją. Pagrindinė tezaurų savybė – ryšiai tarp terminų [1]. Turint konkretų terminą tezauras apibrėš tą pačią prasmę turinčius bei įvairesnių kategorijų terminus.

sopranas

apibrėžimas

Aukščiausias moters balso tipas.

platesni terminai

vokalistas, dainininkas

siauresni terminai

lyrinis sopranas, dramatinis sopranas

susiję terminai

mezzo-sopranas

Toliau analizuojamos kitas žinių atvaizdavimo būdas – semantinis tinklas ir jo sąsaja su *Topic Maps*.

2.2.1.2.4 Topic Maps ir semantinis tinklas

Indeksai, žodynai su aprašymais ir tezaurai – būdas susieti žinių struktūrą. Semantiniam tinklui iškeliamas naujas reikalavimas pristatyti žinias prasmingai t.y. žmonių ir mašinų bendravimui užtikrinti.

Semantinis tinklas susideda iš informacinių viršūnių ir ryšių tarp jų. Viršūnės dažniausiai apibūdina objektus, idėjas arba specifines situacijas. Ryšiai nusako prasminę priklausomybę tarp viršūnių. Tinkle vartotojui suteikiama galimybė modeliuoti ryšius tarp viršūnių. *Topic Maps* gali fiksuoti ryšius tarp informacijos objektų, bet teminiai tinklai taip pat leidžia tiems objektams prieiti prie įvairių duomenų saugojimo vietų. Tokiu atveju žinių bazė gali būti projektuojama ne tik atskleisti ryšius tarp idėjų, bet ir priskirti joms resursus.

Semantinio tinklo pavyzdys galėtų būti šeimos genealoginio medžio sudarymas. 1 pav. pateiktas tokio medžio fragmentas [1].



1 pav. Semantinio tinklo pavyzdys: Reda yra Agnės palikuonis, o Liudvikas - Redos

Semantinis tinklas dažnai naudojamas modeliuoti žinias ekspertinėse sistemose. Tokios sistemos naudoja faktus ir taisykles, kurios panaudojamos dideliems duomenų kiekiams analizuoti. Kompiuterinėse sistemose semantinis tinklas suprojektuojamas taip, kad pagal ryšius būtų galima atsekti viršūnėse esančią informaciją.

Topic Maps ir semantinio tinklo panašumai:

- Tiek *Topic Maps*, tiek semantinis tinklas yra organizuoti į tinklinę informacinių viršūnių ir jų modulių struktūrą;
- *Topic Maps* ir semantinis tinklas leidžia vartotojui modeliuoti ryšius tarp viršūnių;
- *Topic Maps* ir semantinis tinklas leidžia vartotojui pridėti prasmingos informacijos į tinklo viršūnės ir ryšius tarp jų.

Topic Maps ir semantinio tinklo skirtumai:

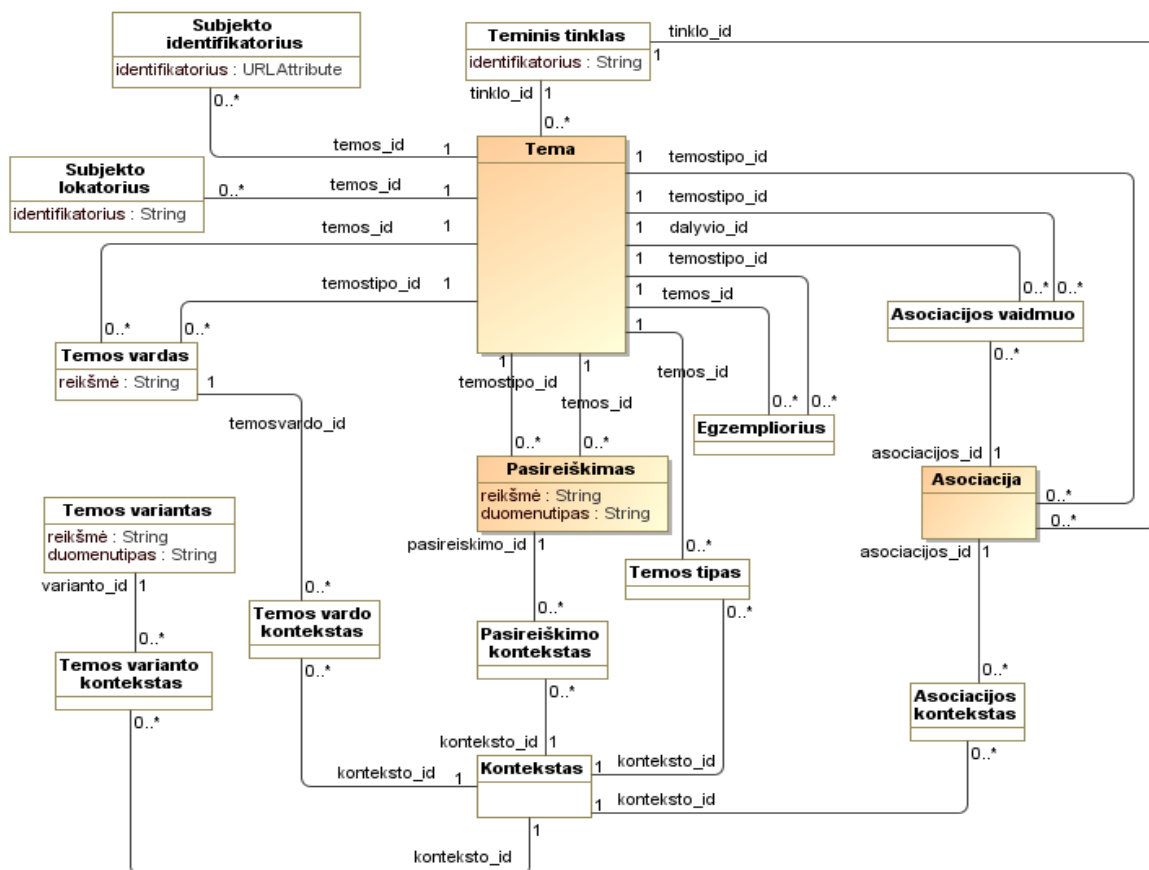
- *Topic Maps* pagrindinė idėja koncentruojasi ties temų navigacija, o ne ryšių tarp temų kūrimu;
- Semantinis akcentuoja prasminius ryšius tarp viršūnių ir žinias;
- *Topic Maps* ryšiai neturi krypties, tuo tarpu semantiniame tinkle kryptis egzistuoja.

Realizuojant semantinį tinklą iš *Topic Maps* reikalinga temos priskirti prasminių savybių. Ryšiai kuriami pagal tranzityvumo, refleksyvumo ir simetrijos savybes [1].

2.2.1.3 Teminio tinklo koncepcinis modelis

Teminio tinklo koncepcinis modelis (metamodelis) sudarytas pagal *TMDM* (angl. *Topic Maps Data Model*) [12] principus. Tokį standartizuotą modelį galima taikyti *Topic Maps* taikomuosiose programose ir saugoti teminio tinklo duomenis reliacinėje duomenų bazėje.

Šis koncepcinis modelis yra lankstus, tačiau gana sudėtingas, nes jis apima tinklo, egzempliorių ir pačio metamodelio informacijos vaizdavimo lygius. Kiekvienas temos egzempliorius, asociacijos, pasireiškimai, temų vardai, asociacijų vadmenys yra laikomi temomis. Tai suteikia neribotą lankstumą atvaizduoti žinias, tačiau didina sudėtingumo taikymą.



2 pav. Teminio tinklo koncepcinis modelis

Antrame paveiksle pateikta lietuviška teminio tinklo koncepcinio modelio esybių-ryšių diagrama. Trys esybės (Tema, Asociacija, Pasireiškimas) yra išskirtos tamsesne spalva, nes jos priskiriamos pagrindinių teminio tinklo elementų aibei.

2.2.1.4 Pagrindiniai *Topic Maps* kūrimo elementai

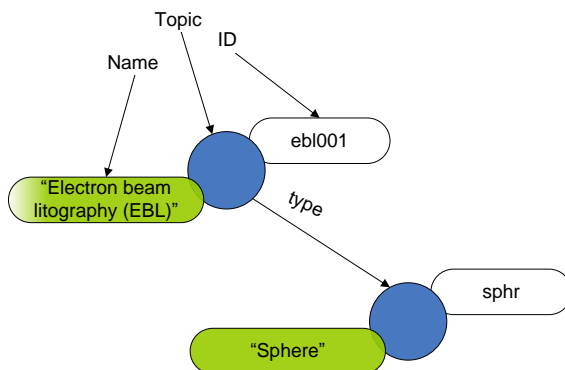
Topic Maps paradigmą sudaro trijų tipų elementai (konceptai): temos (angl. *Topics*), ryšiai (angl. *Association*) ir pasireiškimai (angl. *Occurrences*) (angl. *TAO*). Šiame skyrelyje apžvelgiami šie elementai ir pateikiamos jų grafinės iliustracijos.

2.2.1.4.1 Temos

Temos yra pagrindiniai *Topic Maps* konstravimo blokai. Bendriausiu atveju tema gali būti bet koks daiktas: žmogus, esybė ir t.t. Visai nesvarbu, ar šis daiktas turi tam tikras charakteristikas ar ne. Temos glaudžiai siejasi su subjektu – realiu pasaulio daiktu, kurį pristato, atspindi tema.

Temos dažnai turi vardą (angl. *name*). Triviausias atvejis, kai tema turi tik vieną vardą, bet *Topic Maps* ypatingi yra tuo, kad viena tema gali turėti kelis vardus. Tai – stabilaus žinių pasikeitimo tinkle reikalavimas. *Topic Maps* jungia kelių vardų temas, bet ne pačius vardus.

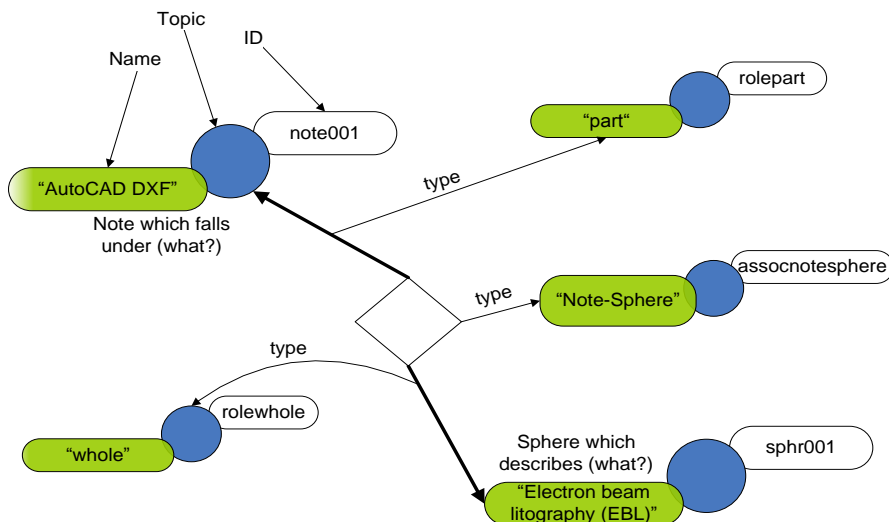
Temos kompiuterinėse sistemose taip pat turi identifikatorių ir tipą, kuriam pagal *Topic Maps* standartą nėra kažkokių specialių apribojimų. Antrame paveiksle pavaizduota tema „Electron beam lithography (EBL)“, priklausanti tipui „Sphere“.



3 pav. *Topic Maps* temos elemento iliustracija

2.2.1.4.2 Asociacijos

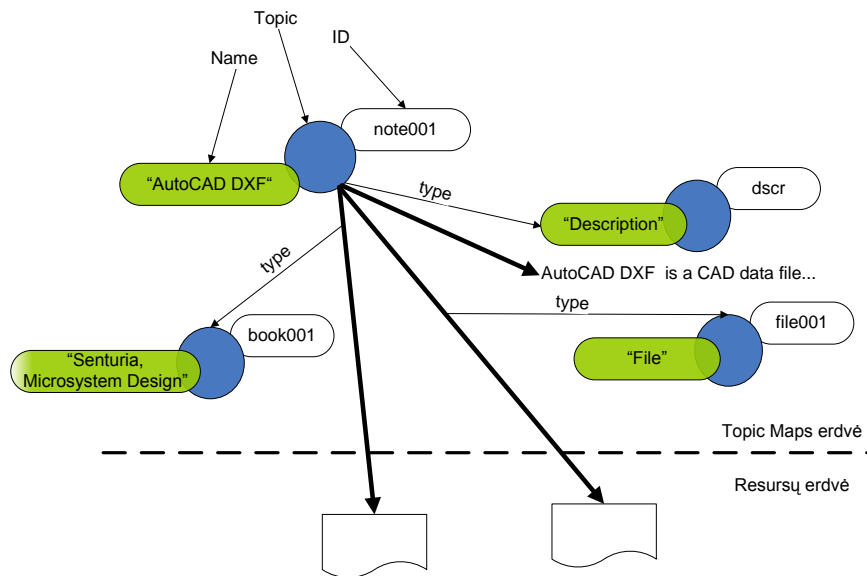
Temos yra jungiamos ryšiais, kurie gali susidėti iš tam tikro vaidmenų skaičiaus. Tiek ryšys, tiek ryšio vaidmuo gali turėti tipą. Asociacijos, kurios gali būti ir *n-narės* (sujungti daugiau nei dvi temas), sujungia *Topic Maps* viršūnes į tinklą. Ketvirtame paveiksle pateiktas temų „AutoCAD DXF“ ir „Elektron beam lithography (EBL)“ jungimo pavyzdys. Specifikuojamas ryšio tipas „Note-Sphere“, tinkantis abiem asociacijos temoms, be to nurodoma, kokius vaidmenis atlieka jungiamos temos asociacijoje (pvz.: tema „AutoCAD DXF“ vaidina rolę „part“). Temos taip pat turi papildomus asociacijos pavadinimus (pasirinkus temą „AutoCAD DXF“ matysis ryšio pavadinimas „Note which falls under (what?)“, o žemiau nurodoma, kuriai sričiai tema priklauso).



4 pav. *Topic Maps* asociacijos elemento iliustracija

2.2.1.4.3 Pasireiškimai

Temos tinkle gali būti sujungiamos su tam tikrais, temai būdingais, informacijos ištekliais. Tokie ištekliai vadinami temų pasireiškimai [12]. Naujienos apie temą iš enciklopedijos, vaizdinė, grafinė temų medžiaga, komentarai, priskiriami pasireiškimų aibei. Kai pasireiškimas siejasi su išoriniu ištekliu, turėtų būti panaudotas pasireiškimo tipas, kuris perduotų temai informaciją, apie išteklio tipą ir ryšį su tema. Jei pasireiškimas - vidinė tekstinė išraiška, būtinas ryšis tik tarp temos ir pasireiškimo.



5 pav. *Topic Maps* pasireiškimo elemento iliustracija

Svarbus aspektas, išryškinantis *Topic Maps* galią, yra tai, kad temos ir pasireiškimai išskiriami į atskirus sluoksnius. Kaip parodyta 5 pav. pasireiškimai peržengia ribą tarp saugomos informacijos *Topic Maps* ir išteklių erdvėse.

2.2.1.5 *Topic Maps* vaidmuo ontologijų kūrime

Nepaisant to, kad teminiai tinklai atlieka informacijos išteklių indeksavimą, ši technologija taip pat realizuoja žinias. Prasmingos žinios padidina *Topic Maps* vertę. Kadangi ontologija yra žinių pristatymas, todėl ji yra svarbi *Topic Maps* technologijoje.

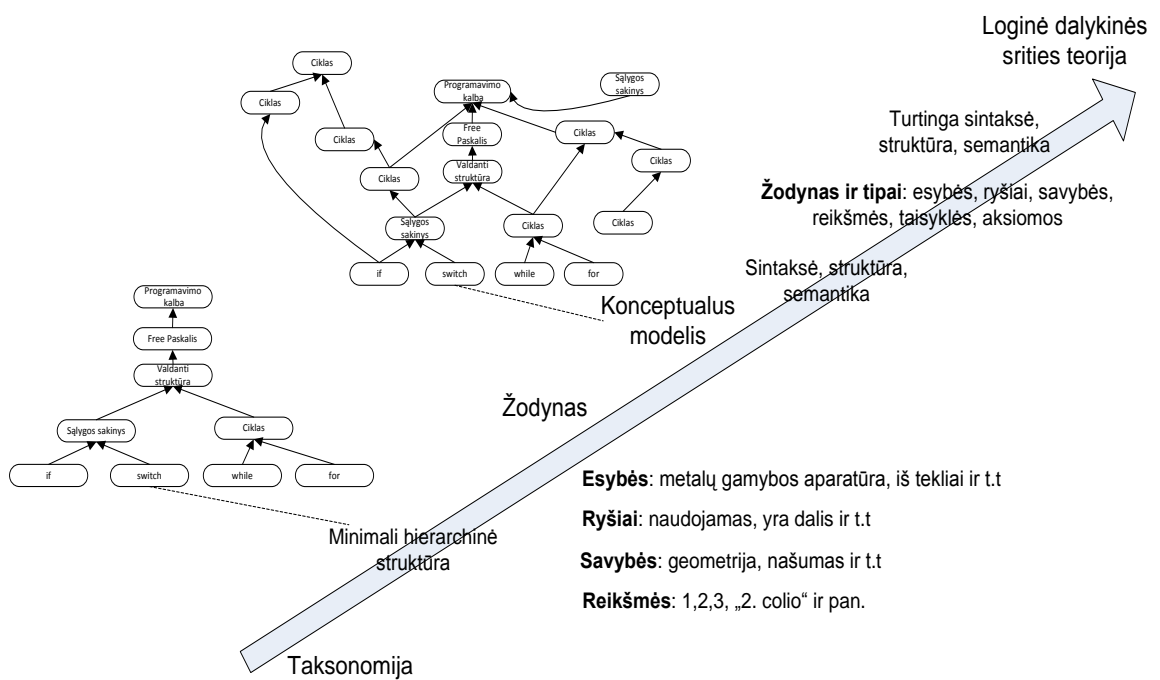
Ontologija gali būti apibrėžiama, kaip tam tikros dalykinės srities idėjų rinkinys, kuriame nustatomi tarpusavio ryšiai. Ontologija skiriasi nuo duomenų modelių tuo, kad ryšiai tarp esybių automatiškai apibrėžia ir pačias esybes. Ontologijos ryšiai apibrėžiami formaliai, išlaikant semantiką. Pagrindiniai ontologijos elementai:

- Esybės (daiktai);
- Ryšiai tarp esybių;
- Esybių savybės ir reikšmės;

- Funkcijos ir procesai;
- Apribojimai ir taisyklės apie esybę;

Topic Maps kalba išreiškia žinias, prasmingai susiejant jas su ištekliais. Ontologijos inžinerija, pritaikyta kurti žinių semantiką perteikiančius teminius tinklus, remiasi šiuo aspektu [1].

6 pav. pateiktas galimas ontologijos spektras, kuris gali kisti nuo paprastos taksonomijos aprašymo (žiniomis su minimalia hierarchine struktūra) iki žodyno (naudojamomis žiniomis su standartizuota terminija ir natūraliu kalbos apibūdinimu). Procesas tęsiamas ir ontologija gali būti traktuojama kaip konceptualus modelis (sudėtingesnis žinių pristatymas) arba suformuluota logine dalykinės srities teorija [1].



6 pav. Ontologijos spektras

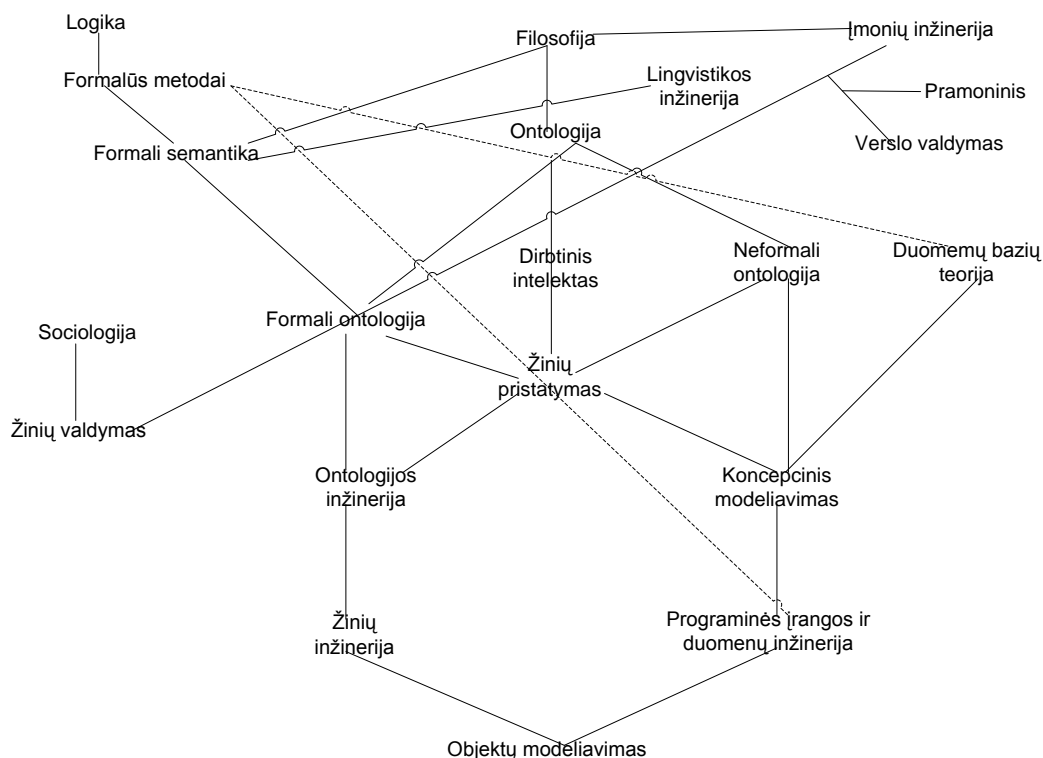
Apsibrėžus ontologijos spektrą, analizuojamos teminių tinklų ir ontologijos inžinerijos sąsajos.

2.2.1.6 Ontologijos inžinerija

Vienas iš didžiausių šių laikų iššūkių saitynui, informacijos gavimui ir žinių bazių našumui yra žinių technologijų realizacija. *Topic Maps* gali būti traktuojama kaip paradigma, padedanti įkelti realaus turinio informaciją interneto sistemose.

Ontologijos inžinerija nagrinėja ontologijų kūrimo ir realizavimo metodus. Ši disciplina apima žinių pristatymą ir valdymą, todėl siejasi su teminiais tinklais.

7 pav. pateikta ontologijos inžinerija ir su ja susijusios disciplinos [1].



7 pav. Ontologijos inžinerija

Ontologijos inžinerijos nagrinėjamos žinių valdymo ir pristatymo sritys yra realizuojamos įvairiomis kalbomis. Kitame skyriuje analizuojamos *OIL (Ontology Interface Layer)*, *DAML (DARPA Agent Markup Language)*, *DAML+OIL*, *RDFS (RDF schema)* kalbos.

2.2.1.7 Topic Maps ir RDF, DAML, OIL, OWL

Terminai *RDF* (angl. *Resource Description Framework*), *DAML*, *OIL* ir *Topic Maps* yra minimi, kai vyksta diskusijos apie žinių ir informacijos valdymą, semantinį tinklą. *OWL* (angl. *Ontology WEB language*) skirta internetinių sistemų ontologijų kūrimui. Retas, kuris gerai išmano daugiau nei vieną iš šių technologijų. Šiame skyrelyje trumpai apžvelgiamos šios technologijos ir kaip jos apibūdina tokias idėjas kaip „daiktas“, „ryšys tarp daiktų“, „daiktų savybė“, „daiktų vardas“ ir pan.

Topic Maps, *RDF*, *DAML*, *OIL* – standartai, kurie geriausiai ir greičiausiai padėtų vartotojui spręsti dalykinės srities problemą [3].

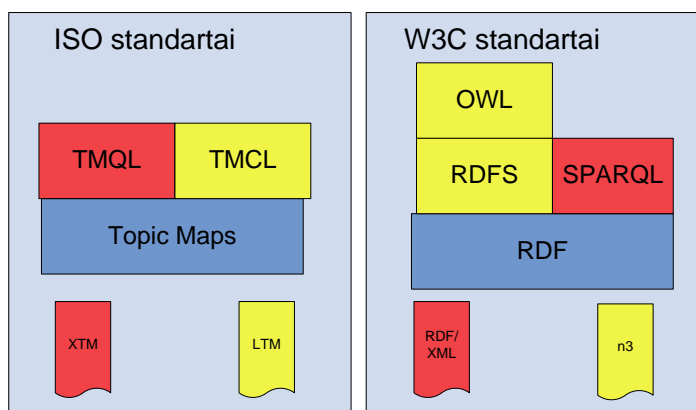
Topic Maps priklauso *ISO 13250* patvirtintam 2000 metų sausio mėnesį. Darbai buvo atliekami toliau ir kompanija *TopicMaps.org* sukūrė *XTM (XML Topic Maps)* žymių kalbą. Tai buvo *Topic Maps* performulavimas *XML* kalba, kuri paremta *XLink* nuorodomis. *Topic Maps* taikomi žinių valdyme, semantiniame tinkle, internetinėse sistemose, įmonių taikomųjų programų integravime.

RDF sukurtas W3C konsorciumo, kuris yra semantinio tinklo dalis. *RDF*, kaip informacijos valdymo technologija, pritaikoma: turinio bei žinių valdyme, portalų technologijoje ir e-prekyboje.

DAML – US valdžios tyrėjų grupės *DARPA* technologija, kurios specifikacija, kaip projekto dalis, aptinkami *daml.org* puslapyje. *DAML* palaiko semantinį tinklą, žinių valdymą ir informacijos paiešką. *DAML*, kitaip nei *RDF*, nėra duomenų modelis, o schemų kalba, kuri aprašo *RDF* duomenų modelio apribojimus. Nors *RDF* ir turi schemų kalbą *RDFS*, *DAML* yra šios kalbos praplėtimas. Tikroji *DAML* vertė yra ta, kad ji apibūdina *RDF* duomenis, todėl galima šiems duomenims suteikti daugiau semantikos. *DAML* taip pat gali būti *RDF* ir *Topic Maps* technologijų supanašėjimo rodiklis, nes pats *Topic Maps* teikia daugiau informacijos apie semantiką. *DAML* leidžia kontroliuoti *RDF* savybių reikšmėmis: *daml:samePropertyAs*, *daml:inverseOf*, *daml:TransitiveProperty* elementais. Taigi *DAML* suteikia *RDF* schemai daugiau semantikos [3].

OIL – Europos sąjungos iniciatyva sukurta žinių valdymo technologija. *OIL* labai panaši į *DAML* ir taip pat praplečia *RDF* schemą. Turi šiek tiek kitų nežymių skirtumų. Paskutinė *DAML* versija įtraukia *OIL* ir pervadina technologiją į *DAML+OIL*. Specifikacija gali būti aptinkama <http://www.daml.org/2001/03/daml+oil-index.html> puslapyje.

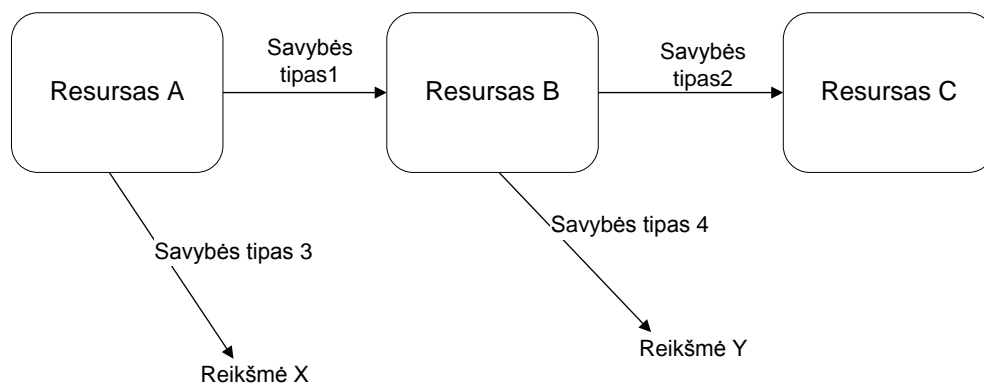
8 pav. pateikiamas ISO ir W3C standartų technologijų skirstymas. Akcentuojami *Topic Maps* ir *RDF*, kurie yra karkasai ant kurių konstruojamos kitos technologijos. *Topic Maps* palaiko *XTM* kalba realizuojamą *TMQL* (*Topic Maps Query Language*), bei *LTM* (*Linear Topic Maps*) kalba - *TMCL* (*Topic Maps Constraint Language*). Šios technologijos skyriuje nedetalizuojamos. Analogiškai *RDF* paliko *SPARQL*, *OWL*, *RDFS* technologijas [3].



8 pav. ISO ir W3C standartų žinių valdymo ir semantinio tinklo technologijos

OWL kalba leidžia išreikšti papildomos naudingos informacijos apie *RDF*. Tai nėra apribojimų kalba, bet kalba optimizuota ontologijos aprašymui. *OWL* yra *RDF* žodynas. *OWL* taip pat suteikia galimybę nusakyti *RDF* savybių apribojimus.

RDF resursas – bet koks objektas, apibūdinimas *URI*. Su ištekliais gali būti susiejamos savybės, jų tipai, reikšmė, apibūdinimas - *RDF* karkaso esminiai elementai [1].



9 pav. Pagrindiniai *RDF* elementai, jų žymėjimai

Topic Maps ir *RDF* panašumai:

- Ieško būdų sistemoms geriau suprasti informacijos semantiką internete;
- Aprašo ir karkase naudoja resursus nuorodomis arba vidiniai elementais;
- *Topic Maps* ir *RDF* gali būti naudojami kuriant semantinius informacijos tinklus. Tai žinių pristatymo technika;
- *RDF* savybės-reikšmės pora atlieka tą pačią funkciją kaip *Topic Maps* egzempliorius;
- *Topic Maps* remiasi temomis grindžiamu požiūriu, o *RDF* – ištekliais. *Topic Maps* pradeda nuo temų ir modeliuoja ryšių tinklą, sluoksniuojamą virš informacijos išteklių. Yra galimybė modeliuoti žinias be nuorodų į apatinį sluoksnį. Tą pačią funkciją gali atlikti ir *RDF*.

Topic Maps ir *RDF* skirtumai:

- *Topic Maps* modelis abstraktesnis;
- Labiausiai pastebimas *Topic Maps* ir *RDF* skirtumas tas, kad *Topic Maps* suteikia galimybę aprašyti tam tikras sąvokas (temas) apibrėžtame kontekste (angl. *scope*);
- Kadangi *Topic Maps* temos gali turėti savas charakteristikas siaurame kontekste, todėl atsiranda galimybė ignoruoti tam tikras nereikalingas savybes. Tai svarbu konstruojant semantinį tinklą. Šios savybės *RDF* neturi [1];
- Sumodeliuotą teminį tinklą perdaryti *RDF* technologija įmanoma, tačiau būtų prarandamos kai kurios prasmingos žinios. Jeigu *Topic Maps* tema būtų *RDF* resursas, tada ryšys tarp temų galėtų būti traktuojamas kaip savybės (ryšio savybė) ir jos reikšmės (kitas resursas) derinys.

RDF notacijoje dalykinės srities objektai apibūdinami ištekliais, turinčiais *URI*, tuo tarpu *Topic Maps* yra temos ir subjektai. 1 lentelėje pateikiama grafinė *RDF* ir *Topic Maps* objektų notacija.

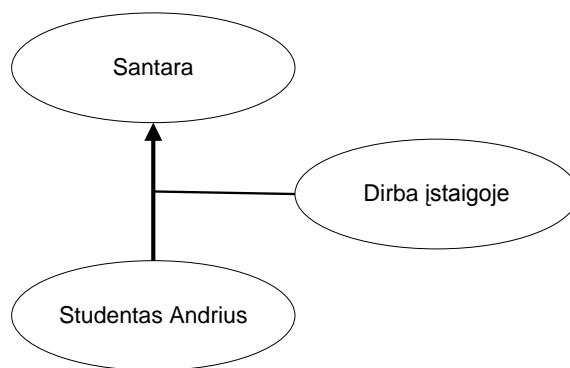
1 lentelė. Topic Maps temos ir RDF ištekliai

Objektas	RDF	Topic Maps
Programavimas	http://programavimopamokos.net	Subject identifier = http://programavimopamokos.net
Studentas Andrius	mailto:atsiliepimai@programavimopamokos.net	Subject identifier = mailto:atsiliepimai@programavimopamokos.net
Video failas	http://www.programavimopamokos.net/ikeltukai/FPS	Subject identifier = http://www.programavimopamokos.net/ikeltukai/FPS

Yra 3 pagrindiniai skirtumai, kaip *RDF* ir *Topic Maps* modeliuoja ryšius:

- Labiausiai pastebimas yra tai, kad *RDF* susieja vieną dalyką su kitu, tuo tarpu *Topic Maps* su tema gali susieti daug tą patį ryšį turinčių dalykų.
- *Topic Maps* ryšiai yra dvikrypčiai. Pvz. jei studentas Andrius dirba Santaroje, tai galima sakyti Santara įdarbina studentą Andrių. *RDF* tokia priklausomybė negalima.
- Nėra įmanoma nusakyti, kada *RDF* ryšiai yra tarp dviejų abstrakčių dalykų, o kada ryšys siejasi su resursu.

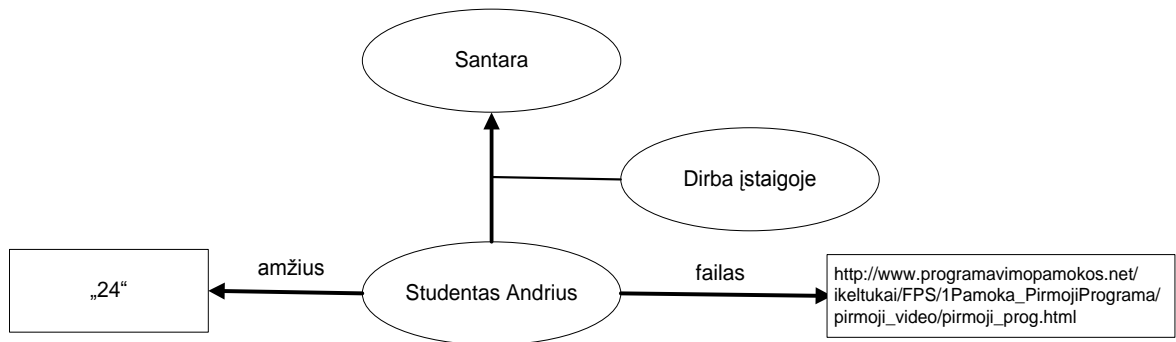
Dešimtame paveiksle pateikiamas *RDF* ryšys. Kaip specifikuoti šį ryšį *Topic Maps* notacija pateikta 2.2.1.3 skyriuje.



10 pav. Ryšys: Studentas Andrius dirba įstaigoje „Santara“ pagal *RDF* standartą

Modeliuojant žinias svarbu žinoti ir dalykinės srities objektų atributus. Pvz.: koks studento Andriaus amžius, kokios įrašytos vaizdinės pamokos. *RDF* standarte šis faktas suprantamas kaip resursas. 11 pav. pateiktas *RDF* atributų užrašymo pavyzdys. Pagrindinis *Topic Maps* ir *RDF*

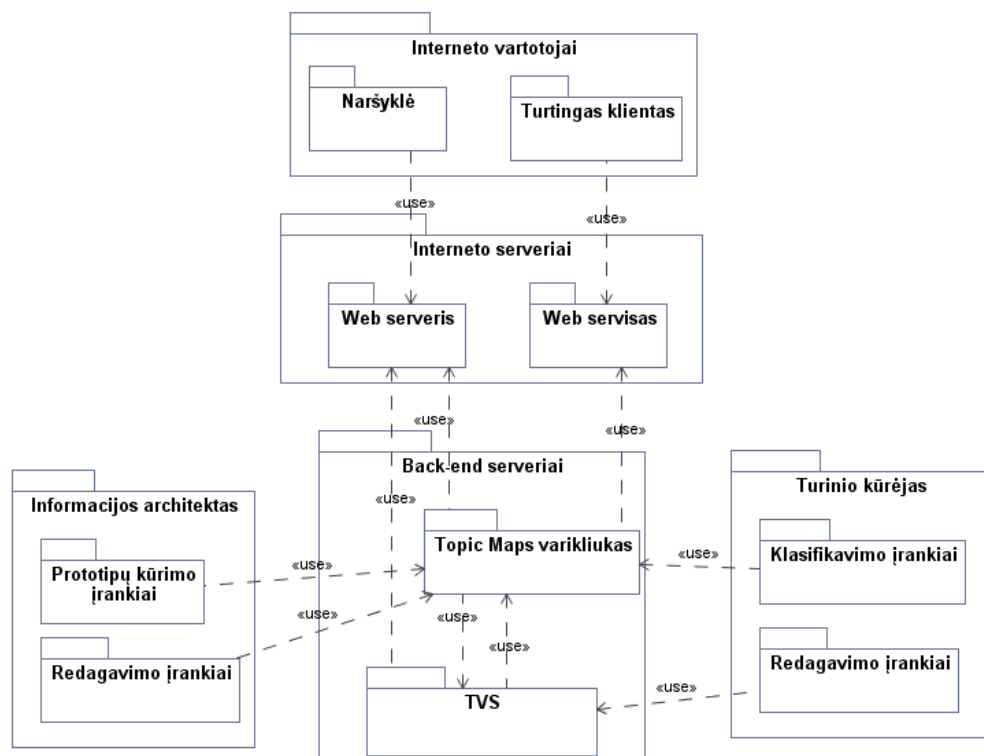
skirtumas specifikuojant resursus: *RDF* nėra specialaus žymėjimo, kad resursas surištas su išoriniu resursu (failu) [1]. *Topic Maps* tai atitiktų pasireiškimą ir jų susiejimą su temomis.



11 pav. Atributų žymėjimas pagal *RDF* standartą

2.2.1.8 *Topic Maps* internetinių sistemų architektūroje

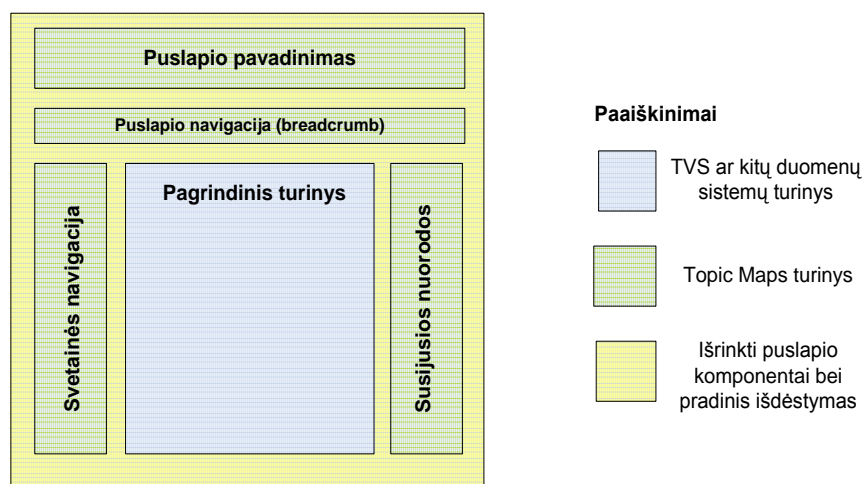
Realizuojant *Topic Maps* internetinių sistemų architektūroje svarbiausias sprendimas – išsiaiškinti, kaip integruoti *Topic Maps* komponentus į turinio valdymo sistemą. Svarstomi šie du aspektai: turinio kūrimo ir indeksavimo integravimą ir *Topic Maps* informacijos užpildymą turiniu. Gerai suprojektuota taikomoji programa leidžia atlikti turinio pakeitimus, nesigilinant į pristatymo ir turinio tvarkymo programos tekstą. 12 pav. rodoma internetinės sistemos, kurioje integruota *Topic Maps* technologija, architektūra[7].



12 pav. *Topic Maps* interneto sistemų architektūros kontekste

Temų žemėlapi tvarko *Topic Maps* varikliukas. Turinio kūrėjas ir informacijos architektas maitina varikliuką duomenimis. Publikuojant turinį yra du *Topic Maps* informacijos integravimo būdai:

- Turinio valdymo sistema (TVS) pasirūpina puslapio struktūra ir paskiria regioną, kuriame pridedama *Topic Maps* informacija [8] (13 pav.);
- Pačiame *Topic Maps* duomenų modelyje nusakoma informacijos iš dėstymo vieta.



13 pav. *Topic Maps* elementų išdėstymas internetiniame puslapyje

2.2.1.9 *Topic Maps* ir reliacinių duomenų bazių ryšys

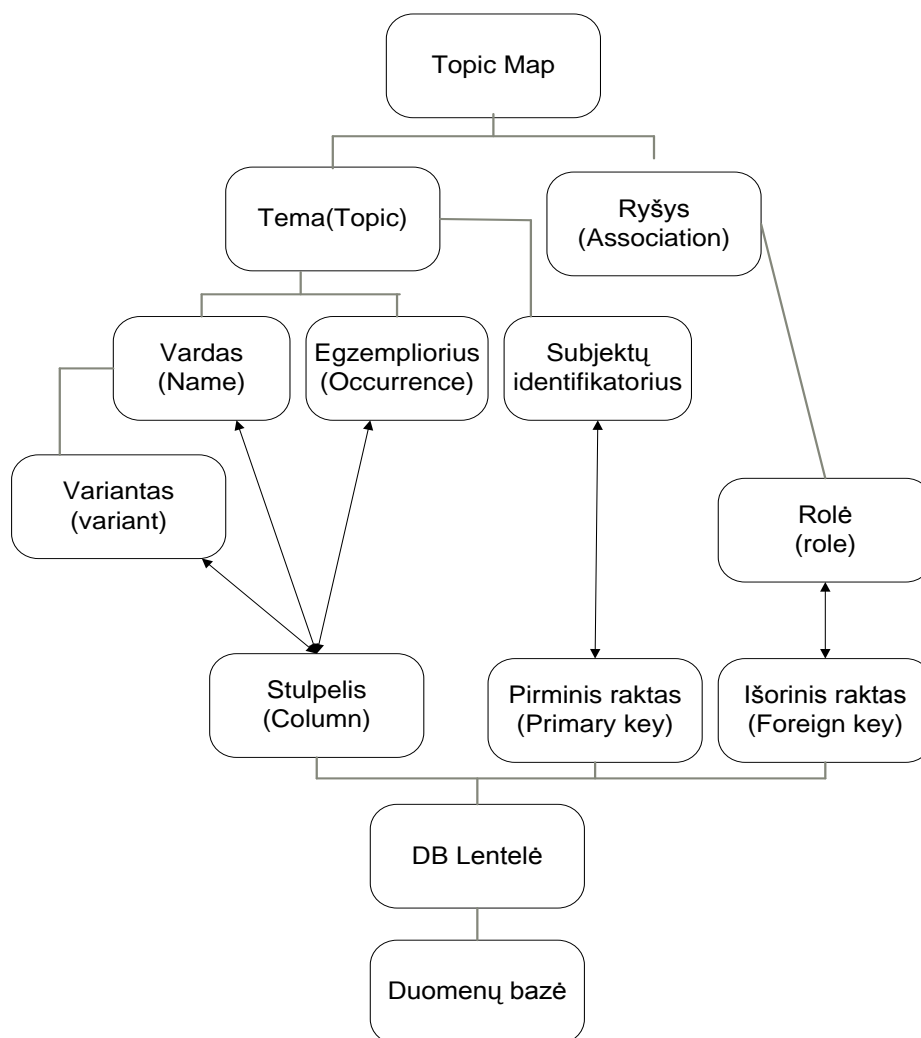
Standartiškai reliacinių duomenų bazių struktūra susideda iš daugelio lentelių. Kiekviena lentelė susideda iš tam tikrų specifinių elementų:

- Pirminio rakto: unikalai apibūdina duomenų elementą duomenų bazės lentelėje;
- Stulpelio: saugo informaciją apie susietą duomenų elementą;
- Išorinis raktas: susieja tarpusavyje duomenų elementus duomenų bazėje

Iš abstraktaus stebėjimo taško *Topic Maps* turi panašumų su standartinėmis duomenų bazių struktūromis, bet ši technologija teikia daugiau, duomenų elemento identifikavimo ir prasmės nusakymo galimybių:

- Tema: apibūdina *Topic Maps* duomenų elementą;
- Elemento identifikatorius: unikalai aprašo *Topic Maps* kūrimo elementą (angl. *TAO*);
- Subjekto identifikatorius: unikalai apibrėžia *Topic Maps* temą;
- Vardas: priskiria temai pavadinimą;
- Variant: kita temos elemento vardo išreiškimo forma;
- Egzempliorius: saugo informaciją apie *Topic Maps* duomenų elementą;
- Ryšys: dviejų ar daugiau temų susiejimas;
- Rolė: ryšio atskiros temos dalis.

14 pav. pateiktas galimas ryšys tarp reliacinės duomenų bazės ir *Topic Maps*: pirminis raktas - subjekto identifikatorius, stulpelis – egzempliorius. Tokia elementų priklausomybė leidžia susieti duomenų bazėje saugomą informaciją ir *Topic Maps* notaciją [8].



14 pav. Duomenų bazės objektų ir *Topic Maps* elementų ryšys

Kaip susieti pasirinktos dalykinės srities teminį tinklą su reliacinėje duomenų pateikiama ketvirtame skyriuje, o toliau analizuojami *Topic Maps* kūrimo įrankiai, pateikiamos jų charakteristikos.

2.2.1.10 *Topic Maps* kūrimo įrankiai

Iš teorinės pusės, apie teminius tinklus galima rašyti įvairių dalykų, nes, kaip minėta ankstesniuose skyriuose, technologija apima daug sričių. Kita vertus, kokia technologija, be praktinio taikymo?

Topic Maps yra nauja technologija, todėl egzistuojantys kūrimo įrankiai [20] keičiasi: išleidžiama nauja produkto versija, įdiegiamos papildomos funkcionalumo galimybės ir pan. Antroje lentelėje pateikiamas populiariausių teminių tinklų naudojimo įrankių sąrašas.

2 lentelė. *Topic Maps* įrankiai

Įrankis	Aprašymas	Nuoroda
Ontopia	Java priemonėmis sukurtas atviro kodo įrankių komplektas kurti ir palaikyti <i>Topic Maps</i> sistemas;	http://www.ontopia.net
Onotoa	teminių tinklų schemų kūrimo įrankis;	http://onotoa.topicmapslab.de/
QuaaxTM	TMAPI skirtas PHP kalbai;	http://quaaxtm.sourceforge.net/
TM++	C++ teminių tinklų varikliukas;	http://tmplusplus.sourceforge.net/
RTM(Ruby Topic Maps)	Ruby teminių tinklų varikliukas;	http://rtm.topicmapslab.de/
TMCore	.NET teminių tinklų varikliukas;	http://www.networkedplanet.com/Products/
Wandora	Teminiu tinklų kūrimo, vizualizavimo įrankis	http://www.wandora.org/wandora/wiki/index.php?title=Main_Page
Perl TM	PERL plėtiniai, pritaikyti teminiams tinklams	http://search.cpan.org/dist/TM/

Kitame skyriuje detaliau nagrinėjamas įrankis *Ontopia*, nes šiai dienai tai bene labiausiai išplėtotas *Topic Maps* kūrimo programų produktas.

2.2.1.11 *Ontopia*

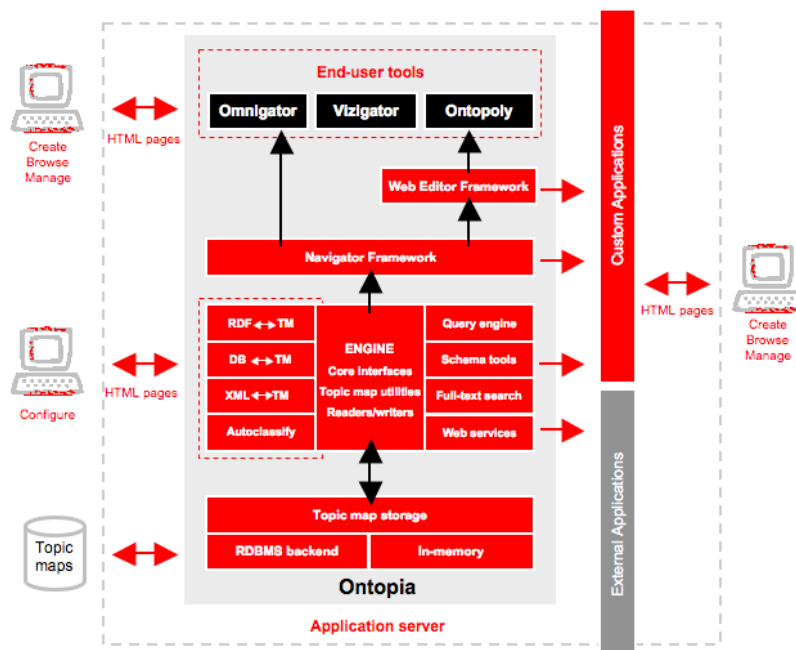
Norvegų firmos *Bouvet* produktas *Ontopia* [16] – Java priemonėmis sukurtas atviro kodo įrankių varikliukas, skirtas kurti ir palaikyti *Topic Maps* sistemas. Vartotojui pateikiami trys pagrindiniai produktai:

- *Omnigator*: leidžia naršyti ir peržiūrėti sukurtą teminį tinklą be jokio papildomo programavimo ar konfigūravimo. Įrankis – mokymo tikslams sukurta technologija, pademonstruojanti *Topic Maps* galią. Gali nuskaityti įvairių formatų (*XTM 1.0*, *XTM 2.0*, *LTM* ir kt.) teminį tinklą. Traktuojamas kaip testavimo įrankis, palaiko *Ontopia* schemas, *tolog*[†] [17] užklausas, indeksavimą, suliejimą, vizualią tinklo analizę.

[†] Loginė teminių tinklų užklausų kalba, išvesta iš Prolog. Yra reliacinių duomenų bazių SQL atitikmuo

- Ontopoly: teminių tinklų redaktorius. Pasirinkus *Topic Maps* kūrimą šiuo įrankiu pridedama teminių tinklų duomenų modelio schema. Naudodamas *Ontopoly* vartotojas turi mąstyti apie savo dalykinės srities ontologijos schemą: kokius temų, asociacijų, pasireiškimų, rolių tipus kurti. Pagal sudarytą modelį užpildoma atitinkamo temos egzemplioriaus informacija: pavadinimai, laukai, ryšiai su kitomis temomis.
- Vizigator: suteikia galimybę analizuoti teminį tinklą grafiškai. Naršant grafiniame lange galima susiaurinti (praplėsti) teminio tinklo vaizdą, įvesti filtravimo kriterijų. Jei užkraunamas teminis tinklas su schema, tai lange matomos ir papildomos „meta lygmens“ temos. Tokiais atvejais teminis tinklas gali tapti nepatogus ir painus, todėl tokį tinklą patariama eksportuoti be schemas, po to užkrauti į *Omnigator* ir tik tada peržiūrėti grafiškai.

Produkto architektūra [21] pateikta 15 pav. *Ontopia* produkte teminiai tinklai gali būti saugomi atmintyje arba reliacinių duomenų bazių sistemoje. Pats varikliukas palaiko *RDF*, *XTM 1.0*, *XTM 2.0*, *LTM*, *CTM*, *OSL Topic Maps* importavimo ir eksportavimo formatus. Bene pažangiausia, tačiau sudėtingiausia produkto dalis – tolog užklausų variklis, leidžiantis ištraukti norimą informaciją iš tinklo.



15 pav. *Ontopia* produkto principinė architektūra

Ontopia priskiriamas teminių tinklų varikliukų (angl. *engine*) kategorijai. Naudojant modulį „*Navigator Framework*“ galima sukurti *Topic Maps* pagrindu veikiančias interneto programas, tačiau šis būdas reikalauja programavimo, gero *tolog* užklausų ir *JSP* technologijos išmanymo.

Kitame skyriuje analizuojamos klasikinių TVS charakteristikos, bei tos turinio valdymo sistemos, kurios bendrauja su teminiais tinklais ir taip palengvina *Topic Maps* portalų kūrimą.

2.2.2 Turinio valdymo sistema (TVS)

Stambios įmonės, įvairiems poreikiams patenkinti internete, naudoja turinio valdymo sistemą. Tokio pobūdžio TVS – sudėtingos, apimančios informaciją iš įvairių žinių šaltinių, todėl reikalauja prasmingo naršymo po turinį. Kiekviena TVS pasižymi šiomis savybėmis:

- TVS suprojektuotos suteikti netechniškiems žmonėms (pvz. rašytojams, redaktoriams ir kt.) įvairiapusę turinio pristatymo kontrolę portale, be jokių specifinių inžinerinių gebėjimų;
- TVS – serveryje patalpintų programų (*Java, PHP* ir t.t.) aibė, pateikianti informaciją per naršyklę
- Leidžia susikoncentruoti į turinį (nesirūpinti papildomu programavimu ir teisėmis);
- Suteikia galimybę kurti, modifikuoti, tvarkyti turinį per internetinę sąsają;
- Įdiegtas teisių mechanizmas įvairiems vartotojams leidžia daryti įtaką turiniui;
- Svarbiausia TVS savybė – turinys atskiriamas nuo pristatymo sluoksnio (pakeitus išvaizdą elementai išlieka);
- Nepaisant daugelio TVS teikiamų galimybių, nėra tinkamo metaduomenų palaikymo, informacijos klasifikavimo, ryšių.

3 lentelėje pateikiamas labiausiai naudojamų turinio valdymo sistemų sąrašas. Šaltinis: *Wikipedia* http://en.wikipedia.org/wiki/List_of_CMSEs

3 lentelė. Populiariausios TVS

Pavadinimas	Platforma	Palaikomos duomenų bazės	Komercinė	<i>Topic Maps</i> palaikymas
Liferay	Java	HSQLDB, MySQL, Oracle, SQL Server, DB2, Apache Derby, Informix, InterBase, JDataStore	Ne	Taip
Nedora	Java	MySQL, Oracle	Ne	Ne
Alfresco	Java	MySQL, Oracle, SQL Server, Ingres	Ne	Ne
Magnolia	Java	MySQL, Oracle, SQL Server, Ingres	Ne	Ne
OpenCMS	Java	HSQL, MySQL, Oracle, SQL Server,	Ne	Ne
Drupal	PHP	MySQL, PostgreSQL, SQLite	Ne	Taip
Joomla	PHP	MySQL	Ne	Taip
PHP-Fusion	PHP	MySQL	Ne	Ne
WordPress	PHP	MySQL	Ne	Taip

Plone	Python/Zope	MySQL, PostgreSQL, Oracle, ZODB per Zope	Ne	Ne
Escenic Content Engine	Java	Oracle, MySQL	Taip	Taip
IBM Lotus Web Content Management	Java	Oracle, SQL Server , DB2	Taip	Ne
SharePoint Server	ASP.Net	SQL Server,SQL Express	Taip	Taip

Pateikta lentelė gali būti pildoma, nes TVS sąrašas informacinių technologijų rinkoje yra ilgas. Darbe siekiama sujungti teminius tinklus su turinio valdymo sistema į vieną derinį, todėl tokio pobūdžio TVS išlieka svarbiausios.

Toliau analizuojama *Liferay* turinio valdymo sistema, pateikiamas šios TVS architektūrinis sprendimas, įvertinimai.

2.2.3 *Liferay* TVS

Turinio valdymo sistema *Liferay* [14] sukurta *Java* technologijomis. Vartotojui suteikta galimybė atsisiųsti nemokamą bendruomenės versiją (angl. *Community Edition*) arba mokamą - įmonės (angl. *Enterprise Edition*) versiją. Abiems versijoms bazinių portletų rinkinys išlieka vienodas. Skirtumas tas, kad įsigijus licenziją suteikiamos papildomos ataskaitų generavimo, testavimo, konsultavimo paslaugos. Už papildomą mokestį *Liferay* bendruomenė organizuoja portalo vystymo, palaikymo, administravimo internetinius seminarus.

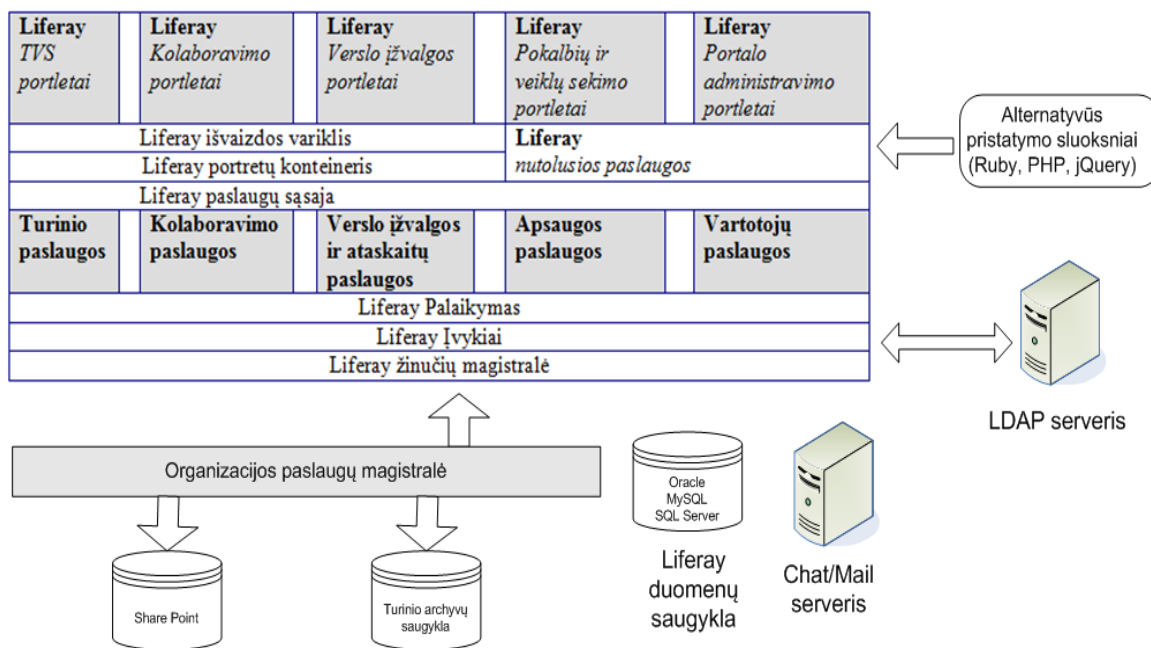
Pagal 2010 metų „*Water & Stone*“ [15] atliktus tyrimus *Liferay* pirmauja, kaip atviro kodo *Java* TVS. Ketvirtoje lentelėje pateikta dešimt svarbiausių tyrime atliktų vertinimo aspektų. Pilną kriterijų sąrašą galima rasti penkioliktame literatūros šaltinyje.

4 lentelė *Liferay* portalo vertinimų suvestinė

Vertinimo kriterijus	Reikšmė	Pozicija tarp <i>Java</i> TVS	Galutinė pozicija
Vidutinis atsisiuntimas per savaitę	9,435	1	6
Naudojimas (įdiegimas)	154 iš 4000 tyrimo dalyvių	1	5
Pritaikymas interneto svetainėse	0,02 %	2	10
Kūrėjų parama, konsultacijos	139, iš jų 59 Guru	1	7
Knygų publikacijos	10, iš jų 1 išleista 2010 metais	2	8
Tyrime dalyvavusių <i>Liferay</i> portalų <i>Google</i> page rank	7	1	3

Prekybinio ženklo populiarumas	77% nesusipažinę 20% yra girdėję 3% susipažinę	3	15
Vartotojų ieškoma TVS pagal <i>Google</i> paieškos duomenis 2009 metų liepos mėnesiui	168 000	1	5
Prekybinio ženklo vertinimas	62,5 % palankus 37,5 % nepalankus iš 2833 respondentų	1	4
Teikiama pirmenybė	118 iš 2833 respondentų	1	5

Liferay populiarumą ir sėkmę lemia ir naudojimosi paprastumas. Architektūra remiasi klasikiniu trijų sluoksnių atvaizdavimu, logikos ir duomenų modeliu. Portalas paliko *MySQL*, *Oracle*, *DB2*, *SQL Server* ir kt. reliacinių duomenų bazių valdymo sistemas. Taikomųjų uždavinių logika grindžiama įvykiais, o žinutės siejais su duomenų saugyklomis per organizacijos paslaugų magistralę. Atvaizdavimas organizuojamas portretais arba kitomis (*jQuery*, *PHP*, *Ruby*) priemonėmis. Šešioliktame paveiksle pavaizduota *Liferay* portalo architektūra [14].



16 pav. *Liferay* architektūra

Apibrėžus norimą naudoti turinio valdymo sistemą, aptiriamos su teminiais tinklais bendraujančios TVS.

2.2.4 Teminiai tinklai ir turinio valymo sistemos

Viena iš įdomiausių ir praktiškiausių teminių tinklų taikymo sričių – turinio valdymas. Kaip minėta darbo pradžioje naudojant *Topic Maps* technologiją su TVS išnaudojamos stipriosios turinio valdymo sistemos savybės ir praplečiamos informacijos organizavimo, ryšių tarp informacijos objektų palaikymo galimybės.

Penktoje lentelėje pateikiamas komercinių turinio valdymo sistemų (išsigyjamos nusipirkus licenziją), su teminių tinklų palaikymu, sąrašas. Šių sistemų detalesnė analizė nepateikiama.

5 Lentelė teminių tinklų palaikymas komercinėse turinio valdymo sistemose

Įrankis	Aprašymas	Nuoroda
<i>Escenic</i>	Komercinis <i>Ontopia</i> ir <i>Escenic</i> turinio valdymo sistemos integravimo variantas	http://www.vizrt.com/online
<i>OfficeNet Knowledge Portal</i>	Uždara Norvegijos turinio valdymo sistema su įdiegtu <i>Ontopia</i> teminių tinklų palaikymu	http://www.officenet.no/lang/no/25
<i>TMCore EPiServer Module</i>	Modulis prideda navigacijos ir paieškos galimybes <i>EpiServer</i> turinio valdymo sistemoje	http://www.networkedplanet.com/Products/TMCore-EPiServer-Module
<i>TMCore SharePoint Module</i>	Modulis integruoja <i>TmCore</i> konceptus į <i>Microsoft SharePoint</i> portalą	http://www.networkedplanet.com/Products/TMCore-SharePoint-Module)

Egzistuoja ir nekomercinių TVS su įdiegtu teminių tinklų palaikymu. Daugelis tokių sistemų yra neišbaigtos, mažai taikomos ir nepopuliarios. Visgi kai kuriuose sprendimuose palaikomos naudingos funkcijos, kurios detalčiau aprašomos šeštoje lentelėje.

6 Lentelė teminių tinklų palaikymas nekomercinėse turinio valdymo sistemose

Įrankis	Kriterijus					Nuoroda
	Naudoja <i>Ontopia</i>	Teminio tinklo formato palaikymas	Užklausų taikymas teminiam tinklui	Naudoja XSLT transformacijas		
<i>Joomla ir Ontopia integracija</i>	Taip, <i>TMRAP</i> paslauga	Taip, <i>XTM</i>	Taip, <i>tolog</i> užklausos	Taip	http://sewebar.vse.cz/	
<i>WordPress Topic Maps (WP2TM)</i>	Ne	Taip, <i>XTM</i>	Ne	Ne	http://wordpress.org/extended/plugins/wordpress-topic-maps-wp2tm/	
<i>Drupal TopicMaps</i>	Ne	Ne	Ne	Ne	http://drupal.org/project/topicmap	
<i>ZTM</i>	Ne	Taip, <i>XTM</i>	Ne	Ne	http://www.ztmproject.org	
<i>Ontopia ir Liferay integracija</i>	Taip	Taip (<i>XTM</i> , <i>LTM</i> ir kt.)	Taip, <i>tolog</i> užklausos	Ne	http://code.google.com/p/ontopia/wiki/HowToBuildLiferayIntegration	

Joomla ir *Ontopia* integracijos sprendime naudojamas *Ontopia Topic Maps* varikliukas, padedantis suformuoti specialiaame redaktoriuje *tolog* užklaudas, jas transformuoti *XSLT* pagalba į HTML fragmentus *Joomla* turinio valdymo sistemoje. Šis sprendimas - speciali biblioteka, užpildanti spragą tarp *PHP TVS* ir *Java*.

WordPress Topic Maps (WP2TM) – turinio valdymo sistemos *WordPress* įskiepis, suteikiantis galimybę susieti informaciją asociacijomis internetiniame tinklaraštyje (galima generuoti *XTM* tipo failus). Sprendimas yra bandymų stadijoje ir vis dar vystomas.

Drupal Topic Maps - *Drupal TVS* įskiepis (modulis), kuriame leidžiama jungti informacinius objektus (temas) ryšiais. Šiai „mini“ sistemai keliami įvairūs reikalavimai (*Drupal 6* versija ir didesnė), kad įskiepis normaliai funkcionuotų. Modulis mažai populiarus ir vis dar plėtojamas.

ZTM (angl. *Zope Topic Maps*) – teminių tinklų pagrindu veikianti *TVS*, kurioje turinys kuriamas kaip *Topic Maps* temos, o susiejamas asociacijų pagalba. Sistemos funkcijos: naršymas po teminį tinklą, jo redagavimas, publikavimas, *XTM* formato palaikymas. Visgi, *ZTM* nėra iki galo išbaigta, o įdiegimas – sudėtingas.

Ontopia ir *Liferay* integracija sujungia lanksčias informacijos specifikavimo ir naudingas turinio tvarkymo bei pateikimo galimybes į vieną efektyvų derinį.

Teminių tinklų taikymas turinio valdymo sistemoje ženkliai palengvina *Topic Maps* pagrindu veikiančių portalų kūrimą, nes esami atskiri įrankiai reikalauja įvairių papildomų programavimo funkcijų iki norimo funkcionalumo.

2.3 Panašių sistemų Lietuvoje ir tarptautiniu mastu analizė

Pateikiamos *Topic Maps* standartą naudojančios sistemos užsienyje ir Lietuvoje. Analizuojami tų sistemų technologiniai sprendimai, pateikiamos iliustracijos.

2.3.1 Italų operos teminis tinklas

Demonstracijai *Ontopia* sukūrė Italų Operos <http://www.ontopia.net/operamap/index.jsp> *Topic Map* (17 pav). Visas turinys saugomas teminiame tinkle. Ši sistema turi operų indeksą, kuris suskirsto jas pagal abėcėlę. Šalia operos pavadinimo yra solistas, apie kurį galima sužinoti papildomą informaciją. Svetainėje galima atlikti paiešką. Italų operos sistemoje egzistuoja teatrų, kompozitorių, regionų ir šalių indeksai. Puslapio navigacija atliekama *Topic Maps* pagalba [4].

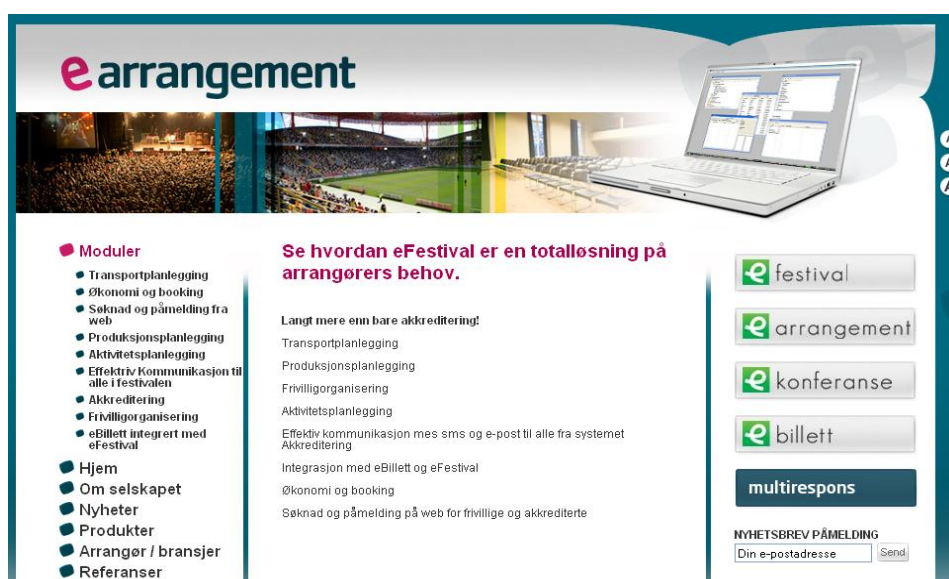
Iš visų sistemoje naudojami 7319 teminio tinklo elementų (2012 temų, 3635 asociacijų bei 1672 pasireiškimų).



17 pav. Italų operos *Topic Map*

2.3.2 Norvegijos kultūrinių įvykių apžvalgos internetinė svetainė

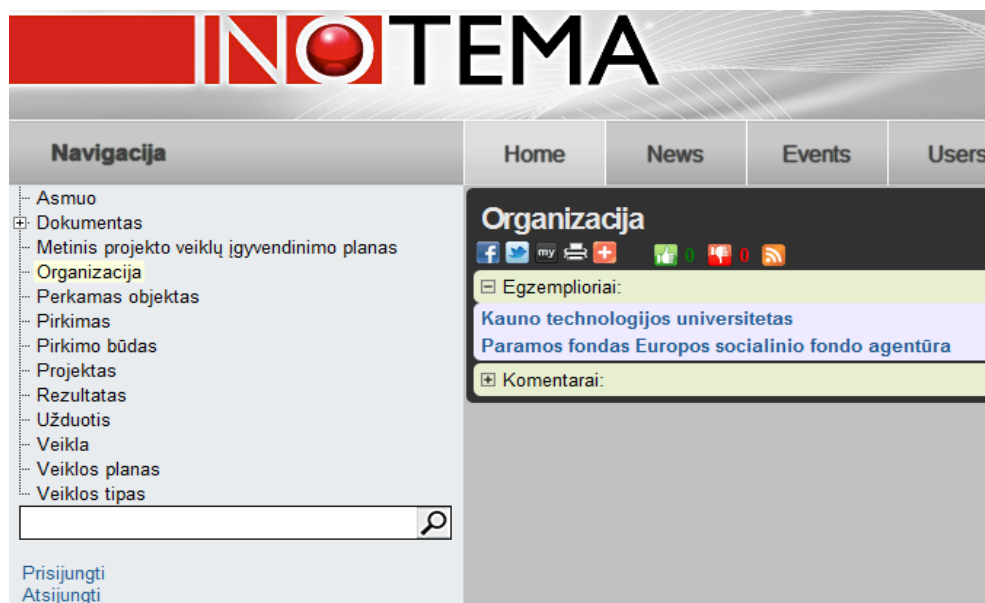
Topic Maps pagrindu sukurta sistema, apibūdinanti Norvegijos kultūrinius įvykius pagal pasirinktą festivalį, programą (18 pav) [6].



18 pav. Norvegijos kultūrinių įvykių *Topic Map*

2.3.3 Projektas „Inotema“

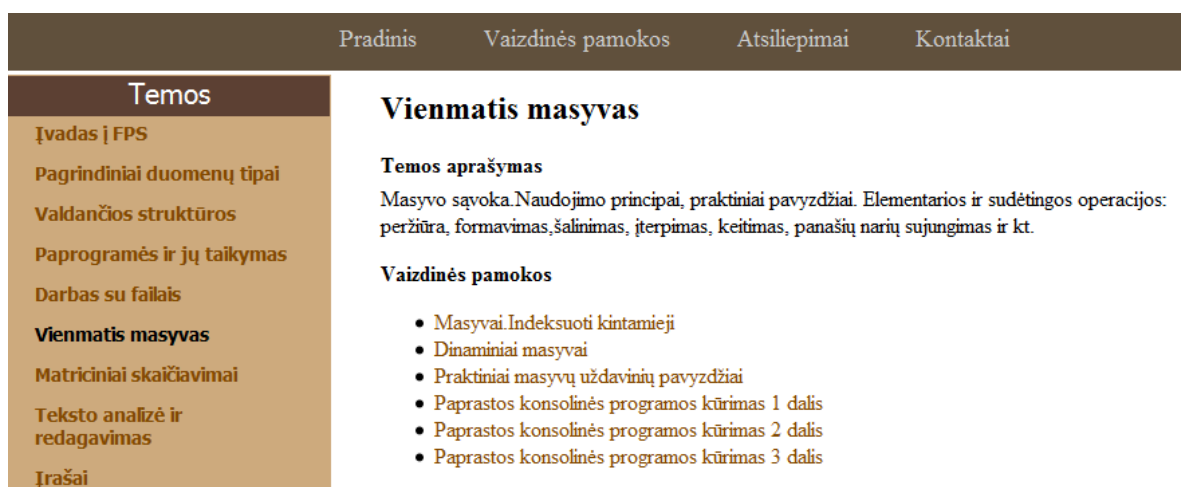
Sistemoje (19 paveikslas) kuriamas *Topic Maps* portalas. Naudojama *Plone* turinio valdymo sistema, prie kurios prilipdyti *ZTM* moduliai. „Inotema“ (<http://88.119.92.15:9000/Inotema>) portale vartotojas, prisijungęs prie sistemos, gali užkrauti norimo formato *Topic Map*. Tada po tokį tinklą galima naršyti, peržiūrėti informaciją grafiškai, atlikti paiešką. Projekte ruošiamasi įdiegti *Topic Maps* redaktorių, sluoksniavimo mechanizmą.



19 pav. Projektas „Inotema“

2.3.4 Teminiai tinklai programavimo vaizdinių pamokų sistemoje

Interneto svetainė [22] veikia *Topic Maps* pagrindu. Sistema realizuota aprašius ontologiją *XTM 1.0* tipo faile, kuri transformuojama *XSLT* pagalba.



20 pav. Teminius tinklus naudojanti programavimo mokymo svetainė

Pasaulyje teminių tinklų taikymas internetinėse sistemose plinta, daugiausiai Skandinavijoje, Vokietijoje. Jungtinėse Amerikos Valstijose puikus *Topic Maps* naudojimo pavyzdys – US Tax Map (<http://taxmap.ntis.gov/taxmap/>). Lietuvoje tokių portalų skaičiuojama vienetais. Pagrindinė to priežastis – *Topic Maps* technologijos naujoviškumas ir sudėtingumas. Kitame skyriuje analizuojama, koks būtų Lietuvoje įdiegto *Topic Maps* portal siekiamas sprendimas ir privalumai.

2.4 Siūlomas sprendimas

Pateikus teminių tinklų, TVS bei panašaus pobūdžio internetinių sistemų analizę, darbe siekiama pateikti klasikinės turinio valdymo sistemos modernizavimo metodiką, kuri iliustruotų praktinį teminių tinklų taikymą lietuviškuose portaluose. Kuriama sistema turėtų pasižymėti šiomis savybėmis:

- Patogi navigacija po TVS informacijos objektus, greitai surandant reikiamą informacijos vienetą ir jo išteklius;
- Atitinkamų TVS informacinių objektų susiejimas pagal aprašytos teminio tinklo ontologijos asociacijas;
- Pagrindinių teminio tinklo elementų (temų, asociacijų, pasireiškimų) panaudojimas sistemoje;
- Grafinė sąsaja, kurios pagalba tvarkoma ir atvaizduojama informacija, administruojami sistemos vartotojai ir pan.

2.5 Darbo tikslas ir uždaviniai

Magistriniame darbe keliamas tikslas: suteikti TVS kūrėjams ir taikomųjų programų vystytojams daugiau praktinių žinių apie teminių tinklų naudojimą turinio valdymo sistemose ir taip populiarinti tokias sistemas Lietuvoje.

Tikslams įgyvendinti formuluojami uždaviniai:

1. Aprašyti pasirinkto teminio tinklo varikliuko ir TVS integravimo reikalavimus metodikai ir kuriamai demonstracinei sistemai;
2. Suprojektuoti pasirinkto teminių tinklų varikliuko ir TVS integravimo modelį;
3. Sukurti programavimo mokymo dalykinės srities teminį tinklą ir iliustruoti pasirinktos ontologijos naudojimą turinio valymo sistemoje konkrečiu pavyzdžiu;
4. Eksperimentiškai ištirti kuriamos programavimo mokymo demonstracinės sistemos apimtį ir sudėtingumą;
5. Eksperimentiškai ištirti rengiamos metodikos kokybę, atsižvelgiant į metodikai ir demonstracinei sistemai iškeltus funkcinis ir nefunkcinis reikalavimus.

2.6 Analizės išvados

Analizėje aptarta teminių tinklų paradigma, vaidmuo turinio valdyje, kūrimo įrankiai. Suformuluotos išvados:

1. Išanalizavus teminių tinklų technologiją, apibendrinama, kad ši technologija nėra triviali, tačiau pasiteisina, kai norima pristatyti sudėtingą žinių modelį, valdyti didelius informacijos kiekius iš įvairių šaltinių.
2. Atliktus teminių tinklų literatūros analizę nustatyta, kad internetinių sistemų kūrimui naudoti vien tik teminius tinklus galima, tačiau tokių programų kūrimas tampa sudėtingesnis, sunkiau palaikomas, nes egzistuojančių *Topic Maps* įrankių (*Ontopia*, *RubyTM* ir kt.) įsisavinimas reikalauja pastangų ir laiko.
3. Išanalizavus panašaus pobūdžio esamas *Topic Maps* sistemas, prieita prie išvados, kad tiesiausias kelias į tokių portalų kūrimą – naudoti turinio valdymo sistemą su teminių tinklų palaikymu. Šių nekomercinių sprendimų egzistuoja nedaug, o iš analizuotų variantų palankiausia naudoti norvegų *Java* priemonėmis sukurtą teminių tinklų valdymo produktą *Ontopia* su *Liferay* portalu, nes šis sprendimas sujungia *Topic Maps* teikiamas lanksčias informacijos organizavimo ir naudingas TVS turinio formatavimo bei pateikimo galimybes į vieną efektyvų derinį.
4. *Topic Maps* (kaip ir *RDF*) paradigma galėtų būti priskiriama prie ateities technologijų, prisidedančių formuojant *Semantic Web 3.0*. Dabartinis pasaulinis tinklas (*www*) sugeba aprūpinti mus įvairiais dokumentais, surasti norimą puslapį. Visgi norint sukurti tobulesnį internetą, reikia remtis semantika, kuri padėtų kompiuteriams suvokti internetinio puslapio informaciją ir susieti ją prasmingais ryšiais.
5. Išnagrinėjus teminius tinklus ir jų taikymo sritis nuspręsta parengti TVS modernizavimo metodiką, kurią iliustruotų lietuviška demonstracinė lietuviška sistema, nes tokio pobūdžio sistemų Lietuvoje beveik nėra. Reikalingi praktiniai pavyzdžiai, padėsiantys populiarinti teminių tinklų taikymą.

3 Reikalavimų specifikacija

Skyriuje pateikiami reikalavimai kuriamai metodikai bei sistemai. Išskiriamos šios reikalavimų specifikacijos dalys:

- Reikalavimai *Liferay* turinio valdymo sistemos modernizavimo metodikai;
- Reikalavimai ruošiamai kurti programavimo mokymo demonstracinei sistemai, įvertinant sistemos sudėtingumą ir apimtį;
- Funkciniai reikalavimai, nurodant kompiuterizuojamų panaudos atvejų (PA) modelį;
- Reikalavimai teminio tinklo duomenims, išskiriant pradinės dalykinės srities modelį;
- Nefunkciniai reikalavimai, kuriuos turi tenkinti metodika, akcentuojant *Topic Maps* standartui keliamus reikalavimus.

3.1 Reikalavimai TVS modernizavimo metodikai

Kuriamos metodikos tikslas – išaiškinti, kaip sujungti *Topic Maps* technologiją su TVS, naudojant teminių tinklų varikliuką *Ontopia* bei *Liferay* turinio valdymo sistemą. Metodikos rezultatas – *Topic Maps* pagrindu veikiantis portalas. Tokio tipo vartotojas susigaudys, kaip naršyti po teminį tinklą, sekant siūlomomis nuorodomis puslapyje.

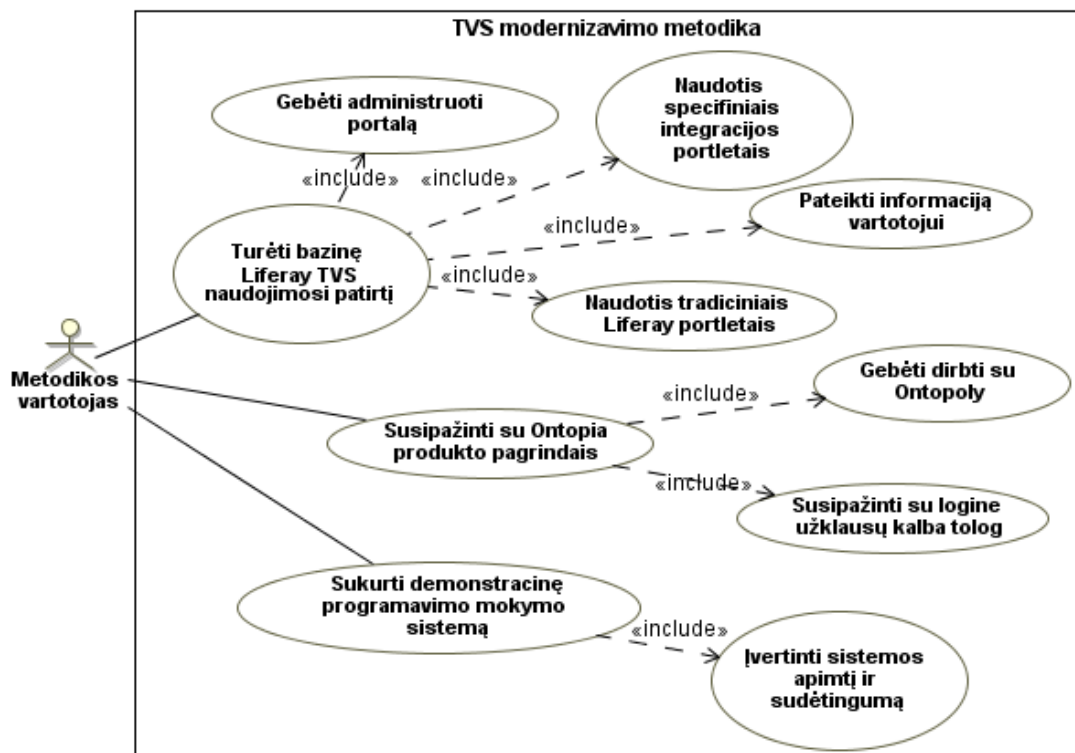
20.1 paveiksle pateikti TVS modernizavimo metodikos reikalavimai. Metodikos vartotojas turi būti susipažinęs su *Ontopia* ir *Liferay* konfigūravimo gidu, turėti bazines šių produktų, loginės užklausų kalbos *tolog* taikymo žinias.

Naudojant *Ontopia* vartotojas turi gebėti kurti teminį tinklą *Ontopoly* redaktoriumi, atlikti būtinus pakeitimus: pridėti (pakeisti) temų, asociacijų, pasireiškimų, vaidmenų tipus, tinkamai parinkti asociacijoje dalyvaujančias temas. *Omnigator* leidžia patikrinti pasirinktų temų charakteristikas, pamatyti daugiau specifinės temų informacijos (*objectid*), kuri gali būti naudinga programuojant *tolog* užklausas. Logine užklausų kalba *tolog* metodikos vartotojas turi gebėti išrinkti tinklo temų tipus, temos egzempliorius, pasireiškimus ir kt.

Liferay pagalba vartotojas turi gebėti kurti norimos struktūros puslapius portale, išdėliojant reikiamus portletus[‡] grafinės sąsajos pagalba, administruoti portalą.

Itin svarbus „Sukurti demonstracinę programavimo mokymo sistemą“ metodinis reikalavimas, kuris nurodo, kad siūloma metodika turi praktinį taikymą. Išpildant šį reikalavimą turi būti atsižvelgta į kuriamos sistemos sudėtingumą ir apimtį (įvertinamos teminio tinklo galimybės, naudojami elementų tipai).

[‡] Portale vartotojui pritaikyti servisai (paieška, meniu, grafikai ir t.t.) pateikiami per taikomuosius uždavinius



20.1 pav. TVS modernizavimo metodikos reikalavimai

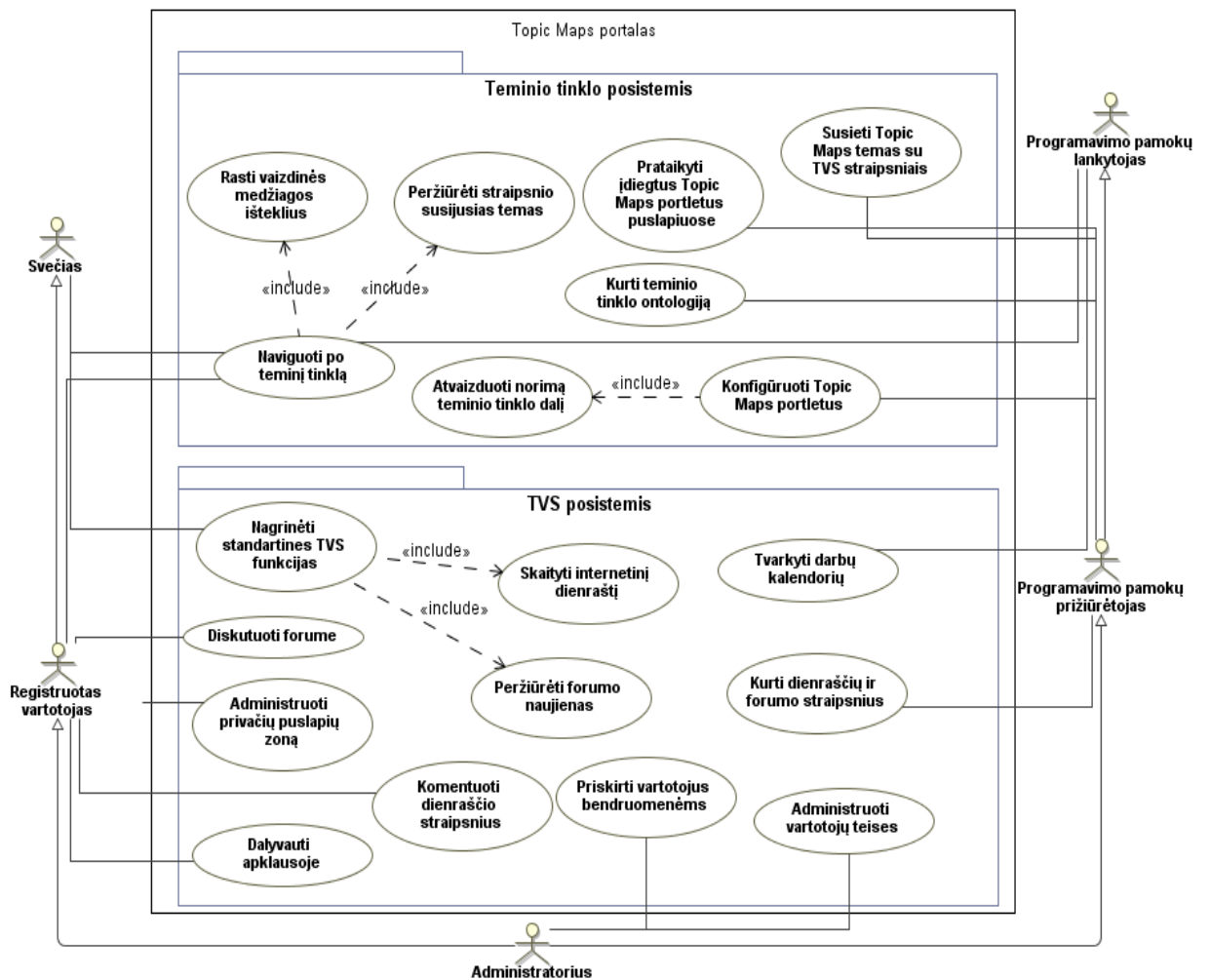
Demonstracinei sistemai keliami reikalavimai, naudojami vartotojų tipai aprašomi 3.2 skyriuje. Nefunkcinių reikalavimų skyriuje (3.3) detaliau pateikiami sistemos sudėtingumui ir apimčiai keliami reikalavimai.

3.2 Reikalavimai programavimo mokymo demonstracinei sistemai

Metodikos taikymui skyriuje pateikiami sistemos (*Topic Maps* portalo) reikalavimai. Pasirinkta dalykinė sritis – vaizdinėmis pamokomis grindžiamas programavimo mokymas.

3.2.1 Kompiuterizuojami panaudos atvejai

Topic Maps portale visi vartotojų panaudojimo atvejai patalpinami į turinio valdymo sistemos ir teminių tinklų posistemius. Išskirti penki vartotojų tipai gali atlikti skirtingas funkcijas pagal nustatymus apribojimus. Pavyzdžiui „Svečiui“ suteikta galimybę naršyti po teminį tinklą, peržiūrėti susijusius temos straipsnius, rasti vaizdinę medžiagą ir t.t. Registruotas vartotojas diskutuoja forume, komentuoja dienraščio straipsnius, dalyvauja apklausoje bei naudojasi „Svečio“ privilegijom, nes modelyje pavaizduota apibendrinimo rodyklė.



20.2 pav. *Topic Maps* portalo kompiuterizuojamų panaudos atvejų modelis

TVS posistemio panaudojimo atvejai (diskutuoti forume, skaityti internetinį dienraštį, dalyvauti apklausoje, tvarkyti darbų kalendorių, kurti dienoraščių ir forumo straipsnius, prižiūrėti forumo naujienas, administruoti vartotojų teises, priskirti vartotojus bendruomenėms) būdingi daugeliui klasikinių turinio valdymo sistemų.

Tuo tarpu teminio tinklo posistemio panaudojimo atvejai – išskirtiniai. Šio posistemio vartotojai privalo turėti specifinių *Topic Maps* žinių konfigūruojant portletus. Kuriant teminio tinklo ontologiją rekomenduojama išmanyti *Topic Maps* duomenų modelį [12], jį sudarančių elementų naudojimą. Norint susieti *Topic Maps* temas su *Liferay* TVS informaciniais objektais nepakanka vien tik publikuoti straipsnių į serverį, bet ir reikia sukurti straipsnių struktūras su šablonais. Atvaizduojant norimą teminio tinklo dalį reikia gebėti pritaikyti loginę užklausų kalbą *tolog* portletams.

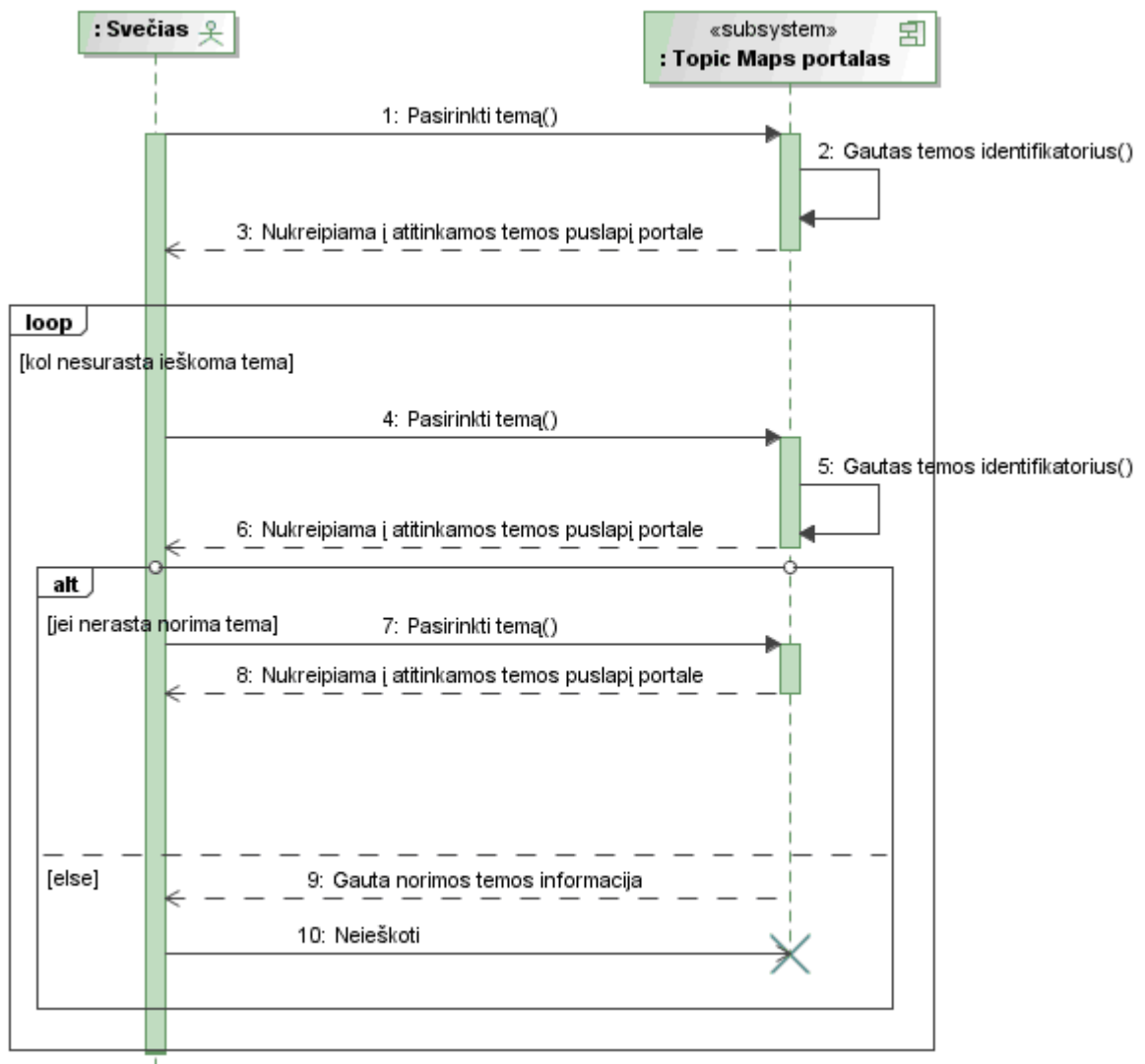
Toliau darbe specifikuojami „Naviguoti po teminį tinklą“ (7 lentelė) ir „Konfigūruoti *Topic Maps* portletus“ (8 lentelė) panaudojimo atvejai, pateikiamos jų sekų diagramos.

7 Lentelė PA „Naviguoti po teminį tinklą“ specifikacija

Panaudojimo atvejis		Naviguoti po teminį tinklą
Tikslas		Pasinaudoti <i>Topic Maps</i> teikiama navigacija portale.
Aprašymas		Portalo lankytojas norėdamas sparčiau surasti norimą informaciją pasirenka galimybe naršyti po turinį teminiu tinklu.
Prieš sąlyga		Turi būti tinkamai sukonfigūruoti <i>Topic Maps</i> portletai bei išdėlioti puslapiuose.
Aktorius		Bet kuris iš penkių portalo vartotojo tipų.
Sistema		<i>Topic Maps</i> portalas
Sužadinimo sąlyga		Prieš pradėdant navigaciją po tinklą, portalo lankytojas turi pasirinkti temą iš pateikto sąrašo.
Veiklos taisyklės		Pradiniame puslapyje pasirenktas temos tipas (angl. <i>topic type</i>) arba egzempliorius (angl. <i>instance</i>) nurodo iš kurio taško pradėta navigacija po tinklą.
Susiję panaudojimo atvejai	Išplečia PA	
	Apima PA	Rasti vaizdinės medžiagos išteklius Peržiūrėti straipsnio susijusias temas
	Specializuoja PA	-
Pagrindinis įvykių srautas		Sistemos reakcija ir sprendimai
1. Portalo lankytojas pasirenka temos tipą ar egzempliorių iš pateikto stulpelio (-ių)		1.1 Nukreipia į atitinkamos temos puslapį portale, perduodant temos identifikatorių kaip parametą per URL. Rodomas temos vardas (-ai), turinys, asociacijos su kitomis temomis.
Po sąlyga		Surasta ieškomos temos informacija ir sąsajos su kitomis temomis
Alternatyvos (nesėkmės atvejai)		1.1. .a. Vartotojas nerado ieškomos temos, todėl iš einamojo puslapio (per duotąsias asociacijas) ieško tol, kol suranda reikiamą temą. 1.1. .b. Užmegzta semantiškai neteisinga sąsaja tarp <i>Topic Maps</i> ontologijos temos ir <i>Liferay</i> turinio. 1.1. .c. Neuždėtas atitinkamas šablonas <i>Topic Maps</i> temai ir <i>Liferay</i> portalas nesugeba nukreipti vartotoją prie atitinkamos temos informacijos.
Specialūs (nefunkciniai) reikalavimai		-
Pastabos		-

Panaudojimo atvejo „Naviguoti po teminį tinklą“ sekų diagrama pateikta 21 paveiksle. Ilustruota sąsaja tarp PA aktoriaus „Svečias“ ir *Topic Maps* portalo posistemio.

Iš pradžių svečias pasirenka temą. Pagal tai *Topic Maps* posistemis nustato temos identifikatorių ir gražina vartotojui tą puslapį, kurio *URL* parametruose perduota parinkta tema. Gali būti, kad portalo svečias nerado ieškomos temos iškart, todėl apibrėžiamas ciklas, kuris vykdomas tol, kol surandama norima tema. Tokiu atveju svečias nebenaršo toliau po teminį, o analizuoja rastos temos informaciją.



21 pav. PA „Naviguoti po teminį tinklą“ sekų diagrama

PA „Konfigūruoti *Topic Maps* portletus“ specifikacija pateikiama 8 lentelėje, o sekų diagrama – 22 paveiksle.

8 Lentelė PA „Konfigūruoti *Topic Maps* portletus“ specifikacija

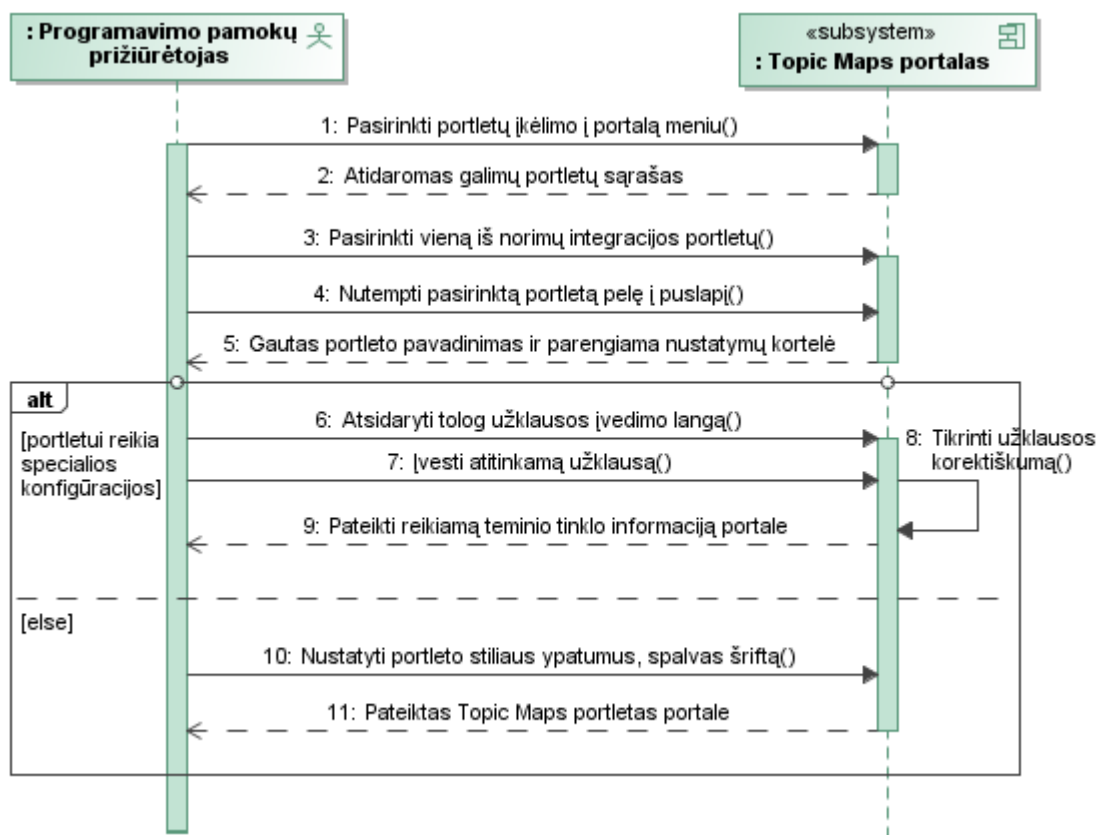
Panaudojimo atvejis	Konfigūruoti <i>Topic Maps</i> portletus
Tikslas	Atvaizduoti reikiamą teminio tinklo informaciją portale.
Aprašymas	Programavimo pamokų prižiūrėtojas pasirenka teminių tinklų informacijai atvaizduoti skirtą portretą, parenka vietą puslapyje. Tada konfigūracijos meniu specifikuoja <i>tolog</i> užklausą. Išsaugojant pakeitimus sistema praneša, ar užklausa teisingai suformuota, kokie duomenys atvaizduoti.
Prieš sąlyga	Portalo vartotojas privalo turėti „Administratoriaus“ arba „Programavimo Pamokų Prižiūrėtojo“ statusą, kad galėtų nustatyti reikiamą konfigūraciją integracijos portletams.

Aktorius	Administratorius, Programavimo pamokų prižiūrėtojas.	
Sistema	<i>Topic Maps</i> portalas	
Sužadinimo sąlyga	Prieš atliekant konfigūracija reikia prisijungti prie valdymo skydo bei pateikti portretą puslapyje.	
Veiklos taisyklės	Jei pasirinkus portletą, jo konfigūracijoje nerandama, kur specifikuoti tolog užklausa, tai tokiam portretui užklausa nereikalinga.	
Susiję panaudojimo atvejai	Išplečia PA	
	Apima PA	Atvaizduoti norimą teminio tinklo dalį
	Specializuoja PA	-
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai	
1. Pasirenkama meniu, iš kurios galima įkelti norimus portletus.	1.1 Atidaro portletų sąrašą.	
2. Pasirenkamas portletas iš sąrašo.		
3. Portletas pele nutempiamas (angl. drag) į norimą puslapio vietą.	3.1 Fiksuoja, koks portletas naudojamas bei paruošia nustatymų kortelę.	
4. Atidaromas <i>tolog</i> užklauso įvedimo langas.		
5. Surenkamas <i>tolog</i> užklauso tekstas įvedimo Lange.	5.1 Tikrinama, ar vartotojo įvesta užklausa korektiška. 5.2 Atvaizduoja reikiamą teminio tinklo informaciją portale	
Po sąlyga	Vartotojui pateikiama teminio tinklo informacija pvz.: duotos temos pasireiškimai, sąsajos su kitomis temomis ir t.t	
Alternatyvos (nesėkmės atvejai)	1.1. a. Portalo konfigūracija atlikta nesėkmingai ir sistema nepateikia galimo portretų sąrašo. 5.1. a. <i>Tolog</i> užklauso tekstas neteisingas ir sistema siunčia klaidos pranešimą. 5.1. b. <i>Tolog</i> užklausa gražina semantiškai neteisingus rezultatus. 5.1. c. <i>Tolog</i> užklausa negražina rezultatų, o vartotojas ekrane mato tuščia portletą.	
Specialūs (nefunkciniai) reikalavimai	<i>Tolog</i> sakniai formuojami, pagal <i>tolog 1.2</i> specifikaciją.	
Pastabos	-	

Panaudojimo atvejo „Konfigūruoti *Topic Maps* portletus“ sekų diagrama pateikta 22 paveiksle. Iliustruota sąsaja tarp PA aktorius „Programavimo pamokų prižiūrėtojas“ ir *Topic Maps* portalo posistemio.

Iš pradžių pasirenkama meniu, kuriame sistema pateikia integracijos portletus. Tada „Programavimo pamokų prižiūrėtojas“ pasirenka norimą portletą ir nustato jo reikiamą vietą puslapyje. Čia sistema identifikuoja portletą ir pagal tai parenka galimus nustatymus. Svarbiausia panaudojimo atvejo stadija – *tolog* užklauso specifikuojimas, nes čia gali būti pridaryta daugiausiai klaidų, o ir ne visada atvaizduojama reikiama teminio tinklo informacija.

Visgi, kai parašoma užklausa bei išsaugomi pakeitimai, portletas pateikia norimą informaciją bei galima parinkti papildomus nustatymus (stilių, spalvas, šriftą).



22 pav. PA „Konfigūruoti *Topic Maps* portletus“ sekų diagrama

Kiti PA, aprašyti kompiuterizuojamų panaudojimo atvejų modelyje, čia neaptariami, nes jų specifikavimas analogiškas.

3.2.2 Reikalavimai sistemos vartotojams

Programavimo mokymo sistemoje kuriami vartotojai, galintys peržiūrėti ir komentuoti tinklo informaciją bei tokie, kurie gali konfigūruoti norimą *Topic Maps* fragmentą portale. Portale numatomi šie penki vartotojų tipai:

9 Lentelė *Topic Maps* portalo vartotojų tipai

Vartotojo tipas	Aprašymas
Administratorius	Vartotojo vaidmuo, leidžiantis tvarkyti visą portalo veiklą: prieiti prie valdymo skydo, administruoti vartotojus, jų roles, bendruomenes ir kita. Administratorius gali naudoti sukurtus portalo portletus ar įdiegti naujus.
Programavimo Pamokų Prižiūrėtojas	Vartotojo rolė, atsakinga už programavimo pamokų bendruomenės veiklą. Šiam vartotojo tipui suteiktos visos reikiamos teisės bendruomenės kontekste: vartotojų įtraukimas į bendruomenę, teisių nustatymas, privačių ir viešų puslapių kontrolė. Programavimo pamokų

	prižiūrėtojas gali kurti portalo bendruomenės elementus(forumą, apklausas, internetinius dienoraščius, straipsnius), nustatyti jų konfigūraciją.
Programavimo pamokų lankytojas	Autentifikuotas vartotojas, priskirtas programavimo pamokų bendruomenei. Šios rolės vartotojas gali prieiti prie jam sukonfigūruoto valdymo skydo (nusistatyti savo viešus, privačius puslapius, paveikslėlių galeriją, turinį). Taip pat lankytojui suteiktos teisės dalyvauti apklausoje, peržiūrėti rezultatus, įkelti internetinio dienoraščio, naujienų straipsnius, atlikti jų paiešką, komentuoti, vertinti. Kita svarbi veikla - forumas, kuriame programavimo pamokų lankytojas gali kurti <i>sub</i> kategorijas, naujus straipsnius, rašyti žinutes, atsakymus. Vartotojui suteikta privačių puslapių erdvė leidžia peržiūrėti aktualijas, tvarkyti savo darbų kalendorių.
Registruotas portalo vartotojas	Gali naršyti po portalo puslapius, peržiūrėti forumo diskusijas, žinutes, internetinio dienoraščio straipsnius, tvarkyti savo privačius, viešus puslapius, talpinti apribotą skaičių portletų.
Svečias	Svečiui suteiktos teisės naršyti po portalo puslapius, peržiūrėti forumo diskusijas, vaizdines pamokas, žinutes, internetinio dienoraščio straipsnius, registruotis.

Devintos lentelės aprašyme pateiktos vartotojų atliekamos funkcijos yra kompiuterizuojamos. Dalis jų kompiuterizuojama išnaudojant standartines turinio valdymo sistemos savybes, kita dalis *Topic Maps* posistemyje. Kitame skyriuje analizuojami kuriamos sistemos nefunkciniai reikalavimai.

3.3 Nefunkciniai reikalavimai ir apribojimai

Išnagrinėjus reikalavimus metodikai bei funkcinius reikalavimus demonstracinei programavimo mokymo sistemai šiame skyriuje pateikiami nefunkciniai reikalavimai. Pirmoje dalyje įvertinami sistemos apimčiai ir sudėtingumui keliai reikalavimai, kurie yra svarbūs rengiant tinkamą metodiką. Antroje dalyje aprašomi *Topic Maps* technologijos reikalavimai: naudojamas standartas, kuriamo tinklo formatas, varikliukas, saugojimo duomenų bazė bei reikalavimai, kuriamos sistemos išvaizdai, saugumui.

3.3.1 Reikalavimai sistemos apimčiai ir sudėtingumui

Sistemos reikalavimai apimčiai ir sudėtingumui vertinami šiais aspektais:

- Praplėstos Liferay TVS galimybių naudojimas: sujungus *Ontopia Topic Maps* varikliuką su *Liferay TVS* atsiranda nauji taikomieji uždaviniai, kuriuos vartotojas turi išbandyti. Tai specifinių *Topic Maps* portletų taikymas *Liferay* portale. Kuo daugiau portletų panaudota, tuo sistema yra sudėtingesnė;

- Teminio tinklo ontologijos pilnumas: demonstracinės programavimo mokymo sistemos teminis tinklas užpildytas turiniu (nėra temų tipų be įrašų). Turi būti panaudoti visi būtiniausi *Ontopoly* redaktoriaus palaikomi teminio tinklo elementai (temų, asociacijų, vaidmenų, pasireiškimų tipai, temų egzemplioriai);
- Tradicinių Liferay TVS portletų naudojimas: pritaikyti standartiniai (forumas, internetiniai dienoraščiai, „wiki“, polls, recent activities, calendar ir kt.) *Liferay* TVS portletai sistemoje. Turi būti bent 2 atskirai nuo turinio navigacijos veikiančys puslapiai pagrindiniame meniu;
- Liferay portalo administravimas: turi būti sukurta bendruomenė kurioje publikuojamas turinys, tvarkomos vartotojų teisės. *Liferay* turinio elementams pritaikytos struktūros ir šablonai;
- Sisteminis informacijos pateikimas vartotojui: turi būti uždėtas sistemos logotipas, parinktos puslapių savybės (turinys kairėje, o susijusios temos dešinėje ir pan.), portletams nustatoma tinkama išvaizda.

Taigi sistema yra tuo sudėtingesnė ir didesnės apimties, kuo labiau išpildomi penki skyriuje išvardinti punktai.

3.3.2 Kiti nefunkciniai reikalavimai

Išskiriami šie nefunkciniai reikalavimai kuriamai metodikai:

- Topic Maps technologija: teminiai tinklai kuriami pagal ISO/IEC 13250 reikalavimus, laikantis bazinio teminių tinklų duomenų modelio ISO/IEC JTC1/SC34 standarto [24];
- Topic Maps duomenų formatas: palaikomi *XTM 1.0*, *XTM 2.0*, *XTM 2.1*, *LTM*, *CTM*, *OSL*;
- Topic Maps varikliukas: iš programos kodo (angl. *source*) suinstaliuota *Ontopia* (≥ 5.1) distribucija;
- Teminio tinklo saugojimas: reikalinga reliacinė duomenų bazių valdymo sistema *MySQL*. Duomenų bazė gal didėti tiek, kiek leidžia *MySQL* server konfigūracijos nustatymai;
- Teminio tinklo ontologija: tinklo temų tipai (angl. *topic type*) kuriami, naudojant *Liferay* struktūras ir šablonus, o temų egzemplioriai – *Liferay* turinio atvaizdavimo elementus. Asociacijų, rolių pasireiškimų tipai kuriami ir palaikomi *Ontopoly* redaktoriumi;
- Operacinė sistema: rekomenduojamos *Linux* bei *Mac OS*, tačiau galima pritaikyti ir *Windows*;
- Turinio valdymo sistema: *Liferay 5.2* ar vėlesnė, komplektuota su *Apache Tomcat* serveriu. Taip pat reikalingas *Liferay-SDK* įskiepis;

- Kita programinė įranga: *Java 1.5* ar naujesnė, Apache Ant (≥ 1.8);
- Puslapių struktūra: Išskirtos antraštės *header* ir *footer*, kurių sritis nekinta nepriklausomai nuo atvėro puslapio. Pagrindinė sritis – skaidoma pagal poreikių į kelis stulpelius, kuriuose informacija keičiasi dinamiškai;
- Išvaizda: standartinė, neišsiskirianti kontrastingomis spalvomis bei neišryškinanti tų vietų, kuriose informacija atvaizduojama iš teminio tinklo;
- Saugumas: TVS ir ypač *Topic Maps* atvaizduojamą informaciją gali tvarkyti tik atitinkamą statusą turintys vartotojai.

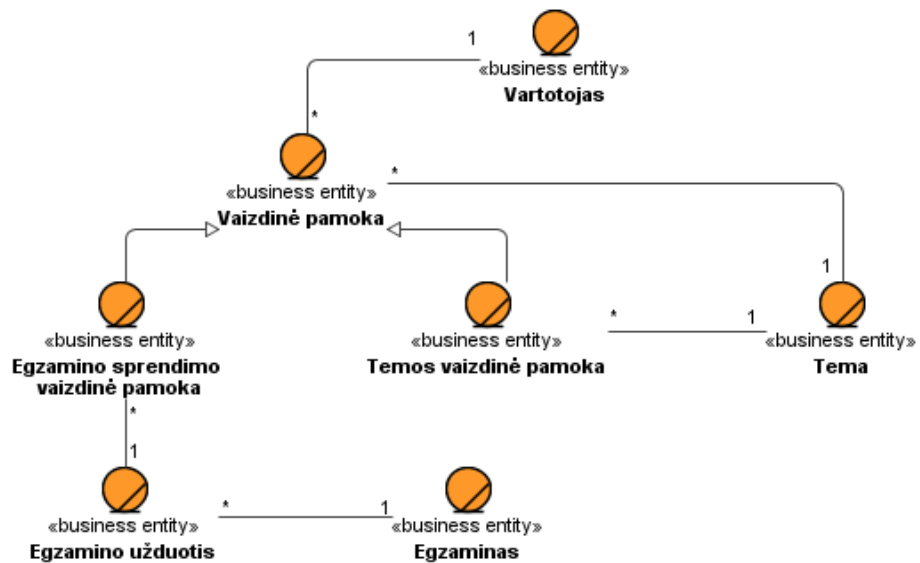
3.4 Reikalavimai teminio tinklo duomenims

Turinio valdymo sistema informaciniams objektams saugoti ir funkcionuoti turi įdiegtą duomenų bazę su nustatytais reikalavimais, o teminio tinklo duomenų reikalavimus reikia apsibrėžti pačiam. Toliau analizuojama, kokie reikalavimai keliami programavimo mokymo teminio tinklo ontologijai.

Teminiame tinkle esybes atitinka temų tipai (angl. *topic type*), kurie diagramoje pavaizduoti kaip veiklos esybės (angl. *business entity*).

23 pav. pateiktas veiklos konceptų pradinis modelis. Dalykinė sritis apima vaizdinėmis pamokomis grindžiamą programavimo mokymą, susidedantį iš temų ir egzaminų užduočių bei jų sprendimų pateikimo. Nurodyti kardinalumai parodo, kad vartotojas gali įrašyti daug vaizdinių pamokų, egzaminas turi kelias užduotis, o egzamino užduotis – daug egzamino vaizdinių pamokų sprendimų.

Projektuojant realų teminį tinklą, nurodoma detalesnė informacija: jungiamų esybių asociacijos tipas, pavadinimai, rolės. Šis veiksnys suteikia modeliui daugiau semantikos, o taip pat realizacijoje atvaizduojama tai, kas suprojektuota.



23 pav. Veiklos esybių (pradinis dalykinės srities) modelis

3.5 Reikalavimų apibendrinimas

Reikalavimų specifikacijos etape išryškėjo, kad svarbiausias reikalavimas – sukurti metodiką grindžiančią sistemą. Reikalavimai suskirstyti į tuos, kurie keliami metodikai ir kuriamam portalui. Vartotojams, norintiems išbandyti metodiką reikalaujama bazinė pažintis su *Topic Maps* standartu, *Ontopia* ir *Liferay* produktais.

Sistemos reikalavimai padalinti į funkcinis ir nefunkcinis. Funkciniai reikalavimai iliustruoti kompiuterizuojamų panaudos atvejų diagrama bei kelių PA specifikacijomis. Sekų diagramos atskleidė, kokius elementus projekte naudoja atitinkami panaudojimo atvejai. Nefunkciniai reikalavimai pabrėžia, kad reikia laikytis *Topic Maps ISO/IEC 13250* standarto, naudoti *Ontopia* varikliuką, saugoti tinklą *MySQL* duomenų bazėje. Teminio tinklo ontologiją kurti ir pildyti *Liferay* bei *Ontopoly* įrankiais.

4 *Ontopia* ir *Liferay* integracijos projektas

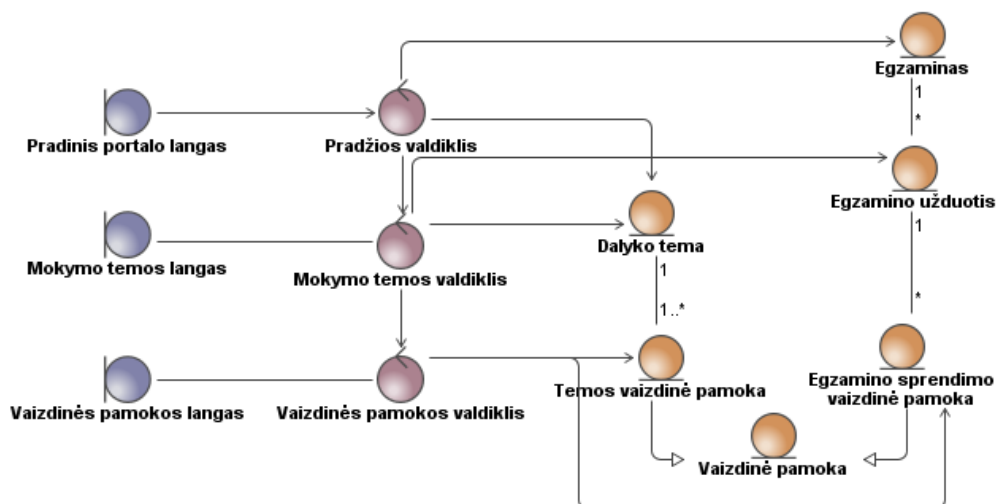
Šiame darbo etape detaliau projektuojama ruošiamą kurti sistema, analizuojant reikalavimus robastiškumo diagramomis, atskleidžiant loginės architektūros posistemų ypatumus. Didelę reikšmę teikia *Liferay* ir *Ontopia* integracijos posistemio projektavimas, padedantis vartotojui greičiau apčiuopti nagrinėjamos metodikos principus. Vartotojo interfeiso modelis nurodo galimą teminio tinklo naršymą, o sistemos elgsenos modelis identifikuoja vaizdinės medžiagos išteklių radimą tinkle.

Skyriaus pabaigoje pateikiama metodikoje naudojama teminio tinklo duomenų bazės schema.

4.1 Reikalavimų analizė

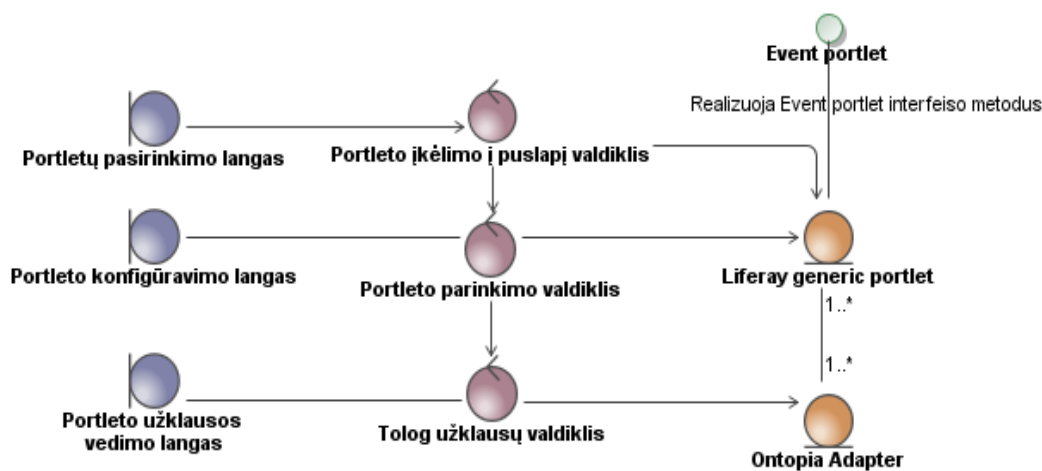
Skyriuje pateikiama specifikuotų panaudojimo atvejų „Naviguoti po teminį tinklą“ ir „Konfigūruoti *Topic Maps* portletus“ analizė robastiškumo diagramos. Šio tipo diagramos nurodo, kokios klasės realizuoja panaudojimo atvejį.

24 paveikslas iliustruoja PA „Naviguoti po teminį tinklą“. Analizės logika tokia: pradiniam lange atvaizduotus duomenis kontroliuoja ir naudoja „Dalyko temos“ ir „Egzamino“ klasės pradžios valdiklis. Todėl vartotojas pradiniam puslapyje gali pasiekti tik pastarųjų esybių informaciją. Dalyko tema siejasi su temos vaizdine pamoka, o egzaminas su egzamino užduotimi. Į mokymo temos langą patenkama iš pradinio puslapio. Čia mokymo temos valdiklis ima duomenis iš „Dalyko tema“ ir „Egzamino užduotis“ esybių, o pasirinkus susijusią temą valdiklis nukreipia į vaizdinės pamokos tipo puslapį. Vaizdinės pamokos puslapis – detaliausias teminio tinklo lygmuo, o vaizdinės pamokos valdiklis siejamas su „Temos vaizdinė pamoka“ ir „Egzamino sprendimo vaizdinė pamoka“ esybėmis.



24 pav. PA „Naviguoti po teminį tinklą“ robastiškumo diagrama

25 paveikslas iliustruoja PA „Konfigūruoti *Topic Maps* portletus“. PA išpildymui naudojami portletų pasirinkimo, konfigūravimo ir užklausų langai, kuriuos apdoroja atitinkami valdikliai. *Liferay* turinio valdymo sistema valdyma portletų mechanizmu, todėl bet koks kuriamas portalo portretas paveldi „Liferay generic portlet“ klasę (atsakinga už „Event portlet“ interfeiso metodų realizaciją). *Tolog* užklausų pertransliavimą ir rezultatų pateikimą portale kontroliuoja *tolog* užklausų valdiklis, naudojantis teminių tinklų varikliuko specialią klasę „*Ontopia Adapter*“. Ši klasė vaidina viena iš svarbiausių vaidmenų, siekiant sujungti teminius tinklus su *Liferay* turinio valdymo sistema.



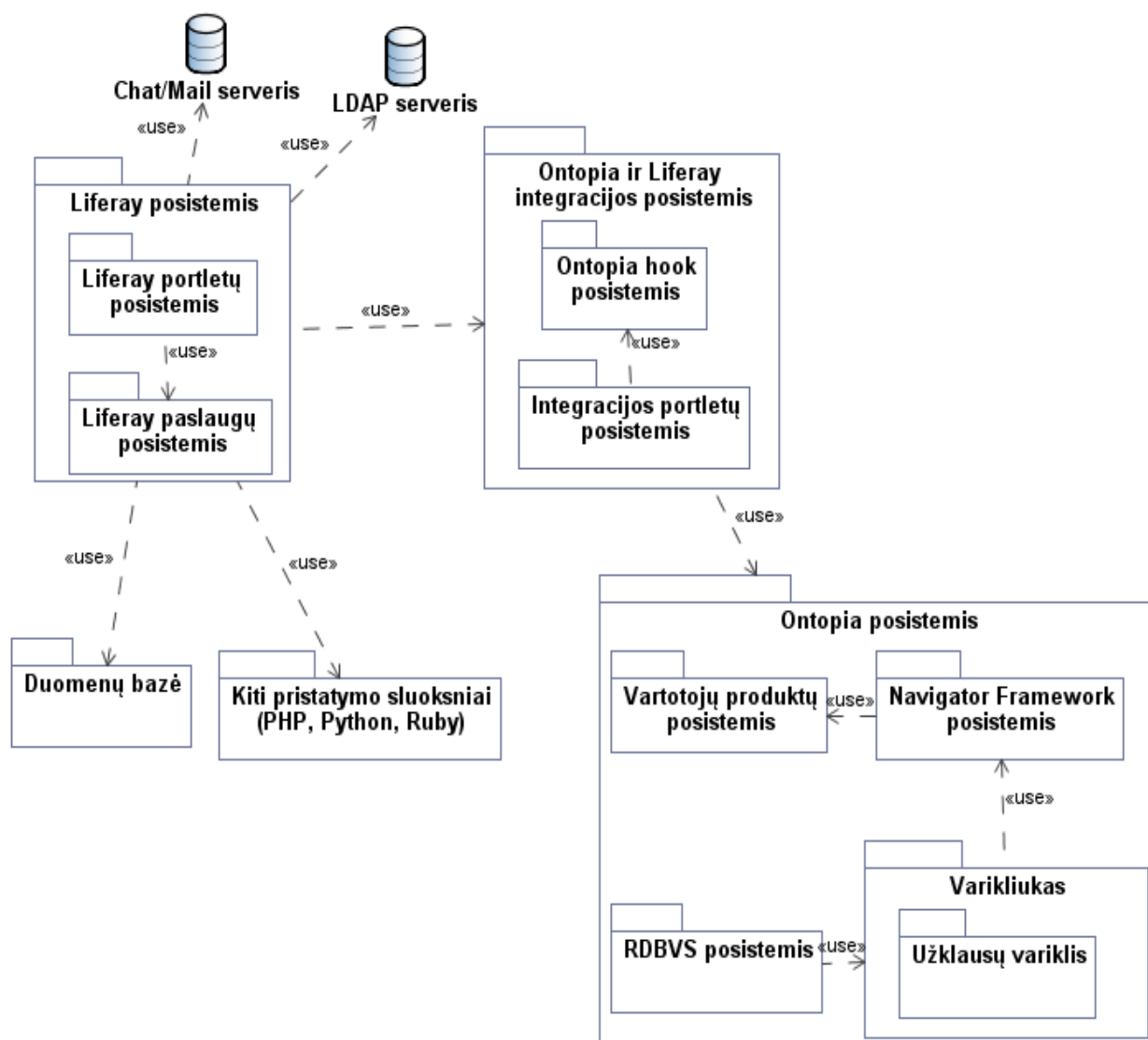
25 pav. PA „Konfigūruoti *Topic Maps* portletus“ robastiškumo diagrama

Išanalizavus ruošiamos kurti sistemos reikalavimus, 4.2 skyriuje nagrinėjama loginė sistemos architektūra.

4.2 Sistemos loginė architektūra

Kuriama metodika apima dvi dideles sistemas: *Liferay TVS* ir *Ontopia*. Abu šie produktai turi šimtus klasių, interfeisų, o metodų skaičius dar gausnis. Visgi sistemose išskiriami tie pagrindiniai posistemiai, kurie atlieka svarbiausius vaidmenis *Liferay* ir *Ontopia* integracijos metodikoje.

26 paveiksle pavaizduota aukščiausio lygio sistemos loginė architektūra. Išskirti trys pagrindiniai posistemiai: *Liferay*, *Ontopia* ir *Liferay* integracijos bei *Ontopia*. *Liferay* posistemis naudoja integracijoje kuriamą „*Ontopia hook*“, kuris padeda įsiskverbti į *Liferay* įvykių sąsają ir vėliau atlikti būtinus metodikos veiksmus. Integracijos posistemis pasiima reikalingus modulius ir metodus iš *Ontopia* - tai suteikia galimybę metodikoje naudoti *tolog* užklausų variklį, RDBVS saugojimo mechanizmą ir kt.



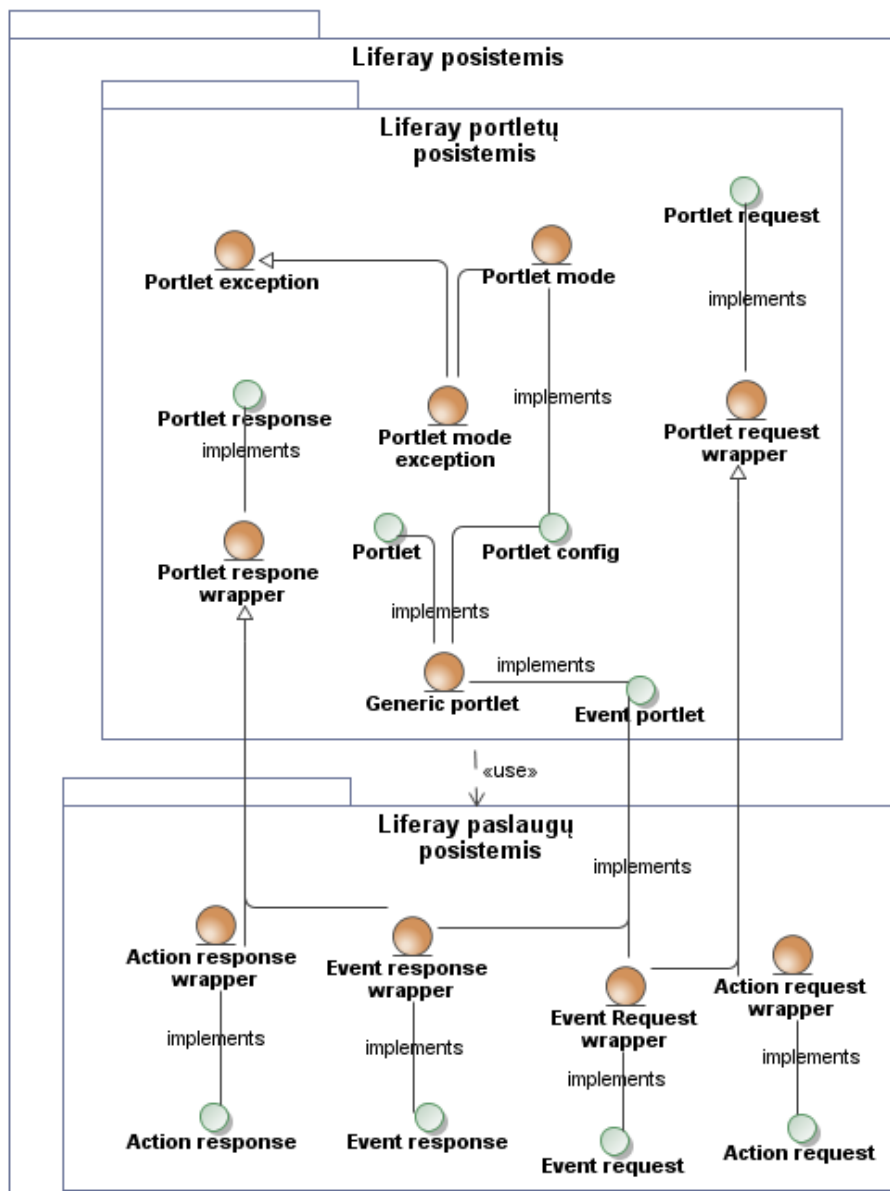
26 pav. Aukščiausio lygio sistemos loginė architektūra

Liferay TVS naudoja portletų programavimo interfeisą [27], kuriam priskiriama aibė klasių. Pagrindinės klasės aptariamos 4.2.1 skyriuje.

4.2.1 *Liferay* posistemis

Liferay sistema valdoma įvykiais, saugomais paslaugų posistemyje. Portletai pagal nustatytus įvykius reaguoja į tai, kaip keičiasi atvaizduojamų taikomųjų uždavinių informacija.

27 paveiksle pavaizduotas *Liferay* posistemio klasių ir jų interfesų poabis. Mažesniais skrituliais pavaizduoti interfeisai, o didesniais – klasės. Asociacija „implements“ nurodo, kokius metodus turi realizuoti siejama klasė.



27 pav. Liferay posistemio klasės ir jų interfeisai

Toliau dešimtoje lentelėje aptariamos svarbiausio *Liferay* posistemio klasės, pagrindiniai veiksmi.

10 Lentelė *Liferay* posistemio klasės

Klasės pavadinimas	Paskirtis
<i>GenericPortlet</i>	Klasė – bendrinio pobūdžio šablonas bet kuriam portletui. Apima inicializavimo, vykdymo, vaizdavimo veiksmus. Parenka portleto pavadinimą. Pagalbinės funkcijos (angl. <i>helper</i> functions) nusako, koku režimu (vykdymo, peržiūros, redagavimo) analizuojamas portletas. Realizuoja <i>Portlet</i> , <i>Portlet Config</i> , <i>EventPortlet</i> interfeisus.
<i>Portlet Response Wrapper</i>	Klasė atlieka tokias funkcijas: prideda, nustato portletui tam tikrą savybę, užkoduoja naudojimo URL adresą. Realizuoja <i>portlet response</i> interfeisą.

<i>Portlet Request Wrapper</i>	Klasė atlieka tokias funkcijas kaip portleto atributų nustatymą, autentifikavimo režimą, naudojamų parametru sąrašą ir reikšmes, serverio portą ir t.t Realizuoja <i>portlet request</i> interfeisą.
<i>Portlet Mode</i>	Reprezentuoja galimus portleto veikimo režimus. Relizuoja <i>portlet config</i> interfeisą.
<i>Portlet Mode Exception</i>	Ši klasė tampa aktyvi, kaip portretas stengiasi panaudoti tokį veikimo režimą, kuris yra neįmanomas.
<i>Portlet Exception</i>	Klasė – bendrinio pobūdžio šablonas bet kuriam portletui. Tampa aktyvus, kai portletas negali atlikti tam tikros operacijos.
<i>Event Response Wrapper</i>	Klasė fiksuoja <i>Liferay</i> sistemos įvykių atsakymus. Relizuoja <i>event response</i> interfeisą.
<i>Event Request Wrapper</i>	Klasė fiksuoja <i>Liferay</i> sistemos įvykių prašymus. Relizuoja <i>event request</i> interfeisą.
<i>Action Response Wrapper</i>	Klasė fiksuoja <i>Liferay</i> sistemos veiksmų atsakymus. Relizuoja <i>action response</i> interfeisą.
<i>Action Request Wrapper</i>	Klasė fiksuoja <i>Liferay</i> sistemos įvykių prašymus. Relizuoja <i>action request</i> interfeisą.

Nustačius *Liferay* posistemio pagrindinius elementus, 4.2.2 skyriuje detalizuojamas svarbiausias – *Ontopia* ir *Liferay* metodikos projektavimo posistemis.

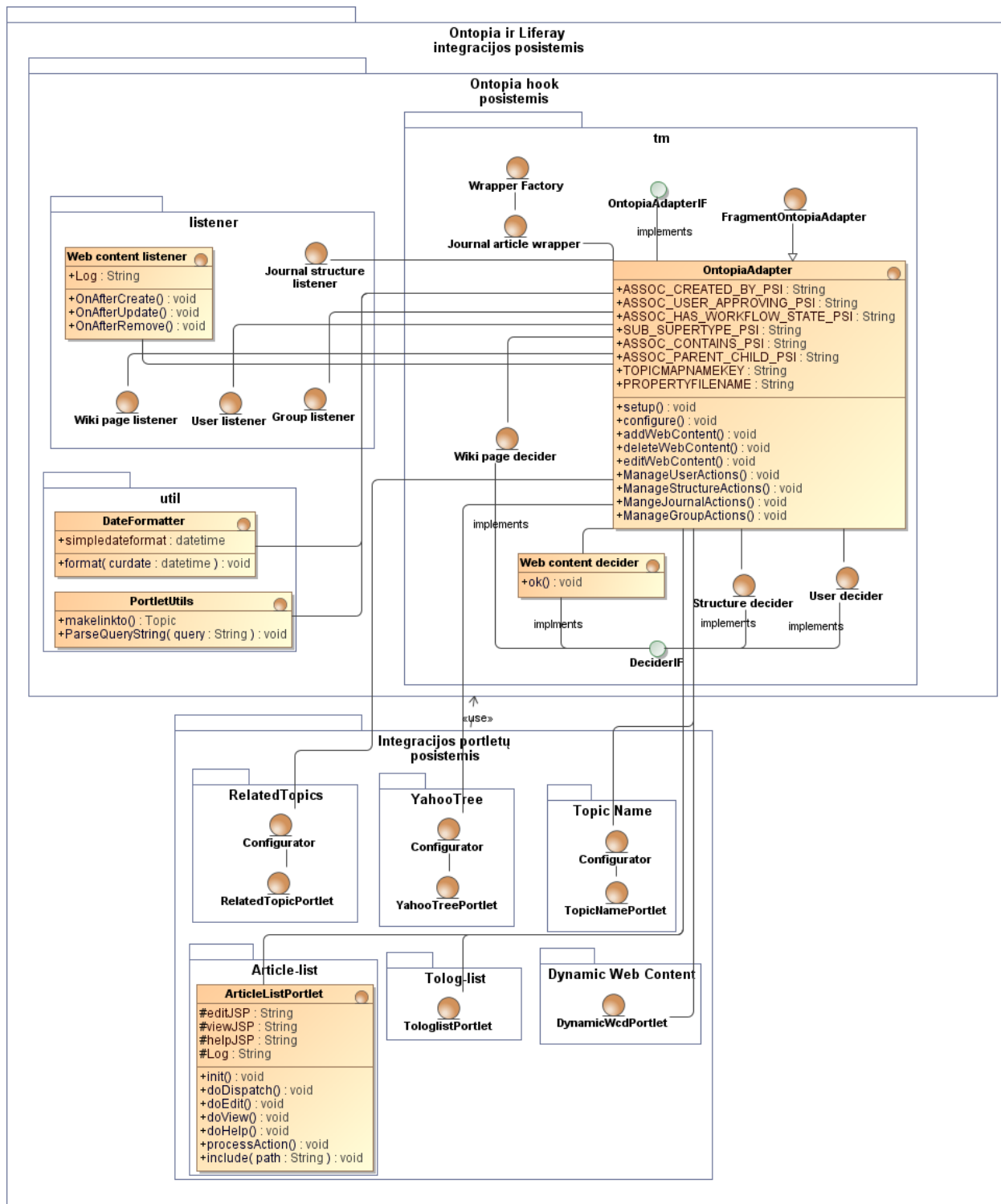
4.2.2 *Ontopia* ir *Liferay* integracijos posistemis

Skyriuje detalizuojami *Ontopia-hook* bei integracijos portretų posistemiai, kurie yra tarpininkai, pasiimti informaciją iš *Ontopia* ir perduodant ją *Liferay* portalui, integracijos metodikoje.

Ontopia-hook sudarytas iš *tm*, *listener*, *util* posistemių. *tm* posistemyje nusprendžiama, kaip atitinkami *Liferay* elementai (*wiki* puslapiai, žurnalai, vartotojai, turinio elementai) yra susiejami su *Topic Maps* temomis. Itin svarbu vaidmenį čia atlieka *OntopiaAdapter*, kuri nulemia ir informacinių objektų modifikavimo veiksmus. Šios klasės atributai ir operacijos specifikuotos 28 paveiksle.

OntopiaAdapter klasė fiksuoja *Liferay* įvykius, kurie registruojami *listener* posistemyje. Šis posistemis geba nustatyti, kada atlikti įterpimo, trynimo, koregavimo veiksmai *Web content*, *Journal structure*, *Wiki page*, *User*, *Group* *Liferay* informaciniams objektams.

util posistemis atsakingas už *tolog* užklausų teisingą sintaksinį išnagrinėjimą, nuorodų parinkimą asociacijoje jungiamoms temoms bei datos tipo laukų formatavimo.

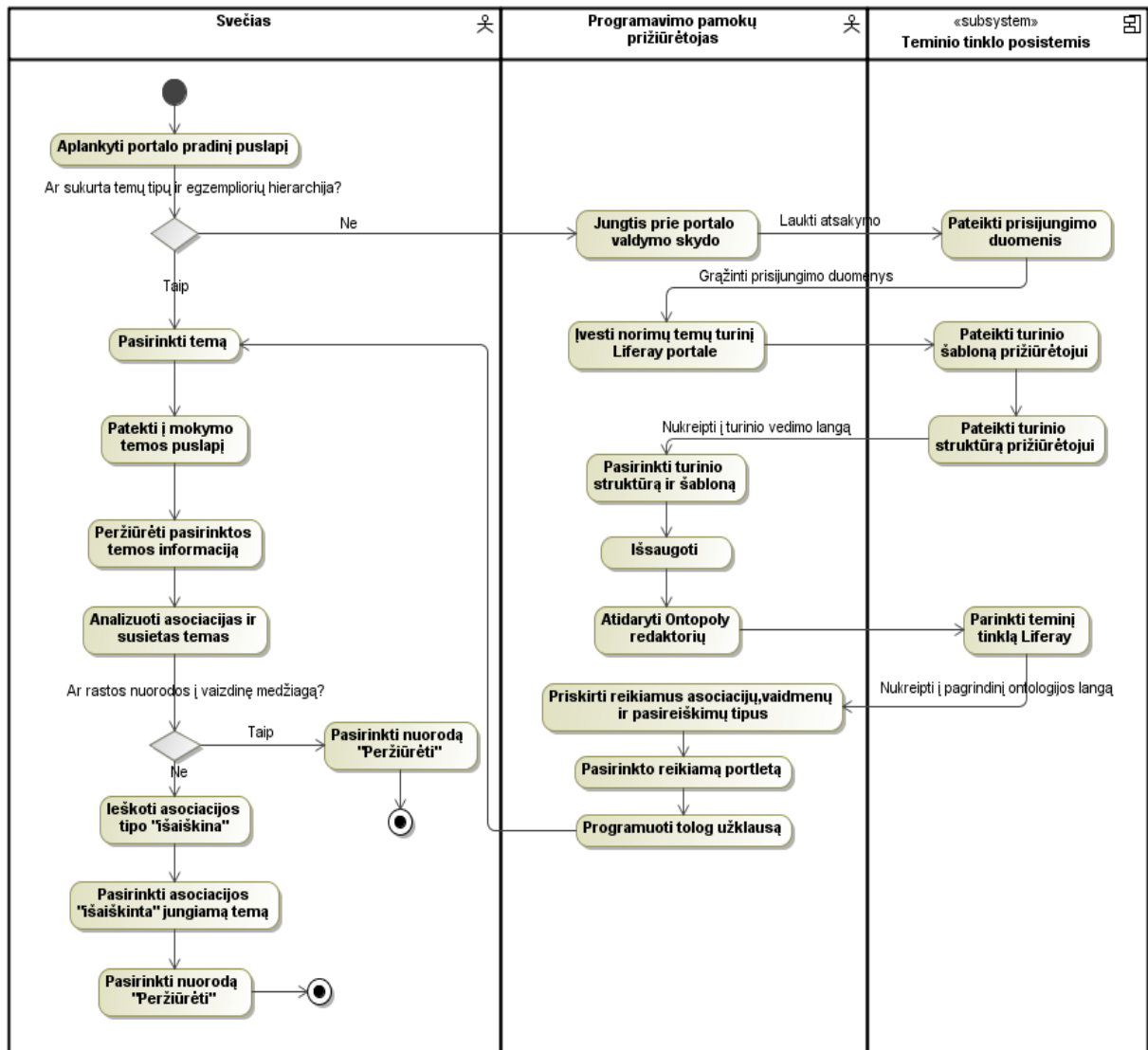


28 pav. *Ontopia ir Liferay* integracijos posistemis

Integracijos portletu posistemis skaidomas į *RelatedTopic*, *YahooTree*, *TopicName*, *Article-list*, *Tolog-list*, *Dynamic Web Content* posistemius, kurie metodikoje tampa *Topic Maps* atvaizdavimo portletais *Liferay TVS*. Posistemiai gali būti sudaryti iš įvairių klasių. 28 paveiksle pateiktas *ArticleListPortlet* klasės atributų ir operacijų sąrašas. Svarbu pabrėžti, kad dėl *Liferay* naudojamų portletų interfeiso, kitų integracijos portletu posistemio klasių atributų ir metodų

4.4 Sistemos elgsenos modelis

30 paveiksle pateiktas sistemos elgsenos modelis iliustruoja vaizdinės medžiagos radimo procesą. Portalo lankytojas ieško *Topic Maps* temų, prie kurių prikabinti specialaus tipo pasireiškimai. Greičiausias kelias iki tokių temų išteklių pradiniam puslapyje pasirinkus temą ir paskui mokymo temos puslapyje surasti asociacijos „išaiškinta“ jungiamą temą bei pasirinkti nuorodą.



30 pav. Portalo vaizdinės medžiagos radimo elgsenos modelis

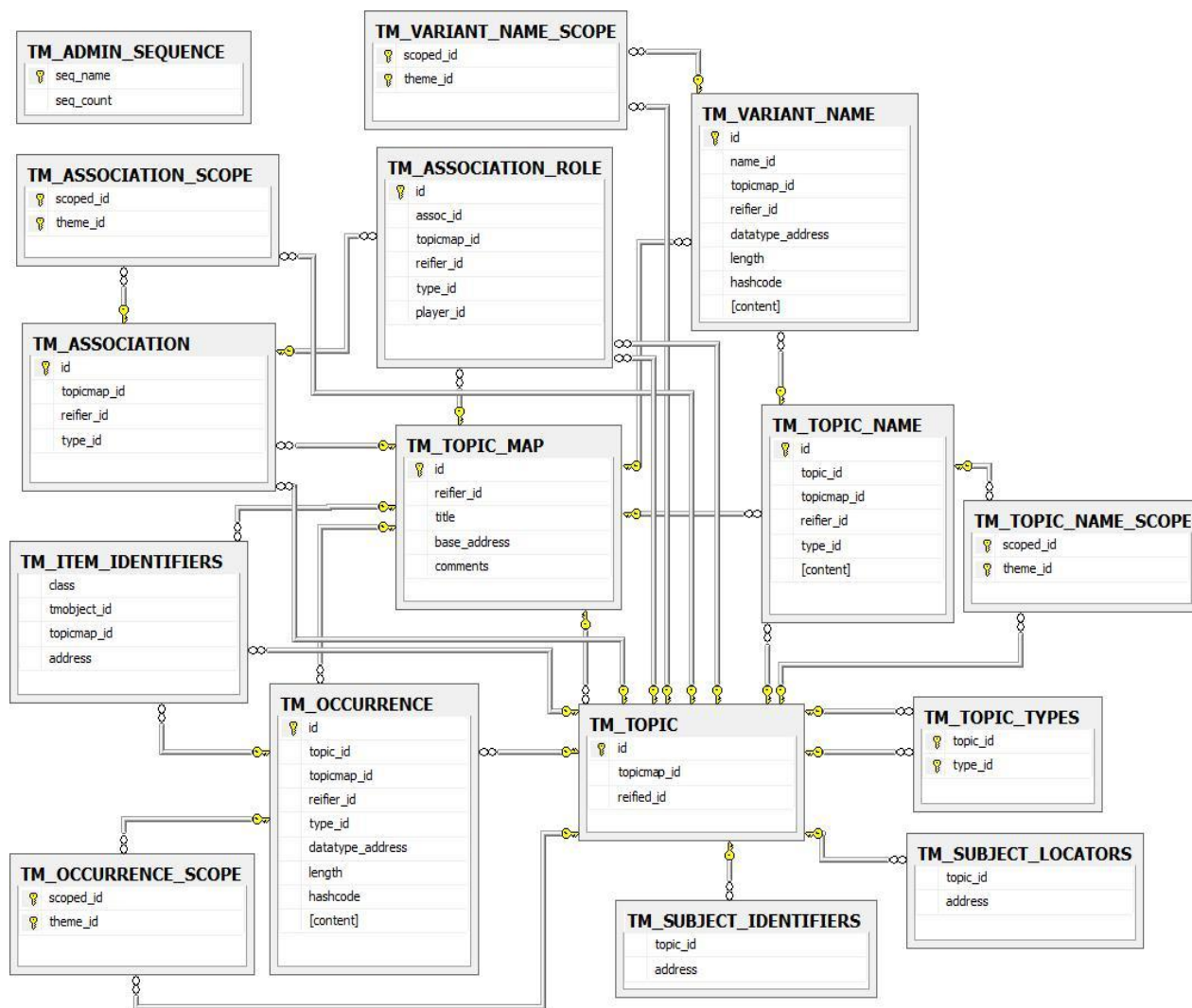
4.5 Duomenų bazės schema teminiam tinklui saugoti

Liferay portalas yra išties didelių galimybių sistema, turinti įvairių *Java* technologijomis realizuotų klasių. Kai duomenų bei informacinių objektų yra daug, jų saugojimui reikalinga duomenų bazė. *Liferay* tokiems dalykams turi sukomplektuotą duomenų bazę, kuri pakankamai sudėtinga ir šiame skyriuje nenagrinėjama.

Darbo metodikai itin aktualus teminių tinklų saugojimas. *Ontopia Topic Maps* varikliukas pagal nutylėjimą teminius tinklus saugo failuose. Kai tinklas išsiplečia, toks saugojimo būdas netinkamas, o *Liferay* sistemoje tai reikštų, kad kiekvieną kartą perkrovus serverį reikėtų iš naujo kurti teminio tinklo ontologiją.

Taigi šiame skyriuje analizuojama, kokios struktūros teminį tinklą galima saugoti reliacinėje duomenų bazėje. Detali instrukcija, kaip prijungti *MySQL* duomenų bazę prie *Liferay* portalo pateikiama *Ontopia* ir *Liferay* integracijos metodikos realizacijoje.

31 paveiksle pateiktas loginis *Topic Maps* modelis. Ši duomenų bazės schema pasiūlyta *Ontopia* produkto ir sugeneruojama įvykdant *.sql* faile nurodytus tekstus pasirinktai duomenų bazių valdymo sistemai. Bazė pildoma įrašais, sukūrus jungtį su atitinkamu teminiu tinklu *Ontopoly* redaktoriuje, kuriame įterpiamos, tvarkomos, šalinamos temos.



31 pav. Teminio tinklo loginis duomenų modelis

Loginis teminių tinklų modelis puikiai nurodo, kokie elementai sudaro *Topic Maps* standartą. 11 lentelė specifikuoja kiekvieną iš duomenų bazės lentelių (aprašant pavadinimą, raktus,

paskirti). Lentelių stulpelių skaičius nurodytas 31 paveiksle: lentelė turi tiek stulpelių, kiek aptinkama teksto etikečių žemiau lentelės pavadinimo.

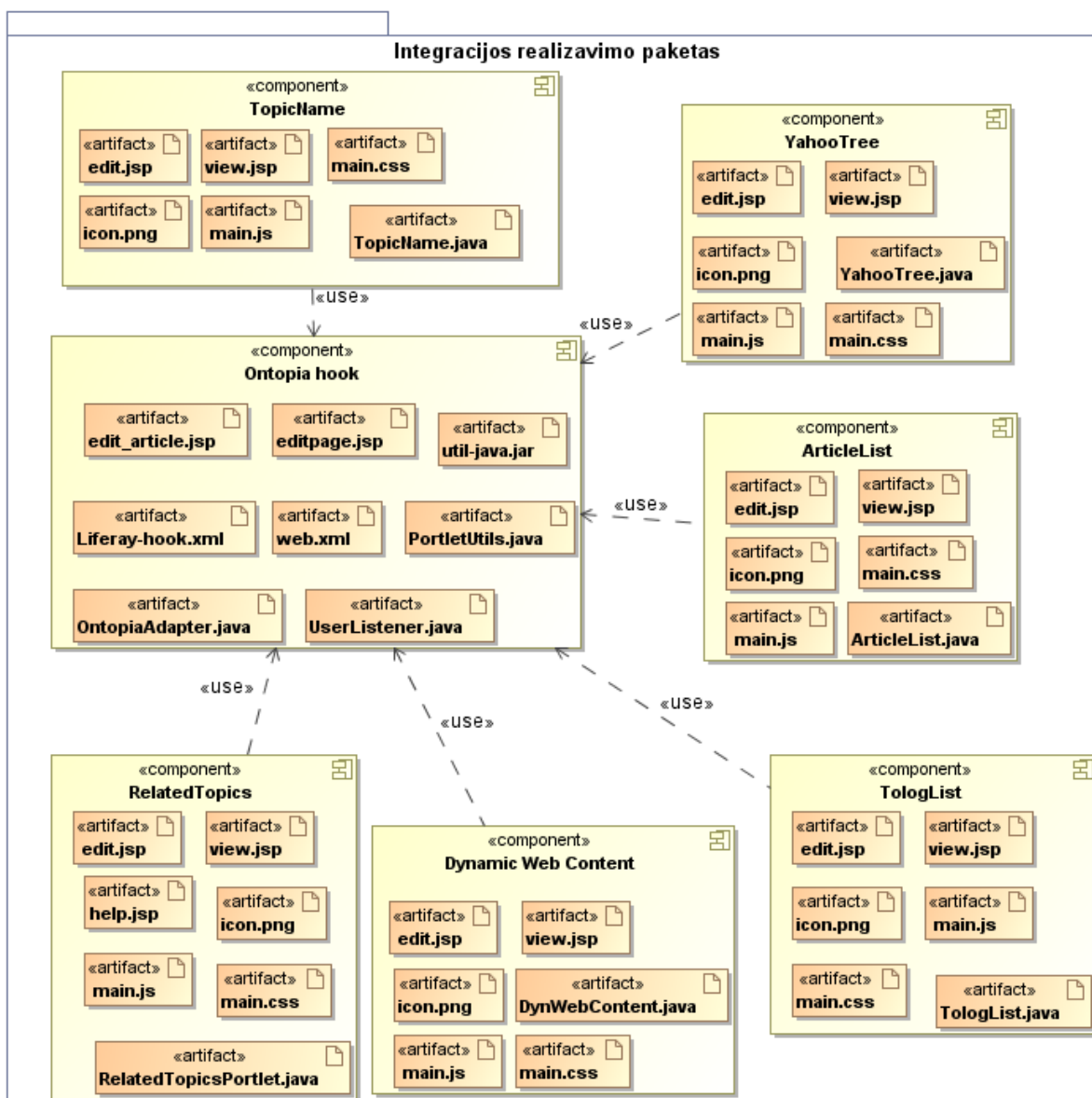
11 Lentelė Topic Maps loginio duomenų modelio specifikacija

Lentelė	Raktai	Lentelės paskirtis
TM_ADMIN_SEQUENCE	Pirminis: seq_count	Atvaizduoti reikiamą teminio tinklo informaciją portale.
TM_TOPIC_MAP	Pirminis: id Išorinis: reifer_id	Saugo informaciją, apie tai kiek ir kokių teminių tinklų yra saugoma duomenų bazėje
TM_TOPIC	Pirminis: id Išorinis: topicmap_id reifer_id	Apibrėžia teminio tinklo temą
TM_TOPIC_NAME	Pirminis: id Išorinis: topicmap_id reifer_id topic_id type_id	Apibrėžia teminio tinklo temos vardą
TM_TOPIC_NAME_SCOPE	Pirminis: scoped_id su theme_id	Saugo, kokiame kontekste (angl. scope) apibrėžti temų vardai
TM_TOPIC_TYPES	Pirminis: topic_id su type_id	Aprašo teminio tinklo temų tipus
TM_SUBJECT_LOCATORS	Išorinis: topic_id	Fiksuoja temų subjektų lokatorius
TM_SUBJECT_IDENTIFIERS	Išorinis: topic_id	Fiksuoja temų subjektų identifikatorius (pvz.: http://psi.liferay.com/tema)
TM_ITEM_IDENTIFIERS	Išorinis: topic_id tm_objectid	Fiksuoja temų identifikatorius (pvz. id156)
TM_VARIANT_NAME	Pirminis: id Išorinis: name_id topicmap_id reifer_id	Aprašo teminio tinklų temų variantus
TM_VARIANT_NAME_SCOPE	Pirminis: scoped_id su theme_id	Saugo, kokiame kontekste (angl. scope) apibrėžti temų vardų variantai
TM_OCCURRENCE	Pirminis: id Išorinis: topic_id type_id topicmap_id reifer_id	Aprašo teminio tinklo temas - pasireiškimus
TM_OCCURRENCE_SCOPE	Pirminis: scoped_id su theme_id	Saugo, kokiame kontekste (angl. scope) apibrėžti pasireiškimai
TM_ASSOCIATION	Pirminis: id Išorinis: type_id topicmap_id reifer_id	Aprašo teminio tinklo asociacijas
TM_ASSOCIATION_SCOPE	Pirminis: scoped_id su theme_id	Saugo, kokiame kontekste (angl. scope) apibrėžtos asociacijos
TM_ASSOCIATION_ROLE	Pirminis: id	Lentelė saugo temų asociacijų

	Išorinis: assoc_id type_id topicmap_id reifer_id player_id	vaidmenis bei nustato, kurios temos dalyvauja asociacijoje.
--	--	---

4.6 Realizacijos modelis

32 paveiksle pateikiamas su *Ontopia* ir *Liferay* integracija įkeliamas paketas, susidedantis iš septynių komponentų. Paketo komponentuose esantys artefaktai – fiziniuose įrenginiuose saugomi *.java*, *.xml*, *.js*, *.css*, *.jsp*, *.png* failai.

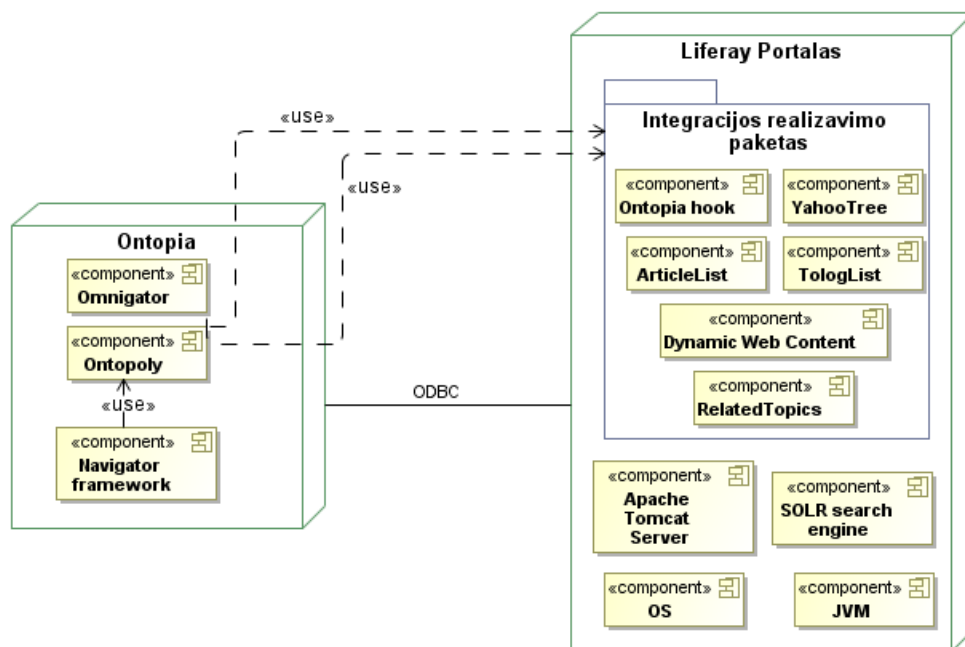


32 pav. Komponentų diagrama

33 paveiksle iliustruojama, kaip įdiegiama *Ontopia* ir *Liferay* integracija. Tam į *Liferay* portalą perkeliamas integracijos paketas su reikalingais komponentais. *Liferay* sistema veikia,

naudojant tam tikrą operacinę sistemą (OS komponentas), Java virtualią mašiną (JVM), *Apache Tomcat* serverį, *SOLR* paieškos variklį.

Prie portalo per *ODBC* jungtį prijungiama *Ontopia* sistema, joje naudojami *Omnigator*, *Ontopoly* moduliai.



33 pav. *Ontopia* ir *Liferay* integracijos įdiegimas

Aprašius metodikos projektavimo etapą, penktame skyriuje nagrinėjamos *Ontopia* ir *Liferay* integracijos realizavimo detalės.

5 *Ontopia* ir *Liferay* integracijos metodikos realizacija

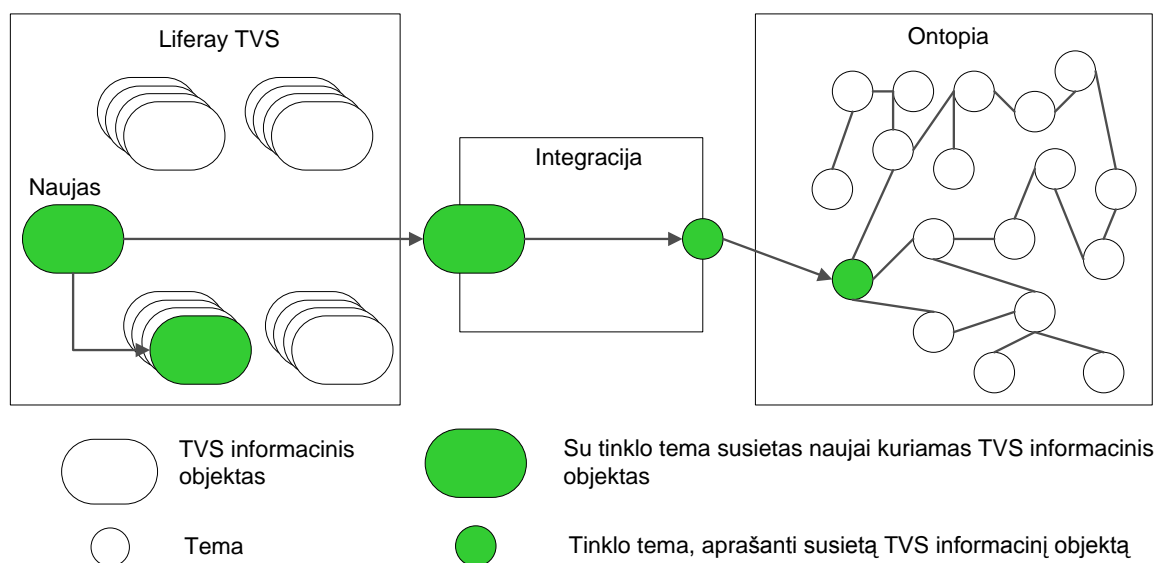
Antrame skyriuje buvo nagrinėjama panašaus pobūdžio sistemų analizė. Prieita prie išvados, kad klasikinė turinio valdymo sistema turi trūkumų informacijos organizavime. Problemą galima spręsti taikant teminių tinklų technologiją, tačiau šiai dienai daugelis TVS ir *Topic Maps* integracijos sprendimų yra arba komerciniai, arba neišbaigti. Dėl šių priežasčių pasirinktas *Ontopia* su *Liferay* variantas, kuriam specifikuoti reikalavimai, parengtas projektas.

Šiame skyriuje aprašoma, kaip realizuoti *Topic Maps* varikliuko *Ontopia* ir *Liferay* portalo integraciją. Metodikoje pateikiami svarbiausi etapai, pagal kuriuos integruojama *Ontopia* su *Liferay*. Kiekvienas etapas susideda iš sau būdingų žingsnių, o praleidus tam tikrą etapą metodika tampa negalima. Realizacijoje atskleidžiamas demonstracinis programavimo mokymo portalo prototipas: *Liferay* turinio susiejimas su teminio tinklo temomis, ontologijos kūrimas ir paruošimas *Ontopoly* įrankiu, integracijos portletų panaudojimas, *tolog* užklausų sakiniai. Šis pavyzdys iliustruoja, kad metodika gali būti tinkamai taikoma bet kokiai apsibrėžtai dalykinei sričiai.

Norint praplėsti informacijos organizavimo galimybes *Liferay* sistemoje, reikia:

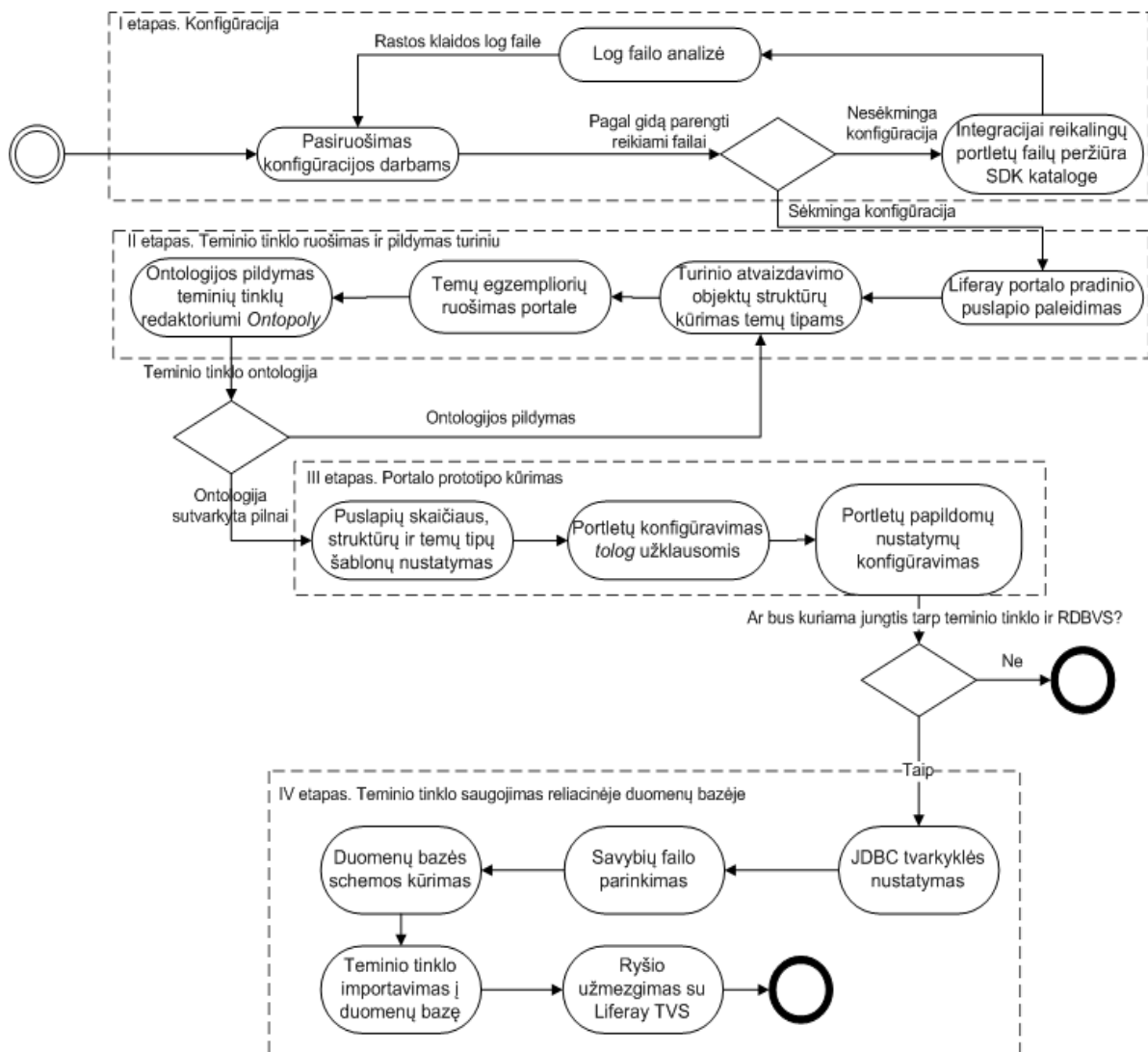
- Apibūdinti *Liferay* turinio objektus *Topic Maps* temomis;
- Leisti vartotojams sujungti norimą turinį su dalykinės srities ontologija;
- Atvaizduoti teminio tinklo informaciją portale.

Liferay portale aptinkama TVS informacinių objektų aibė. Integracija suteikia galimybę susieti *Ontopia* varikliuku sukurtas *Topic Maps* temas su *Liferay* informaciniais objektais [25]. Ketvirtame skyriuje integracijos architektūra pateikta detaliau, o realizacijoje 34 paveiksle primenama principinė schema:



34 pav. *Ontopia* ir *Liferay* integracijos principinė architektūra

Iš technolinės pusės *Ontopia* ir *Liferay* sistemų integracija yra ganėtinai paprasta, nes abu produktai sukurti *Java* pagrindu (lengviau užtikrinamas dviejų sistemų komunikavimas, nesurūpinant internetinėmis paslaugomis ir galutinių rezultatų pristatymu). Kuriamas portalo prototipas iliustruoja *Liferay* TVS modernizavimo metodiką (35 pav).



35 pav. *Ontopia* ir *Liferay* integracijos metodika

Norint pritaikyti teminius tinklus *Liferay* sistemoje pirmiausiai reikia suprasti kaip skirtingų sistemų (*Ontopia* ir *Liferay*) dalys formuoja vieningą darnią sistemą ir tik tada pereiti prie praktinių veiksmų. Kituose poskyriuose detalizuojami *Ontopia* ir *Liferay* integracijos etapai ir žingsniai.

5.1 Konfigūracija

Pirmasis iteracinis metodikos etapas – integracijos konfigūracija, kuri nėra triviali, o be to reikalauja nemažai rankinio kruopštaus darbo. Visą reikalingą konfigūracijos informaciją galima rasti gide [23], o šiame skyriuje aptariami svarbiausi *Ontopia* ir *Liferay* konfigūracijos žingsniai.

5.1.1 Pasiruošimas konfigūracijos darbams

Siekiant išbandyti, kaip *Liferay* ir *Ontopia* sistemos veikia kartu, reikia atlikti pasiruošimo veiksmus. Darbai prasideda nuo konstravimo blokų ieškojimo internete. Prieš atsisiunčiant *Liferay* portalą (nemokamą bendruomenės versiją) rekomenduojama į naudojamą operacinę sistemą įdiegti *Ontopia* varikliuką. *Ontopia* turi būti įdiegta iš programos kodo (angl. *source*), nes daugelis *Liferay* integracijos (portletai ir t.t.) failų yra patalpinti *svn* kataloge.

Šiame žingsnyje taip pat įdiegiama *Apache Ant* programa, kuri organizuoja *Liferay SDK* įskiepių naudojimą. *Liferay SDK* failų tvarkymas aprašomas 5.1.2 skyriuje.

5.1.2 Integracijai reikalingų portletų failų peržiūra SDK kataloge

Liferay SDK katalogas yra būtinas, norint patiems kurti ir vystyti portletus. Integravus *Ontopia*, reikalingi specifiniai portletai, atvaizduojantys teminio tinklo informaciją. Šiame žingsnyje svarbu sukurti *ontopia-liferay.jar* failą bei jį nukopijuoti į *svn* kataloge esančių portletų portlets-> *pavadinimas*->*docroot*->*Web-INF*->*lib* pakatalogį. Papildytos struktūros bylos kopijuojamos į *Liferay* portalo *SDK* katalogą. Tokiu būdu, atsiranda galimybė įdiegti naujus portletus į tradicinį *Liferay* portalą, taikant *ant build* komandą. Įvykus klaidai, operacinė sistema informuoja vartotoją bei išveda pranešimą. Pagal tai galima orientuotis, kas buvo atlikta neteisingai. Neįdiegus *Topic Maps* portletų į *Liferay TVS*, modernizavimo metodika tampa neprasminga.

Toliau nagrinėjama, kaip nesėkmingos konfigūracijos atveju gali padėti *log* failas.

5.1.3 Log failo analizė

Log failas *catalina.out* randamas *portal*->*liferay-portal6*->*tomcat-6.0.2.6*->*logs* kataloge. Šis failas seka *Liferay* įvykius ir informuoja vartotoją apie klaidas. *Catalina.out* faile reikia atskirti vietas, turinčias *ERROR* žymę, nes dėl jų sistema gali veikti neteisingai.

Žemiau pateikiamas klaidos pavyzdys. Pagal tai galima atsekti, kokios klasės metoda reikia koreguoti, kad sukelta klaida būtų panaikinta.

```
09:26:34,036 ERROR [PortletRequestDispatcherImpl:136] org.apache.jasper.JasperException:
net.ontopia.utils.OntopiaRuntimeException:
net.ontopia.topicmaps.query.core.BadObjectReferenceException: No topic with subject identifier
'urn:uuid:ca5e9ddb-ac5c-4fbd-aeb1-ab46fe98ef22' found
org.apache.jasper.JasperException: net.ontopia.utils.OntopiaRuntimeException:
net.ontopia.topicmaps.query.core.BadObjectReferenceException: No topic with subject identifier
'urn:uuid:ca5e9ddb-ac5c-4fbd-aeb1-ab46fe98ef22' found at
org.apache.jasper.servlet.JspServletWrapper.handleJspException(JspServletWrapper.java:491) at
org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:419)
    at org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:313)
    at org.apache.jasper.servlet.JspServlet.service(JspServlet.java:260)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
```

Apžvelgus *Ontopia* ir *Liferay* konfigūracijos žingsnius, pateikiama, kaip paleisti *Liferay* portalą, kurti ir tvarkyti teminį tinklą.

5.2 Teminio tinklo ruošimas ir pildymas turiniu

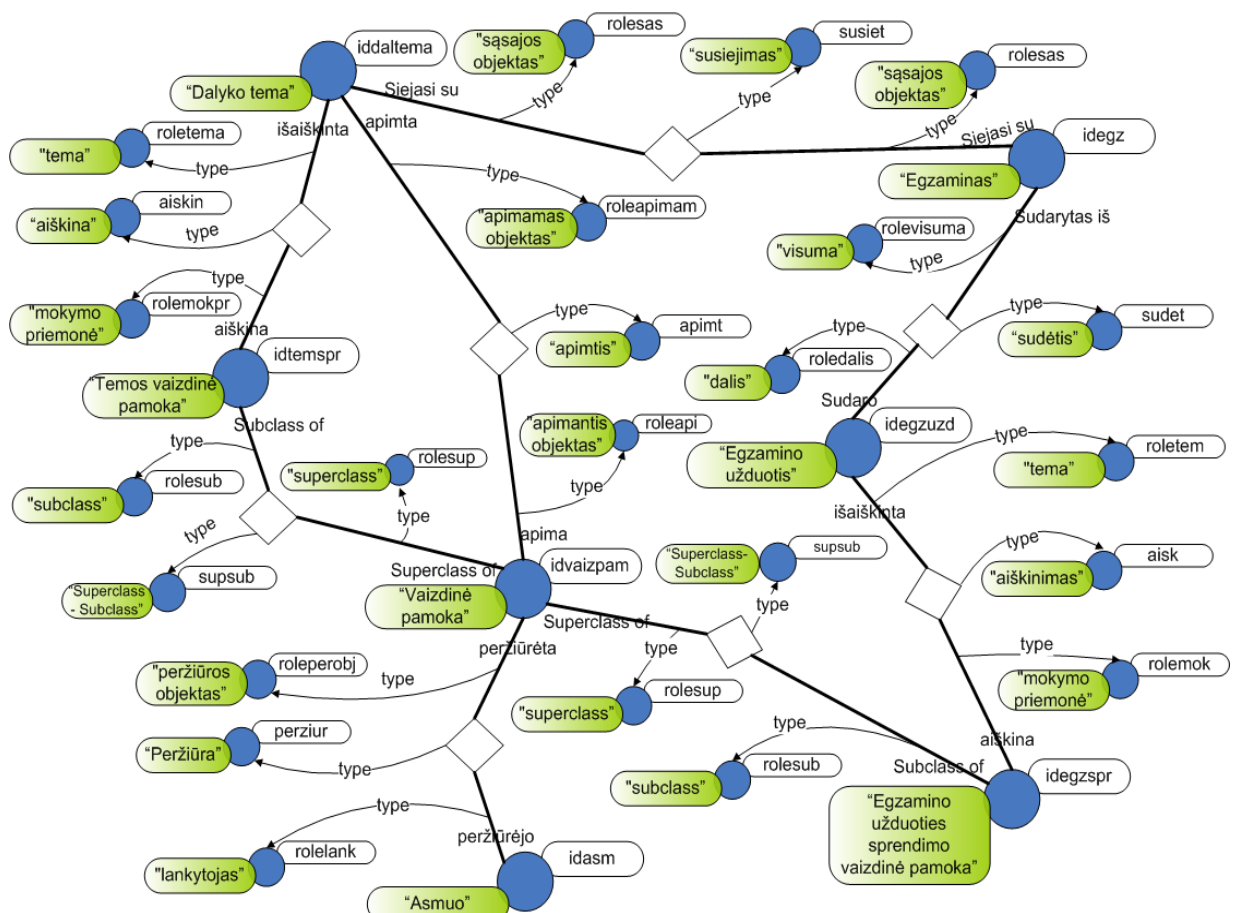
Antrasis metodikos etapas apima teminio tinklo ruošimą ir pildymą turiniu. Etape aptinkamas grįžtamasis ryšis, nes dalykinės srities ontologija dažnai keičiama ir tobulinama.

5.2.1 *Liferay* portalą pradinio puslapio paleidimas

Sėkmingai įdiegus *Topic Maps* portletus į *Liferay* sistema, turi būti paleistas *Liferay* portalas, nes priešingu atveju nepavyks sukurti demonstracinės sistemos. *Liferay* paleidžiama kataloge *liferay-portal6/tomcat/bin* įvedus komandą *sh startup.sh* (*Linux* sistemose). Tada, priklausomai nuo turimo kompiuterio resursų, reikia truputį palaukti, kol naršyklėje pasirodys pradinis *Liferay* puslapis.

5.2.2 Turinio atvaizdavimo objektų struktūrų kūrimas temų tipams

Šiame žingsnyje svarbu žinoti, kad prieš pradėdant dirbti su teminiu tinklu *Liferay* sistemoje reikia pasirinkti dalykinę sritį. Demonstracijai pasirinkta programavimo mokymo sritis.



36 pav. Programavimo mokymo ontologijos teminio tinklo schema

Teminis tinklas pradedamas kurti nuo ontologijos: pirmiausiai sukuriamos tos temos-tipai, kurios apibūdina egzempliorius, tada ryšių tarp temų tipai, vaidmenų, pasireiškimų tipai. Tai panašu į esybių-ryšių modelio kūrimą.

36 paveiksle pavaizduota programavimo mokymo ontologija – atspirties taškas teminio tinklo realizacijai *Liferay* portale. Norint sukurti tokią ontologiją portale reikia prisijungti prie administravimo srities (angl. „*Control Panel*“) bei sukurti tiek TVS informacinių objektų atvaizdavimo struktūrų, kiek yra temų tipų. Programavimo mokymo ontologijos atveju turi būti sukurtos tokios atvaizdavimo struktūros: „Asmuo“, „Egzamino užduoties sprendimo vaizdinė pamoka“, „Temos vaizdinė pamoka“, „Egzamino užduotis“, „Egzaminas“, „Dalyko tema“. (hierarchiją nusakantis temos tipas „Vaizdinė pamoka“ kuriamas įrankiu „*Ontopoly*“).

ID	Description	Actions
<input type="checkbox"/> 14501	Dalyko tema Čia dedamos FPS stambesnės temos. Jose yra aprašymas bei gali būti daug vaizdinių pamokų.	Actions
<input type="checkbox"/> 16804	Egzaminas Ši struktūra skirta talpinti trumpus aprašymus apie informacinių technologijų egzaminų metus, kuriuose gali būti daug vaizdinių pamokų.	Actions
<input type="checkbox"/> 18310	Vaizdinė pamoka Struktūra skirta talpinti vaizdinių pamokų aprašymus.	Actions
<input type="checkbox"/> 19202	Egzamino užduotis Struktūra (temos tipas) skirtas kaupti informaciją apie egzaminų užduotis (salygas).	Actions
<input type="checkbox"/> 19239	Egzamino užduoties sprendimo vaizdinė pamoka Temos tipas, saugantis informaciją apie egzaminų vaizdines pamokas, kurios aiškina egzaminus.	Actions
<input type="checkbox"/> 19241	Temos vaizdinė pamoka Temos tipas, saugojantis informaciją apie FPS dalykų vaizdines pamokas.	Actions
<input type="checkbox"/> 19243	Asmuo Temos tipas saugo svetainės vartotojus ir lankytojus	Actions

37 pav. *Liferay* straipsnių struktūros

Liferay straipsnių struktūrų pavadinimai turi sutapti su teminio tinklo temų tipais (angl. *Topic Type*). Atlikus šį veiksma galima kurti *Liferay* turinį ir tuo pačiu ruošti teminio tinklo egzempliorius.

5.2.3 Temų egzempliorių ruošimas portale

Kiekviena struktūra turi savo šabloną, nusakantį *Liferay* TVS informacinių objektų laukų sudėtį. Kuriant TVS atvaizdavimo informacinį objektą nurodomi struktūros ir šablono pavadinimai, ruošiamas norimas turinys (tekstas, paveiksliukai ir kt.) bei publikuojama į serverį – tokiu būdu įvyksta *Topic Maps* temos susiejimas su turiniu portale.

38 paveiksle pavaizduotas temos „Egzamino užduotis“ egzemplioriaus (angl. *instance*) kūrimo pavyzdys. Tokiu pat principu ruošiami ir kitų temų egzemplioriai.

2010 metų brandos egzamino užduotis Nr.2

«Back

Content [History](#)

ID: 19971 Version: 1.0 Status: **Approved**

Name
2010 metų brandos egzamino užduotis Nr.2

Language **Default Language**
English (United States) English (United States)

Ontopoly:

Tittel
2010 metų brandos egzamino užduotis Nr.2

«Basic

Structure

Name
Egzamino užduotis (Use Default)

[Choose](#)

Template

Template
 EgzaminoUzduociai

38 pav. Temos „Egzamino užduotis“ egzempliorius

Kai turinys užpildomas, atsiradę TVS informaciniai objektai gali būti redaguojami, šalinami. Tvarkant tą patį TVS informacinį objektą kelis kartus, teminio tinklo redaktoriuje *Ontopoly*, kuriama to paties pavadinimo tik kito identifikatoriaus tema (fiktyvus dublis). Tačiau nerekomenduojama jo koreguoti *Ontopoly* priemonėmis, o viską daryti *Liferay* portale.

Kitame skyriuje aprašoma, kokiomis situacijomis reikia naudoti *Ontopoly* redaktorių.

5.2.4 Ontologijos pildymas teminių tinklų redaktoriumi *Ontopoly*

Teoriškai *Ontopoly* redaktoriaus nėra svarbus, nes vartotojas turi galimybę apsirrašyti norimą ontologiją vien tik naudojant *Liferay* portalą ir redaguojant atitinkamus straipsnius. Šiai dienai toks variantas nėra įmanomas, nes *Ontopia* ir *Liferay* integracija nėra iki galo išbaigta.

Ontopoly redaktorių reikia naudoti, kai norima specifiuoti temų asociacijas, vaidmenis, nustatyti temų pasireiškimų laukus. Siūloma įsidėmėti adresą <http://localhost:8080/ontopoly> ir tvarkyti teminį tinklą *Liferay*.

Atsidarius *Liferay* teminiam tinklui redaktoriuje matomi sukurti temų tipai. Tačiau tokią ontologiją dar reikia pildyti t.y. susieti temas asociacijomis (ryšiais), priskirti išteklius, nustatyti asociacijų vaidmenis. Tam vartotojas *Ontopoly* redaktoriuje turi naudoti „*Association types*“, „*Role types*“, „*Occurrence types*“ korteles.

Specifikuojant asociacijas nurodomas pavadinimas (-ai), identifikatorius, jungiamos temos ir vaidmenys, kardinalumai. Aprašant pasireiškimus nurodomas, pavadinimas, identifikatorius, tipas bei tema, naudojanti pasireiškimą.

39 paveiksle pateiktas temos „Temos vaizdinė pamoka“ egzempliorių sąrašas. Šis sąrašas gali būti didinamas (mažinamas) sukūrus, pašalinus atitinkamos struktūros *Liferay* straipsnį.

Temos vaizdinė pamoka














Expand all Collapse all

- Cikliniai algoritmai
- Dinaminiai masyvai
- Dvimatis masyvas - matrica
- Funkcija Sqrt
- Funkcija Trim
- Funkcijos
- Funkcijos reikšmių lentelė
- Kintamieji
- Kvadratinė matrica
- Lietuviškųjų simbolių naudojimas FPS 0.6.4a ir FPS 0.9.6 versijose
- Masyvai.Indeksuoti kintamieji
- Matricos eilučių, stulpelių šalinimas ir įterpimas
- Paprastos konsolinės programos kūrimas 1
- Paprastos konsolinės programos kūrimas 2
- Paprastos konsolinės programos kūrimas 3
- Pirmoji Programa
- Praktiniai masyvų uždavinių pavyzdžiai
- Praktinis dvimačio masyvo uždavinio pavyzdys
- Praktinis kintamųjų panaudojimas
- Praktinis įrašų duomenų tipo uždavinio pavyzdys
- Procedūros
- String Funkcijos
- Tekstiniai failai
- Teksto analizė 1
- Teksto analizė 2
- Teksto analizė 3
- Teksto analizė 4

39 pav. Temos „Egzamino užduotis“ egzempliorius

40 paveiksle pateiktas pilnai aprašytos ontologijos temos egzempliorius. *Ontopoly* redaktoriuje „Instances“ kortelėje pasirinkus konkrečią temą, matoma atitinkama informacija, ryšiai.

Praktinis įrašų duomenų tipo uždavinio pavyzdys

Name:	<input type="text" value="Praktinis įrašų duomenų tipo uždavinio pavyzdys"/>	
Created_by (Work):	<input type="text" value="bruno@7cogs.com"/>	 
ArticleID:	<input type="text" value="21226"/>	
Display Date:	<input type="text" value="2010-12-01 18:36:00"/>	
Create Date:	<input type="text" value="2010-12-01 18:38:09"/>	
NuorodaAtsisiusti:	<input type="text" value="http://localhost:8080/FPS/29Pamoka_irasas_praktinis/irasas.zip"/>	
Contains (Containeer):	<input type="text" value="Guest"/>	 
topikotipui (dalis):	<input type="text" value="Temos vaizdinė pamoka"/>	 
Subject identifier:	<input type="text" value="urn:uuid:2e8b5633-5d71-4b7b-88b3-194f573b9c5"/>	
Modified Date:	<input type="text" value="2010-12-01 18:38:10"/>	
Has_workflow_state (Work):	<input type="text" value="Approved"/>	 
aiškina (mokymo priemonė):	<input type="text" value="Įrašai"/>	 
NuorodaProgramosTekstas:	<input type="text" value="http://localhost:8080/FPS/29Pamoka_irasas_praktinis/irasas_prak"/>	
Version:	<input type="text" value="1.0"/>	
Approved_by (Work):	<input type="text" value="bruno@7cogs.com"/>	 
Approved Date:	<input type="text" value="2010-12-01 18:38:10"/>	
NuorodaPerziureti:	<input type="text" value="http://localhost:8080/FPS/29Pamoka_irasas_praktinis/irasas/irasa"/>	

40 pav. „Temos vaizdinė pamoka“ egzemplioriaus pavyzdys

Sukūrus ontologiją ir turinį, reikia nuspręsti kaip teminio tinklo informacija bus atvaizduojama portale. Čia svarbiausią vaidmenį atlieka *Topic Maps* teikiamo funkcionalumo portletai.

5.3 Portalo prototipo kūrimas

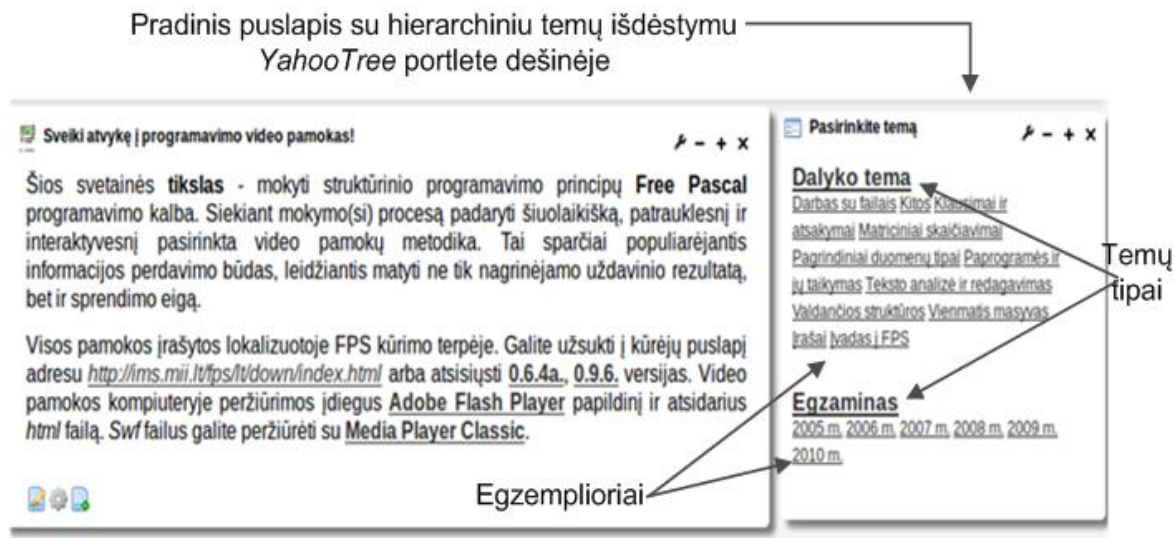
Trečiajame etape rūpinamasi portalu, kurio kūrimui reikia nustatyti puslapių skaičių ir struktūrą, atvaizduoti norimą teminio tinklo informaciją, konfigūruoti *Topic Maps* portletus *tolog* užklausomis. Visi papildomi teminių tinklų portletai *Liferay* sistemoje randami pasirinkus *Add*→*More*→*Sample* meniu.

5.3.1 Puslapių skaičiaus, struktūrų ir temų tipų šablonų nustatymas

Programavimo vaizdinių pamokų portalo prototipo turinys valdomas teminiu tinklu. Puslapiuose naudojami integracijos portletai ir rodyklėmis pavaizduoti *Topic Maps* elementai pagerina navigaciją portale.

Liferay turinio valdymo sistemoje *Topic Maps* palaikymui naudojami keturių tipų puslapiai:

- ***Pradinis***. Portalui pristatyti geriausiai tinka *YahooTree* portletas, atvaizduojantis nurodytų temų ir jų egzempliorių hierarchiją. Vartotojui suteikta galimybė rinktis arba temas tipą (pvz.: „Egzaminas“), arba potemę. Abiem atvejais nueinama į puslapį, kuriame rodomas atitinkantis turinys ir ryšiai su kitomis temomis.



41 pav. Pradinis puslapis

- ***Temos puslapis***. Skirtas parodyti temų tipų informaciją. Puslapio parametruose (per URL) perduodamas analizuojamos temos identifikatorius, todėl *Ontopoly* tokioms temas uždedamas šablonas: *http://localhost:8080/tema&topic=%topicid%*. Struktūrą sudaro *Topic Name* portletas kairėje viršuje, žemiau jo *Article List*, o dešinėje – *RelatedTopics*.

PROGRAMAVIMO VIDEO PAMOKOS Sign In

Pradžia Forumas Apklausa Programavimo pamokų dienraštis

Egzaminas

Skyriuje pateikiami valstybinių informatikos egzaminų užduočių video siužetai. Šios pamokos skirtos abiturientams, norintiems perprasti egzaminų programų kūrimo specifiką. Jais išanalizavus galima tikėtis sėkmingesnio IT VBE rezultato.

Taip pat galite žiūrėti [Free Paskalio](#) pamokų kursą.

- sudarytas iš temų
 - [2005 m.](#)
 - [2006 m.](#)
 - [2007 m.](#)
 - [2008 m.](#)
 - [2009 m.](#)
 - [2010 m.](#)

Visos teisės saugomos. © 2009-2011 <http://www.programavimopamokos.net> puslapį sukūrė **Andrius Lauraitis**

42 pav. Temos puslapis

- Mokymo temos puslapis. Atvaizduoja temų egzempliorių informaciją. Puslapio parametruose perduodamas temos identifikatorius, todėl *Ontopoly* tokioms temoms uždedamas šablonas: `http://localhost:8080/mokymotema&topic=%topicid%`. Struktūrą sudaro *Dynamic Content* portletas kairėje, o dešinėje – *RelatedTopics*.

PROGRAMAVIMO VIDEO PAMOKOS Sign In

Pradžia Forumas Apklausa Programavimo pamokų dienraštis

2010 m.

Aptariami 2010 metų egzaminų ir bandomųjų užduočių sprendimai. Pateikiami bandomojo egzamino du praktiniai uždaviniai: Žirmiai ir pataisyta versija bei Pasirinkimas. Taip pat pagrindinės sesijos uždaviniai: Sachmatai (2 versijos) bei Gimtadienis. **Iš viso užduočių: 4, vaizdinių pamokų: 6, – 3,5 val.**

- temos tipas
 - [Egzaminas](#)
- sudarytas iš
 - [2010 metų bandomoji užduotis Nr.1](#)
 - [2010 metų bandomoji užduotis Nr.2](#)
 - [2010 metų brandos egzamino užduotis Nr.1](#)
 - [2010 metų brandos egzamino užduotis Nr.2](#)

Visos teisės saugomos. © 2009-2011 <http://www.programavimopamokos.net> puslapį sukūrė **Andrius Lauraitis**

43 pav. Mokymo temos puslapis

- Vaizdinės pamokos puslapis. Pateikia Free Pascal vaizdinės pamokos temą. *Ontopoly* įrankyje nustatomas šablonas: `http://localhost:8080/vaizdinepamoka&topic=%topicid%`. Struktūrą sudaro *Dynamic Content* portletas kairėje, *RelatedTopics* dešinėje ir *Tolog list* apačioje.

PROGRAMAVIMO VIDEO PAMOKOS

Mokėjimai Pradžia Forumas Apklausa Programavimo pamokų dienraštis

Dvimatis masyvas - matrica + DYNAMIC CONTENT

Tai teorinio pobūdžio pamoka. Aptariama, kas yra dvimatis masyvas, kaip jis vaizduojamas kompiuterio atmintyje. Aiškinama, kaip peržiūrėti matricos elementus ir spausdinti matricą. Pabaigoje susipažįstama su dinaminio dvimatio masyvo sąvoka.
Įkėlimo data: **2009-05-14**, trukmė: **17min. 35sek.**

[\[Atsisijusti\]](#) [\[Peržiūrėti\]](#) [\[Užduočių sąlygos ir programų tekstai\]](#) + TOLOG LIST

- temos tipas + RELATEDTOPICS
 - [Temos vaizdinė pamoka](#)
- aiškina + RELATEDTOPICS
 - [Matriciniai skaičiavimai](#)

Visos teisės saugomos. © 2009-2011 <http://www.programavimopamokos.net> puslapį sukūrė **Andrius Lauraitis**

44 pav. Vaizdinės pamokos puslapis su nuorodomis į informacijos resursus

Išskyrus pradinį, kiti puslapiai pasiekiami tik tada, kai puslapio parametruose (per *URL*) perduodamas analizuojamos temos identifikatorius. Mokymo temos ir vaizdinės pamokos puslapiai portale *header* dalyje nerodomi (savybė *hidden*), nes turinys igyja prasme tik tada, kai pasirenkama konkreti tema, o ne *Liferay* puslapis.

Apžvelgus portalo prototipo puslapių struktūrą ir portletų išdėstymą juose, kitame skyriuje analizuojamos, kaip atvaizduoti norimą informaciją portale.

5.3.2 Portletų konfigūravimas *tolog* užklausomis

Kadangi pagrindinė portletų funkcija – rodyti norimą teminių tinklų informaciją, integracijos portretams reikia nurodyti, kokie *Topic Maps* elementai turi būti atvaizduojami. Šiam tikslui turi būti suprogramuotos *tolog* užklausos į teminį tinklą.

12 lentelėje pateikiama metodikoje naudojamų portletų konfigūracija, pritaikius *tolog* loginę kalbą.

12 Lentelė *Topic Maps* portletų konfigūravimas

Portleto pavadinimas	<i>Tolog</i> užklausa	Aprašymas
<i>YahooTree</i>	<p>Temų tipų: select \$B from instance-of(\$A, \$B), not(instance-of(\$B, i"http://psi.ontopia.net/ontology/system- topic")), \$B /= i"http://psi.ontopia.net/liferay/category", \$B /= i"http://psi.ontopia.net/liferay/group", \$B /= i"http://psi.ontopia.net/liferay/community", \$B /= i"http://psi.ontopia.net/liferay/article", \$B /= i"http://psi.ontopia.net/liferay/webcontent", \$B /= i"http://psi.ontopia.net/liferay/user" order by \$B?</p> <p>Egzempliorių: instance-of(\$A, %self%) order by \$A desc?</p> <p>Stulpeliai: 2</p>	<p>Sukuria aukščiausio lygio temų ir jų potemių (egzempliorių) hierarchinį išdėstymą stulpeliais. Aukščiausio lygio temos vaizduojamos stambesniu šriftu viršuje, o potemės – žemiau stulpelyje.</p> <p><i>%Self%</i> - nurodo, kad bus formuojami \$A kintamojo egzemplioriai. Tėvinėje užklausoje filtruojamos <i>Ontopoly</i> sisteminės temos</p> <p>(<i>i"http://psi.ontopia.net/ontology/system-topic"</i>) bei tos temos, kurios neturi būti įtrauktos į rezultatų aibę (/= operatorius).</p>
<i>Article list</i>	<p>using lr for i"http://psi.ontopia.net/liferay/" lr:sudetis(%topic% : lr:visuma, \$ART : lr:dalis), lr:article_id(\$ART, \$ARTID), lr:contains(\$ART : lr:containeer, \$GRP : lr:container), lr:groupid(\$GRP, \$GRPID) order by \$ART desc?</p>	<p>Skirtas atvaizduoti turinį temų tipams pagal užmegztą ryšį „sudetis“ ontologijoje.</p> <p>Kiekvienas straipsnis portale turi identifikatorių: (<i>i"http://psi.ontopia.net/liferay/article_id"</i>). Perdavus atitinkamą temą, gaunami egzemplioriai (pvz. Egzaminas temos atveju: 2010 m., 2009m. ir t.t).</p>
<i>Tolog list</i>	<p>select \$TITLE,\$URL from { type(\$O, i"http://psi.ontopia.net/liferay/nuorodaatsisiusti"</p>	<p>Atvaizduoja temos pasireiškimų elementus portale. Čia</p>

), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Atsisiųsti]" type(\$O, i"http://psi.ontopia.net/liferay/nuorodaperziureti"), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Peržiūrėti]" type(\$O, i"http://psi.ontopia.net/liferay/nuorodaprogttekstas"), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Uzduočių sąlygos ir programų tekstai]" }?	i"http://psi.ontopia.net/liferay/nuorodaperziureti" temos pasireiškimo, skirto peržiūrėti vaizdinę pamoką identifikatorius, o %topic% - temos parametras, perduodamas per URL.
--	---	--

Integracijoje naudojamiems *Related Topics, Dynamic web content, Topic Name* portletams *tolog* užklausa nereikalingos.

13 lentelėje pateikiama naudota užklausa, įėjimo, išėjimo parametrai. Pateikiami užklausoje naudojami predikatai[§].

13 Lentelė naudotų tolog užklausių analizė

<i>Tolog</i> užklausa	Įėjimo parametrai	Rezultatai	Predikatai			
			Bendriniai	<i>Topic Maps</i>	Operatoriai	Kintamieji
select \$B from instance-of(\$A, \$B), not(instance-of(\$B, i"http://psi.ontopia.net/ontology/system-topic")), \$B /= i"http://psi.ontopia.net/liferay/category", \$B /= i"http://psi.ontopia.net/liferay/group", \$B /= i"http://psi.ontopia.net/liferay/community", \$B /= i"http://psi.ontopia.net/liferay/article", \$B /= i"http://psi.ontopia.net/liferay/webcontent", \$B /= i"http://psi.ontopia.net/liferay/user" order by \$B?	Liferay Topic Map	Eilučių:2 Trukmė: 30 ms. Išėjimo param.: B	11(972+969+967+965+948+922+869+836+816+799+791 naudotų elementų)	2 (669+972 naudotų elementų)	14 (12 projekcija + order + not)	2
using lr for i"http://psi.ontopia.net/liferay/" lr:sudėtis(@389 : lr:visuma, \$ART : lr:dalis), lr:article_id(\$ART, \$ARTID), lr:contains(\$ART : lr:containeer, \$GRP : lr:container), lr:groupid(\$GRP, \$GRPID) order by \$ART desc?	Tema, kurios id lygus 389	Eilučių:1 Trukmė: 17 ms Išėjimo param.: ARTID, GRP, ART, GRPID	-	4(1+1+1+1 naudotų elementų)	5(4 projekcija + order)	4
select \$TITLE,\$URL from { type(\$O, i"http://psi.ontopia.net/liferay/nuorodaatsisiusti"), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Atsisiųsti]" type(\$O, i"http://psi.ontopia.net/liferay/nuorodaperziureti"), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Peržiūrėti]" type(\$O, i"http://psi.ontopia.net/liferay/nuorodaprogttekstas"), occurrence(@784, \$O), resource(\$O,\$URL),\$TITLE="[Uzduočių sąlygos ir programų tekstai]" }?	Tema, kurios id lygus 784	Eilučių:3 Trukmė: 1ms. Išėjimo param.: TITLE, URL	3 (1+1+1 naudotų elementų)	9 (9+1+1+9+1+1+9+1+1 naudotų elementų)	3 or alternatyvos	2

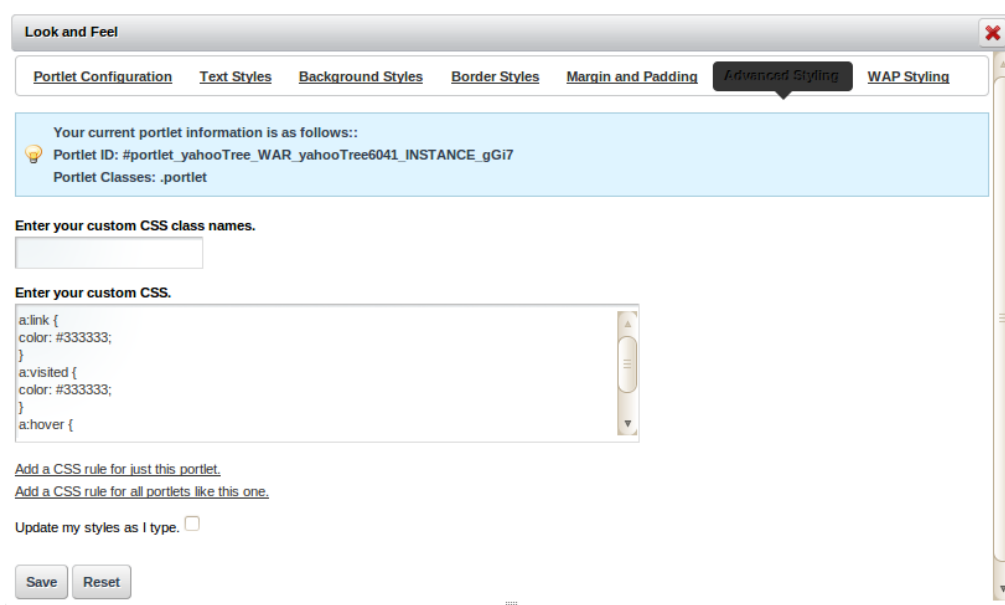
[§] sakiny su kintamaisiais, kuris gali būti teisingas arba klaidingas priklausomai nuo kintamųjų reikšmių.

Tolog užklauso padidina programavimo mokymo sistemoje naudojamas teminio tinklo galimybes. Pastebima, kad užklauso sudėtingumas bei predikatų skaičius lemia veikimo spartą. Užklauso gali tapti dar sudėtingesnės priklausomai nuo užklauso formulavimo, teminio tinklo apimties.

5.3.3 Portletų papildomų nustatymų konfigūravimas

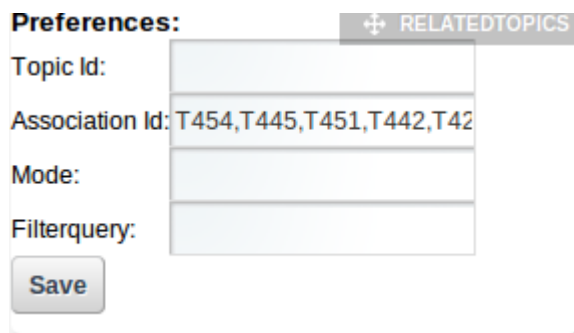
Pagrindinis darbas konfigūruojant specifinius *Ontopia* ir *Liferay* integracijos portletus atliekamas programuojant *tolog* užklauso. Visgi atsiranda ir papildomų portletų tvarkymo poreikių:

- Stilių nustatymo. 45 paveiksle parodytas *YahooTree* portleto CSS stiliaus parinkimas:



45 pav. *YahooTree* CSS stiliai

- Kitų poreikių. 46 paveiksle *RelatedTopics* portletui nurodomi tie asociacijų identifikatoriai, kurie nerodomi vartotojui portale:



46 pav. *RelatedTopics* papildomų nustatymų parinkimas

Apžvelgus *Topic Maps* puslapių pritaikymą *Liferay* portale, analizuojamos teminio tinklo saugojimo galimybės.

5.4 Teminio tinklo saugojimas reliacinėje duomenų bazėje

Pateikti konfigūracijos, teminio tinklo ontologijos kūrimo ir atvaizdavimo portale metodiniai žingsniai yra esminiai, tačiau tampa beverčiais, kai neapmąstytos teminio tinklo saugojimo alternatyvos. Standartiniai *Liferay TVS* objektai palaikomi duomenų bazių valdymo sistemos (DBVS). Teminiai tinklai – priešingai, o jų nesaugant kiekvieną kartą perkrovus serverį tektų iš naujo kurti dalykinės srities ontologiją.

Ontopia suteikia galimybę saugoti teminius tinklus reliacinėse duomenų bazėse. Palaikomos *Oracle*, *MS SQL Server*, *MySQL*, *PostgreSQL* DBVS. Visos reikiamos konfigūravimo detalės pateikiamos [26] literatūros šaltinyje.

5.4.1 JDBC tvarkyklės nustatymas

Įdiegus *MySQL* serverį į operacinę sistemą, reikia ieškoti *mysql JDBC* tvarkyklės, nes per ją nustatomas *MySQL* ir Java priemonėmis sukurto paketo *Ontopia* ryšys. Radus failą, būtina jį įtraukti į *\$CLASSPATH* aplinkos kintamąjį. *Ubuntu* sistemoje konsolėje surinkus komandą *echo \$CLASSPATH* turėtų matytis panaši eilutė: */home/\${vartotojas}/portal/liferay-portal-6.0.4/tomcat-6.0.26/lib/ext/mysql-connector-java-5.3.1.10.jar*

5.4.2 Savybių failo parinkimas

Kitas žingsnis – savybių failo sukūrimas. Jame turi būti nurodyta *JDBC* tvarkyklė, *root* vartotojo prisijungimas prie *MySQL server*, naudojama *DVBS*:

```
net.ontopia.topicmaps.impl.rdbms.Database=mysql
net.ontopia.topicmaps.impl.rdbms.ConnectionString=jdbc:mysql://localhost/topicmaps?useUnicode=yes&characterEncoding=utf8&relaxAutoCommit=true
net.ontopia.topicmaps.impl.rdbms.DriverClass=com.mysql.jdbc.Driver
net.ontopia.topicmaps.impl.rdbms.UserName=root
net.ontopia.topicmaps.impl.rdbms.Password=slaptas
net.ontopia.topicmaps.impl.rdbms.ConnectionPool=true
```

5.4.3 Duomenų bazės schemas kūrimas

Teminio tinklo elementai turi atitikmenis struktūrizuotose duomenų bazėse. *Ontopia* instaliacijos kataloge reikia rasti *rdbms* katalogą, kuriame yra *setup*. Jame pasirenkamas atitinkamai *DVBS* pritaikytas *DDL* komandų rinkinys. Naudojant *MySQL* server naudinga žinoti komandą *SOURCE*, kuri importuoja į serverį *SQL* faile nurodytą duomenų bazės struktūrą. Pakanka nurodyti kelią iki failo. *Topic Maps* schemą sudaro šešiolika lentelių, o grafinė interpretacija ir lentelių specifikacijos pateiktos 4.5 skyriuje.

5.4.4 Teminio tinklo importavimas į duomenų bazę

Sukūrus schema, duomenų bazės lenteles reikia užpildyti duomenimis. Pradžioje turi būti naudojamas sisteminis *liferay.ltm* teminis tinklas, turintis integracijai reikiamą informaciją. Importavimas atliekamas konsolėje panaudojus komandą (patartina savybių ir teminio tinklo failus laikyti tame pačiame kataloge):

```
java net.ontopia.topicmaps.cmdlineutils.rdbms.RDBMSImport db.mysql.props
liferay.ltm. Gaunamas ID (M1)
```

Sėkmingai atlikus operaciją, gaunamas identifikatorius (pvz.: *M1*), kurį reikės nurodyti *Liferay* sistemai.

5.4.5 Teminio tinklo duomenų bazės ryšio užmezgimas su *Liferay* TVS

Ryšys užmezgamas kataloge `/home/${vartotojas}/portal/liferay-portal-6.0.4/tomcat-6.0.26/lib`. Jame rekomenduojama taip pat turėti savybių ir teminio tinklo failą. Svarbiausias katalogo failas – *tm-sources.xml*, saugantis nuorodas į teminių tinklų šaltinius. Šiame faile taip pat nurodomas ryšys su DVBS. Galima konfigūracija:

```
<?xml version="1.0"?>
<repository>
  <source class="net.ontopia.topicmaps.impl.rdbms.RDBMSTopicMapSource">
    <param name="propertyFile" value="db.mysql.props"/>
    <param name="id" value="mysql"/>
    <param name="title" value="MySQL database"/>
    <param name="supportsCreate" value="true"/>
    <param name="supportsDelete" value="true"/>
  </source>
  <source class="net.ontopia.topicmaps.impl.rdbms.RDBMSSingleTopicMapSource">
    <param name="propertyFile" value="db.mysql.props"/>
    <param name="topicMapId" value="M1"/>
    <param name="referenceId" value="liferay.ltm"/>
    <param name="title" value="Liferay"/>
  </source>
</repository>
```

Atlikus pakeitimus, paleidžiamas *Liferay* portalas. Jei pasiruošimo veiksmai atlikti teisingai, sukūrus naują temą *Ontopoly* redaktoriumi *MySQL server* lentelės pasipildo įrašais.

Teminių tinklų saugojimas tampa pastovus ir *liferay.ltm* failą galima pašalinti iš `/home/${vartotojas}/portal/liferay-portal-6.0.4/tomcat-6.0.26/lib` katalogo.

Detaliai išanalizavus *Liferay* turinio valdymo sistemos modernizavimo metodiką ir jos taikymą, pateikiamas eksperimentinis metodikos tyrimas.

6 Eksperimentinis demonstracinės sistemos tyrimas

Skyriuje aprašomas realizuotos *Liferay* TVS modernizavimo metodikos eksperimentinis tyrimas, eiga ir rezultatai. Tyrimu siekiama parodyti *Topic Maps* galimybes sukurtoje programavimo mokymo demonstracinėje sistemoje. Tam analizuojamas sistemos teminis tinklas, jį sudarantys elementų tipai. Kitas tyrimo aspektas išsiaiškinti, ar pakankamai išnaudotos kuriamos programavimo mokymo sistemos galimybės, atsižvelgiant į aprašomus kriterijus. Šioje tyrimo dalyje analizuojama, kokie *Topic Maps* bei *Liferay* portalo portletai panaudoti. Taip pat tiriamos *Liferay* TVS administravimo bei informacijos sistemingo pateikimo vartotojui galimybės. Pabaigoje nagrinėjama parengtos metodikos kokybė, kuri priklauso nuo to, kaip išpildyti metodikos ir demonstracinės sistemos funkciniai ir nefunkciniai reikalavimai.

6.1 Sukurto teminio tinklo apimties įvertinimas

Ontopia ir *Liferay* integracijoje didelę reikšmę teikia teminis tinklas, sudarantis lanksčias galimybes naršyti turinio valdymo sistemoje. Norint ištirti teminį tinklą ir jo apimtį reikia suskaičiuoti programavimo mokymo sistemoje naudojamų *Topic Maps* elementų bei jų tipų skaičių.

Teminiai tinklai yra vieni iš pažangiausių ir sudėtingiausių žinių pristatymo modelių [12]. Programavimo mokymo sistemoje teminis tinklas tvarkomas *Ontopoly* įrankiu. Šis redaktorius sukurtas pagal *ISO/IEC 13250:2003* standartą, tačiau ne visos *Topic Maps* duomenų modelio ypatybės yra palaikomos. Todėl atliekant eksperimentinį tyrimą priimta, kad šių teminio tinklo savybių panaudoti neįmanoma:

- Konteksto (angl. *scope*) taikymas;
- Temų vardų variantų (angl. *variant name*) taikymas;
- Reifikacijos naudojimas (veiksmas kai tema reprezentuoja subjektą iš kito teminio tinklo esamajame tinkle);
- Temų sujungimas (angl. *merging*).

Visgi *Ontopoly* redaktorius, įvertinus nepalaikomas aukščiau išvardintas funkcijas, yra pakankamai išplėtotas produktas. Keturioliktoje lentelėje nagrinėjamos demonstracinėje sistemoje taikytos *Ontopoly* savybės.

14 Lentelė *Ontopoly* tyrimo kriterijai

Nr.	Savybė	Savybės aprašymas	Pritaikyta sistemoje
1.	Asociacijos tipų specifikavimas	Reikalingas, formuojant ryšius tarp tinklo temų. Specifikuojant nurodomas asociacijos tipas, jungiamos temos, vaidmenys, kardinalumai bei papildomi ryšio pavadinimai.	Taip

2.	Temų pasireiškimų tipų specifikavimas	Nurodomi teminio tinklo pasireiškimai specifikuojant: duomenų tipą, kardinalumą.	Taip
3.	Temų asociacijos vaidmenų tipų parinkimas	Reikalingi siekiant korektiškai specifiuoti teminio tinklo asociacijas. Nurodant suteikiamas vaidmens vardas, priskiriama temai.	Taip
4.	Ontopoly administravimo režimo įjungimas	Rėžimas suteikia galimybę prieiti prie <i>Topic Maps</i> metamodelio elementų, uždėti ryšius temų tipams.	Taip
5.	Temų šablonų nustatymas	Ypač svarbus kriterijus (<i>occurrence</i> laukas) nurodomas temų tipams ar egzemplioriams. Pagal šabloną nusakoma koku URL adresu vartotojas bus nukreiptas pasirinkus atitinkamą temą.	Taip

15 lentelėje pateikiamas pagrindinių teminio tinklo tyrimo kriterijų sąrašas. Pateikiami naudoti *Topic Maps* elementų tipai ir jų dažniai.

15 Lentelė Teminio tinklo ontologijos pilnumo tyrimo kriterijai

Nr.	Kriterijus	Kiekis	Kriterijaus paaiškinimas
Teminio tinklo elementai (angl. TAO): 1826			
1.	Temų skaičius	198	Bendras temų skaičius tinkle
2.	Asociacijų skaičius	700	Bendras asociacijų skaičius tinkle
3.	Pasireiškimų skaičius	928	Bendras pasireiškimų skaičius tinkle
Skirtingi temų tipai: 10 (+3 <i>superclass</i> temos)			
4.	Article	17	<i>Liferay</i> sisteminė tema
5.	Community	2	<i>Liferay</i> sisteminė tema
6.	Dalyko tema	11	Programavimo mokymo ontologijos tema
7.	Egzaminas	6	Programavimo mokymo ontologijos tema
8.	Egzamino užduotis	26	Programavimo mokymo ontologijos tema
9.	Egzamino užduoties sprendimo vaizdinė pamoka	20	Programavimo mokymo ontologijos tema
10.	Group	2	<i>Liferay</i> sisteminė tema
11.	Liferay_WebContent	17	<i>Liferay</i> sisteminė tema
12.	Temos vaizdinė pamoka	33	Programavimo mokymo ontologijos tema
13.	User	8	<i>Liferay</i> sisteminė tema
14.	Vaizdinė pamoka	53	Programavimo mokymo ontologijos tema

15.	Workflow state	3	<i>Liferay</i> sisteminė tema
16.	(Asmuo)	0	Temos tipas neišnaudotas
Skirtingi asociacijų tipai: 13			
17.	Aiškina	33	Programavimo mokymo ontologijos ryšys
18.	Aiškinimas	29	Programavimo mokymo ontologijos ryšys
19.	Approved_by	113	<i>Liferay</i> sisteminis ryšys
20.	Contains	120	<i>Liferay</i> sisteminis ryšys
21.	Created by	120	<i>Liferay</i> sisteminis ryšys
22.	Has_workflow_state	120	<i>Liferay</i> sisteminis ryšys
23.	Is-about	5	<i>Liferay</i> sisteminis ryšys
24.	Panašumas	1	Programavimo mokymo ontologijos ryšys
25.	Sudėtis	26	Programavimo mokymo ontologijos ryšys
26.	Superclass/subclass	4	<i>Topic Maps</i> duomenų modelio <i>Superclass/Subclass</i> ryšys
27.	Susiejimas	25	Programavimo mokymo ontologijos ryšys
28.	Topikotipui	96	Nusako duotos temos tipą.
29.	Supertype/subtype	8	<i>Topic Maps</i> duomenų modelio <i>Supertype/Subtype</i> ryšys
Skirtingi pasireiškimų tipai: 14			
30.	Aproved date	113	<i>Liferay</i> straipsnio patvirtinimo data
31.	ArticleID	120	<i>Liferay</i> straipsnio identifikacijos numeris
32.	Create date	120	<i>Liferay</i> straipsnio sukūrimo data
33.	Creator	1	Teminio tinklo kūrėjas
34.	Description	39	<i>Liferay</i> straipsnio aprašymas
35.	Display date	120	<i>Liferay</i> straipsnio atvaizdavimo data
36.	GroupID	2	<i>Liferay</i> grupės identifikacijos nuemris
37.	Modified Date	120	<i>Liferay</i> straipsnio redagavimo data
38.	NuorodaAtsisiusti	53	Pamokos atsisiuntimo nuoroda
39.	NuorodaPerziureti	53	Pamokos peržiūros nuoroda

40.	NuorodaProgramosTekstas	48	Pamokos programos teksto atsiuntimo nuoroda
41.	Structure-id	8	<i>Liferay</i> straipsnio struktūros identifikacijos numeris
42.	URL-template	10	Temos URL šablonas
43.	Version	121	Teminio tinklo versija

Demonstracinėje sistemoje naudota 16 asociacijų vaidmenų tipų. Jų sąrašas pateikiamas 47 paveiksle.



47 pav. Programavimo mokymo sistemos teminio tinklo asociacijų vaidmenų tipai

Taigi atsižvelgus į *Topic Maps* redaktoriaus *Ontopoly* apribojimus, penkioliktos lentelės bei panašaus pobūdžio sistemas (italų operos teminio tinklo programoje panaudoti 7316 elementai ir visi pagrindiniai jų tipai) duomenis, galima teigti, kad su turimais įrankiais teminio tinklo galimybės programavimo mokymo demonstracinėje sistemoje išnaudotos pakankamai.

6.2 Demonstracinės sistemos apimties ir sudėtingumo įvertinimas

Liferay portalas – turinio valdymo sistema, kurios įgūdžių lavinimui reikia laiko ir pastangų. Integravus į *Liferay Ontopia* atsiranda papildoma aibė įgūdžių, kuriuos reikia įsisavinti. Šiuo eksperimentiniu tyrimu siekiama atskleisti, ar pakankamai iliustruotos *Liferay* TVS galimybės, nulemiančios kuriamos sistemos apimtį, išpildymą, kokybę ir sudėtingumą.

Tyrimas atliekamas, įvertinus šiuos aspektus:

- Naudojamų specifinių *Liferay* ir *Ontopia* integracijos metu įdiegtų *Topic Maps* portletų skaičius;
- Naudojamų standartinių *Liferay* portletų skaičius;
- Išnaudotų *Liferay* portalo administravimo galimybių įvairovė;
- Išnaudotų *Liferay* portalo informacijos pateikimo vartotojui galimybių įvairovė.

Ontopia gamintojai, integruojant savo produktą su *Liferay* portalu, pateikė 5 specialius porteltus. Kiekvieno iš jų naudojimas suteikia sistemai papildomo sudėtingumo, o netaikant tam tikrų portletų navigacija po teminių tinklą taptų nepilna. Demonstracinėje sistemoje buvo išnagrinėti ir pritaikyti visi penki portletai, o be to vienas sukurtas savarankiškai „*Tolog-list*“, atvaizduojants temos pasireiškimų reikšmes.

16 lentelėje aprašomas *Liferay Topic Maps* portletų tyrimas. Pateikiamas portleto pavadinimas, paskirtis bei naudojimo statistika. Portletus naudoti sistemoje gali tik prisijungę vartotojai.

16 Lentelė Praplėstos *Liferay* TVS galimybių naudojimo tyrimas

Nr.	Portletas	Portleto paskirtis	Sistemos puslapiai, naudojančys portletą	Panaudojimų skaičius
1.	Related Topics	Reikalingas parodyti susijusias temas	Tema, Mokymo tema, Vaizdinė pamoka	5
2.	Dynamic Web Content	Skirtas atvaizduoti prie temos prikabinatą <i>Liferay</i> turinį	Mokymo tema, Vaizdinė pamoka	2
3.	Article-list	Pateikia <i>Liferay</i> straipsnių sąrašą	Tema	1
4.	Tolog-list	Atvaizduoja duotos temos pasireiškimą (-us)	Vaizdinė pamoka	1
5.	YahooTree	Atvaizduoja temų ir jų egzempliorių išdėstymą stulpeliais	Pradinis	1
6.	TopicName	Parodo temos vardą (-us)	Tema	1

17 lentelėje aprašomas *Liferay* tradicinių portletų tyrimas. Pateikiamas portleto pavadinimas, paskirtis ir taikymas demonstracinėje sistemoje. Kiekvieną iš portletų į sistemą galima įsikelti tiek per pagrindinį meniu, tiek per administravimo langą (angl. *Control Panel*).

17 Lentelė Liferay tradicinių portretų naudojimo tytimas

Nr.	Portletas	Portleto paskirtis	Panaudojimų skaičius
1	Web Content Display	<i>Liferay</i> TVS informacijos saugojimo ir atvaizdavimo objektas	109
2.	Message Boards	Forumų diegimas	1
3.	Blogs	Internetinių dienraščių kūrimas	1
4.	Polls	Apklausų ruošimas	1
5.	Tags	Turinio žymėjimas	1
6.	Category	Turinio skirstymas į kategorijas	1
7.	Search	Portalo paieška	1
8.	Document library	<i>Liferay</i> dokumentų saugykla	1
9.	RSS	Puslapių prenumeravimui	1
10.	Recent activities	Neseniai vykusių įvykių pateikimui	1
11.	Calendar	<i>Liferay</i> kalendorius	1

18 lentelėje aprašomas *Liferay* administravimo galimybių išnaudojimas demonstracinėje sistemoje. Pateikiamas naudojamos funkcijos pavadinimas, paskirtis ir taikymo principai.

18 Lentelė Liferay portalo administravimo galimybių tyrimas

Nr.	Kriterijus	Kriterijaus paaiškinimas	Panaudojimų skaičius
1.	Šablonų nustatymas	<i>Liferay</i> šablono (<i>velocity</i>) parinkimas	6
2.	Struktūrų kūrimas	<i>Liferay</i> straipsnio laukų parinkimas	6
3.	Šablonų pritaikymas struktūroms	Reikalinga pagal nutylėjimą	6
4.	Struktūrų pritaikymas <i>Liferay</i> straipsniams	Sukuriamas <i>Topic Maps</i> temos tipas pagal struktūros pavadinimą	109
5.	<i>Liferay</i> bendruomenių tvarkymas	Sukurti, administruoti bendruomenę (kaupiti turinį bendruomenėje)	Panaudota. Skaičius keičiasi priklausomai nuo poreikių
6.	<i>Liferay</i> organizacijų tvarkymas	Sukurti, administruoti organizaciją (priskirti bendruomenę, vartotojus)	Panaudota. Skaičius keičiasi priklausomai nuo poreikių
7.	<i>Liferay</i> vartotojų administravimas	Suteikti vartotojams statusą	Panaudota. Skaičius keičiasi priklausomai nuo poreikių
8.	<i>Liferay</i> vartotojų teisių administravimas	Nustatyti prie kokios informacijos gali prieiti atitinkamą statusą turinti vartotojas	Panaudota. Skaičius keičiasi priklausomai nuo poreikių

19 lentelėje aprašomas informacijos pateikimo vartotojui *Liferay* portale galimybių tyrimas. Galimybės susideda iš portalo logotipo uždėjimo, puslapių savybių parinkimo, portletų papildomų savybių konfigūravimo.

19 Lentelė Sistemingo informacijos pateikimo vartotojui *Liferay* portale tyrimas

Nr.	Kriterijus	Kriterijaus paaiškinimas	Panaudota sistemoje
1.	Portletų išdėstymas puslapiuose	Išsirinkti portletų įkėlimo meniu ir nutempti pele į reikiamą puslapio vietą	Taip
2.	Puslapių struktūros (<i>layout</i>) parinkimas	Nustatyti kaip bus rodoma informacija puslapyje (1 stulpelis per lapą, 1 stulpelis kairėje, 2 dešinėje, laisva forma ir pan.)	Taip
3.	Portalo logotipo nustatymas	Parinkti portalą pristatantį paveikslėlį	Taip
4.	Puslapių savybių nustatymas	Parinkti, kurie puslapiai pagal nutylėjimą nepasiekiami iš pradinio meniu (integracijos puslapiai)	Taip
5.	Portletų išvaizdos konfigūravimas	Nustatyti portletų šrifto dydį, parinkti stilius, spalvas.	Taip
6.	Portletų papildomų savybių konfigūravimas	Suteikti prieigos teises, konfigūruoti specifinėmis <i>tolog</i> užklausomis	Taip

Sistemos kokybę nusako naudojamų specifinių *Liferay Topic Maps* portletų skaičius. Teminių tinklų baziniam funkcionalumui (integraliam naršymui po turinį) pademonstruoti būtina panaudoti bent penkis (*Related Topics, Dynamic Web Content, Article-list, YahooTree, TopicName*) portletus. Kadangi programavimo mokymo sistemoje išnaudoti visi šie portletai bei papildomas „*Tolog-list*“ temos ištekliams išgauti, galima teigti, kad sistema yra reikiamos apimties ir sudėtingumo. Pateikus *Liferay* taikytų standartinių portletų, išnaudotų administravimo ir informacijos pateikimo vartotojui savybių statistinius duomenis priimta, kad *Liferay* portalo galimybės iliustruotos pakankamai.

6.3 Sukurtos demonstracinės sistemos įvertinimas

Trečioje eksperimento dalyje pateikiamas reikalavimų išpildymo tyrimas. Perkeliame 20.1 ir 20.2 paveiksle specifiškai demonstracinės sistemos ir metodikos reikalavimai. Dvidešimtoje lentelėje aprašomas kiekvienas reikalavimas bei nurodomas tą reikalavimą iliustruojantis elementas: tai gali būti darbe nagrinėtas skyrius, paveikslas ar lentelė. Pagal tai nustatoma, ar reikalavimas yra išpildytas.

Tokio tipo eksperimentinis tyrimas padės nustatyti, ar parengta turinio valdymo sistemos modernizavimo metodika yra kokybiška.

20 Lentelė Sistemos įvertinimo tyrimas

Nr.	Reikalavimo pavadinimas	Aprašymas	Iliustruojantis elementas	Išpildyta
1.	Naviguoti po teminį tinklą	Suteikta galimybė naršyti po <i>Liferay</i> sistemoje sukurtą turinį	41, 42, 43, 44 paveikslai	Taip
2.	Rasti vaizdinės medžiagos išteklius	Pasirinkti temą su nuoroda į išteklių	44 paveikslas	Taip
3.	Peržiūrėti straipsnio susijusias temas	Matyti puslapio dešinėje siūlomas pasirinkti temas	42, 43, 44 paveikslai	Taip
4.	Pritaikyti įdiegtus Topic Maps portletus puslapiuose	Išnaudoti Topic Maps portletus demonstracinėje sistemoje	41, 42, 43, 44 paveikslai	Taip
5.	Kurti teminio tinklo ontologiją	Sukurti reikiamas temas, asociacijas, pasireiškimus.	39,40 paveikslai	Taip
6.	Susieti Topic Maps temas su TVS straipsniais	Liferay straipsniams parinkti struktūrą	38 paveikslas	Taip
7.	Konfigūruoti Topic Maps portletus	Nurodyti portletams įvairias savybės, filtrus	45,46 paveikslai 12 lentelė	Taip
8.	Atvaizduoti norimą teminio tinklo dalį	Programuoti tolog užklausas	12 lentelė	Taip
9.	Nagrinėti standartines TVS funkcija	Išsiaiškinti, kokie įprastiniai <i>Liferay</i> portretai gali būti taikomi sistemoje	5 skyriaus paveikslai	Taip
10.	Peržiūrėti forumo naujienas	Analizuoti diskusijų forume parašytas naujienas	42 paveikslas kortelė „Forumas“	Taip
11.	Skaityti internetinį dienraštį	Analizuoti tinklaraštyje parašytus skelbimus	42 paveikslas kortelė „Programavimo pamokų dienraštis“	Taip
12.	Diskutuoti forume	Dalyvauti diskusijų forume (atsakyti į žinutes ir pan.)	42 paveikslas kortelė „Forumas“	Taip
13.	Administruoti privačių puslapių zoną	Nustatyti elementus ir parinkti portletus, kuriuos matys vartotojai savo privačioj zonoj	Iliustruojama prisijungus prie sistemos ir pasirinkus valdymo skydą	Taip
14.	Dalyvauti apklausoje	Atsakyti į sistemos administratoriaus suformuluotus klausimus	42 paveikslas kortelė „Apklausa“	Taip
15.	Komentuoti dienraščio straipsnius	Atsakyti į parašytą skelbimą, jį vertinti balais	42 paveikslas kortelė „Programavimo pamokų dienraštis“	Taip
16.	Administruoti vartotojų teises	Nustatyti kokius veiksmus gali atlikti atitinkamas portalo vartotojas	Iliustruojama prisijungus prie sistemos ir pasirinkus valdymo skydą	Taip
17.	Priskirti vartotojus bendruomenėms	Nustatyti, kokiai portalo bendruomenei priklauso vartotojas	Iliustruojama prisijungus prie sistemos ir pasirinkus valdymo skydą	Taip

18.	Kurti dienraščių ir forumo straipsnius	Pateikti sistemos lankytojams aktualią informaciją tinklaraštyje	42 paveikslas kortelė „Programavimo pamokų dienraštis“	Taip
19.	Tvarkyti darbų kalendorių	Priskirti atitinkamoms dienoms svarbius įvykius	Iliustruojama prisijungus prie sistemos ir pasirinkus valdymo skydą	Taip
20.	Turėti bazinę <i>Liferay</i> TVS naudojimosi patirtį	Gebėti orientuotis sistemoje, konfigūruoti portletus ir jais naudotis	41,45 paveikslai, 2.2.3, 5.1 skyriai	Taip
21.	Gebėti administruoti portalą	Susipažinti su <i>Liferay</i> valdymo skydu, vartotojų teisių mechanizmu, bendruomenėmis	18 lentelė	Taip
22.	Naudotis specifiniais integracijos portletais	Išdėlioti teminio tinklo portletus, kad išeitų naviguoti po informaciją	41,42,43,44 paveikslai 16 lentelė	Taip
23.	Naudotis tradiciniais <i>Liferay</i> portletais	Sukurti sistemoje diskusijų forumo, apklausų, tinklaraščių modulius	17 lentelė	Taip
24.	Pateikti informaciją vartotojui	Atvaizduoti informaciją pagal puslapių struktūrą, kad vartotojas suprastų, kur temos turinys, o kur navigacija	19 lentelė	Taip
25.	Susipažinti su <i>Ontopia</i> produkto pagrindais	Gebėti naudoti <i>Ontopoly</i> redaktoriumi, tolog užklausų varikliu	39,40 paveikslai 2.2.1.11 skyrius	Taip
26.	Gebėti dirbti su <i>Ontopoly</i>	Kurti teminio tinklo ontologijos elementus: asociacijas, pasireiškimus ir t.t	39,40 paveikslai	Taip
27.	Susipažinti su logine užklausų kalba <i>tolog</i>	Gebėti atvaizduoti reikiamą teminio tinklo informaciją portale	12, 13 lentelės	Taip
28.	Sukurti demonstracinę programavimo mokymo sistemą	Pateikti TVS modernizavimo metodiką iliustruojantį sistemos prototipą	41,42,43,44 paveikslai	Taip
29.	Įvertinti sistemos apimtį ir sudėtingumą	Pateikti ir įvertinti apimtį ir sudėtingumą apibūdinančius kriterijus	6.2 skyrius	Taip

20 lentelė įrodo, kad demonstracinei sistemai kelti reikalavimai yra išpildyti. Ši sistema iliustruoja TVS modernizavimo metodiką, todėl galima teigti, kad darbe užsibrėžtas tiklas yra įvykdytas.

7 Išvados

1. Išnagrinėjus teminius tinklus ir jų taikymo sritis nuspręsta parengti TVS modernizavimo metodiką, kurią iliustruotų lietuviška demonstracinė sistema, nes tokio pobūdžio sistemų Lietuvoje beveik nėra. Reikalingi praktiniai pavyzdžiai, padėsiantys populiarinti teminių tinklų taikymą.
2. Išanalizavus panašaus pobūdžio esamas teminių tinklų sistemas ir įrankius, prieita prie išvados, kad daugeliui sukurtų įrankių iki reikiamo funkcionalumo trūksta papildomo programavimo, todėl tiesiausias kelias į tokių portalų kūrimą – naudoti turinio valdymo sistemą su teminių tinklų palaikymu. Šių nekomercinių sprendimų egzistuoja nedaug, o iš analizuotų variantų palankiausia naudoti norvegų *Java* priemonėmis sukurtą teminių tinklų valdymo produktą *Ontopia* su *Liferay* portalu, nes šis sprendimas teminių tinklų teikiamas lanksčias informacijos organizavimo ir naudingas TVS turinio formatavimo bei pateikimo galimybes į vieną efektyvų derinį.
3. Pateikta *Liferay* TVS modernizavimo metodika išaiškina, kaip vartotojams, naudojant tik atviro kodo įrankius, kurti teminių tinklų portalus. Sistemos prototipas leidžia išbandyti naršymą programavimo mokymo teminiu tinklu turinio valdymo sistemoje *Liferay*, grafinės sąsajos pagalba puslapiuose pele išdėstant norimus portletus. Veikiantis portalas nėra iki galo išbaigtas, o tolimesnės darbų kryptys būtų orientuotos į naujų portletų (semantinės paieškos, navigacijos kelio, temų išdėstymo medžiu) kūrimą.
4. Atlikus eksperimentinį tyrimą nustatyta, kad sistemos apimties ir sudėtingumo įvertinimui keltas reikalavimas yra pakankamai išpildytas, nes:
 - Sukurtas 1826 elementų dydžio teminis tinklas, kuriame pritaikyti temų, asociacijų, pasireiškimų, asociacijų vaidmenų elementų tipai;
 - Sistemoje panaudoti visi penki modernizuotos *Liferay* TVS teminių tinklų portletai ir vienas sukurtas papildomai;
 - Išbandytos *Liferay* portalo administravimo galimybės, sukuriant skirtingus sistemos lankytojų tipus, bendruomenę, vartotojų teisių ir sistemingo informacijos valdymo mechanizmus;
5. Atlikus demonstracinės sistemos eksperimentinį tyrimą nustatyta, kad kiekvienas parengtai metodikai ir sistemai iškeltas reikalavimas turi jo išpildymą iliustruojantį elementą. Taigi TVS modernizavimo metodika paruošta pagal reikalavimus, o darbe užsibrėžtas tikslas yra pasiektas – populiarinti teminių tinklų pagrindu veikiančias sistemas Lietuvoje.

8 Literatūra

- [1]. **Jack Park, Sam Hunting XML Topic Maps:** Creating and Using Topic Maps for the Web 2003.
- [2]. **Topic Maps, RDF, DAML, OIL**
[žiūrėta 2010-01-21] Prieiga per Internetą: <http://www.ontopia.net/topicmaps/materials/tmrdfoidaml.html>
- [3]. **Ontopia firmos produktai ir Topic Maps**
[žiūrėta 2010-01-20] Prieiga per Internetą: <http://www.ontopia.net/topicmaps/index.html>
- [4]. **Susipažinimas su Topic Maps paradigma**
[žiūrėta 2010-01-10] Prieiga per Internetą: http://www.slideshare.net/LEler/tutorial-introduction-to-topic-maps?src=related_normal&rel=2553829
- [5]. **Topic Maps portalai**
[žiūrėta 2010-01-20] Prieiga per Internetą <http://www.slideshare.net/LEler/topic-maps-portals>
- [6]. **Sal Mangano XSLT CookBook** 2002.
- [7]. **NetworkedPlanet:** White Paper: Topic Maps in Web site architecture 2005.
- [8]. **Thomas Neidhart, Rani Pinchuk, and Bernard Valentin:** Semantic Integration of Relational Data Sources With Topic Maps
- [9]. **TMRA oficiali svetainė:**
[žiūrėta 2010-01-20] Prieiga per Internetą: <http://www.topicmapslab.de/>
- [10]. **Topic Maps ir miglotoji inžinerija:**
[žiūrėta 2010-01-19] Prieiga per Internetą: <http://www.semantic-web.at/1.36.resource.206.roy-lachica-x22-topic-maps-is-very-applicable-in-collaborative-solutions-because-they-rese.htm>
- [11]. **TM vaidmuo turinio valdyme:** <http://www.ontopia.net/topicmaps/materials/itms.html>
- [12]. **Teminių tinklų duomenų modelis** <http://www.isotopicmaps.org/sam/>
- [13]. **Liferay architektūros psl.** http://www.liferay.com/community/forums/-/message_boards/message/1210342
- [14]. **Liferay TVS puslapis** <http://www.liferay.com/>
- [15]. **Water and Stones tyrimai, įvertinantys Liferay TVS** <http://www.liferay.com/web/guest/about-us/awards>
- [16]. **Ontopia Topic Maps engine puslapis:** <http://www.ontopia.net>
- [17]. **tolog gidas:** <http://www.ontopia.net/doc/current/doc/query/tutorial.html>
- [18]. **tolog predikatai:** <http://www.ontopia.net/doc/current/doc/query/predicate-reference.html>
- [19]. **TM TAO** <http://www.ontopia.net/topicmaps/materials/tao.html>
- [20]. **TM įrankių sąrašas:** <http://www.topicmap.com/tools>
- [21]. **Principinė Ontopia architektūros schema:** <http://www.ontopia.net/section.jsp?id=ontopia-the-product>
- [22]. **Programavimo mokymo internetinė svetainė**
http://programavimopamokos.net/ikeltukai/Magistrinis/vienmatis_masyvas.html
- [23]. **Garshol L. M .** How to build Liferay Integration . 2010
<http://code.google.com/p/ontopia/wiki/HowToBuildLiferayIntegration>
- [24]. **Garshol L. M. Moore G. .** Topic Maps — Data Model . 2008
<http://www.isotopicmaps.org/sam/sam-model/>
- [25]. **Garshol L. M .** Ontopia/Liferay Integration . 2010 <http://ontopia.wordpress.com/tag/liferay/>
- [26]. **Ontopia .** The RDBMS Backend Connector . 2010
<http://www.ontopia.net/doc/current/doc/rdbms/install.html>
- [27]. **Java portlet-api** <http://www.bluesunrise.com/portlet-api/index.html>