



KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

INFORMACIJOS SISTEMŲ KATEDRA

Milda Mickevičiūtė

**Veiklos žodyno ir veiklos taisyklių formavimo  
automatizuotu būdu iš veiklos procesų modelio  
metodas**

Magistro darbas

Darbo vadovas      prof. dr. Rimantas Butleris

Darbo konsultantė      dokt. Lina Tutkutė

Kaunas, 2012



KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA

Milda Mickevičiūtė

**Veiklos žodyno ir veiklos taisyklių formavimo  
automatizuotu būdu iš veiklos procesų modelio  
metodas**

Magistro darbas

Recenzentas

doc. dr. Dalius Makackas  
2012-05-28

Vadovas

prof. dr. Rimantas Butleris  
2012-05-28

Konsultantė

dokt. Lina Tutkutė  
2012-05-28

Atliko

IFME-0/4 gr. stud.  
Milda Mickevičiūtė  
2012-05-28

Kaunas, 2012

## Summary

There are many organizations that want to represent their business processes (BP) using modeling notations, such as BPMN, UML and Workflow but usually the analyst cannot identify all necessary constraints of BP. Therefore responsibilities for the representatives of the organization cannot be correctly specified which leads to poor quality of work. In order to avoid such problems business process diagrams (BPD) must meet the real world processes.

The research area of thesis covers various BP and business rules (BR) modeling notations, the intersection of BP and BR (the relevance and formation of BP and BR, applicability and integration of BR, the extraction of BR and business vocabulary (BV) from BPD). The main issue is the automated extraction of BV and BR from BPD which is used not only to compare business model to real organization processes but also it is useful to have BV in organization for more effective collaboration.

The main tasks to solve the identified issue are the analysis of BP with their constraints, analysis of the structures of BR and their applicability in BP, the analysis of structures of BR in BP and finally the formation of the algorithm for the automated result's extraction, creation and evaluation of prototype.

The analysis of existing methods revealed that there is no such method that allows user to form BP and BR effectively integrating all necessary aspects. BPMN and SBVR standards are chosen as main knowledge resources because of their ability to express information in natural language, intuitive modeling style, and compatibility between them.

The final result of the research is the designed prototype of the method which represents the business vocabulary which consists of words' list with their name, identified element, word type, also facts list and BR list. The user can evaluate, manage and change all necessary aspects of modeling of the real world process using business process model, business vocabulary and business rules. Such integration minimize user's work time, ability to make mistakes, identify business vocabulary in the departments of the organization, which leads to successful growth.

## **Terminų ir santrumpų sąrašas**

VP – Veiklos Procesai  
BP – Business Process  
VPD – Veiklos Procesų Diagrama  
BPD – Business Process Diagram  
VT – Veiklos Taisyklė  
BR – Business Rule  
VŽ – Veiklos Žodynas  
BV – Business Vocabulary  
BPMN2 – The Business Process Modeling and Notation  
UML – The Unified Modeling Language  
Workflow – darbų sekų diagrama  
SBVR – Semantics of Business Vocabulary and Business Rules  
SRML – Simple Rule Markup Language  
PRR – Production Rule Representation  
XML - Extensible Markup Language  
MDA – The Model-Driven Architecture  
CIM – Computation-Independent Model  
PIM – Platform-Independent Model  
PSM – Platform-Specific Models  
BPM - Business process management  
EPC – Event-driven Process Chains  
RADs – Role-Activity Diagrams  
BPMI – Business Process Management Initiative  
BPML – Business Process Modeling Language  
WfMC – The Workflow Management Coalition  
DB – Duomenų bazė  
OCL – Object Constraint Language  
OMG – Object Management Group  
SOA – Service Oriented Architecture  
MOF – The Meta-Object Facility  
FT – Fakto Tipas

# Turinys

<b>SUMMARY .....</b>	<b>3</b>
<b>TERMINŲ IR SANTRUMPŲ SĄRAŠAS .....</b>	<b>4</b>
<b>1 ĮVADAS .....</b>	<b>10</b>
<b>2 VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ MODELIO METODŲ ANALIZĖ ....</b>	<b>12</b>
2.1 ANALIZĖS TIKSLAS .....	12
2.2 ANALIZĖS METODAI .....	12
2.3 VEIKLOS PROCESŲ ANALIZĖ .....	13
2.3.1 Veiklos procesų valdymas .....	14
2.3.2 Veiklos proceso modeliavimas .....	14
2.3.3 Veiklos proceso gyvavimo ciklas .....	15
2.3.4 Modeliais grįsta architektūra .....	16
2.4 VEIKLOS PROCESŲ MODELIAVIMO STANDARTAI .....	18
2.4.1 Veiklos procesų modeliavimo notacija (BPMN) .....	18
2.4.2 Unifikuota modeliavimo kalba .....	20
2.4.3 Darbų sekos modeliavimo kalba .....	23
2.4.4 Veiklos procesų modeliavimo standartų palyginimas .....	26
2.5 VEIKLOS ŽODYNO IR TAISYKLIŲ ANALIZĖ .....	27
2.5.1 Veiklos žodynas .....	27
2.5.2 Veiklos taisyklės .....	27
2.6 VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ STANDARTAI .....	30
2.6.1 Veiklos žodyno ir veiklos taisyklių semantika (SBVR) .....	30
2.6.2 Bendrinė taisyklių žymėjimo kalba (SRML) .....	32
2.6.3 Našumo taisyklių vaizdavimo standartas (PRR) .....	34
2.6.4 Veiklos taisyklių standartų palyginimas .....	37
2.7 VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ MODELIO METODAI .....	38
2.7.1 UML veiklos diagramos elementų žymėjimas SBVR taisyklėse .....	39
2.7.2 Veiklos taisyklių ir veiklos procesų realizavimas, naudojant SOA .....	41
2.8 ANALIZĖS IŠVADOS .....	42
<b>3 VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS METODAS .....</b>	<b>44</b>
3.1 VEIKLOS ŽODYNO AUTOMATIZUOTO IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS ALGORITMAS .....	44
3.1.1 I. Sudaryti veiklos procesų diagramą (VPD) .....	45
3.1.2 II. Automatizuotu būdu formuoti VŽ .....	45
3.1.3 III. Rankiniu būdu sutvarkyti VŽ .....	51
3.2 VEIKLOS TAISYKLIŲ AUTOMATIZUOTO IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS ALGORITMAS .....	52
3.2.1 I. Automatizuotu būdu formuoti VŽ .....	53
3.2.2 II. Automatizuotu būdu formuoti veiklos elementų jungimo struktūras .....	53
3.3 VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ STRUKTŪRŲ IŠGAVIMAS IŠ VEIKLOS PROCESŲ ELEMENTŲ .....	57
3.3.1 VŽ ir VT išgavimas iš srautų objektų elementų .....	57
3.3.2 VŽ ir VT išgavimas iš duomenų elementų .....	72
3.4 AUTOMATIZUOTO VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS ALGORITMO SUDARYMO REZULTATAI .....	74
<b>4 APRIBOJIMAI VEIKLOS PROCESŲ DIAGRAMOS ELEMENTŲ PAVADINIMŲ UŽRAŠYMU IŠGAVIMUI .....</b>	<b>76</b>
4.1 BASEINO (ANGL. POOL) IR JUOSTŲ (ANGL. LANE) ELEMENTAI .....	76
4.2 VEIKLOS (ANGL. ACTIVITY) ELEMENTAI .....	76
4.3 DUOMENIMIS GRĮSTI VARTAI (ANGL. DATA – BASED GATEWAY) .....	76
4.4 SEKOS SRAUTŲ (ANGL. SEQUENCE FLOW) ELEMENTAI .....	78
4.5 ĮVYKIŲ (ANGL. EVENT) ELEMENTAI .....	78
4.6 DUOMENŲ (ANGL. DATA) ELEMENTAI .....	79
4.7 KITI SVARBŪS APRIBOJIMAI .....	80
<b>5 AUTOMATIZUOTO VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS ALGORITMO KŪRIMAS, TESTAVIMAS IR REZULTATAI .....</b>	<b>81</b>

5.1.	VARTOTOJŲ ANALIZĖ .....	81
5.2.	REIKALAVIMŲ AUTOMATIZUOTAM VEIKLOS ŽODYNO IR VEIKLOS TAISYKLIŲ IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS ALGORITMUI SPECIFIKACIJA .....	82
5.2.1.	<i>Veiklos modeliai</i> .....	82
5.2.2.	<i>Reikalavimų modeliai</i> .....	83
5.2.3.	<i>Dalykinės srities esybių klasių modelis</i> .....	85
5.2.4.	<i>Realizacijos modelis. Komponentų diagrama</i> .....	86
5.3.	ALGORITMO TESTAVIMO ETAPAI .....	87
5.4.	PROTOTIPO TESTAVIMAS IR REZULTATŲ ANALIZĖ .....	89
5.4.1.	<i>Duomenų nuskaitymas</i> .....	89
5.4.2.	<i>Duomenų atvaizdavimas</i> .....	90
5.4.3.	<i>Veiklos žodyno testavimas</i> .....	93
5.4.4.	<i>Prototipo vertinimas kiekybiniais ir kokybiniais aspektais</i> .....	94
5.5.	PROTOTIPO KŪRIMO IR TESTAVIMO REZULTATAI.....	103
<b>6</b>	<b>IŠVADOS</b> .....	<b>104</b>
<b>7</b>	<b>LITERATŪRA</b> .....	<b>106</b>
<b>8</b>	<b>PRIEDAI</b> .....	<b>108</b>
8.1	1 PRIEDAS. BPMN ELEMENTŲ APRAŠYMAS .....	108
8.2	2 PRIEDAS. UML ELEMENTŲ APRAŠYMAS.....	109
8.3	3 PRIEDAS. WORKFLOW ELEMENTŲ APRAŠYMAS.....	112
8.4	4 PRIEDAS. KOMPIUTERIZUOJAMŲ PANAUDOJIMO ATVEJŲ SPECIFIKACIJOS .....	113
8.5	5 PRIEDAS. DALYKINĖS SRITIES ESYBIŲ KLASIŲ MODELIO SPECIFIKACIJA .....	116
8.6	6 PRIEDAS. VEIKLOS „KLIENTO IDENTIFIKAVIMAS“ PROCESO TESTAVIMO ATVEJIS .....	118
8.7	7 PRIEDAS. VEIKLOS „SUTARTIES KEITIMAS, NUTRAUKIMAS“ PROCESO TESTAVIMO ATVEJIS .....	119
8.8	8 PRIEDAS. VEIKLOS „PRANEŠIMO SIUNTIMAS DĖL MOKĖJIMO TERMINŲ ATIDĖJIMO“ PROCESO TESTAVIMO ATVEJIS .....	120
8.9	9 PRIEDAS. STRAIPSNIS XVII TARPUNIVERSITETINĖJE MAGISTRANTŲ IR DOKTORANTŲ MOKSLINĖJE KONFERENCIJOJE „INFORMACINĖS TECHNOLOGIJOS 2012“ .....	121

## Paveikslų sąrašas

2.1 pav. BPM gyvavimo ciklas.....	15
2.2 pav. Bendra MDA schema .....	17
2.3 pav. Veiklos procesų ir veiklos taisyklių modeliavimo kalbų pasiskirstymas .....	18
2.4 pav. BPMN diagramos elementai.....	19
2.5 pav. Prekės apmokėjimo veiklos proceso pavyzdys BPMN diagrama .....	20
2.6 pav. UML veiklos diagramos pagrindiniai elementai .....	22
2.7 pav. Užsakymo pristatymo ir apmokėjimo veiklų atvaizdavimo pavyzdys .....	22
2.8 pav. Trys darbų sekų diagramos dimensijos.....	23
2.9 pav. Darbų sekų diagramos pagrindiniai elementai.....	25
2.10 pav. Veiklos taisyklių kalbų evoliucija.....	29
2.11 pav. SBVR taisyklės loginė formuluotė .....	31
2.12 pav. Mokesčių apskaičiavimo priežiūros teiginio pateikimas, naudojantis taisyklėmis .....	42
2.13 pav. Mokesčių apskaičiavimo priežiūros teiginio pateikimas proceso forma .....	42
3.1 pav. Pagrindiniai VŽ išgavimo iš VP modelio algoritmo žingsniai .....	44
3.2 pav. Pasirinkti veiklos procesų diagramą veiklos diagrama (1 – as žingsnis).....	45
3.3 pav. Formuoti VŽ veiklos diagrama (2 – as žingsnis).....	47
3.4 pav. Veiklos „II.I. Atrinkti terminus” 3.3 paveiksle algoritmo diagrama .....	48
3.5 pav. Veiklos „II.II. Nustatyti tipus veiklos (angl. Activity) ir įvykio (angl. Event) elementams” 3.3 paveiksle algoritmo diagrama .....	48
3.6 pav. Veiklos „II.III. Nustatyti tipus vartų (angl. Gateway) elementams” 3.3 paveiksle algoritmo diagrama .....	49
3.7 pav. Veiklos „II.IV. Nustatyti tipus baseino (angl. Pool) ir juostos (angl. Lane) elementams” 3.3 paveiksle algoritmo diagrama .....	50
3.8 pav. Veiklos „II.V. Nustatyti tipus duomenų objektų (angl. Data Object), vartų (angl. Gateway) ir įvykio (angl. Event) elementams” 3.3 paveiksle algoritmo diagrama .....	50
3.9 pav. VŽ redagavimo rankiniu būdu veiklos diagrama (3 – ias žingsnis) .....	51
3.10 pav. Pagrindiniai veiklos taisyklių išgavimo iš VP modelio sudarymo veiklos diagrama (žingsniai) .....	52
3.11 pav. Detalizuojamos veiklos „Nustatyti įvykio (angl. Event) elemento struktūras“ veiklos diagrama.....	54
3.12 pav. Detalizuojamos veiklos „Nustatyti veiklos (angl. Activity) elemento struktūras“ veiklos diagrama.....	55
3.13 pav. Detalizuojamos veiklos „Nustatyti veiklos su įvykiu (angl. Boundary Activity) elemento struktūras“ veiklos diagrama.....	56
3.14 pav. Detalizuojamos veiklos „Nustatyti vartų (angl. Gateway) elemento struktūras“ veiklos diagrama.....	56
3.15 pav. Veiklos klasės metamodelis.....	58
3.16 pav. Situacijos nr. 1 sprendimas (liet.) .....	58
3.17 pav. Situacijos nr. 1 sprendimas (angl.).....	58
3.18 pav. Vartų (angl. Gateway) klasės metamodelis .....	60
3.19 pav. Situacijos nr. 2 sprendimas (liet.) .....	61
3.20 pav. Situacijos nr. 2 sprendimas (angl.).....	61
3.21 pav. Situacijos nr. 3 sprendimas (liet.) .....	62
3.22 pav. Situacijos nr. 3 sprendimas (angl.).....	62
3.23 pav. Situacijos nr. 4 sprendimas (liet.) .....	63
3.24 pav. Situacijos nr. 4 sprendimas (angl.).....	63
3.25 pav. Situacijos nr. 5 sprendimas (liet.) .....	64
3.26 pav. Situacijos nr. 5 sprendimas (angl.).....	64
3.27 pav. Įvykio (angl. Event) klasės metamodelis.....	66
3.28 pav. Situacijos nr. 6 sprendimas (liet.) .....	67

3.29 pav. Situacijos nr. 6 sprendimas (angl.).....	67
3.30 pav. Situacijos nr. 7 sprendimas (liet.) .....	70
3.31 pav. Situacijos nr. 7 sprendimas (angl.).....	70
3.32 pav. Situacijos nr. 8 sprendimas (liet.) .....	72
3.33 pav. Situacijos nr. 8 sprendimas (angl.).....	72
5.1 pav. Veiklos panaudojimo atvejų modelis.....	83
5.2 pav. Tiriamąjo darbo tikslų modelis.....	83
5.3 pav. Kompiuterizuojamų panaudojimo atvejų modelis.....	84
5.4 pav. Veiklos proceso diagrama.....	84
5.5 pav. Duomenų bazės klasių modelis .....	85
5.6 pav. Komponentų modelis.....	87
5.7 pav. Pagrindinis vartotojo sąsajos langas .....	89
5.8 pav. Failo pasirinkimo forma .....	90
5.9 pav. Atliktų veiksmų sąrašas ir rezultatai.....	90
5.10 pav. Nuskaitytų duomenų pateikimo vartotojams langas.....	91
5.11 pav. Pavienių žodžių pateikimo langas.....	92
5.12 pav. Faktų sąrašo pateikimo langas .....	92
5.13 pav. Veiklos žodyno redagavimo langas .....	93
5.14 pav. Veiklos taisyklių pateikimo langas .....	93
8.1 pav. Grafinis banko veiklos proceso „Kliento identifikavimas“ pateikimas.....	118
8.2 pav. Grafinis banko veiklos proceso „Sutarties keitimas, nutraukimas“ pateikimas .....	119
8.3 pav. Grafinis banko veiklos proceso „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ pateikimas .....	120

### **Lentelių sąrašas**

2.1 lentelė. Veiklos procesų modeliavimo standartų palyginimo lentelė.....	26
2.2 lentelė. Veiklos taisyklių standartų palyginimas .....	37
2.3 lentelė. Veiklos taisyklių pagal pateiktą pavyzdį priskyrimas .....	40
2.4 lentelė. Fakto tipai ir jai priskiriami veiksmai.....	40
3.1 lentelė. Situacijos nr. 1 veiklos žodynas.....	59
3.2 lentelė. Situacijos nr. 1 veiklos taisyklių sąrašas.....	59
3.3 lentelė. Situacijos nr. 2 veiklos žodynas.....	61
3.4 lentelė. Situacijos nr. 2 veiklos taisyklių sąrašas.....	61
3.5 lentelė. Situacijos nr. 3 veiklos žodynas.....	62
3.6 lentelė. Situacijos nr. 3 veiklos taisyklių sąrašas.....	62
3.7 lentelė. Situacijos nr. 4 veiklos žodynas.....	63
3.8 lentelė. Situacijos nr. 4 veiklos taisyklių sąrašas.....	63
3.9 lentelė. Situacijos nr. 5 veiklos žodynas.....	64
3.10 lentelė. Situacijos nr. 5 veiklos taisyklių sąrašas.....	65
3.11 lentelė. Situacijos nr. 6 sprendimo veiklos žodynas.....	67
3.12 lentelė. Situacijos nr. 6 sprendimo veiklos taisyklių sąrašas.....	68
3.13 lentelė. Situacijos nr. 7 sprendimo veiklos žodynas.....	70
3.14 lentelė. Situacijos nr. 7 sprendimo veiklos taisyklių sąrašas.....	71
3.15 lentelė. Situacijos nr. 8 sprendimo veiklos žodynas.....	72
3.16 lentelė. Situacijos nr. 8 sprendimo veiklos taisyklių sąrašas.....	73
4.1 lentelė. Veiklos elementų pavadinimų pavyzdžiai .....	76
4.2 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 1 .....	77
4.3 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 2 .....	77
4.4 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 3 .....	77
4.5 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 5 .....	77
4.6 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 6 .....	78
4.7 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 7 .....	78



4.8 lentelė. Sekos srautų pavadinimų pavyzdys .....	78
4.9 lentelė. Tarpinių įvykių pavadinimų pavyzdžiai 1 .....	79
4.10 lentelė. Tarpinių įvykių pavadinimų pavyzdžiai 2 .....	79
4.11 lentelė. Būdvardžių naudojimo pavyzdžiai .....	80
4.12 lentelė. „Jungtinių veiksmazodžių“ ir raktažodžių atvaizdavimo pavyzdžiai .....	80
5.1 lentelė. DB klasių modelio lentelių specifikacija .....	85
5.2 lentelė. Komponentų specifikacija .....	87
5.3 lentelė. VŽ iš VP „Kliento identifikavimas“ analizė.....	96
5.4 lentelė. Faktų iš VP „Kliento identifikavimas“ analizė.....	97
5.5 lentelė. VŽ iš VP „Sutarties keitimas, nutraukimas“ analizė .....	98
5.6 lentelė. Faktų išVP „Sutarties keitimas, nutraukimas“ analizė .....	99
5.7 lentelė. VŽ iš VP „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ analizė.....	100
5.8 lentelė. Faktų iš VP „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ analizė.....	101
5.9 lentelė. Apibendrinti eksperimento rezultatai.....	101
8.1 lentelė. BPMN elementų aprašymas.....	108
8.2 lentelė. UML veiklos diagramos elementai.....	109
8.3 lentelė. UML panaudojimo atvejų diagramos elementai.....	111
8.4 lentelė. Workflow diagramos elementai.....	112
8.5 lentelė. PA „Patikrinti BPMN diagramos atitikimą realiems organizacijos procesams“ specifikacija .....	113
8.6 lentelė. PA „Specifikuoti ir analizuoti įmonės reikalavimus“ specifikacija.....	113
8.7 lentelė. PA „Patikrinti veiklos žodyno teisingumą“ specifikacija.....	114
8.8 lentelė. PA „Patikrinti veiklos taisyklių teisingumą“ specifikacija.....	114
8.9 lentelė. PA „Patvirtinti BPMN teisingumą“ specifikacija.....	115
8.10 lentelė. DB klasių modelio klasės „BPMN VP modelis“ specifikacija.....	116
8.11 lentelė. DB klasių modelio klasės „BPMN elementas“ specifikacija .....	116
8.12 lentelė. DB klasių modelio klasės „Elemento tipas“ specifikacija.....	116
8.13 lentelė. DB klasių modelio klasės „VZ elementas“ specifikacija .....	116
8.14 lentelė. DB klasių modelio klasės „Zodžio tipas“ specifikacija.....	116
8.15 lentelė. DB klasių modelio klasės „Faktas“ specifikacija .....	116
8.16 lentelė. DB klasių modelio klasės „Fakto tipas“ specifikacija .....	117
8.17 lentelė. DB klasių modelio klasės „VT“ specifikacija .....	117
8.18 lentelė. DB klasių modelio klasės „VT struktūra“ specifikacija .....	117
8.19 lentelė. DB klasių modelio klasės „Raktažodis“ specifikacija .....	117

## 1 Įvadas

Šiuo metu vis daugiau įmonių nori apibrėžti organizacijose vykstančius veiklos procesus (toliau VP) atvaizduojant juos grafiškai, tačiau VP modeliuojantis analitikas ne visuomet nustato visus procese esančius apribojimus. Kadangi šių apribojimų jis negali peržiūrėti ir jų patvirtinti sumodeliuotame VP modelyje, padidėja rizika atsirasti neatitikimams ir klaidoms. Tai gali sutrikdyti organizacijos veiklą. Vadovaudamasis pateiktais VP modeliais kiekvienas dalyvis žino, kokia yra jo pozicija organizacijoje bei kokios jam deleguotos atsakomybės. Todėl ir yra svarbu, kad nubraižyti VP modeliai atitiktų realius organizacijos procesus.

Taip pat svarbu išlaikyti informacijos vieningumą, modeliuojant kitus veiklos aspektus bei aiškumą bendradarbiaujant organizacijoje. Dėl šių priežasčių reikalingas veiklos žodynas (toliau VŽ), apibrėžiantis organizacijoje naudojamus pagrindinius terminus, faktus ir sinonimus, bei veiklos taisyklės (toliau VT), apibrėžiančios visus apribojimus organizacijoje. Dėl VP pokyčių ir didelės apimties, VŽ ir VT sudarymui turi būti naudojamas automatinis VŽ formavimo ir VT sudarymo metodas, kurio pagalba verslo atstovas sutaupo laiko ir užtikrina jo atitikimą realiai vykstantiems VP.

Metodo realizacijai analizuojami VP modeliavimo (BPMN2, UML, Workflow) ir VŽ bei VT (SBVR, SRML ir PRR) formavimo standartai, įvertinant suprantamumą vartotojams, informacijos pateikimą, bei kitas funkcijas (pavyzdžiui, modelių transformavimas į vykdomąjį kodą, XML failo generavimas ir pan.), kurių rezultatas gali būti panaudojamas kaip pradinis duomenų šaltinis, taip pat struktūras, rezultatų pateikimo vartotojams būdus ir pan.

Šio darbo metu nustatyti labiausiai apibrėžtus vertinimo kriterijus atitinkantys standartai – BPMN2 ir SBVR – yra natūralia kalba išreikštos priemonės organizacijoje vykstantiems procesams atvaizduoti. Visgi viena svarbiausių tarpusavio bendravimo priemonių tarp skirtingų organizacijos asmenų, analizuojant VP modelius, išlieka veiklos žodynas ir veiklos taisyklės. Todėl minėti standartai naudojami algoritmo realizacijai, kurios pagrindinis tikslas yra automatizuotu būdu suformuoti VŽ (analizuojant pavienius elementus, žodžius, jų tipus bei pozicijas pavadinime) ir VT (analizuojant VP elementus, jų seką).

Realizavus metodą Java technologija vartotojas gali pats kontroliuoti organizacijos VP: patikrinti jų atitikimą realiems vykdomiems procesams ir, esant poreikiui, keisti VŽ bei VT. Taip įvertinami visi VP apribojimai, o gauti rezultatai – suformuoti VŽ elementai: terminai, faktai („*characteristic fact type*“, „*binary fact type*“, „*partitive fact type*“, „*is property of fact type*“, „*categorization fact type*“), veiksmazodžiai, raktažodžiai ir vardiniai pavadinimai bei VT – gali būti sėkmingai panaudojami ne tik tolimesniuose IS kūrimo etapuose, tačiau ir naudingi organizacijos atstovams vykdam jiems deleguotas atsakomybes.

## **Tyrimo objektas ir sritis**

**Objektas.** Veiklos žodyno ir veiklos taisyklių išgavimas iš veiklos procesų diagramos.

**Sritis.** VP ir VT sankirta. Tai yra:

- VP ir VT aktualumas bei sudarymas;
- VT pritaikomumas ir integravimas;
- VŽ ir VT išgavimo iš VPD galimybės.

## **Tyrimo tikslas ir uždaviniai**

**Tikslas** – suteikti galimybę vartotojui kontroliuoti savo veiklos procesų apribojimus bei padidinti veiklos procesų diagramos atitikimą organizacijos veiklos procesams, sudarant automatizuotą veiklos žodyno ir veiklos taisyklių išgavimo metodą.

### **Uždaviniai:**

1. Išanalizuoti VP ir juose modeliuojamus apribojimus.
2. Išanalizuoti VT struktūras ir jų pritaikomumą veiklos procesuose.
3. Sudaryti VT, esančių veiklos procesuose, struktūrą.
4. Sudaryti VŽ ir VT automatizuoto išgavimo iš VPD algoritmą.
5. Pagal sudarytą algoritmą sukurti prototipą.
6. Ištestuoti sukurtą prototipą ir nustatyti sukurto metodo naudingumą.

## **Tyrimo planas**

Tyrimą sudaro 4 etapai: analizė, projektavimas, eksperimentas ir magistro tezių rašymas:

- analizės etapo metu turi būti išanalizuotos ir palygintos BPMN2, UML bei darbų sekų diagramos, jų elementai ir juose saugoma informacija, kuri nusako veiklos apribojimus. Taip pat išanalizuotos SBVR, SRML VT struktūros bei VT pritaikomumas pasirinktose VPD.
- projektavimo etapo metu pagal atliktą analizę sudaryti VT struktūras, kuriose bus saugomos taisyklės, išgautos iš VPD. VT išgavimui sukurti automatizuoto išgavimo algoritmą, kuris identifikuos diagramos elementus, saugančius apribojimus bei veiklos raktinius žodžius (veiklos žodynui).
- eksperimento etapo metu pagal sudarytą algoritmą sukurti prototipą, tokiu būdu įvertinant algoritmo teisingumą. Antrasis pagrindinis šio etapo žingsnis yra ištestuoti sukurto prototipo veikimą bei naudingumą.
- ketvirtame darbo etape parengti magistro tezes ir įvertinti, ar atlikti visi iškelti uždaviniai bei pasiektas užsibrėžtas tikslas.

## 2 Veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų modelio metodų analizė

### 2.1 Analizės tikslas

Šios analizės tikslas yra išanalizuoti BPMN2, UML, darbų sekų diagramas, nustatyti bei palyginti juose esančius elementus bei saugomą informaciją, kuri nusako veiklos apribojimus. Taip pat išanalizuoti VT modeliavimo standartus, tokius kaip SBVR, SRML ir PRR.

VP modeliavimo ir veiklos žodyno bei VT standartai analizuojami remiantis tam tikrais kriterijais. Kadangi kiekvienas iš jų priklauso tam tikram MDA lygmeniui, tai analizuojama kiekvieno standarto paskirstymas CIM, PIM ir PSM lygmenyje (plačiau apie MDA 2.3.4 poskyryje). Taip pat analizuojant standartus vertinamas ir suprantamumas vartotojams (t.y. ar standartas iš vartotojo papildomai reikalauja techninių žinių).

Modeliuojant VPD svarbiausia yra elementai, todėl pirmiausiai vertinamas elementų skaičius standartuose. Taip pat svarbios ir tokios funkcijos kaip: modelių transformavimas į vykdomąjį kodą, XML failo transformavimas. Standartai vertinami ir pagal diagramų hierarchiškumą, apribojimus – jų sudėtingumą, struktūrą bei aprašymą natūralia kalba.

VŽ bei VT standartai analizuojami atsižvelgiant į taisyklių struktūrą, apribojimų sudėtingumą. Kaip ir analizuojant VP, taip ir šiuo atveju atkreipiamas dėmesys į XML failo generavimo funkciją. Papildomai analizuojamas VŽ, t.y. kokią funkciją jis atlieka standarte (pavyzdžiui, skirtas aprašyti VT elementams ar apibūdinti VT struktūrai).

Pagal atliktą analizę, turi būti nustatyta VP modeliavimo ir VT tinkamiausia kombinacija, užtikrinant efektyvų VŽ ir VT iš VP diagramos išgavimą.

### 2.2 Analizės metodai

Tyrimo analizei atlikti analizuojami literatūros šaltiniai, kurie padeda išspręsti tiriamojo darbo problemą. Problemą išspręsti ir įgyvendinti yra nustatyti tikslai, analizuojami ne tik VP, VŽ ir VT, tačiau ir esami sprendimai, kurie skirti panašioms problemoms spręsti.

Problemos sprendimui atliekama tyrimo objekto, vartotojų bei lyginamoji analizė.

#### 1. Objekto analizė

Tyrimo objekto analizei atlikti yra tiriamos BPMN2, UML ir Workflow VP modeliavimo kalbos, taip pat SBVR, SRML ir PRR veiklos taisyklių standartai.

Taigi siekiant ištirti tyrimo objektą yra atliekama artefaktų (darinių ir modelių) analizė. Darinių **analizės metu** turi būti išnagrinėti ir tiriami: metamodelio elementai, metamodelio elementų tarpusavio ryšiai, metamodelio elementų ir ryšių sintaksė ir semantika bei taikymo

taisyklės ir apribojimai. **Modelių** analizė atliekama tokiais aspektais: modelio modeliujama sritis, modelio paskirtis, modelio sintaksė ir semantika, naudojama terminija, modelio pritaikomumas.

## 2. Vartotojų analizė

Vartotojus, kurie naudojami tiriamais objektais, taip pat nustatomi remiantis keliais aspektais: vartotojų tipais, atsakomybėmis (savybėmis), kvalifikacija, kiekiu ir rinka, tikslais ir problemomis.

## 3. Lyginamoji analizė

Šios analizės metu yra ieškomi, nagrinėjami ir vertinami esami sprendimai. Nagrinėjimui ir vertinimui yra pasirinkti du esami sprendimai. Visa tai įtakoja siūlomą sprendimą, VT ir VŽ iš VPD išgauti.

### Lyginamosios analizės procesas:

1. Surasti esamus sprendimus;
2. Apibrėžti sprendimų palyginimo kriterijus;
3. Atrinkti lyginamus sprendimus ir juos išanalizuoti;
4. Įvertinti ir palyginti esamus sprendimus, pasiūlyti savo sprendimą.

Lyginami sprendimai atrenkami atsižvelgiant į tai, ar esami yra taikomi panašioms problemoms spręsti, nauji, praktiškai patikrinti. Dariniai, modeliai ir esami sprendimai problemai išspręsti yra analizuojami ir aprašomi remiantis tokiu šablonu:

- **Dariniai:** Pavadinimas, akronimas; Informacijos šaltinis; Paskirtis; Kur darinys pritaikytas; Darinio metamodelis.
- **Modeliai:** Pavadinimas; Informacijos šaltinis; Modeliuojama sritis, apimtis ir paskirtis; Kur modelis pritaikytas; Modelio schema.
- **Esami sprendimai:** Pavadinimas; Diagramos elementų žymėjimas taisyklėse.

## 2.3 Veiklos procesų analizė

Šiandienos [9] įmonės privalo įvertinti ne tik produktus ar teikiamas paslaugas, tačiau ir naudojamas informacines sistemas, kurios tampa verslo sudėtine dalimi. Verslas yra tai, kas galiausiai kuria informacinių sistemų reikalavimus, o projektuoti programinę įrangą tinkamai nesuvokiant, kokia bus sistemos paskirtis, yra pavojinga.

Siekiant tai suprasti svarbu sukurti veiklos modelį. *Modelis* yra supaprastintas sudėtingos tikrovės vaizdas, kuriuo pašalinamos nereikšmingos detalės ir sutelkiamas dėmesys į vieną ar daugiau svarbių aspektų. Tinkamai suprojektuoti modeliai palengvina derybas tarp skirtingų veikloje dalyvaujančių aktorių (angl. *Stakeholders*), taip suteikiant jiems galimybę sutarti dėl to, kas svarbiausia, ir toliau dirbti siekiant pagrindinių tikslų. Veiklos modelis pateikia informaciją organizacijos nariams, kuri padeda suprasti VT ir VP.

### **2.3.1 Veiklos procesų valdymas**

Veiklos [25] procesų valdymas (angl. *Business Process Management (BPM)*) – tai įvairių organizacinių procesų valdymas, taikant metodus ir įrankius. Taip siekiama palaikyti VP projektavimą, vykdymą, valdymą ir analizę.

VP yra suderinta tos pačios krypties veiklų grandinė, skirta pateikti veiklos taisyklės, arba pasikartojantis ratas, kuriuo siekiami organizacijos tikslai. Siekiant visiškai išbaigti procesą įtraukiami skirtingų skyrių ir organizacijų žmonės.

Procesai gali būti paprasti ir apriboti (skirti tam tikram ratui) funkcinėi organizacijos grupei arba sudėtingi ir kertantys (persidengiantys) kelis verslo partnerius (angl. *Cutting across several business partners*).

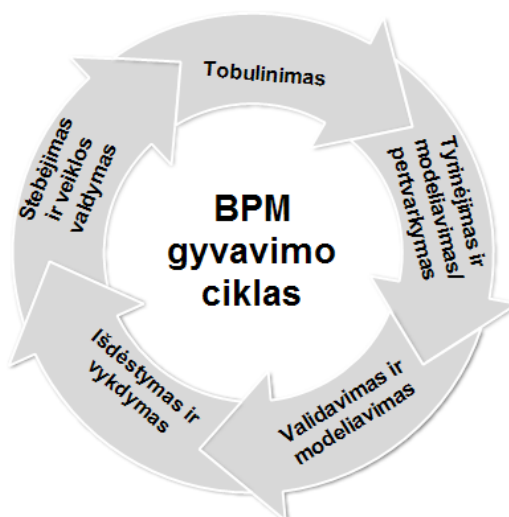
### **2.3.2 Veiklos proceso modeliavimas**

Veiklos proceso modeliavimas [1, 25] (angl. *Business Process Modelling*) yra VP grafiškas atvaizdavimas ir dokumentavimas. Tai yra struktūrizuotas metodas, kuris padeda organizacijos veikloje dalyvaujantiems aktoriams analizuoti procesus ir nustatyti galimus patobulinimus. Proceso, kuris paprastai apibrėžia veiklas, modeliavimas yra atliekamas skirtingų šalių, įtrauktų į kuriamą procesą. VP (ir tai kas yra aplink jį) sudarytas iš veiklų, įvykių, išteklių, rolių ir aktorių, funkcijų, organizacijos ir hierarchijos. Be to, VP kontekstas apima proceso ir įmonės tikslus bei priemones ir procesų rezultatus. Todėl naudojantis VP modeliavimo notacijomis turėtų būti suteikta galimybė modeliuoti minėtus proceso aspektus. VP modeliavimo kalba taip pat turi suteikti galimybę atsakyti į penkis klausimus – kodėl, kas, kas (apie žmones), kur ir kada. Keturi paskutiniai klausimai padeda apibrėžti patį procesą, o pirmasis – kodėl – padeda apibrėžti tikslus ir reikalavimus, esančius už proceso ribų. Tikslų modeliavimas ir atsekamumas (angl. *Traceability*) tarp pradinių reikalavimų ir įgyvendinto proceso yra svarbios savybės proceso valdymui, todėl egzistuoja keletas, tikslo modeliavimui skirtų, proceso modeliavimo notacijų.

### 2.3.3 Veiklos proceso gyvavimo ciklas

VP apima skirtingus ir apibrėžtus etapus. Norint visiškai optimizuoti visas proceso galimybes, svarbu ne tik juos žinoti, bet taip pat reikia mokėti juos pritaikyti. 2.1 paveiksle pavaizduoti etapai ir rolės, siekiant pastovaus proceso tobulinimo.

BPM gyvavimo ciklas dažniausiai susideda iš keleto pasikartojančių žingsnių, kurių tikslas – pagerinti VP kokybę.



2.1 pav. BPM gyvavimo ciklas

2.1 paveiksle pavaizduotas tipiškas, modeliais grįstas BPM gyvavimo ciklas [27], prasidedantis verslo procesų *tyrinėjimu ir modeliavimu* (angl. *Discovery & Model / Remodel*). Šiame etape rekomenduojama naudoti naujus proceso modelius tam, kad jie nekliudytų kitiems BPM projekto žingsniams, pradėdant nuo optimalaus rezultato ir vertės pateikimo.

Taip pat svarbi ir integruota bei tam tikrus įrankius palaikanti metodologija, kuri naudinga vartotojams, *validuojantiems* bei *modeliuojantiems* (angl. *Validation & Simulation*) VP.

BPM projekto *išdėstymo ir vykdymo* (angl. *Deployment and Execution*) etapo pagrindinis akcentas yra sumodeliuotų procesų automatizavimas. Jei tai yra integruota su informacinėmis sistemomis, kurios teikia duomenis, naudingus *stebėjimo* (angl. *Monitoring*) ir *vertinimo* (angl. *Performance*) valdymo etapams, tai, remiantis *išdėstymo ir vykdymo* etapu, gerinami vėlesni etapai.

*Tobulinimo* (angl. *Improvement*) etape gali būti naudojami veiklos vertinimo rezultatai, gauti stebėjimo metu. Kitaip tariant, gali būti siūloma pertvarkyti modelius, kurie vėliau labiau atitiktų proceso tikslą. Minėti modeliai yra paremti problema, atrasta stebėjimo etape, tam, kad būtų galima lengviau sulygtinti procesą su tikslais, aprašytais modeliavimo etape. Jei procesas vykdomas puikiai, tada jo tikslai taip pat gali būti persvarstyti.

Norint [3] atvaizduoti VP ir jų galimus srautus bei perėjimus grafiškai, yra naudojami grafiniai standartai. Jie taip pat leidžia vartotojui grafiškai išreikšti VP roles. Grafiniai standartai lengvai nagrinėjami ir suprantami be pradinio techninio pasiruošimo. UML veiklos diagramos (angl. *UML Activity Diagrams*), BPMN (angl. *Business Process Modelling Notations*), EPC (angl. *Event-driven Process Chains*), RADs (angl. *Role-Activity Diagrams*) ir darbų sekų diagramos (angl. *Workflow*) yra dažnai pasitaikančios metodikos, naudojamos VP grafiniam atvaizdavimui.

Šios metodikos svyruoja nuo įprastų (pvz.: sekų diagramos) iki standartinių (pvz.: BPMN) notacijų. Iš standartų, UML AD ir BPMN2 šiuo metu yra du išraiškingiausi, lengviausiai integruojami apsisikeitimo ir vykdymo lygmenyse. Dėl šių priežasčių, labiau atkreipiamas dėmesys į UML AD (2.0 versija) ir BPMN (2.0 versija).

#### 2.3.4 Modeliais grįsta architektūra

Modeliais [15] grįsta architektūra (angl. *The Model-Driven Architecture*) (toliau MDA) apibrėžia modeliu grindžiamo programinės įrangos kūrimo metodą. MDA specifikacija sudaryta iš trijų sluoksnių: CIM, PIM ir PSM (schema pateikta 2.2 paveiksle).

1. CIM (angl. *Computation-Independent Model*) sluoksnį sudaro veiklos modelis arba, kitaip, nuo skaičiavimų nepriklausomas modelis (CIM), kuris gali būti naudojamas kaip gairė tiksliai apibrėžiant nuo platformos nepriklausantį modelį (PIM). To pavyzdys yra veiklos semantikos veiklos taisyklėms pasiūlymas.

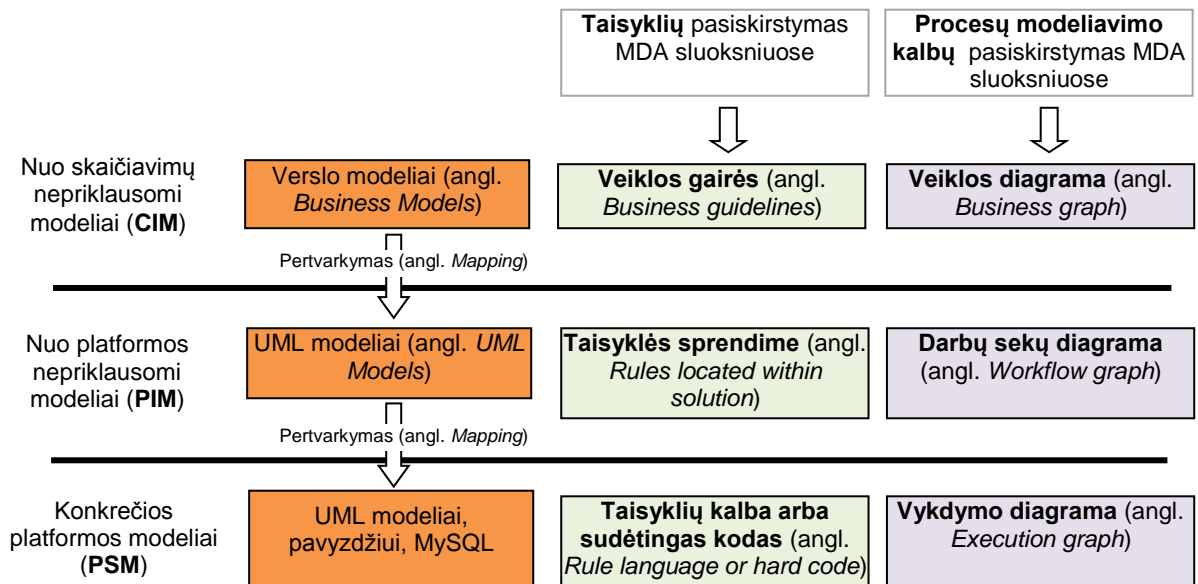
CIM sluoksnyje atvaizduojamos VT, procedūros ir apribojimai bei VP, kurie yra užrašyti natūralia kalba. Veiklos modeliai, esantys CIM sluoksnyje, yra transformuojami (angl. *Mappable*) į UML modelio elementus.

2. PIM (angl. *Platform-Independent Model*) sluoksnį sudaro galutinis (nekeičiamas), nuo platformos nepriklausantis, UML modelis.

Naudojant formalias taisykles nuo platformos nepriklausantis modelis (PIM) nuosekliai transformuojamas į konkrečios platformos modelį (PSM), pertvarkant nuo platformos nepriklausantį modelį į kokią nors vykdymo kalbą ar platformą (pavyzdžiui, Java).

3. PSM (angl. *Platform-Specific Models*) sluoksnį sudaro vienas ar daugiau konkrečios platformos modelių (PSM) ir sąsajos sąvokų rinkiniai, kurių kiekvienas aprašo, kaip pagrindinis modelis yra realizuotas skirtingose platformose.





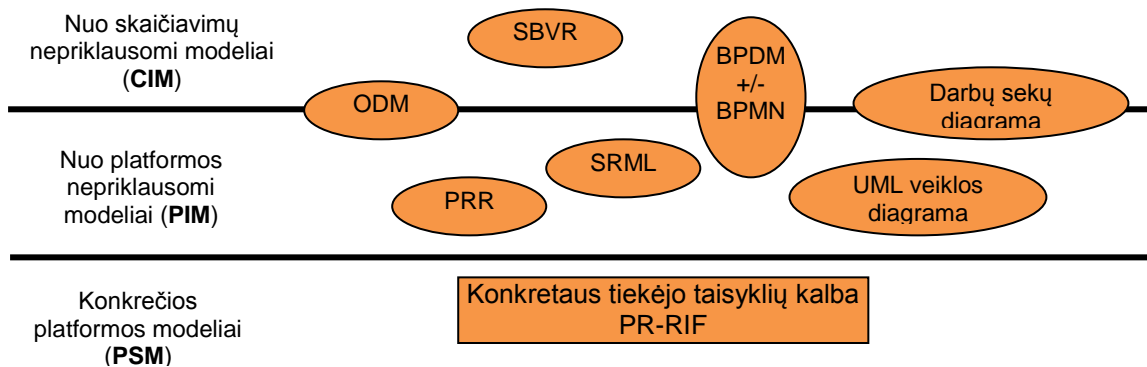
2.2 pav. Bendra MDA schema

Kiekviena VP (tokia kaip BPMN2, UML veiklos diagrama, darbų sekų diagrama), VT (SBVR, SRML ir PRR) modeliavimo kalba priklauso vienam (ar daugiau) MDA sluoksnių. Pavyzdžiui, BPMN2 gali priklausyti CIM ir PIM sluoksniams, SBVR priklauso CIM sluoksniui, PRR ir SRML – PIM sluoksniui. Tai pavaizduota 2.3 paveiksle.

- SBVR, BPMN2 ir kitų standartų pasiskirstymą CIM sluoksnyje lemia tai, jog šios VT ir VP modeliavimo kalbos yra lengvai suprantamos vartotojui.

PRR [16] reprezentuoja PIM sluoksnį. PRR yra orientuota į našumo (gamybos) taisyklės sistemą klasė - platforma (angl. *Class-to-platform*) ir nepriklauso nuo konkretaus tiekėjo sistemos. PRR apsiriboja reikalavimų našumo (gamybos) taisyklėmis, orientuotoms į tiesioginio išvedimo taisyklių ir procedūrinę klasė - platforma sistemos formas, nurodymu. Šios dvi formos apima visus sprendimus, pradedant nuo sprendimo priėmimo ir baigiant VP valdymo palaikymu.

SRML taisyklių kalbos formatas taip pat reprezentuoja PIM sluoksnį, nes yra pagrįstas XML kalba ir nenaudoja konkretaus tiekėjo patentuotų kalbų.



2.3 pav. Veiklos procesų ir veiklos taisyklių modeliavimo kalbų pasiskirstymas

## 2.4 Veiklos procesų modeliavimo standartai

### 2.4.1 Veiklos procesų modeliavimo notacija (BPMN)

BPMN [2] (angl. *The Business Process Modelling and Notation*) – VP modeliavimo notacija – yra naujas VP srautų ir Web servisų modeliavimo standartas, sukurtas BPMI (angl. *Business Process Management Initiative*).

**Paskirtis (tikslai).** Pirmasis BPMN tikslas yra teikti notaciją, kuri lengvai suprantama verslo vartotojams. Šie vartotojai yra analitikai, kurie sudaro pirminius procesų projektus technikos kūrėjams, atsakingiems už technologijų, kurios vėliau bus naudojamos aprašomiems VP atlikti, realizaciją.

Antrasis, taip pat svarbus, tikslas – užtikrinti, kad XML kalbos, tokios kaip BPEL4WS (angl. *Business Process Execution Language for Web Services*) ir BPML (angl. *Business Process Modelling Language*), skirtos VP įforminti, galėtų būti vaizdžiai išreikštos, naudojant įprastą, į verslą orientuotą, notaciją.

Be to, BPMN apibrėžia veiklos proceso diagramą (angl. *Business Process Diagram (BPD)*), kuri buvo suprojektuota siekiant šių tikslų:

1. BPD galima naudoti norint greitai ir lengvai sumodeliuoti VP. Ši diagrama taip pat lengvai suprantama techninių žinių neturintiems vartotojams (dažniausiai vadybos srities asmenims).
2. BPD siūlo išraiškumą, modeliuojant labai sudėtingus VP, ir gali būti susietas su veiklos (angl. *Business*) vykdymo kalbomis.

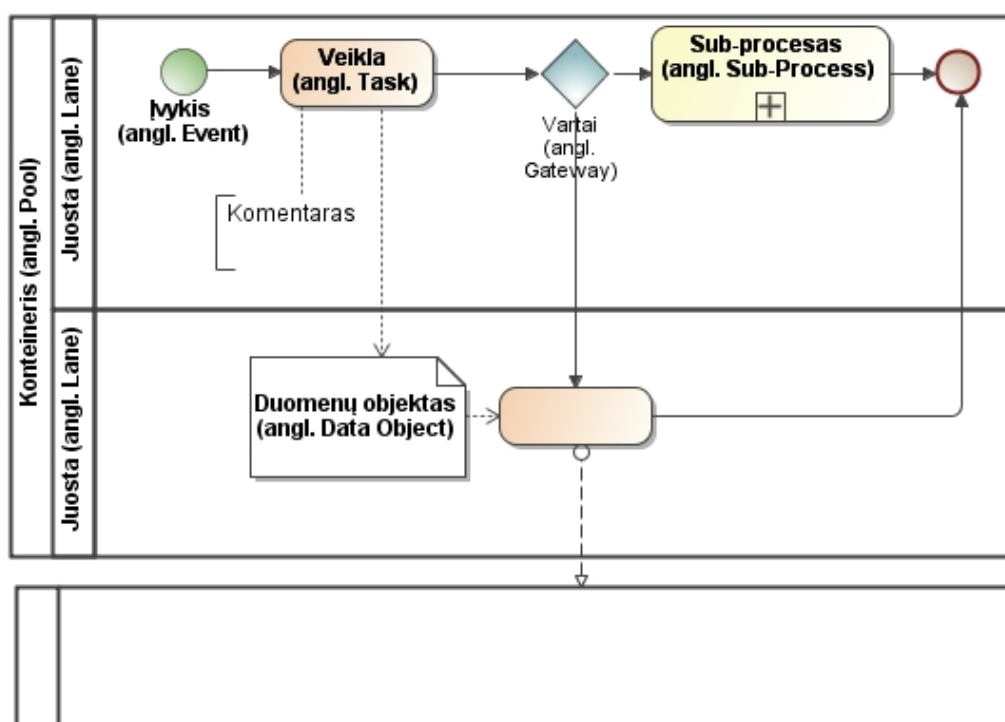
**BPMN elementai.** Reikėtų [8] dar kartą pabrėžti, kad vienas iš BPMN2 kūrimo tikslų yra sukurti paprastą ir suprantamą mechanizmą VP modelių kūrimui, tuo pat metu sugebant valdyti VP būdingą sudėtingumą. Sprendimas, siekiant išspręsti šiuos du konfliktiškus reikalavimus, buvo suskirstyti grafines notacijas į kategorijas. Jos pateikia notacijų rinkinį, kad BPMN2

diagramos skaitytojas galėtų lengvai atpažinti pagrindinius elementus ir taip suprasti modelį.

Taigi yra sukurtos 5 pagrindinės elementų kategorijos:

1. Srautų objektai (angl. *Flow Objects*)
2. Duomenys (angl. *Data*)
3. Jungimo objektai (angl. *Connecting Objects*)
4. Konteineriai ir juostos (angl. *Pool, Lane*). Bendrai tai vadinama plaukimo takeliais (angl. *Swimlanes*)
5. Artefaktas

Kiekvieną iš išvardintų kategorijų sudaro elementai. Keletas iš jų (t.y. pagrindiniai) pateikiami 2.4 paveiksle.



2.4 pav. BPMN diagramos elementai

**Srautų objektai** yra pagrindiniai grafiniai elementai, apibrėžiantys VP elgesį. Yra 3 srautų objektai:

1. Įvykiai (angl. *Events*)
2. Veiklos (angl. *Activities/Tasks*)
3. Vartai (angl. *Gateways*)

**Duomenys** aprašomi 5 elementais:

1. Duomenų objektai (angl. *Data Objects*)
2. Duomenų įėjimai (angl. *Data Inputs*)
3. Duomenų išėjimai (angl. *Data Outputs*)
4. Duomenų saugyklos (angl. *Data Stores*)

## 5. Savybės (angl. *Properties*)

Srautų objektams vienas su kitu arba su kita informacija sujungti yra aprašomi 4 būdai (**jungimo objektai**):

1. Sekos srautas (angl. *Sequence Flow*)
2. Žinučių srautas (angl. *Message Flow*)
3. Ryšiai (angl. *Association*)
4. Duomenų ryšiai (angl. *Data Association*)

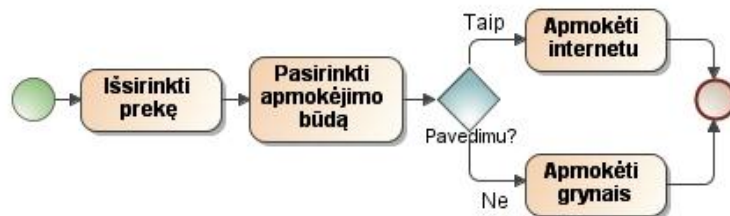
**Plaukimo takeliams** yra 2 būdai aprašyti:

1. Konteineriai (angl. *Pools*)
2. Juostos (angl. *Lanes*)

**Artefaktai** yra naudojami teikiant papildomą informaciją apie procesą. Yra 2 standartizuoti artefaktai, tačiau modeliuotojai ar modeliavimo įrankiai gali pridėti artefaktų kiek tik reikia. Artefakto elementais gali būti:

1. Grupavimo elementai (angl. *Group*)
2. Komentavimo elementai (angl. *Text Annotation*)

Šių elementų pritaikymo kokiai nors organizacijos veiklai pavyzdys pateikiamas 2.5 paveiksle. Jame vaizduojama VP (šiuo atveju prekės apmokėjimo) diagrama, kurioje pateikiami pagrindiniai elementai: įvykiai, veiklos, vienas iš jungimo objektų (sekos srautas).



2.5 pav. Prekės apmokėjimo veiklos proceso pavyzdys BPMN diagrama

Išanalizavus 8.1 lentelę (žiūrėti 8.1 priede) ir 2.5 paveikslą, įsisavinamas kiekvieno elemento panaudojimas. Šiuo atveju didžiausias dėmesys skiriamas jungimo objektams. Kiekvienas jų naudojamas skirtingais tikslais, todėl ir žymėjimas yra skirtingas. Pavyzdžiui, 2.4 paveiksle matoma, kad žinučių srautas yra naudojamas tik tarp vidinių ir išorinių dalyvių.

BPMN2 modelio elementai yra aprašyti ir pateikiami 8.1 priede.

### 2.4.2 Unifikuota modeliavimo kalba

UML [5,6] (angl. *The Unified Modelling Language*) yra specifikuojama kalba, bet ne metodas ar procedūra.

**Paskirtis (tikslai).** UML yra naudojama apibrėžti programinės įrangos sistemas – detalizuoti artefaktus sistemoje, aprašyti ir struktūrizuoti loginę struktūrą. UML siūlo standartinį būdą, skirtą aprašyti sistemos projektus, įtraukiant abstrakčius dalykus, tokius kaip VP ir sistemos funkcijos, taip pat realius dalykus, tokius kaip programavimo kalbos teiginiai, duomenų bazės schemas ir daugkartinio naudojimo programinės įrangos komponentai.

**UML struktūra, elementai.** UML [10] vis dažniau vertinama kaip programinės įrangos modeliavimo ir projektavimo de-facto standartas. Naujausioje versijoje (UML 2.0) yra 13 skirtingų modeliavimo notacijų pradedant nuo aukšto lygio panaudojimo atvejų diagramų, kurios atvaizduoja sąveikas ir ryšius tarp aktorių ir pagrindinių veiklos funkcijų, baigiant žemo lygio objektų diagramomis, kurios kaupia atskirus duomenų objektus, jų sudedamųjų duomenų elementų ir reikšmių bei jų ryšių su kitais duomenų objektais atvejus.

Skirtingos modeliavimo notacijos yra skirstomos į tris pagrindines grupes:

1. **Elgsenos diagramos** (angl. *Behaviour diagrams*), kurios aprašo programinės įrangos funkcionalumą santykinai aukštame abstrakcijos lygmenyje.
2. **Sąveikos diagramos** (angl. *Interaction diagrams*), kurios dar labiau praplečia elgsenos diagramas ir pateikia detalesnį sistemos funkcionavimo (objektinės sąveikos išraiška) aprašymą.
3. **Struktūrinės diagramos** (angl. *Structure diagrams*), kurios užfiksuoja pagrindinę programinės įrangos sistemos statinę struktūrą įvairiuose lygmenyse, pradedant nuo atskirų objektų ir baigiant bendrais taikymo paketais.

Iš [9] minėtų kategorijų elgsenos diagramos, tokios kaip UML veiklos diagrama ir panaudojimo atvejų diagrama, yra dažnai naudojamos modeliuojant VP.

UML [2,9] siūlo objektiškai orientuotą programų, VP modeliavimo būdą. Dauguma UML metodų siūlo pirmiausia išsiaiškinti objektus, naudojant statines struktūrines diagramas, ir tik tada projektuoti dinamišką elgsenos diagramą, taip atvaizduojant objektų sąveiką. Tai yra, sistemos kūrimas turi prasidėti nuo panaudojimo atvejų modeliavimo, taip apibrėžiant funkcinius reikalavimus sistemai. Panaudojimo atvejai aprašo vieno ar daugiau aktorių konkretų sistemos panaudojimą. Šios diagramos tikslas yra identifikuoti ir aprašyti visus sistemos panaudojimo atvejus, kurių reikalauja vartotojas. Vėliau šie aprašymai analizuojami ir projektuojama sistemos architektūra, kuri atitinka aprašytus panaudojimo atvejus. Panaudojimo atvejų diagramos elementai yra aprašyti ir pateikiami 8.2 priede.

## UML veiklos diagrama (angl. *UML Activity diagram*) elementai [6,10]

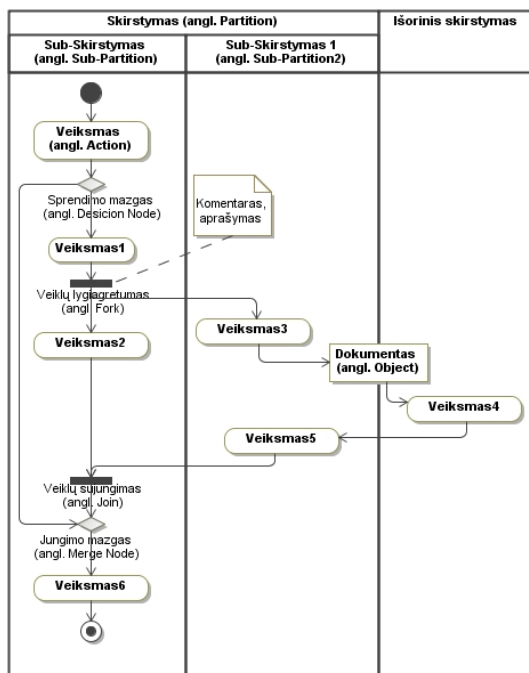
Ši [6] diagrama naudojama, siekiant pavaizduoti darbų seką nuo pradžios iki pabaigos, išsamiai aprašant daugybę sprendimo būdų, kurie egzistuoja veiklos įvykių sekoje. Veiklos diagramos gali būti naudojamos detalizuojant situacijas, kur gali atsirasti lygiagrečių veiklų. Taip pat naudingos veiklos modeliavimui, detalizuojant organizacijos veiklų procesus.

Tam, kad būtų galima detalizuoti situacijas, UML veiklos diagramose taip pat naudojama daug elementų, kurie, kaip ir BPMN2 elementai, yra suskirstyti į kategorijas. Šiuo atveju jos yra:

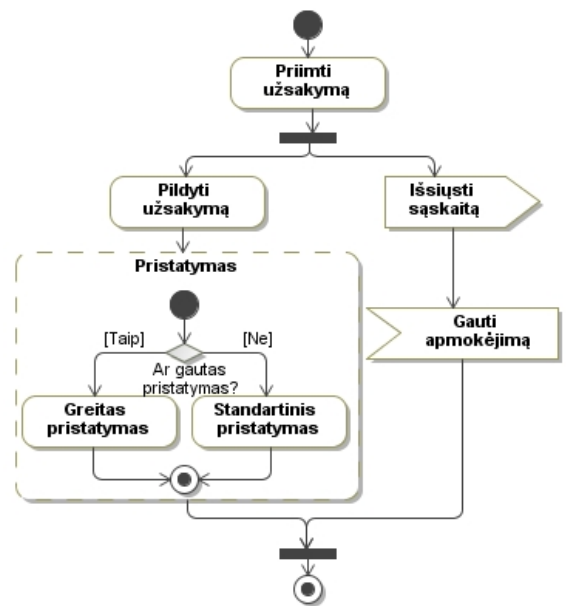
1. Veiksmai (angl. *Actions*)
2. Kontrolės mazgai (angl. *Control nodes*)
3. Objektai ir objektų srautai (angl. *Objects and Object Flow*)

Šių kategorijų elementų vaizdavimas pateikiamas 2.6 paveiksle.

2.6 paveiksle pavaizduoti elementai gali būti naudojami įvairiuose organizacijos VP tokiuose kaip pavyzdžiui, užsakymo pristatymo ir apmokėjimo VP (žiūrėti 2.7 pav.).



2.6 pav. UML veiklos diagramos pagrindiniai elementai



2.7 pav. Užsakymo pristatymo ir apmokėjimo veiklų atvaizdavimo pavyzdys

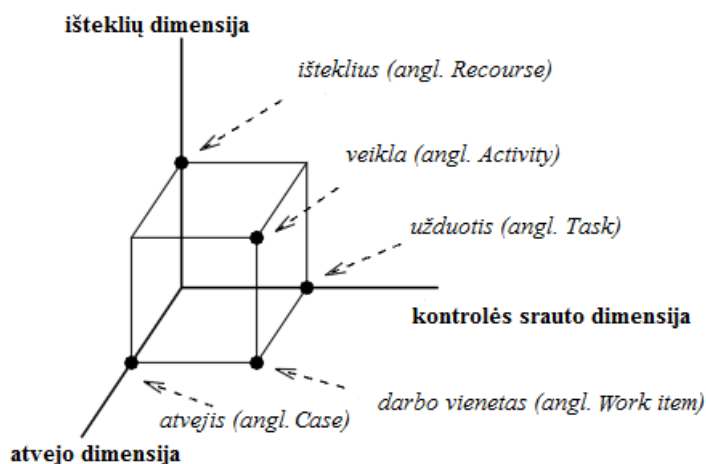
Diagramose pateikti pagrindiniai elementai – kontrolės mazgai, veiklos, objektai – plačiau aprašyti 8.2 priede.

### 2.4.3 Darbų sekos modeliavimo kalba

WfMC [7] (angl. *The Workflow Management Coalition*) darbų seką (angl. *Workflow*) apibrėžia taip: darbų seka – tai VP automatizavimas, kurio metu vienas dalyvis perduoda dokumentus, informaciją ar užduotis kitam dalyviui, atsižvelgiant į procesinių taisyklių rinkinį. Taip siekiama galutinio organizacijos tikslo. Nors [22] darbų sekų diagrama gali būti organizuota rankiniu būdu, tačiau praktikoje dauguma sekų yra organizuojamos naudojantis IT konteksto sistemomis, taip siekiant pateikti kompiuterizuotą procedūrinio automatizavimo palaikymą.

Darbų sekų diagramos [7] procesai yra pagrįsti atvejais (angl. *Case-driven*). Pavyzdžiui, užduotys yra vykdomos tam tikrais atvejais. Paskolų patvirtinimas, apdorojamų draudimo patvirtinimas, sąskaitų išrašymas, mokesčių deklaravimo tvarkymas, eismo pažeidimų ir nekilnojamo turto įkeitimo tvarkymas yra atvejais pagrįsti procesai, kurie dažnai yra palaikomi WFM sistemų. Tokie procesai dar vadinami darbų sekų procesais ir yra apibrėžti trimis aspektais:

1. Kontrolės srauto (angl. *Control-flow*)
2. Išteklių (angl. *Resource dimension*)
3. Atvejo (angl. *Case dimension*)



2.8 pav. Trys darbų sekų diagramos dimensijos

**Kontrolės srauto** aspektas yra susijęs su daliniu užduoties užsakymu, pavyzdžiui, darbų sekų procesais. Užduotys, kurios turi įvykti, yra identifikuojamos. Taip pat parenkamas tų užduočių atvejų maršrutas (angl. *Routing*). Sąlyginis, nuoseklus, lygiagretus ir iteratyvus maršrutas yra tipinė struktūra, apibrėžta kontrolės srauto dimensijoje. Užduotys vykdomos išteklių pagalba. **Ištekliai** yra žmogiškieji (pvz.: darbuotojai) arba materialūs (pvz.: įrenginiai, programinė įranga, techninė įranga). Jie klasifikuojami identifikuojant roles (t.y., išteklių klasės, paremtos funkcinėmis charakteristikomis) ir organizacinius vienetus (t.y., grupės, komandos ir skyriai). Ir kontrolės srauto ir išteklių dimensijos yra suderintos (angl. *Generic*) – nėra

pritaikytos konkrečiam atvejui. Paskutinioji (**atveju**) **dimensija** yra susijusi su atskirais atvejais, vykdomais atsižvelgiant į tam tikro resurso (antroji dimensija) proceso aprašymą (pirmoji dimensija).

Svarbiausia WFM sistemos užduotis yra priimti atvejais pagrįstus VP, sujungiant keletą perspektyvų [7,23]:

1. Kontrolės srauto (arba proceso) perspektyva. Darbų sekų schemas yra apibrėžiamos, siekiant aprašyti, kokios užduotys ir kokia tvarka jos turi būti įvykdytos.
2. Išteklių (arba organizacijos) perspektyva. Čia aprašoma organizacinė struktūra ir bendruomenė. Organizacinė struktūra aprašo ryšius tarp rolių (t.y., išteklių klasės, paremtos funkciniais aspektais) ir grupių (t.y., išteklių klasės, paremtos organizaciniais aspektais).
3. Duomenų (arba informacijos) perspektyva.
4. Užduoties (arba funkcijos) perspektyva.
5. Operacijų (arba taikymo) perspektyva. Ši perspektyva aprašo paprastas operacijas, kurias atlieka ištekliai ar programos. Šios operacijos naudojamos proceso perspektyvoje kurti, skaityti arba redaguoti kontrolinius ir gamybinius (angl. *Production*) duomenis duomenų perspektyvoje. Dauguma operacijų yra realizuotos programomis.

**Paskirtis (tikslai).** Darbų sekų modelis [24] pristato VP ir skirtas vartotojams, kurie supranta veiklos (angl. *Business*) terminiją ir klausimus. Paprastai tokie modeliai kuriami seminaruose, konsultuojantis su tos srities ekspertais. Modeliai reikalingi kaip komunikavimo įrankis tarp verslo ekspertų kaip patikros priemonė, siekiant užtikrinti teisingą supratimą, ir kaip dokumentacija, perkeliant rezultatus. Modeliai gali būti naudojami daugumoje projektų, siekiant tokių tikslų:

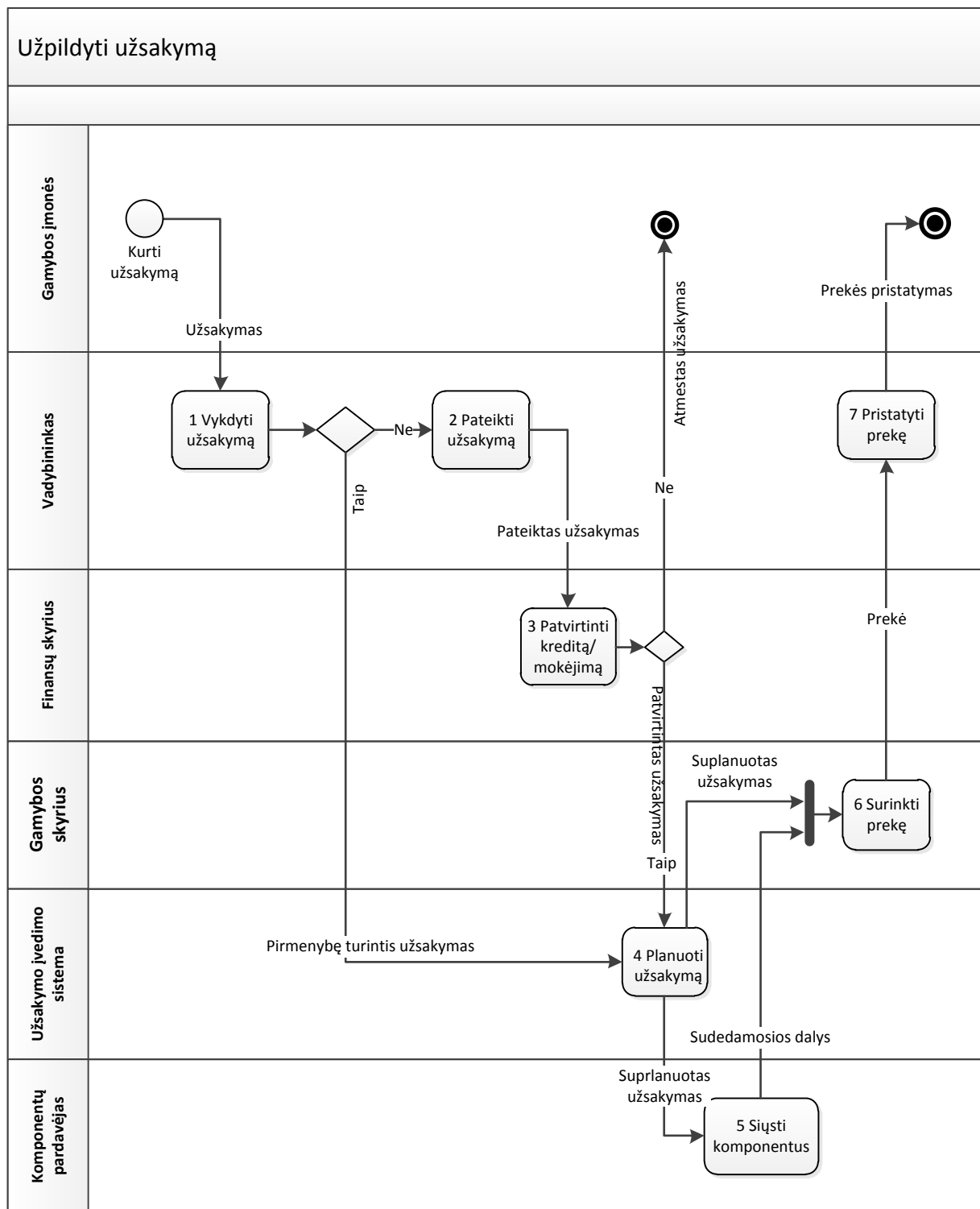
- Pertvarkyti VP, siekiant pagerinti kokybę, mažinti kainą, padidinti klientų reakciją ir t.t.;
- Formalizuoti VP, nenuosekliai atliekamus visoje įmonėje;
- Pertvarkyti organizaciją, siekiant efektyvesnio VP vykdymo;
- Įsisavinti taikomąsias sistemas, taip sėkmingai palaikant pertvarkytus VP ir organizacijos struktūras.



Visgi, pagrindinis darbų sekų proceso tikslas – valdyti panašius atvejus. Darbų sekų proceso modeliu aprašoma, kokios veiklos ir kokia tvarka jos turi būti vykdomos.

### Darbų sekų diagramos elementai

Darbų sekų diagrama sudaryta iš skirtingų komponentų, siekiant atvaizduoti skirtingus proceso aspektus. Pagrindiniai komponentai yra pavaizduoti diagramoje (žiūrėti 2.9 pav.):



2.9 pav. Darbų sekų diagramos pagrindiniai elementai

Paveiksle matomi elementai plačiau aprašomi 8.3 priede.

Paprastai atskiros veiklos yra gerai organizuotos, bet procesas baigiamas, kai darbas yra perduodamas iš vienos organizacijos į kitą. Dažnai veiklos rezultatai nėra tai, kas naudojama vėlesnėje veikloje. Kartais rezultatai yra visai nenaudojami. Akivaizdūs „atsijungimai“ procese gali būti identifikuojami ir ištaisyti naudojant tik darbų sekų modelį.

#### 2.4.4 Veiklos procesų modeliavimo standartų palyginimas

2.1 lentelėje pateikiama apibendrinta BPMN2, UML veiklos diagramos ir darbų sekų diagramos informacija. Taip siekiama išskirti kiekvienos privalumus ir trūkumus. Pliuso ženklas žymi, kad funkcija galima, minusas – negalima, o +- reiškia, kad reikia pastangų ar kitų resursų atitinkamo aspekto įgyvendinimui.

2.1 lentelė. Veiklos procesų modeliavimo standartų palyginimo lentelė

	BPMN2	UML veiklos diagrama	Darbų sekų diagrama
Elementų diagramoje skaičius	Daug	Vidutiniškai	Mažai
Suprantamumas			
Įmonės atstovai	+	+-	+-
Analitikai	+	+	+
Galimybė transformuoti į vykdomą kodą	+	+	+
Diagramų hierarchija	+	+	+
Galimybė generuoti XML failą	+	+	+
Apribojimai			
Sudėtingas apribojimas	-	+-	+-
Formali struktūra	-	+	-
Natūrali kalba	-	-	-
MDA lygmuo	CIM	PIM	CIM/PIM

Standartai tiriami keliais aspektais: elementų diagramoje skaičiumi, suprantamumu ir pan. Kiekvienas iš standartų turi ir plusų ir minusų. Pagrindiniai skirtumai yra tai, kad BPMN2 yra suprantamas daugumai vartotojų, o norint suprasti kitus standartus reikalingos papildomos techninės žinios. Taip pat BPMN2 yra galimybė generuoti XPDL failą, UML veiklos diagramą naudoti XMI formatu. Išnagrinėjus standartus lentelėje išvardintais pjūviais matoma ir panašumų: nei vienas standartas nenaudoja natūralios kalbos (modeliai pateikiami grafiškai, naudojant skirtingas atvaizdavimo galimybes), kiekvieno standarto modelius galima transformuoti į vykdomąjį kodą bei modeliavimo kalbos yra suprantamos analitikams, kadangi jie turi atitinkamų žinių.

## 2.5 Veiklos žodyno ir taisyklių analizė

### 2.5.1 Veiklos žodynas

VŽ yra kalbos pagrindas ir atramos taškas. Tai yra „sąvokų katalogas“, be kurio bendravimas būtų neįmanomas. Turint VŽ, kuris pagrinde sudarytas iš terminų – pagrindinių žodžių ar frazių, naudojamų versle bei faktų, lengviau bendrauja ir skirtingų sričių atstovai (pavyzdžiui, sistemos (ar VP) modeliotojas ir samdomas ar mažai IT žinių turintis darbuotojas).

VŽ sudaromas visose srityse, pradedant nuo neformalių ir neužfiksuotų bendravimo žodynų, formaliai aprašytų ir vartotojams pateiktų kalbų žodynų (anglų – lietuvių ir pan.), baigiant versle naudojamais terminais. Pirmuoju atveju žodynas „gimsta“ bendraujant susitikimuose, kur aiškūs elementarūs terminai/ žodžiai, o paskutiniuoju – iš organizacijoje naudojamų veiklos duomenų, VP bei kitų modelių (tikslų, pareigybių ir pan.), techninės darbuotojų kalbos.

VŽ naudojamas ne veiksmams, operacijoms, procedūroms apibrėžti, o nustatyti jose naudojamus terminus (organizacijoje ir ne tik) bei jų reikšmes. Žodyne dažniausiai išskiriami terminai, vardai, veiksmazodžiai, raktažodžiai ir faktai. Kiekvienas iš jų naudojamas skirtingais tikslais, kaip pavyzdžiui, **terminai** – veiklos esybėms nurodyti, **veiksmazodžiai** – fakto tipui pažymėti, apjungti keletą terminų bei, įtraukiant **raktažodžius** (naudojamus korektiškam sakiniui sudaryti), sudaryti **faktą**, kuriuo aprašomi ryšiai tarp terminų. Faktai ir terminai sudaro faktų modelį, kuris reprezentuoja pagrindinį žodyną veiklos taisyklėms apibrėžti.

VŽ (ir VT) apiforminti egzistuoja standartai, savyje talpinantys VŽ, kuris gali būti formuojamos iš DB (PRR, SRML standartai), kai terminams, faktams išgauti naudojamos programavimo kalbos, taip pat gali būti formuojamas atskirai ir galiausiai vizualiai pateikiamas vartotojui (standartas su savimi gali turėti tam tikrą struktūrą, spalvas ir kitus skiriamuosius požymius, kaip pavyzdžiui, simbolius: „/“, „<“, „/>“ ir t.t.).

### 2.5.2 Veiklos taisyklės

VP [5] modeliavimo kalbos aprašo procedūrinę veiklų seką, įtraukiant sprendimus (angl. *Decisions*) ir sutapimus (lygiagretumus) (angl. *Concurrency*). VT modeliavimo kalbos dažnai remiasi faktų aprašymais, sąlygomis ir apribojimais. Jų egzistavimas yra vienas iš reikalavimų VT formavimui.

VT yra pirminis vaizdas, kaip vykdomas VP, organizuojami duomenys duomenų bazėje ar pan. Tai yra galutinis rezultatas – tikslas, pagal kurį sistemos kūrėjai, organizacijos nariai privalo dirbti, norint pasiekti paskirtų tikslų.

VT gali būti išgaunamos rankiniu ar automatiniu būdu. Pastarasis nereikalauja daug techninių žinių bei laiko. Nepriklausomai nuo to, koku būdu suformuotas VT sąrašas, pagrindinis jo bruožas yra aiškumas ir išbaigtumas. Tai yra, VT sąrašas turi būti pateiktas aiškia (natūralia kalba, įtraukiant žodžius, pabrėžiančius galimumą, būtinumą, draudžiamumą ir pan.) verslo kalba, kuri galutiniam vartotojui gali būti pateikiama vizualia forma, t.y., panaudojant specifinius simbolius, įvairaus tipo pabraukimus ir pan. Tam, kaip ir VŽ formavimui, naudojamos įvairios VT modeliavimo kalbos, kuriose saugomų VT struktūros skiriasi. Viena kalba gali saugoti siaurą, o kita sudėtingą (platų) VT žodyną (t.y. VT elementai, žymėjimas). Taip pat VT gali būti aiškios ir nesiremti kitais žodynais, gali reikalauti vartotojų techninių žinių (kai taisyklėse naudojami specifiniai simboliai), bei gali būti formuojamos natūralia, vartotojui suprantama, kalba.

### 2.5.2.1 Veiklos taisyklių klasifikacija

Veiklos [5] taisyklė yra teiginys, kuriuo siekiama įtakoti ar valdyti elgesį ir informaciją organizacijoje. Jos turi būti išskiriamos tada, kai faktuose norima naudoti apribojimus. Pavyzdžiui, „klientas negali pateikti daugiau nei 15 užsakymų“ yra daugiau nei faktas – jame naudojamas apribojimas fakto egzemplioriui. Todėl tai yra taisyklė – verslo sistemos valdymo, o ne struktūrinio aspekto, dalis.

Atsižvelgiant į VT struktūrą, skiriamos įvairios VT rūšys:

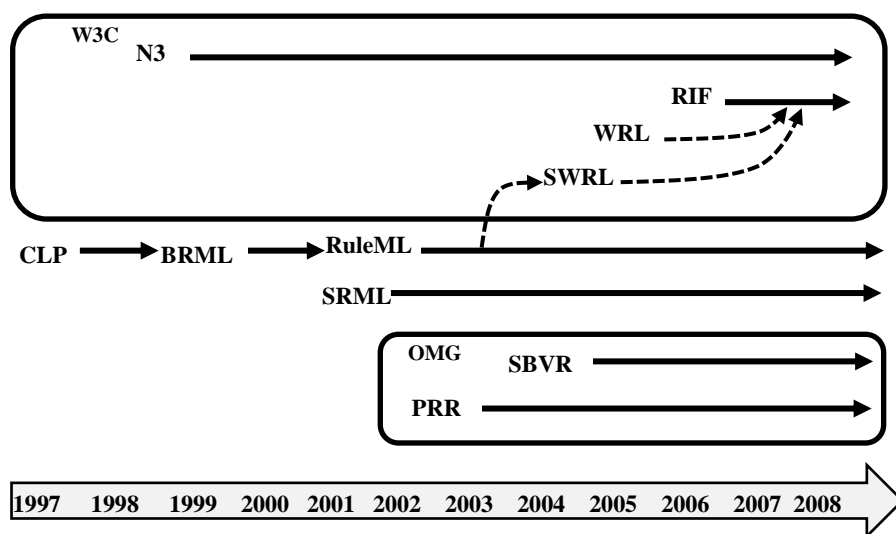
1. **Integralumo taisyklės** (angl. *Integrity rules*) apribojimams išreikšti. Šios taisyklės paprastai aprašo priimtinus/laukiamus ryšius tarp duomenų elementų. Pavyzdžiui, kiekvienas projektas privalo turėti vieną ir tik vieną projekto vadovą.
2. **Išvedimo taisyklės** (angl. *Derivation rules*) sąlygoms, kurios atsiranda pabaigoje, išreikšti. Šios taisyklės apibūdina faktų pagrįstumą ir gali būti naudojamos išvesti naujus faktus, paremtus žinomais faktais. Pavyzdžiui, labai svarbus klientas gauna 5% nuolaidą. John Doe yra tas klientas ir dėl to gauna 5% nuolaidą.
3. **Atoveikio (reakcijos) taisyklės** (angl. *Reaction rules*) arba dar žinomos kaip įvykio sąlygos veiksmo (angl. *Event-Condition-Action (EAC)*), alternatyvaus varianto veiksmo taisyklės ( angl. *Alternative-action rules*) arba sąlygos, kad tik po kai kurių kodo vykdymo dalių viskas turi būti tiesa (angl. *Post-conditions*). Tai tiksliai apibrėžia trigerį (priežastį), kuris aktyvuoja taisyklės įvertinimą, sąlygą, kuri bus įvykdyta, ir sekančią veiklą, kuri bus vykdoma, jei tik apibrėžta sąlyga priimta. Pavyzdžiui, atoveikio (reakcijos) taisyklės vertinimas yra sukeliamas (angl. *Triggered*) tuo metu,

kai nauja sąskaita yra gaunama. Jei sąskaita yra daugiau nei 1000 litų, tuomet inicijuojama darbų vadovo peržiūra .

4. **Našumo (gamybos) taisyklės** (angl. *Production rules*) taip pat žinomos kaip veiksmo taisyklės (angl. *Action rules*). Tai yra atitikmuo atoveikio (reakcijos) taisyklėms, tačiau neapibrėžia tam tikrų aplinkybių, dėl kurių atliekamas vertinimas. Pavyzdžiui, jei paskutiniuose 10 įtaisų nerasta jokių defektų, tada visa įtaisų grupė yra tvirtinama kaip kokybiška.
5. **Transformacijos taisyklės** (angl. *Transformation rules*) apriboja objektų būsenos kitimą. Pavyzdžiui, darbuotojų amžius gali kisti nuo 30 iki 31, bet ne nuo 31 iki 30.

Bendrai, VT apibrėžia, kas privalo įvykti, o ne kaip kažkas yra atlikta.

Veiklos taisyklėms pristatyti laikui bėgant buvo sukurta daug formatų (žiūrėti 2.10 pav.). Šios kalbos skiriasi tarp (angl. *Vary between*) tyrimų prototipų (pvz.: N3), konkrečių tiekėjo formatų (pvz.: Fair Isaac's SRL or ILOG's IRL) ir pasiūlymų VT, pagrįstų XML kalba, mainams (pvz.: SRML, PRR ir SBVR). Taip pat skiriasi VŽ ar VT struktūrų sudėtingumu, suprantamumu vartotojams ir pan.



2.10 pav. Veiklos taisyklių kalbų evoliucija

Taigi objekto analizei atlikti parinktos trys taisyklių modeliavimo specifikacijos, kurios teikia bendrus savo žodynų paaiškinimus:

- SBVR nėra vykdomoji kalba (angl. *Executable language*). Šis standartas pateikia žodyną, kuriuo gali būti pagrįsta dalis gramatikos. Dėl šios priežasties, SBVR įtraukimas į ontologijų analizę yra naudingas.

- SRML standartas įtraukiamas į analizės sąrašą kaip pristatomasis taisyklių modeliavimo kalbos su mažu žodynu pavyzdys. Ši kalba taip pat yra aiškios loginės struktūros ir nepriklausoma nuo kitų žodynų.
- PRR standartas parinktas, nes jis buvo sukurtas tos pačios organizacijos kaip ir SBVR, tačiau su ketinimu veikti kaip apsisikeitimo formatas komercinėse PRM (angl. *Production Rule Management*) sistemose, apimančiose VT įvairovę.

## 2.6 Veiklos žodyno ir veiklos taisyklių standartai

### 2.6.1 Veiklos žodyno ir veiklos taisyklių semantika (SBVR)

SBVR [1] (angl. *Semantics of Business Vocabulary and Business Rules*) – VŽ ir VT semantika – sukurtas OMG organizacijos. Tai yra sprendimas, kuris leidžia sukurti vartotojui lengvai suprantamą veiklos schemą (projektą), naudojantis VŽ ir VT.

SBVR [11] iniciatyva siekiama užfiksuoti veiklos faktus ir veiklos taisykles, kurios gali būti išreikštos formaliai arba neformaliai. VT išraiškos yra *formalios* tik tada, jei jos išreikštos tik faktų tipais iš anksto pripažintoje veiklos srities schemoje, taip pat tam tikrais loginiais / matematiniais operatoriais, kvantatoriais (kiekiniais žodžiais) (angl. *Quantifiers*) ir pan. Formalūs taisyklių sakiniai gali būti transformuojami į logines formuluotes, kurios yra naudojamos mainams su kitais taisyklėmis pagrįstais programinės įrangos įrankiais. *Neformalūs* taisyklių sakiniai gali būti pakeisti neinterpretuojamais komentarais.

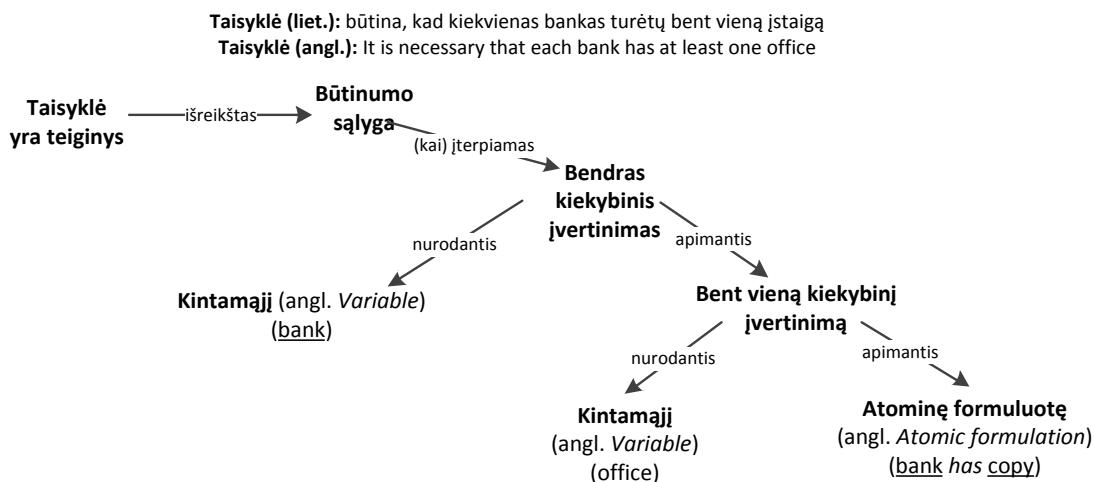
**SBVR tikslai, galimybės.** SBVR [1] taip pat sukurtas siekiant, kad skirtingos bendruomenės (veikloje dalyvaujantys asmenys, IT asmenys) galėtų bendrai naudotis veiklos semantika. SBVR leidžia veiklos analitikams ar kitiems veikloje dalyvaujantiems asmenims, kurie suinteresuoti VT rašymu, veiklos artefaktus išreikšti natūralia kalba. Veikloje dalyvaujantys asmenys veiklos taisykles gali rašyti savo mokama kalba ir gali sukurti jom semantinį modelį.

#### **Loginiai formulavimai (SBVR veiklos taisyklės ir struktūra)**

SBVR VT yra sakiniai, kurie apibūdina organizacijos struktūrą ir elgseną. Taisyklės, apibūdinančios struktūrą, yra vadinamos struktūrinėmis taisyklėmis (angl. *Structural Rules*), o apibūdinančios elgseną – operacinėmis taisyklėmis (angl. *Operational Rules*).

SBVR leidžia 4 VŽ ir VT kūrimo išraiškų tipus. Siekiant išanalizuoti SBVR veiklos modelį, reikia nustatyti 4 išraiškų tipus: terminą, vardą, veiksmožodžius, raktažodžius (tai paaiškinta „**SBVR veiklos žodyno terminologija**“ skyrelyje).

Kiekviena taisyklė aprašo veiklos artefakto semantiką. Tam atlikti, SBVR pateikia struktūrą, kuri dar vadinama loginiu formulavimu. Tai yra abstrakti ir nuo kalbos nepriklausanti sintaksė, aprašanti taisyklės prasmę. To pavyzdys pateikiamas 2.11 paveiksle.



2.11 pav. SBVR taisyklės loginė formulotė

### SBVR veiklos žodyno terminologija

SBVR VŽ yra veiklos esybių, jų egzempliorių ir ryšių tarp jų rinkinys, kurį savo raštuose ar kalboje veiklos kurso metu gali naudoti bet kokia organizacija.

SBVR turi specialios paskirties žodžių ir terminologijos rinkinį, skirtą siekiant aprašyti VŽ ir veiklos taisykles. Rinkinį sudaro:

- **Terminai** (angl. *Terms*). Tai yra daiktavardis arba žodžių grupė, kuri bendrai gali būti naudojama veiklos esybėms nurodyti. Pavyzdžiui: „bankas“ arba „investicinis bankas“.
- **Vardas** (angl. *Name*). Tai yra žodžiai, naudojami pristatant tam tikro termino (esybės) egzempliorių. Tarkim, „SBI“ yra esybės **bankas** egzempliorius.
- **Veiksmazodis** (angl. *Verb*). Naudojamas fakto tipui – dažniausiai veiksmazodžiui, prielinksniui ar jų kombinacijai pažymėti. Pavyzdžiui, *turi, naudoja, dažo*.
- **Raktažodis** (angl. *Keyword*). Naudojamas kitiems žodeliams, iš kurių sudaromas sakiny, pažymėti. Pavyzdžiui, **bent, jei, ar, kai, nors** (angl.: **the, an, a, in, into**, etc.) ir t.t.
- **Fakto tipas** (angl. *Fact Type*). Tai yra sakiniai, kuriais aprašomi ryšiai tarp terminų. Nustatant ryšį tarp dviejų terminų, naudojamas šablonas „terminas – veiksmazodis – terminas“. Taip yra visiškai aišku, kad bendri ryšiai tarp 3 esybių gali būti lengvai išskirstyti į daugiausiai 3 dvejetainius ryšius (angl. *3 binary entities*). Pavyzdžiui, fakto tipas „klientas turi sąskaitą yra narys“ teigia, kad klientas yra susijęs su sąskaitą ir

sąskaita yra susijusi su nariu, o asmuo, kuriam priklauso sąskaita, bus nariu. Šis ryšys gali būti išskaidytas į du ryšius, aprašytus dviem fakto tipais: „[klientas turi sąskaitą](#)“ ir „[klientas yra narys](#)“.

## 2.6.2 Bendrinė taisyklių žymėjimo kalba (SRML)

SRML [12] ( angl. *Simple Rule Markup Language*) yra XML kalba pagrįstas taisyklių kalbos formatas, priimtas ILog programinės įrangos prekiautojų kompanijos. Tai [14] – suderinta (angl. *Generic*) taisyklių kalba, sudaryta iš kalbos struktūrinių elementų (angl. *Constructs*) poaibio, panašiu į populiarius tiesioginio išvedimo taisyklių variklius (angl. *Engines*). Ji nenaudoja konkretaus tiekėjo patentuotų kalbų (angl. *Proprietary language*), todėl taisyklės gali būti lengvai transformuotos į kitą taisyklės sistemos kalbą. Ji tampa naudinga tarpine kalba (angl. *Interlingua*) taisyklių (tarp Java taisyklės sistemų) mainams.

SRML [13] yra santykinai abstrakčiame žymėjimo, kuris gali būti lengvai suprantamas programuotojams, lygmenyje.

### SRML veiklos žodynas, veiklos taisyklės ir jų analizavimas

SRML struktūros [21] aprašymas pateikiamas XML formatu. SRML žodyno struktūros pagrindą sudaro taisyklių rinkinys (angl. *Ruleset*). Tai yra šakninis SRML XML dokumento elementas, kuris sudarytas iš taisyklių sąrašo. Kiekviena [20] taisyklė turi aprašymą, taip pat sąlygos ir veiksmų dalis.

### Taisyklė, naudojant tradicinius terminus [14,19]

```
Rule: Discount
Description: If
              the total purchase amount of a shopping cart is > 100$
              Then
                Set the discount for the shopping cart to 0.1%
```

### Tradicinė sintaksė

```
rule Discount {
    when {
        ?s:ShoppingCart(purchaseAmount > 100);
    } then {
        update ?s { discount = 0.1; }
    }
}
```

### SRML struktūra:

```
<rule name="Discount">
  <conditionPart>
    <simpleCondition className="ShoppingCart" objectVariable="s">
      <binaryExp operator="gt">
        <field name="purchaseAmount"/>
        <constant type="float" value="100"/>
      </binaryExp>
    </simpleCondition>
  </conditionPart>
  <actionPart>
    <modify>
      <variable name="s"/>
      <assignment>
```



```

        <field name="discount"/>
        <constant type="float" value="0.1"/>
    </assignment>
</modify>
</actionPart>
</rule>

```

### SRML veiklos žodyno terminologija

Taisyklių rinkinyje SRML, kaip ir kitų standartų VT, turi specialios paskirties žodžių ir terminologijos rinkinį, skirtą siekiant aprašyti VŽ ir VT. Rinkinį sudaro 2 pagrindinės dalys:

- **Sąlygos** [20] (angl. *Condition part*). Kiekviena taisyklės sąlygos dalis privalo būti sudaryta iš mažiausiai vienos sąlygos. Rezultate (SRML sugeneruotame faile) yra atskiriamos paprastos sąlygos (<*simple conditions*>), „ne“ sąlygos (angl. *Not condition*), peržvalgos sąlygos (<*lookup conditions*>) ir daugiapakopės sąlygos (<*multistage conditions*>). Trys pirmieji sąlygų tipai susideda iš išraiškų, kurių pagrindu nagrinėjami ir aprašomi duomenys apie jutiklių (angl. *Sensor*) adapterius. Daugiapakopės sąlygos susieja sąlygą, nurodančią duomenis, kuriuos teikia jutiklio adapteriai (paprasta sąlyga arba „ne“ sąlyga), su sąlyga, kuri tiksliai nusako parametrus atitinkamo adapterio aptikimui (peržvalgos sąlyga). Sąlygos [21] yra sudarytos iš bandomųjų išraiškų (angl. *Test expressions*) ir gali būti paprastos sąlygos (angl. *Simple conditions*) arba neturėti jokių sąlygų („ne“ sąlygos (angl. *Not conditions*)).
- **Veiksmai** [20] (angl. *Action part*). Ši dalis susideda iš veiksmų, kurie gali būti kintantys aprašai (angl. *Declarations*) arba priskyrimai (<*assignments*>), taip pat kaip ir objektų kūrimo (<*create*>), redagavimo (<*modify*>) ar šalinimo (<*delete*>) instrukcijos. Šios išraiškos, naudojamos visose taisyklės dalyse, gali būti kintamieji (angl. *Variable*) arba laukai (angl. *Fields*), kuriems gali būti priskiriamos reikšmės (angl. *Values*), konstantos (angl. *Constants*), aritmetinės (angl. *Arithmetic*) arba taip-ne (angl. *Boolean*) išraiškos, taip pat operacijos (angl. *Operations*) ir objektai (angl. *Objects*).

Aukščiau pateiktame [14] pavyzdyje pagrindiniai elementai – sąlygos ir veiksmai – yra pradedami ir užbaigiami žymėmis <...> </...>. Tai yra, sąlygos dalis prasideda <*conditionPart*> ir baigiasi </*conditionPart*> žymėmis, veiksmo dalis prasideda <*actionPart*> ir baigiasi </*actionPart*> žymėmis.

### 2.6.3 Našumo taisyklių vaizdavimo standartas (PRR)

PRR [15] (angl. *Production Rule Representation*) sukurta siekiant nustatyti našumo taisyklių atvaizdavimo UML modeliuose poreikį (pavyzdžiui, VT modeliavimas kaip modeliavimo procesas).

Našumo taisyklė [15] (angl. *Production Rules*) yra programavimo logikos teiginys, kuris tiksliai apibrėžia vieno ar kelių veiksmų vykdymą tuo atveju, jei įvykdytos jos (taisyklės) sąlygos. Todėl šiose taisyklėse naudojama operacinė semantika (angl. *Operational semantic*), formalizuojanti būsenos pasikeitimus.

Yra du našumo taisyklių tipai, kurie turi būti sumodeliuoti PRR specifikacijai, atsižvelgiant į tai, ar našumo taisyklė yra:

- Nepriklausomai nuo tvarkos vykdoma tiesioginio išvedimo taisyklių sistemoje (angl. *In a forward chaining rule engine*), tokioje kaip *Rete-based* taisyklių sistemos.
- Vykdoma nuosekliai procedūrinėje taisyklių sistemoje.

#### Tiesioginio išvedimo taisyklės (angl. *Forward-chaining Inference rules*)

Tiesioginio išvedimo taisyklė yra našumo taisyklė, apibrėžta neatsižvelgiant į vykdymo tvarką, nors vykdymo tvarka išvadų sistemoje (angl. *Inference engine*) yra valdoma.

Tokia taisyklė yra aprašoma taip:

$$\text{if [condition-list] then [action-list]}^1$$

Kartais [15] ši taisyklė yra išplėsta įtraukiant žodį „else“ :

$$\text{if [condition-list] then [action-list] else [alternative-action-list]}$$

nors ši forma nėra svarstoma PRR; visos taisyklės, kuriose naudojamas „else“ teiginys, gali būti sutrauktos iki anksčiau minėtos formos (be „else“) ir semantika, interpretuojant kai „else“ veiksmai yra vykdomi, gali būti sudėtinga kai kuriose galutinėse (išvadų) schemose.

*Sąlygų sąrašas* (angl. *Condition-list*) [15] aprašomos taisyklės sąlygos, kurios, savo ruožtu, nusako:

- Ryšius, kurie aprašo duomenų sąrašo sekas iš numatyto konteksto (situacijos), aprašyto lokalaus objekto modeliu ir nurodyto taisyklės veiksmuose (angl. *In the rule actions*).

---

<sup>1</sup> Vietoj *if ... then* dažnai naudojamas junginys *while ... then*

- Sąlygų išraiškas, kurios apriboja (angl. *Constrain*) duomenų sąrašo sekas į kai kuriuos konteksto (situacijos) poaibius.

*Veiksmų sąrašų* (angl. *The action-list*) aprašomi:

- Taisyklės veiksmi. Tai yra išsireiškimai, kurie apibūdina elgseną, kas pakeičia sistemos būklę. Šie išsireiškimai apibūdinami kaip ryšiai (angl. *Bindings*) ir vykdomi tik su tomis sekomis, kurios atitinka taisyklės sąlygos apribojimus, jos vykdymo (apdoravimo) metu.

**Tiesioginio nuoseklaus išvedimo taisyklės (angl. *Forward-chaining sequential production rules*)**

- Tiesioginio procedūrinio išvedimo taisyklė apibrėžiama atsižvelgiant į vykdymo tvarką, nes taisyklės yra vykdomos iš anksto apibrėžta tvarka. Tokios taisyklės taip pat gali būti tiksliai apibrėžtos kaip ryšiai (angl. *Bindings*), sąlygos ir veiksmi.

### **PRR standarto tikslai**

PRR [18] specifikacija sukurta siekiant sunorminti (standartizuoti) VT modeliavimą. Pagrindinis tikslas – tiekti VT sistemas (angl. *Business Rules Engines*) su standartiniu apsikeitimo formatu.

PRR [15] taip pat buvo kuriama siekiant:

1. pagreitinti našumo taisyklių komponentų priėmimą kiekvieną dieną naudojamose programinės įrangos sistemose.
2. pagerinti našumo taisyklių modeliavimą, ypač susijusių su UML kalba (angl. *Especially with respect to UML*).
3. leisti skirtingiems tiekėjams, teikiantiems našumo taisyklės realizavimo paslaugas, bendradarbiauti (sąveikauti).

PRR [16] atitinka reikalavimus, susijusius su veiklos taisyklėmis, programinės įrangos sistemomis, OMG standartais ir kitais taisyklių standartais.

### **PRR veiklos žodynas, veiklos taisyklės ir jų analizavimas [16,17]**

Kadangi PRR priklauso našumo taisyklių grupei, tai pirmiausiai reikia paminėti, kad jų pagrindas yra našumo taisyklių rinkinys. Tai yra, jį apibrėžia:

- taisyklių, susijusių su VP ar veiklomis, kaip funkcinių vienetų kaupimas.
- taisyklės iškvietimo (angl. *Invocation*) sąsaja.

Žvelgiant iš architektūrinės ir struktūrinės pusės, taisyklių rinkinys UML požiūriu yra elgsena (angl. *Behaviour*). Taisyklės įtakoja objektų rinkinį, o objektus apibūdina taisyklių rinkinio parametrai, kontekstas (iškvietimo metu).

Vykdyimo pabaigoje pakeistos reikšmės pateikia rezultatą arba taisyklių rinkinio iškvietimo (angl. *Invocation*) išvestis.

### **Taisyklė, naudojant tradicinius terminus: [16]**

„If the shopping cart contains between 2 and 4 items and either the purchase value is greater than \$100 and the customer category is gold or the purchase value is greater than \$200 and the customer category is Silver then apply a 15% discount on the shopping cart value”.

#### **Tradicinė sintaksė:**

```
rule discount {
  when
  {
    ?customer: Customer();
    ?shoppingCart: ShoppingCart(customer == ?customer);
    evaluate((?shoppingCart.containsItemsInRange(2, 4)) &&
      (((?shoppingCart.getValue() > 100d) &&
        (?customer.category equals "Gold")) ||
        ((?shoppingCart.getValue() > 200d) &&
          (?customer.category equals "Silver"))));
  }
  then
  {
    modify ?shoppingCart
    {
      shoppingCart.discountValue = shoppingCart.discountValue + 15f);
    }
  }
}
```

#### **PRR OCL struktūra:**

```
Rule discount
ruleVariable:
  ?customer: Customer = Customer->any()
  ?shoppingCart: ShoppingCart = ShoppingCart->any(c: customer |
c=?customer)
Condition:
  (?shoppingCart.containsItemsInRange(2, 4)
  and
  (((?shoppingCart.items->collect(i:Item|i.value))->sum())>100
  and
  ?customer.category == "Gold")
  or
  ((?shoppingCart.items->collect(value))->sum() > 200
  and
  ?customer.category == "Silver")))
Action:
  shoppingCart.discountValue = shoppingCart.discountValue+15f
```

### **PRR veiklos žodyno terminologija [16, 17]**

Taisyklių rinkinyje PRR turi specialios paskirties žodžių ir terminologijos rinkinį, skirtą siekiant aprašyti VŽ ir taisykles. Rinkinį sudaro:

- **Taisyklės kintamasis** (angl. *Rule Variable*). Yra dviejų tipų – *standartinis* ir *taisyklės* – kintamieji. Standartinis turi **tipą** ir pasirenkamą pradinę išraišką. Tokie kintamieji yra apibrėžiami taisyklių rinkinio lygyje. Taip pat turi **tipą** ir sritį (angl. *Domain*), kurią apibrėžia filtras, taikomas duomenų šaltiniams. Toks kintamasis gali būti apibrėžtas taisyklės lygyje arba taisyklių rinkinio lygyje; pastaruoju atveju kintamojo aprašymai yra prieinami visoms taisyklėms, esančioms taisyklių sąrašė.
- **Sąlygos** (angl. *Conditions*). Sąlygos teiginiai yra taip-ne išraiškos, kurios filtruoja taisyklės kintamųjų egzempliorius ir yra naudojamos kaip sujungimai tarp jų (išraiškų).
- **Veiksmai** (angl. *Actions*). Šioje dalyje aprašant taisyklę, nurodoma, ką reikia daryti kiekvienai sekai, kuri tenkina sąlygas.

#### 2.6.4 Veiklos taisyklių standartų palyginimas

2.2 lentelėje pateikiama apibendrinta SBVR, SRML ir PRR informacija. Taip siekiama išskirti kiekvieno standarto privalumus ir trūkumus. Pluso ženklas žymi, kad funkcija galima, minusas – negalima, o +- reiškia, kad reikia pastangų ar kitų papildomų resursų to aspekto įgyvendinimui.

2.2 lentelė. Veiklos taisyklių standartų palyginimas

	<b>SBVR</b>	<b>SRML</b>	<b>PRR</b>
Kūrėjas	OMG (angl. <i>Object Management Group</i> )	IBM ILOG	OMG (angl. <i>Object Management Group</i> )
Suprantamumas vartotojams (natūrali kalba)	+	-	-
Formali struktūra	+	+	+
XML failo generavimas	-	+	+
Sistemos kūrimo etapas	reikalavimų	projektavimo	projektavimo
Sudėtingas apribojimas	+	+-	+-
Vartotojai	Visi vartotojai	Techninių žinių IT srityje turintys vartotojai	Techninių žinių IT srityje turintys vartotojai
MDA lygmuo	CIM	PIM	PIM
Veiklos žodynas	Skirtas apibūdinti VT žodžius	Skirtas VT struktūrai apibūdinti	Skirtas VT struktūrai apibūdinti

Standartai tiriami keliais aspektais: suprantamumas vartotojams, XML failo generavimas, sistemos kūrimo etapas ir pan. Kiekvienas iš standartų turi ir plusų ir minusų. SBVR išskyrė tuo, kad tik šios modeliavimo kalbos taisyklės yra aprašytos natūralia kalba. Todėl jos lengvai suprantamos vartotojams, kurie gali neturėti techninių ar kitų žinių, reikalaujamų veiklos

taisyklėms bei VŽ suprasti. SBVR standartas teikia lengvai suprantamas taisykles ir VŽ. Šiuo atveju žodynas skirtas aprašyti kiekvieną VP modelio elemento pavadinimą, kai tuo tarpu SRML ir PRR teikiama žodyno terminologija labiau skirta VT struktūrai apibūdinti. Kas liečia VŽ kaip galutinį rezultatą, pateikiamą vartotojui, tai SBVR standarte VŽ suformuojamas aiškiai, o kituose nagrinėjamuose standartuose žodžiai yra saugomi duomenų bazėse kaip klasės, todėl VŽ tampa sunkiau suprantamas. Be visa to, SRML ir PRR yra paremtos XML kalba, todėl gali būti generuojamas XML failas taip pat modeliais grindžiamoje architektūroje (MDA) pasiskirstę nuo platformos nepriklausomame (PIM) lygmenyje, kuriame VŽ bei VT aprašytos specifiniais sakiniais, žodžiais, t.y., naudojant specifinius simbolius. Dar SRML ir PRR yra panašūs tuo, kad naudojami sistemos projektavimo metu.

## 2.7 Veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų modelio metodai

VP modelių diagramos gali turėti tekstinius apribojimus (formalius arba neformalius) bei detalius aprašymus (VŽ). Formalūs apribojimai naudojami OCL notacijoje, kuri UML vartotojams suteikia daugiau tikslumo, o neformalūs apribojimai (dažniausiai tai yra komentarai) naudojami BPMN notacijos. Apribojimai turi būti aprašomi grafiškai. Tokiu būdu vartotojas gali identifikuoti vietą, kurioje turi būti apribojimas.

Dažnai prie VP modelio (grafiškai pateiktų apribojimų) palanku (patogu) pridėti ir detalių aprašą – VŽ, kuris, kaip ir apribojimai, gali būti formuojamas įvairiais būdais, pradedant nuo:

- formavimo rankiniu būdu, kada sudaromas dažniausiai organizacijoje naudojamų terminų, faktų ir kitų žodžių sąrašas;
- taip pat rankiniu būdu išgaunant detalių aprašą iš veiklos modelio, kas užima daugiau laiko ir reikalauja daugiau dėmesio bei vartotojo techninių žinių;
- išgavimo iš žinių bazės, kurioje saugoma informacija naudojama plačiau, nei tik pateikti vartotojams organizacijoje naudojamus terminus ir baigiant automatiniu VŽ bei apribojimų išgavimu.

Gero duomenų modelio kūrimo terminologijos svarba yra suprantama visose organizacijose, tačiau sudėtinga sudaryti išsamų dokumentą, kuris galėtų aiškiai apibrėžti VT. Kai taisyklių skaičius siekia šimtus ar netgi tūkstančius, tai VŽ turi būti išbaigtas (išsamus). Tokio dokumento struktūrą daugiausiai sudaro du pagrindiniai komponentai: *terminai* ir *faktai*, kurie turi identifikuoti daiktus. Veiklos žodyne terminas (dažniausiai daiktavardis) atitinka tam tikrą reikšmę, kuri suteikiama konkrečiam kontekstui, versle. Nuo kiekvienos reikšmės priklauso terminą apibūdinantys faktai bei taisyklės. Aiškiam VŽ sudaryti, terminas turi atitikti kelis kriterijus, t.y., jis turi reprezentuoti:

- pagrindinius verslo dalykus, t.y., terminus, kurie gali būti išgaunami iš kitų terminų. Tokie terminai vadinami *pagrindiniais*.
- dalykus, kurie yra nedalomi, t.y., ne sudėtiniai. Terminai *atominiai*.
- dalykus, kurie yra pažįstami, o ne galimus įvykius (procedūras, veiksmus, procesus, kurie kuria žinias). *Pažinūs* terminai.

Faktai turi atvaizduoti/apibūdinti terminus (pavyzdžiui, „klientas“ (angl. *Customer*) ir „užsakymas“ (angl. *Order*)), bet ne procesus („priimti kliento užsakymą“ (angl. *Take customer order*)).

Kad terminai būtų aiškiai suprantami įvairių sričių vartotojams, faktai turi būti paprasti, konstatuojami sakiniai, kurie susiję su atitinkamais terminais. Paprastai faktas išreiškiamas užbaigtu sakiniu, dažniausiai griežtos struktūros: subjektas – veiksmažodis – subjektas (pavyzdžiui, „Customer places order“ arba „Order is included into shipment“). Veiksmažodžiai turi apjungti keletą terminų. Be visa to, faktas apibūdina tai, kas žinoma, jame naudojami apribojimai, nes tai jau yra VT dalis, o taip pat faktai išplečia pagrindinį VŽ skirtingais būdais.

Taigi VŽ, reprezentuojantis VP modelius, nepriklausomai nuo to, ar jis sudaromas rankiniu ar automatinu būdu iš modelio, apibūdina ir kiekvieną modelio elementą.

### 2.7.1 UML veiklos diagramos elementų žymėjimas SBVR taisyklėse

Šioje [1] dalyje aptariamas SBVR komponentų žymėjimas UML veiklos diagramos komponentuose (angl. *Mapping of SBVR components to UML activity diagram components*).

1. **Pirminis mazgas** (angl. *Initial Node*). Tai yra veiklos diagramos pradžios simbolis. Jis neturi didelės įtakos, bet yra reikšmingas, nes parodo scenarijaus atskaitos tašką. Pagal nutylėjimą jis vadinamas „pradžia“ (angl. „*Start*“).
2. **Veiklos mazgas** (angl. *Activity Node*). Šis simbolis reiškia tam tikrus veiksmus, vadinamus indeksiniais (angl. *Transitive*) veiksmažodžiais. Pavyzdžiui, faktas (angl. *Fact type*) „vartotojas įdeda kortelę“ yra teiginys, kuriame indeksinis veiksmažodis „įdeda“. Tai parodo veiksmą „kortelės įdėjimas į automatą“. Šie tipai gali būti traktuojami kaip veiklos.
3. **Veiklos ryšys** (angl. *Activity Edge*). Tai yra įvykių rinkinys, apsauginės (angl. *Guard*) sąlygos ir veiksmai, kurių įvykdymas įgalina perėjimą iš vieno veiklos mazgo į kitą. Ši perėjimą sukelia įvykis. Po perėjimo sukėlimo tikrinamos sąlygos: jei sąlyga yra teisinga, tada atsiranda atsakomasis veiksmas ir vykdoma kita veikla.

Paprastai SBVR taisyklė gali būti aprašyta taip:

(ankstesnis) įvykis, jei <siūloma išraiška 2>, tada  
<siūloma išraiška 1> <sup>2</sup>.

Vis dėlto veiklos taisyklėms aprašyti yra naudojamas toks formatas:

<siūloma išraiška 1>[jei <siūloma išraiška 2>]<sup>3</sup>.

Taigi, toks SBVR taisyklių tipo UML AD elementų žymėjimas sukurs trumpesnę sandūrą. Siūloma išraiška 2 naudinga aiškinantis atsargumo sąlygas (angl. *Guard condition*), o siūloma išraiška 1 naudingas siekiant išsiaiškinti veiksmą.

4. **Atsargumo sąlyga** (angl. *Guard Condition*) įgauna „privaloma, kad kiekvienas automatas reikalauja tik vieno slaptažodžio, jei vartotojas įdeda kortelę“ (angl. „**it is obligatory that each atm request exactly one password if a user inserts card**“) VT formatą. Siūloma išraiška 2 „jeigu“ (angl. *If*) sąlygoje nurodo fakto tipą „vartotojas įdeda kortelę“. Priklausomai nuo fakto tipo kuriamas taip-ne kintamasis, o tada atsargumo sąlyga tampa „įdeda kortelę = true“.

2.3 lentelė. Veiklos taisyklių pagal pateiktą pavyzdį priskyrimas

Fakto tipas (faktas - veikla)	Atitinkamas taip/ne kintamasis	Atitinkama sąlyga/būklė
Vartotojas įdeda kortelę	Įdeda_kortelę	įdeda kortelę = true
Automatas reikalauja slaptažodžio	Reikalauja_slaptažodžio	reikalauja slaptažodžio = true

5. **Veiksmas** (angl. *Action*). Jais remiamasi pereinant nuo vienos veiklos prie kitos. Pavyzdžiui, jei „vartotojas įdeda kortelę“, tada automatas neatliks „reikalauja slaptažodžio“ veiksmo. Čia, tiesiog, imama [siūloma išraiška 1] ir ieškomas atitinkamas fakto tipas. Sujungiant veiksmą ir paskutinį fakto žodį, sukuriamas veiksmo vardas, pavyzdžiui, „reikalauja\_slaptažodžio()“ kaip parodyta 2.4 lentelėje.

2.4 lentelė. Fakto tipai ir jai priskiriami veiksmai

Fakto tipas (faktas - veikla)	Atitinkamas veiksmas
Vartotojas įdeda kortelę	įdeda_kortelę()
Automatas reikalauja slaptažodžio	reikalauja_slaptažodžio()

6. **Išsišakojimo/sujungimo mazgai** (angl. *Fork/Join Node*). Tai struktūra, kai įeina viena veikla, o lygiagrečiai išeina kelios veiklos. SBVR taisyklė „privaloma, kad kiekvienas automatas atspausdina tik vieną kvitą ir kiekvienas automatas išstumia kortelę, jei bankas pateikia netinkamą sąskaitos pranešimą“ (angl. „*It is obligatory that each atm print exactly one receipt and each atm eject the card if bank return bad account*“).

2 (angl. *upon event, if <propositional expression 2>, then <propositional expression 1>*)

3 (angl. *<propositional expression 1> [if <propositional expression 2>]*).



*message*“) aprašo dviejų veiklų – „automatas atspausdina kvitą“ ir „automatas išstumia kortelę“ – vykdymą, jei bankas pateikia netinkamą sąskaitos pranešimą. Esant tokiai situacijai, naudojamas *išsišakojimo mazgas*. Pagal pavyzdį tai būtų: vienas įėjimo perėjimas (veikla) su atsargos sąlyga „pateikia netinkamą sąskaitos pranešimą == true“ ir du išeinantys, lygiagrečiai vykstantys, perėjimai nukreipiami į dvi veiklas: „automatas atspausdina kvitą“ ir „automatas išstumia kortelę“. *Sujungimo mazgas* yra atvirkščias aptartajam. Šiuo atveju daug lygiagrečių įėjimų ir tik vienas išėjimas, t.y., daug atsargos sąlygų ir vienas išeinantis perėjimas (angl. *Transition*).

7. **Veiklų grupė** (angl. *Activity Group*). UML veiklos diagramose grupė yra žinoma kaip plaukimo takelis. Šiuo elementu aprašomi asmenys, organizacijos ir pan., kurie vykdo veiklą. Esysbė, atliekanti veiklą, nurodoma kaip jos pradininkas. Veiklos pateikimas gali būti dviejų tipų: aktyvios ir pasyvios formos. Sakinyje su aktyvios formos veiksmoždžiu, veiklos pradininkas yra sakinio objektas (pavyzdžiui, „klientas“ yra sakinio „klientas pateikia užsakymą“ veiklos pradininkas). Veikla gali būti ir pasyvios formos (pavyzdžiui, „užsakymas yra pateikiamas kliento“). Abi šios formos reiškia tą patį (t.y., pradininkas yra tas pats – „klientas“).
8. **Veiklos pabaigos mazgas** (angl. *Activity Final Mode*). Tai yra taškas, kuriame visos veiklos, esančios veiklos diagramoje, baigiasi.

### 2.7.2 Veiklos taisyklių ir veiklos procesų realizavimas, naudojant SOA

Vienas [26] iš svarbiausių stimulų norint naudotis į servisu orientuotą architektūra (angl. *Service Oriented Architecture*) yra įmonės judrumas (angl. *Agility*) ir neišvengiamų pasikeitimų įmonėje minimizavimas. Tai pasiekama atskiriant realizuojamus artefaktus. Pagrindinė technologija, siekiant tai įvykdyti, yra dekompozicija ir paketavimas (angl. *Encapsulation*). SOA dekompozicijos atveju apibrėžiami servisi, reprezentuojant labiau nusistovėjusius artefaktus. Be to, SOA dekompozicija apibrėžia VP, reprezentuojant nepastovesnius artefaktus.

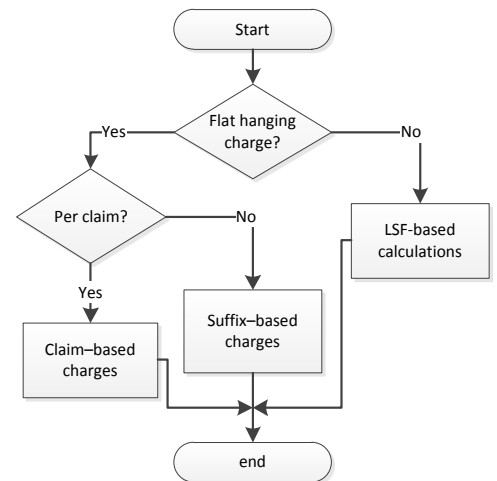
#### Veiklos komponentų realizavimas

Kaip pavyzdys aptariamas „mokesčių apskaičiavimo priežiūros“ teiginys. Jis gali būti priklausomas nuo buto priežiūros mokesčio (angl. *Flat handling charge*) arba nuostolio konversijos koeficiento (angl. *Loss conversion factor (LCF)*). Pirmuoju atveju, buto mokestis gali būti taikomas arba visam reikalavimui arba atskiriems reikalavimo indeksams. Šių trijų skirtingų pasirinkimų kombinacija sukuria tris skirtingas (aiškias) mokesčių apskaičiavimo priežiūros taisykles. Taisyklų rinkinys gali būti pristatomas kaip taisyklių rinkinys (2.12 pav.) arba kaip procesas (2.13 pav.).

```

If Flat handling Amount and Per Suffix
  For each suffix calculate
    Claim Handling Charge (Paid Loss) = Flat Amount
End If
If Flat handling Amount and not(Per Claim)
  For each claim calculate
    Claim Handling Charge (Paid Loss) = Flat Amount
    Distribute to each Suffix Pro Rata
End If
If not(Flat handling Amount) and not(LCF Factor)
  For each Suffix
    Claim Handling Charge (Paid Loss) =
      (Loss Subject to Claim Handling Charge (Paid Loss) * LCF Factor)
      - Loss Subject to Claim Handling Charge (Paid Loss)
  .....
Endif

```



2.12 pav. Mokesčių apskaičiavimo priežiūros teiginio pateikimas, naudojantis taisyklėmis

2.13 pav. Mokesčių apskaičiavimo priežiūros teiginio pateikimas proceso forma

Nagrinėjant VP modeliavimo priemones ir VT variklius nustatyta, kad kol kas siūloma VP ir VT kurti atskirai. Tai yra, sukurti VP ir VT ir tik tuomet tai integruoti. Vis dėl to, šis principas neefektyvus, kadangi negali būti užtikrinamas pilnas suderinamumas tiek VŽ lygmenyje, tiek pačiuose vartotojų reikalavimuose, kuriant VPD.

## 2.8 Analizės išvados

1. Analizės metu išanalizuoti VP ir VT modeliavimo standartai. Išnagrinėjus VP modeliavimo metodologijas (BPMN2, UML veiklos diagrama, Darbų sekų diagrama), nustatytas kiekvieno iš jų panaudojimo tikslas. BPMN2 metodologija skirta modeliuoti VP, orientuota į įmonės atstovus. UML projektuoja informacines sistemas, vartotojas privalo turėti techninių žinių, norėdamas tinkamai atvaizduoti realaus pasaulio procesus. Darbų sekų diagrama automatizuoja VP ir taip pat reikalauja techninių žinių.
2. Kiekvienas nagrinėtas standartas buvo vertinamas pagal tokius kriterijus kaip elementų skaičius standarte, suprantamumas tiek įmonės atstovams, tiek ir analitikams, galimybė transformuoti modelį į vykdomąjį kodą bei generuoti XML failą, apribojimų sudėtingumas, formalių struktūrų egzistavimas bei apribojimų aprašymas natūralia kalba. Svarbi funkcija yra XML failo generavimas, tai gali būti panaudojama formuojant VŽ bei VT. Nors kiekvienas standartas turi panašiai privalumų ir trūkumų, tačiau BPMN2 labiausiai atitiko vertinamus kriterijus, kadangi, priešingai nei kituose standartuose, BPMN2 nėra neapibrėžtumo, t.y. yra tik pliusai ir minusai, o ne plius/minusai ir pan.
3. Išanalizavus VT modeliavimo standartus (SBVR, SRML, PRR) nustatyta, kad kiekvienas iš nagrinėtų standartų turi formalią struktūrą, tačiau jos yra skirtingai suprantamos vartotojams. SBVR taisyklės yra orientuotos į IT žinių neturinčius

virtotojus, kadangi jos yra suformuotos natūralia kalba. Įvertinus VŽ bei VT pritaikymą, nustatyta, kad geriausiai kriterijus atitinka SBVR standartas.

4. Kaip ir VP modeliavimo, taip ir VT standartai išanalizuoti tokiais pjūviais kaip: virtotojas, suprantamumas virtotojams bei kokių žinių reikia virtotojams, norint perskaityti veiklos taisykles, formali VT struktūra, XML failo generavimas, kuriame sistemos kūrimo etape dalyvauja bei VŽ (jo paskirtis standarte). Svarbu yra ir tai, kad VŽ turi būti skirtas VT naudojamiems žodžiams apibūdinti.
5. BPMN2 ir SBVR priklauso MDA architektūros CIM lygmeniui. Nuo skaičiavimų nepriklausančiame lygmenyje (CIM) atvaizduojamos VT, procedūros ir apribojimai bei VP užrašyti natūralia kalba. Tiek SBVR, tiek BPMN metamodelius apjungia OMG grupės meta-meta modelis (MOF), dėl šios priežasties yra užtikrinama tinkama VŽ, VT ir VP sinchronizacija.
6. Kadangi VP modeliai ir iš jų suformuotas VŽ bei VT aktualios ne tik analitikams, bet ir kitiems organizacijos atstovams, todėl galima teigti, kad VP modeliavimo standarto kriterijus labiausiai atitinka BPMN2 standartas, o VT – SBVR. Pagrindinė to priežastis yra tai, kad tiek BPMN2 tiek SBVR standartai nereikalauja papildomų techninių žinių, taip pat būtent SBVR VŽ skirtas VT naudojamiems žodžiams apibūdinti, o ne tik VT struktūrai apibrėžti (PRR, SRML).
7. Analizuojant esamus sprendimus, VP modeliavimo priemonės ir VT variklius, nustatyta, kad jų (sprendimų) nėra daug. Šiuo metu siūloma VP ir VT kurti atskirai, t.y. sukurti VP bei VT ir tik tuomet juos integruoti. Šis principas yra neefektyvus, kadangi negali būti užtikrinamas pilnas suderinamumas tiek VŽ lygmenyje, tiek pačiuose virtotojų reikalavimuose, kuriant VPD.

### 3 Veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų diagramos metodas

Kuriamas metodas padeda išspręsti iškeltą problemą – išvengti veiklos procesų diagramoje atvaizduojamų procesų klaidų ir neatitikimų organizacijoje vykstantiems realiems procesams. Šiuo metodu užtikrinama sinchronizacija tarp VP ir VT (kartu su VŽ) modelių.

Siekiant išvengti nesusipratimų dėl skirtingų terminų/sąvokų sudaromas VŽ. Jis yra pagrindas aiškiam bendradarbiavimui tiek įmonės viduje, tiek su klientais, tiek su diegėjais. Veiklos taisyklėje gali būti žodžių ar jų junginių, kurie yra naudojami keliose BPMN modelio vietose. Šie atskiri junginiai naudojami vienoje taisyklėje, kuri vartotojui gali pasirodyti neaiški dėl to, kad skirtingi vartotojai skirtingai supranta terminus.

Naudodamas metodą, vartotojas (tiek IT analitikas, tiek užsakovas) galės patikrinti sumodeliuotą VPD. Užsakovas, neturintis pakankamai techninių žinių, sekdamas išgautas VT bei VŽ galės ją patvirtinti, valdyti, keisti.

Metodo įgyvendinimui yra suformuotas algoritmas, kurį sudaro VŽ bei VT išgavimo sudedamosios dalys. Abiem atvejais yra analizuojami BPMN2 ir SBVR standartų elementai bei tam tikri numatyti atvejai (elementų jungimo, vardinimo). Sudaryti BPMN2 elementų kombinacijų modeliai, kurie pateikti 3.3 poskyryje.

Kadangi SBVR ir BPMN2 standartai yra glaudžiai tarpusavyje susiję, todėl informacija, gauta iš VPD elementų, naudojama ir analizuojant bei sudarant SBVR žodyną bei VT struktūras.

#### 3.1 Veiklos žodyno automatizuoto išgavimo iš veiklos procesų diagramos algoritmas

Algoritmo sudarymui nagrinėjami BPMN2 metamodelio elementai ir juose saugoma informacija, kuri gali būti naudinga išgaunant VŽ.

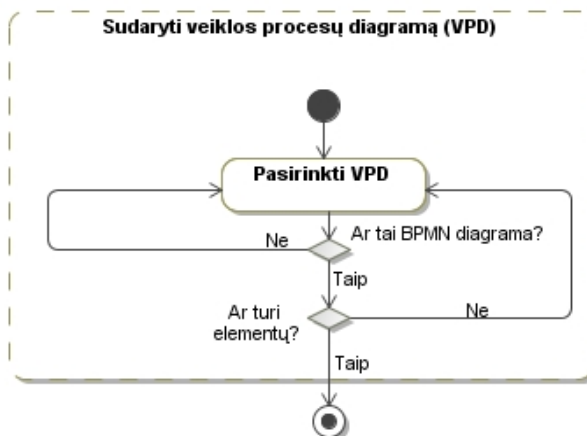
Pagrindiniai VŽ išgavimo algoritmo žingsniai pateikiami 3.1 paveiksle.



3.1 pav. Pagrindiniai VŽ išgavimo iš VP modelio algoritmo žingsniai

### 3.1.1 I. Sudaryti veiklos procesų diagramą (VPD)

Kadangi analitikai braižo ne vieną organizacijos procesų modelį, todėl vartotojas turi pasirinkti VPD, su kuria planuoja dirbti. 3.2 paveiksle pateiktas šio žingsnio algoritmas, kuriame matoma, kad sistema automatiškai patikrina ar diagramoje yra elementų, kurie priklauso BPMN standartui. Jei yra, tuomet galima pereiti prie antrojo žingsnio (II. Automatizuotu būdu formuoti VŽ).



3.2 pav. Pasirinkti veiklos procesų diagramą veiklos diagrama (1 – as žingsnis)

### 3.1.2 II. Automatizuotu būdu formuoti VŽ

Išanalizavus BPMN2 standartą sudarytas (kiek galima labiau) automatizuotas VŽ išgavimo iš VPD algoritmas. 3.3 paveiksle matoma, kad algoritmą sudaro 2 etapai: „Atrinkti terminus“ ir „Nustatyti tipus“. Pirmame etape išrenkami visi elemento pavadinimo terminai (žodžiai) ir tokia informacija išsaugoma. Antrajame etape tikrinamas kiekvienas elementas (pagal grupę – vieno elemento visi žodžiai). Pagrindiniai algoritmo žingsniai:

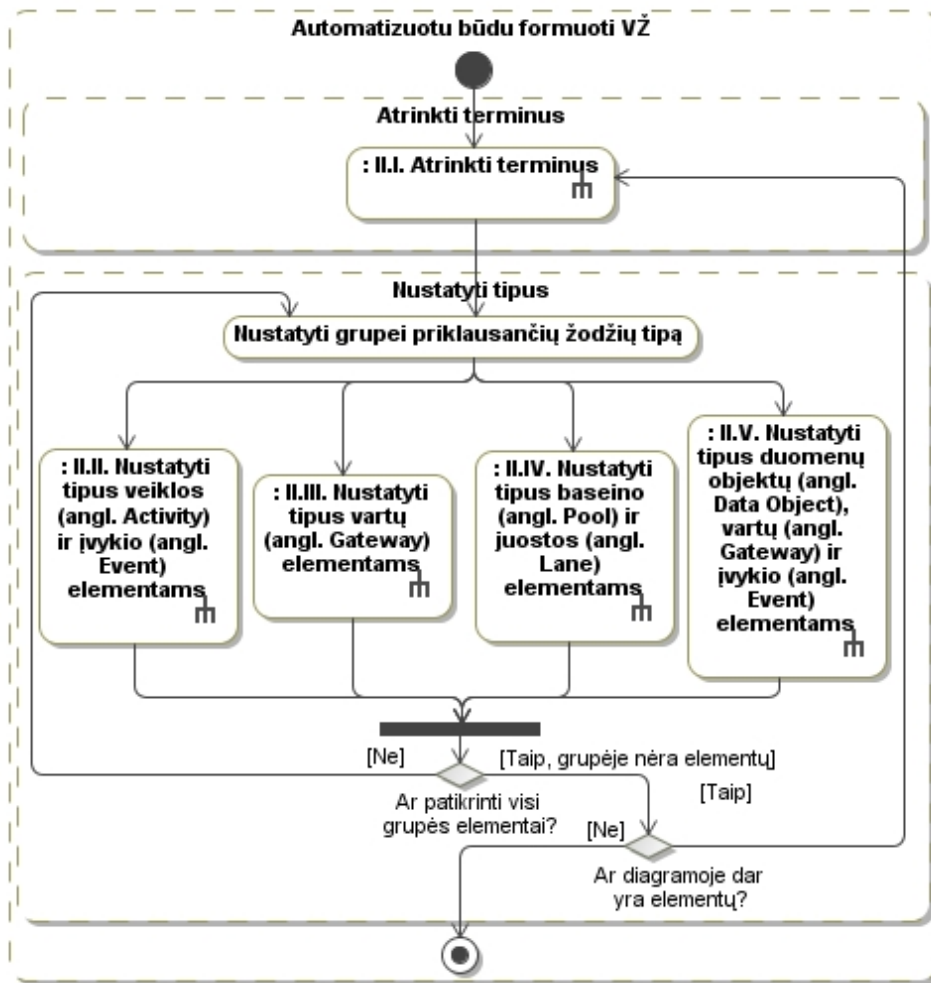
- ✓ analizuojama, iš kokio elemento išrenkami žodžiai. Pagal tai sudaryti skirtingi algoritmai, kurie 3.3 paveiksle atvaizduoju kaip veiklos:
  - „II.II Nustatyti tipus veiklos (angl. Activity) ir įvykio (angl. Event) elementams“
  - „II.III Nustatyti tipus vartų (angl. Gateway) elementams“
  - „II.IV Nustatyti tipus baseino (angl. Pool) ir juostos (angl. Lane) elementams“
  - „II.V. Nustatyti tipus duomenų objektų (angl. Data Object), vartų (angl. Gateway) ir įvykio (angl. Event) elementams“.

Pastarajame algoritme vartų elementui nustatomas tipas, kai nagrinėjami lygiagretūs vartai (angl. *Parallel Gateway*), o įvykių elementui – tarpiniai įvykiai (angl. *Intermediate Events*).

- ✓ analizuojama, koks pirmas elemento pavadinimo žodis. Šio etapo metu analizuojama:

- ar pirmas žodis yra skaičius;
  - ar pirmas žodis yra artikkelis („the“, „an“);
  - žodžio struktūra (t.y., žodis gali būti su apostrofa (pvz.: Client’s);
  - išskiriami variantai (kai elemento tipas yra vartai, kartais tarpiniai įvykiai), kai pirmas žodis gali būti „Can“, „Need“.
- ✓ ieškoma žodžio vieta elemento pavadinime:
- kai elemento tipas yra vartai, įvykiai ar duomenų objektai, tuomet ieškoma jungtuko/ raktažodžio (angl. *Keyword*) vieta elemento pavadinime. Pagal paskutinę rastą poziciją, nustatoma šalia esančių elementų tipai, prieš tai patikrinant pastarųjų tipą (skaičius, artikkelis ir pan.).
- ✓ analizuojama, kiek žodžių sudaro elemento pavadinimą:
- nuo žodžių kiekio pavadinime priklauso algoritmo sudėtingumas. Tai yra, veiklos (angl. *Activity*) elemento pavadinimą gali sudaryti 2 žodžiai. Tokiam pavadinimui nereikia papildomų tikrinimų, kadangi veiklos elemento pavadinimas turi būti sudaromas iš veiksmažodžio ir daiktavardžio (elementų užrašymo reikalavimai pateikiami 4 skyriuje). Tokiu atveju nereikia papildomai tikrinti, ar pirmas žodis yra skaičius, jungtukas/raktažodis, artikkelis ar pan..

Atlikus pagrindinius žingsnius nustatomas elemento žodžio tipas, kuris, esant poreikiui, gali būti redaguojamas.

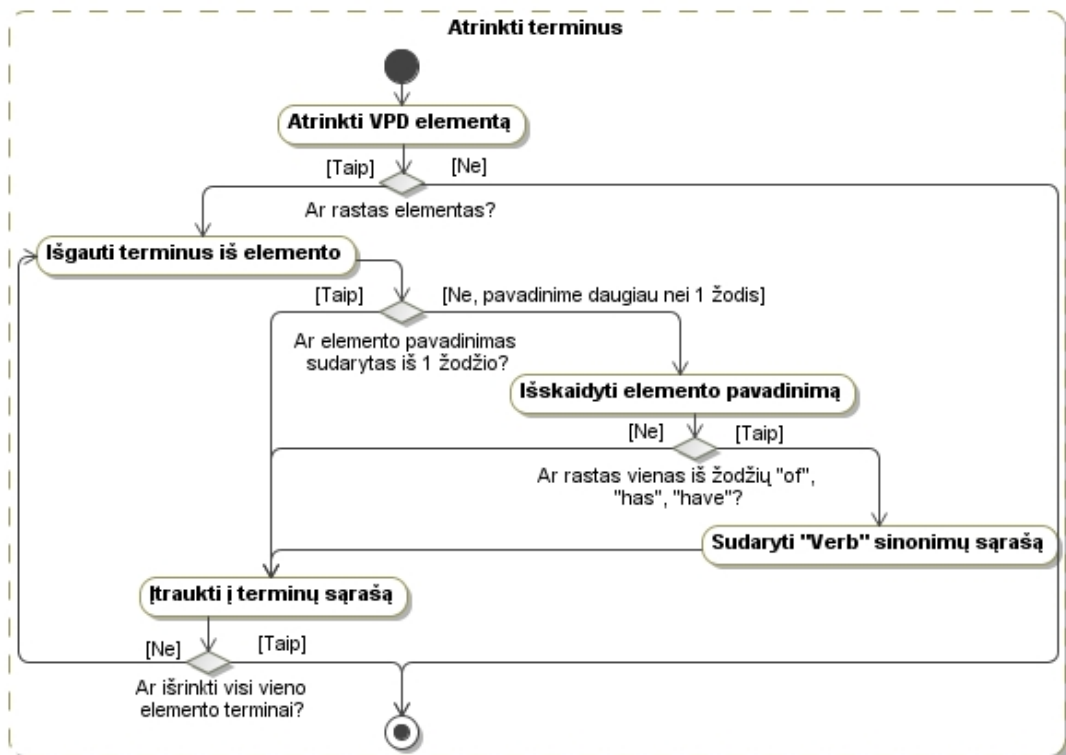


3.3 pav. Formuoti VŽ veiklos diagrama (2 – as žingsnis)

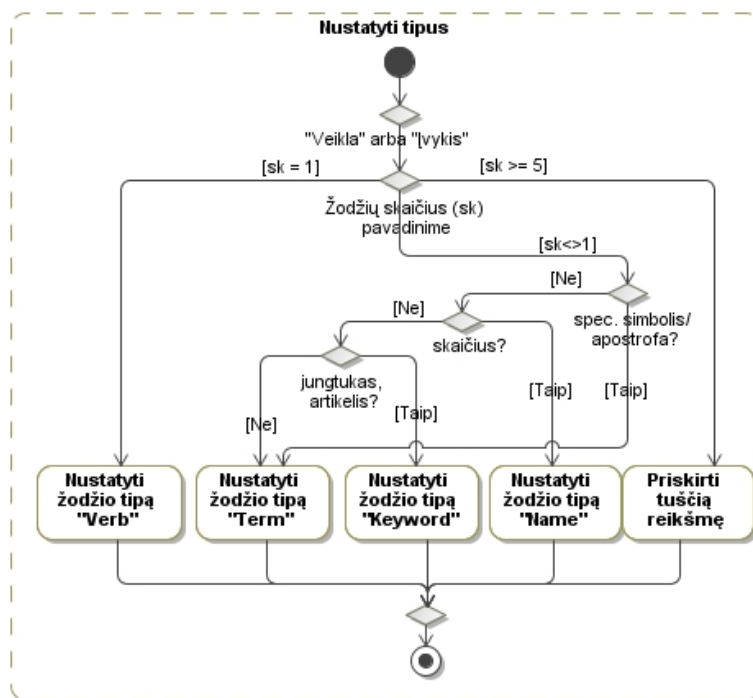
Atskiri algoritmai, susiję su II-uoju etapu, pateikiami 3.4 – 3.8 paveiksluose.

3.4 paveiksle pavaizduotas terminų atrinkimo algoritmas, kurio pagalba išskaidomi pavadinimai į pavienius žodžius, kurie vėliau naudojami VŽ sudarymui.

Kai atliekami visi veiksmai, pavaizduoti 3.4 paveiksle, tuomet parenkamas vienas iš algoritmų (žiūrėti 3.5 – 3.8 pav.) pagal elemento tipą (veikla, įvykis, vartai ir pan.). 3.5 paveiksle pavaizduotas algoritmas veiklos arba įvykio elementų pavadinimo žodžių tipams nustatyti. Nustačius tipus, vėliau sistema įvertindama prieš tai buvusį bei esantį žodžio tipą automatinio būdu juos pakeičia.



3.4 pav. Veiklos „II.I. Atrinkti terminus“ 3.3 paveiksle algoritmo diagrama

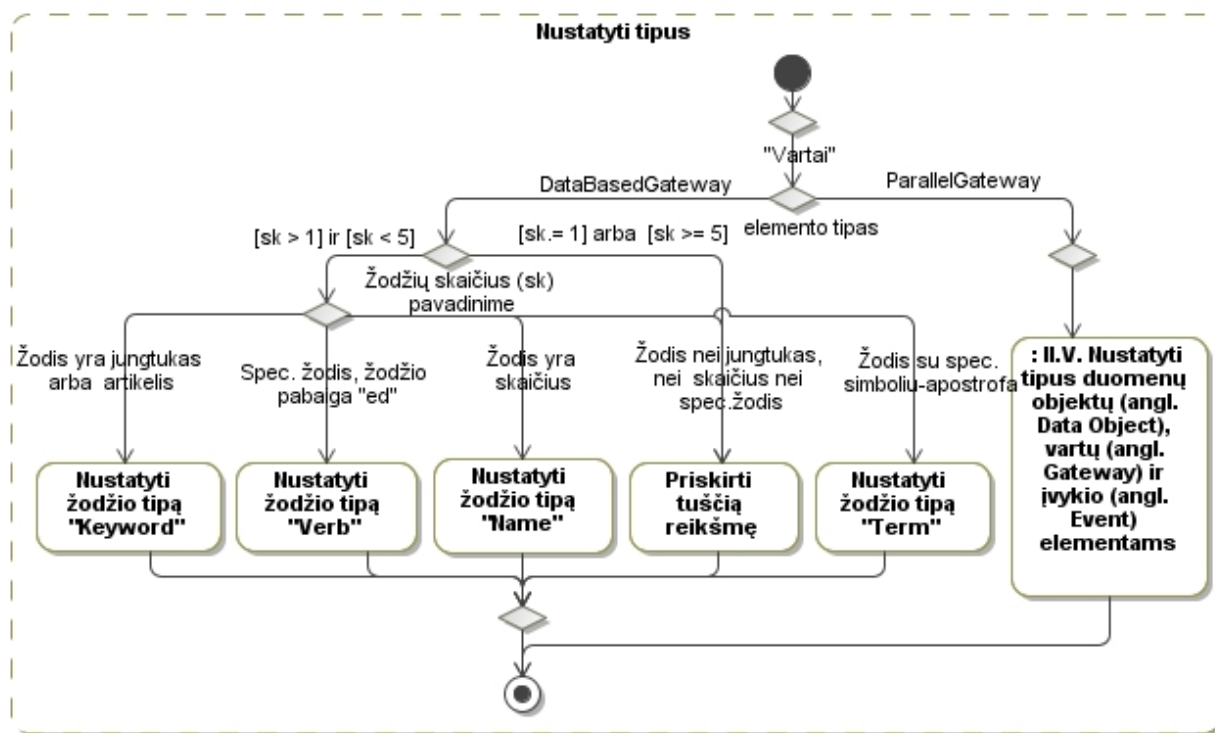


3.5 pav. Veiklos „II.II. Nustatyti tipus veiklos (angl. Activity) ir įvykio (angl. Event) elementams“ 3.3 paveiksle algoritmo diagrama

Vartų elementų pavadinimų žodžiai vadinami remiantis 3.6 ir 3.8 paveiksluose pavaizduotais algoritmais. Pastarasis naudojamas tada, kai aptinkamas lygiagrečių vartų elementas. Pirmuoju atveju (3.6 pav.) tikrinamas žodžių skaičius pavadinime. Taip pat, kaip ir anksčiau aptartu atveju, tikrinama, ar žodis yra jungtumas, skaičius. Papildomai patikrinama ir žodžio galūnė (jei pabaiga yra „ed“, tai laikoma, kad žodis yra veiksmažodis. Nors gali



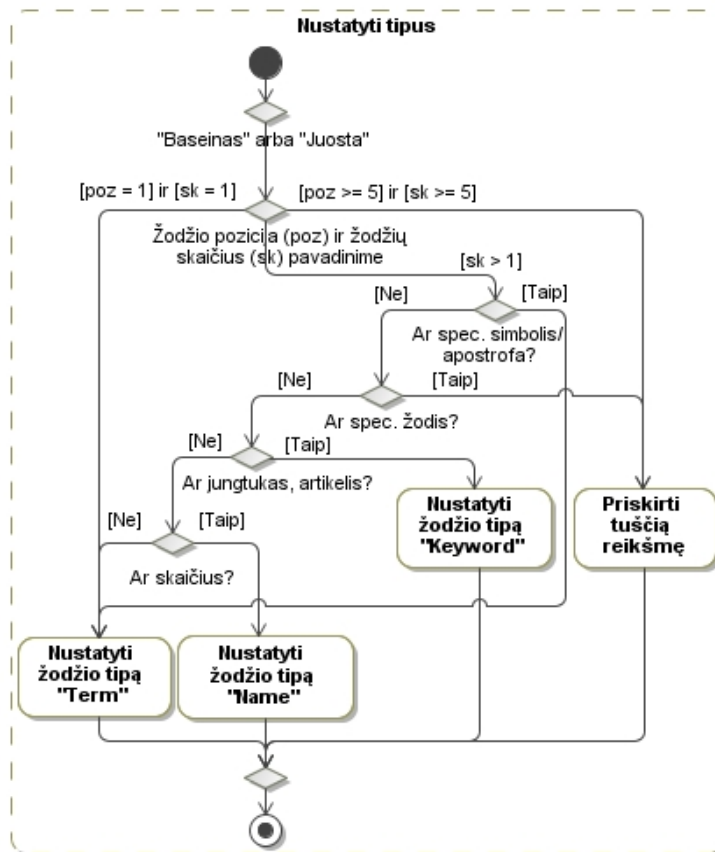
pasitaikyti ir daiktavardžių, kurie baigiasi tokia galūne. Tam vartotojas rankiniu būdu gali pakeisti tipą (III – iasis algoritmo žingsnis). Antrasis atvejo aprašymas pateiktas žemiau (prie 3.8 paveikslo).



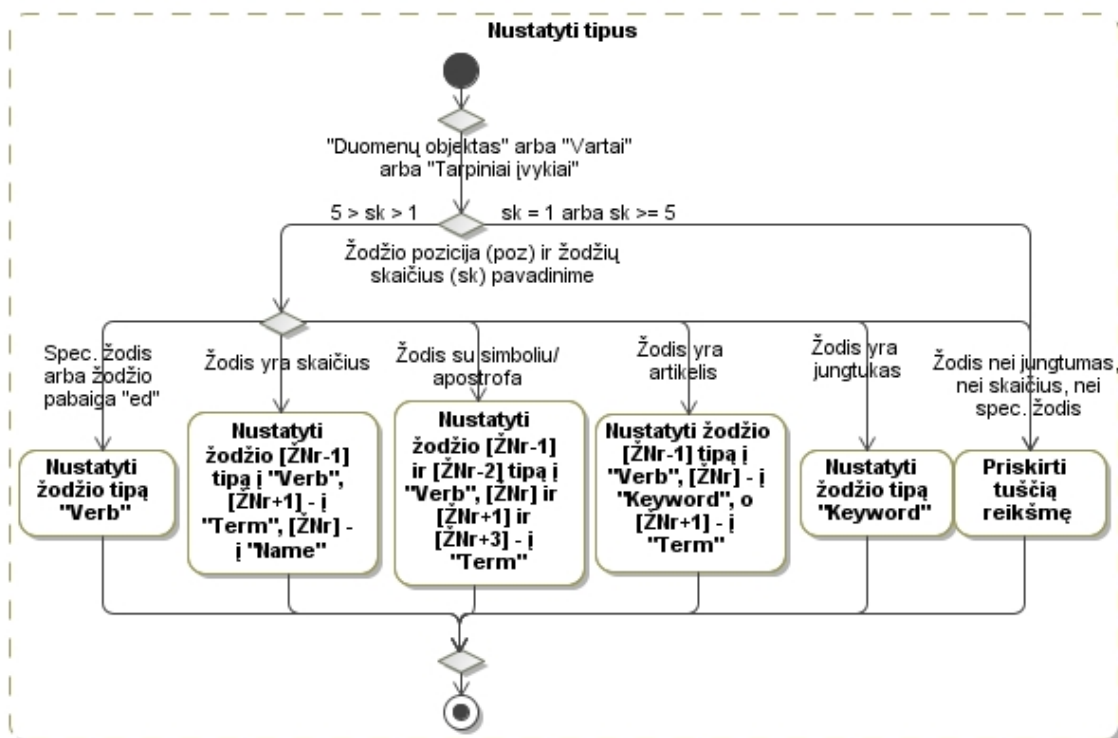
3.6 pav. Veiklos „II.III. Nustatyti tipus vartų (angl. Gateway) elementams” 3.3 paveiksle algoritmo diagrama

3.7 paveiksle pavaizduotas BPMN standarto baseino ir juostos elemento pavadinimo žodžių tipų nustatymo algoritmas. Tai algoritmas, kurio metu nenumatyta žodžiui priskirti veiksmažodžio reikšmę, kadangi baseinai ir juostos apibūdina veiksmo atlikėją, o ne veiksmą. Dėl šios priežasties pavadinimą sudaro daiktavardžiai (terminai) ir jungiamieji žodžiai (raktažodžiai).

Apibūdinant 3.6 paveikslą, jau aptartas vienas žodžių tipo nustatymas vartų elemento pavadinimui. 3.8 paveiksle pateiktas algoritmas, tinkamas ir duomenų objektų, ir vartų (lygiagrečių), ir tarpinių įvykių elementų analizei. Šiuo atveju papildomai nustatomi ir ankstesnių žodžių tipai. Algoritmas skiriasi tuo, kad nagrinėjami elementų tipai apibūdina faktą (pavyzdžiui, „Report is received“, „Document has copy“), todėl struktūra yra pakankamai panaši. Todėl, projektuojant algoritmą, numatyta iškart priskirti žodžių tipus prieš ir po nagrinėjamo žodžio esantiems elementams (t.y., jei antras žodis, remiantis SBVR terminija yra pavadinimas (angl. *Name*), tai dažniausiai (jei analitikas laikosi nurodymų, pateiktų 4 skyriuje) prieš skaičių esantis žodis yra veiksmažodis, o sekantis – terminas).



3.7 pav. Veiklos „II.IV. Nustatyti tipus baseino (angl. Pool) ir juostos (angl. Lane) elementams” 3.3 paveiksle algoritmo diagrama



3.8 pav. Veiklos „II.V. Nustatyti tipus duomenų objektų (angl. Data Object), vartų (angl. Gateway) ir įvykio (angl. Event) elementams” 3.3 paveiksle algoritmo diagrama

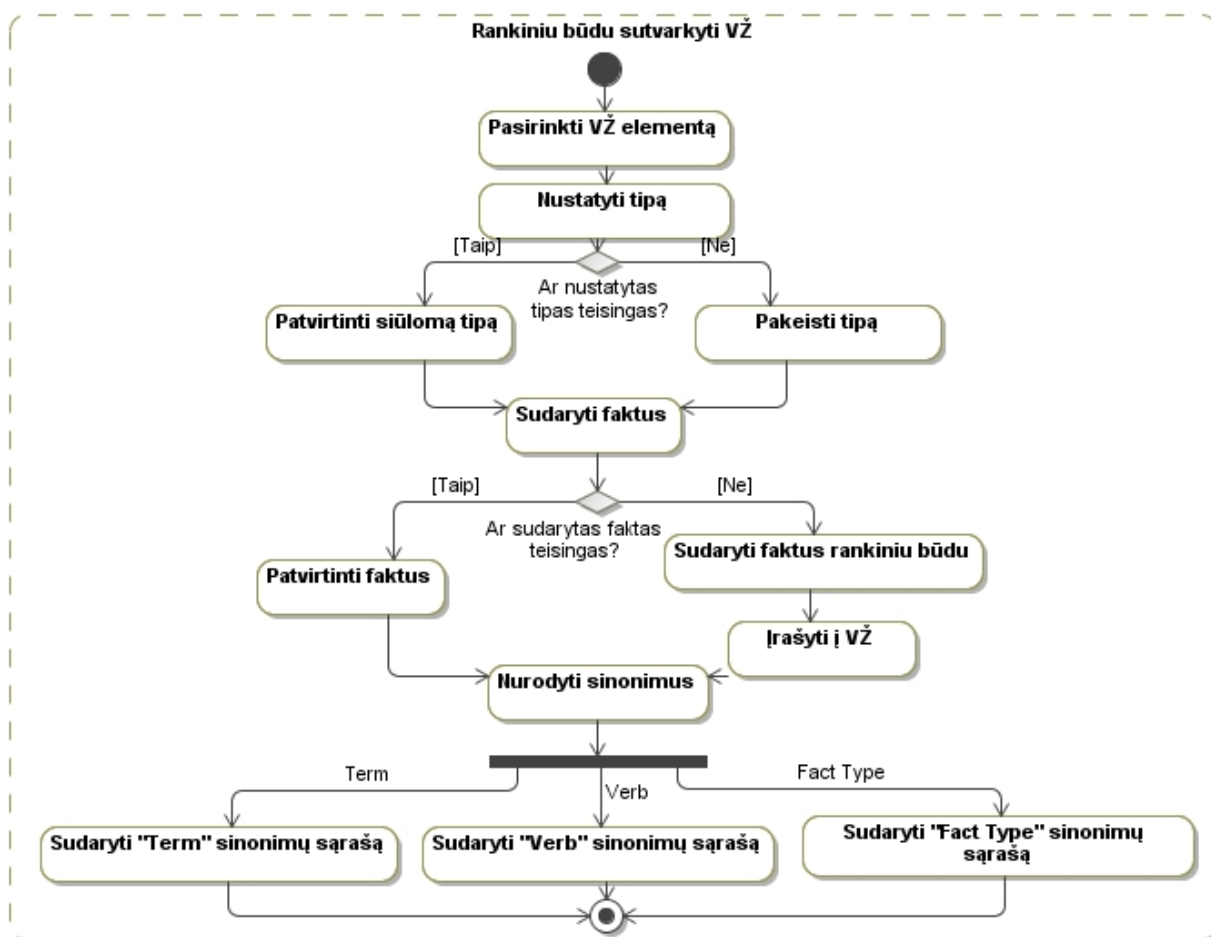
Algoritmuose, analizuojančiuose duomenų objekto (tuo pačiu ir vartų bei tarpinių įvykių) (žiūrėti 3.8 pav.), vartų (žiūrėti 3.6 pav.) bei baseino ir juostos (žiūrėti 3.7 pav.) elementus, papildomai analizuojamas ir žodžių skaičius pavadinime. Nuo to priklauso ankstesnių ar vėlesnių žodžių tipai.

### 3.1.3 III. Rankiniu būdu sutvarkyti VŽ

Automatizuotu būdu sudarant žodyną gali atsirasti klaidų ar neatitikimų (VŽ gali būti sudarytas ne taip, kaip tikėjosi IT analitikas). Taip gali atsitikti dėl rekomendacijų, aprašant elemento pavadinimą, nesilaikymo.

Tokiu atveju vartotojui pateikiamas redaguojamas žodynas, kuriame jis gali nustatyti kitokią reikšmę (pvz.: veiksmazodis → daiktavardis).

3.7 paveiksle pateikiamas trečiasis VŽ išgavimo žingsnis.



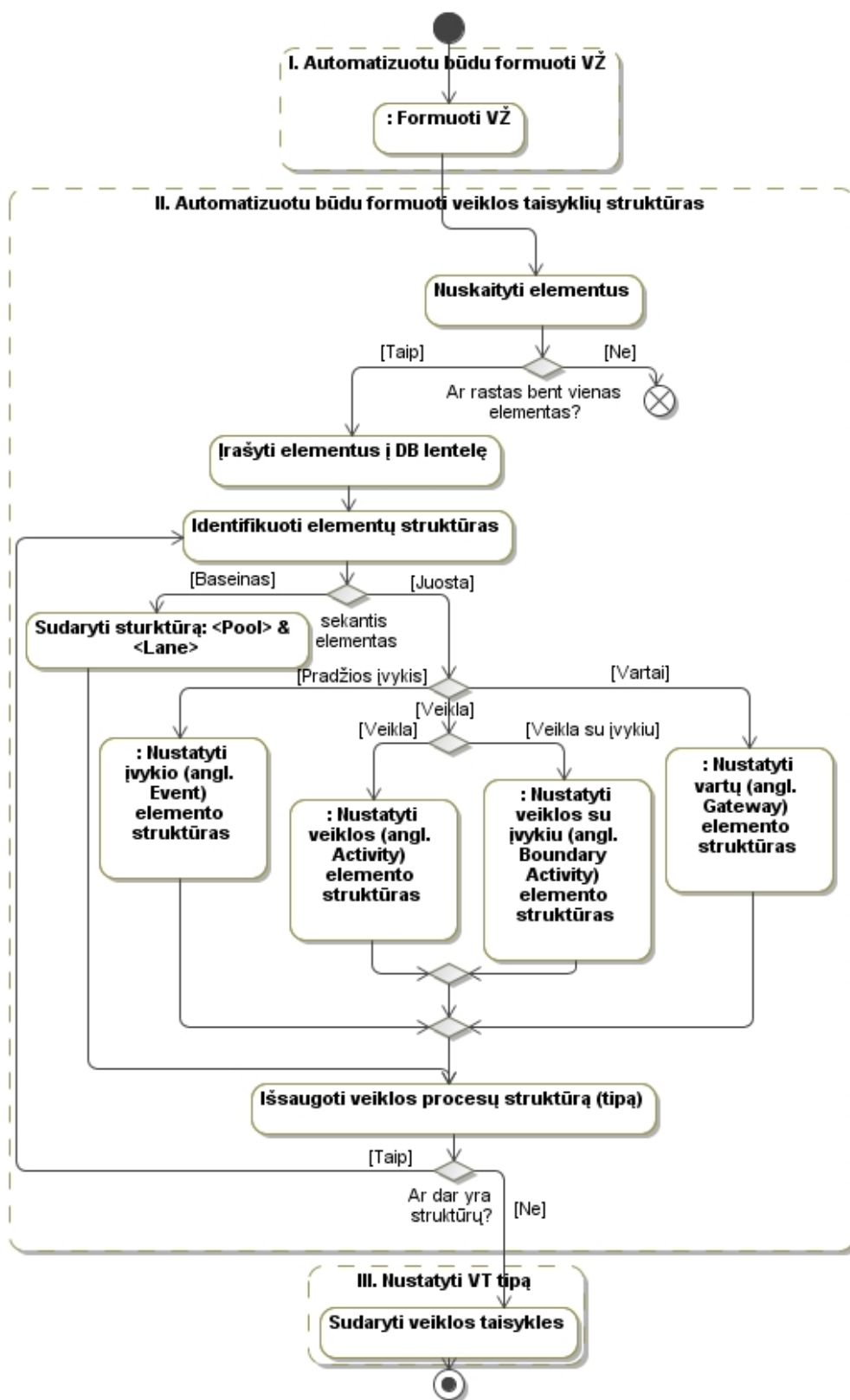
3.9 pav. VŽ redagavimo rankiniu būdu veiklos diagrama (3 – ias žingsnis)

#### Algoritmo veikimas:

1. Užsakovas patikrina, ar nustatytas elemento tipas teisingas. Jei ne pakeičia esantį tipą.
2. Patikrina, ar faktas sudarytas teisingai, ir juos įrašo arba redaguoja sistemoje.
3. Sutvarko visus elementus ir išsaugo pakeistą informaciją.

### 3.2 Veiklos taisyklių automatizuoto išgavimo iš veiklos procesų diagramos algoritmas

Algoritmo sudarymui taip pat nagrinėjami BPMN2 standarto elementai ir juose saugoma informacija, kuri gali būti naudinga sudarant veiklos taisykles.



3.10 pav. Pagrindiniai veiklos taisyklių išgavimo iš VP modelio sudarymo veiklos diagrama (žingsniai)

Norint jas atvaizduoti natūralios kalbos sakiniiais (kaip matoma, pavyzdžiui, 3.2 lentelėje) buvo analizuojamas ir SBVR standartas: taisyklių tipai, žodžių (esančių VŽ) išdėstymas taisyklėse. Pagal skirtingas BPMN2 elementų struktūras sudaromos skirtingos taisyklės.

Pagrindiniai VŽ išgavimo algoritmo žingsniai (I. Automatizuotu būdu formuoti VŽ, II. Automatizuotu būdu formuoti veiklos taisyklių struktūras, III. Nustatyti VT tipą) pateikiami 3.10 paveiksle.

Šis algoritmas sudaromas remiantis BPMN2 elementais bei žodžiais, išgautais VŽ automatinio išgavimo iš VPD metu.

### 3.2.1 I. Automatizuotu būdu formuoti VŽ

Šiame žingsnyje iš VPD išgaunami visi elementai bei juose saugoma informacija, kuri toliau naudojama sudarant veiklos taisyklės. Ši informacija išsaugoma duomenų bazės lentelėse (duomenų bazės klasės modelis pateikiamas 5.5 paveiksle).

Visi šio žingsnio etapai aprašyti 3.1 poskyryje.

### 3.2.2 II. Automatizuotu būdu formuoti veiklos elementų jungimo struktūras

Šiame žingsnyje nuskaityti elementai (pirmame žingsnyje) naudojami sudarant jų struktūras (nurodoma, kokie elementai jungiami tarpusavyje), iš kurių vėliau sudaromos VT. Šiame etape sudaromos įvairios kombinacijos (veiklos elementų struktūros), kurios skirstomos į grupes, kurios 3.10 paveiksle atvaizduojamos kaip detalizuojamos veiklos:

- „Nustatyti įvykio (angl. Event) elemento struktūras“;
- „Nustatyti veiklos (angl. Activity) elemento struktūras“;
- „Nustatyti veiklos su įvykiu (angl. Boundary Activity) elemento struktūras“;
- „Nustatyti vartų (angl. Gateway) elemento struktūras“.

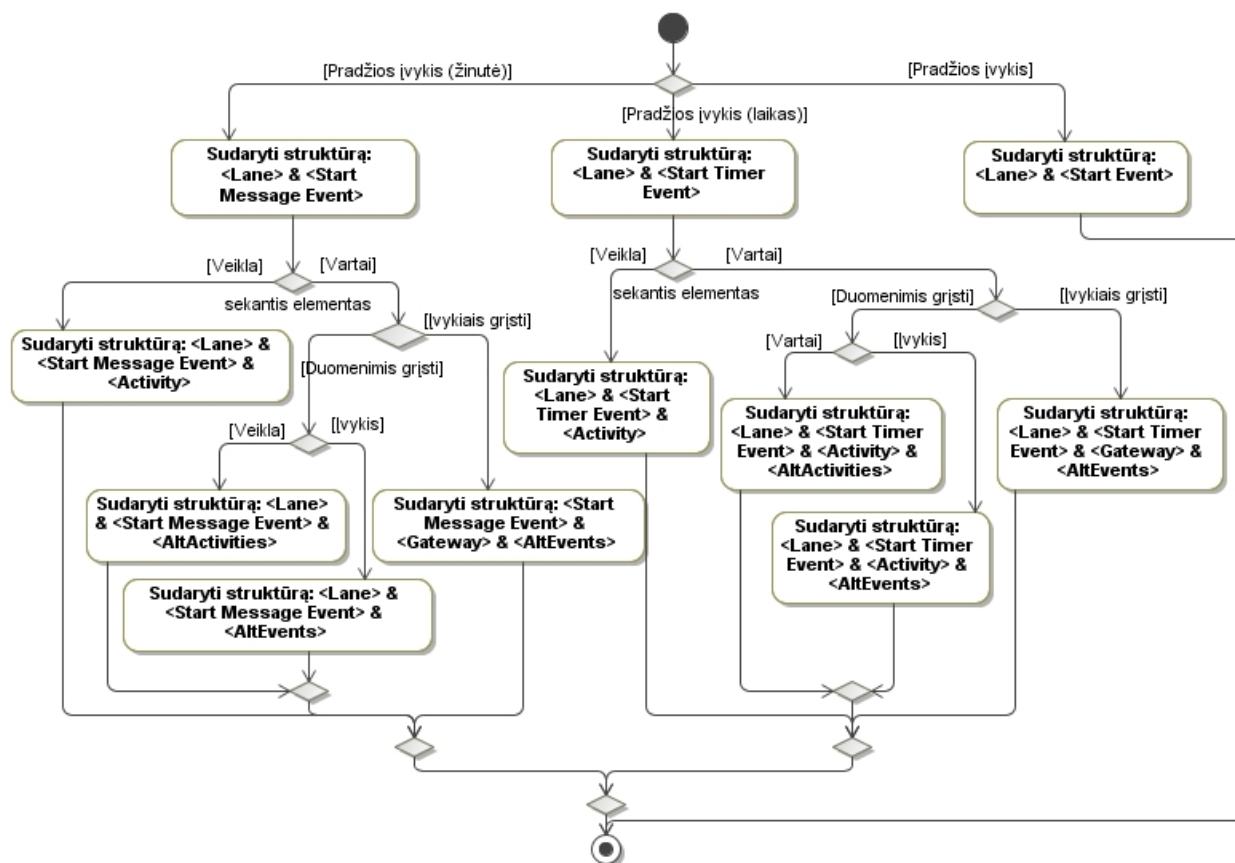
Tai yra atskiri algoritmai, kurie naudojami nustatant elementų seką (nustatomi elementai, einantys vienas po kito (nuosekliai)). Pagal išgautas struktūras nustatomas VT tipas bei ji (VT) išsaugoma duomenų bazės lentelėje natūralios kalbos sakiniiais.

Taisyklės struktūra labiausiai priklauso nuo to, kokia tvarka elementai nuskaityti iš duomenų bazės lentelės ir ar naudojami konteineriai (angl. *Pool*) bei juostos (angl. *Lane*). Jei pastarieji elementai nenaudojami, tuomet, pavyzdžiui, vykdomoji įsipareigojimo/įgalinimo taisyklė atrodo taip: „*It is obligatory to*“. Kitu atveju (dažniausiai) – „*It is obligatory that*“.

Minėtų detalizuojamų veiklų algoritmai pateikti 3.11 – 3.14 paveiksluose. Visuose algoritmuose galima pastebėti, kad nagrinėjamas einamas ir sekantis elementai ir kiekvieno

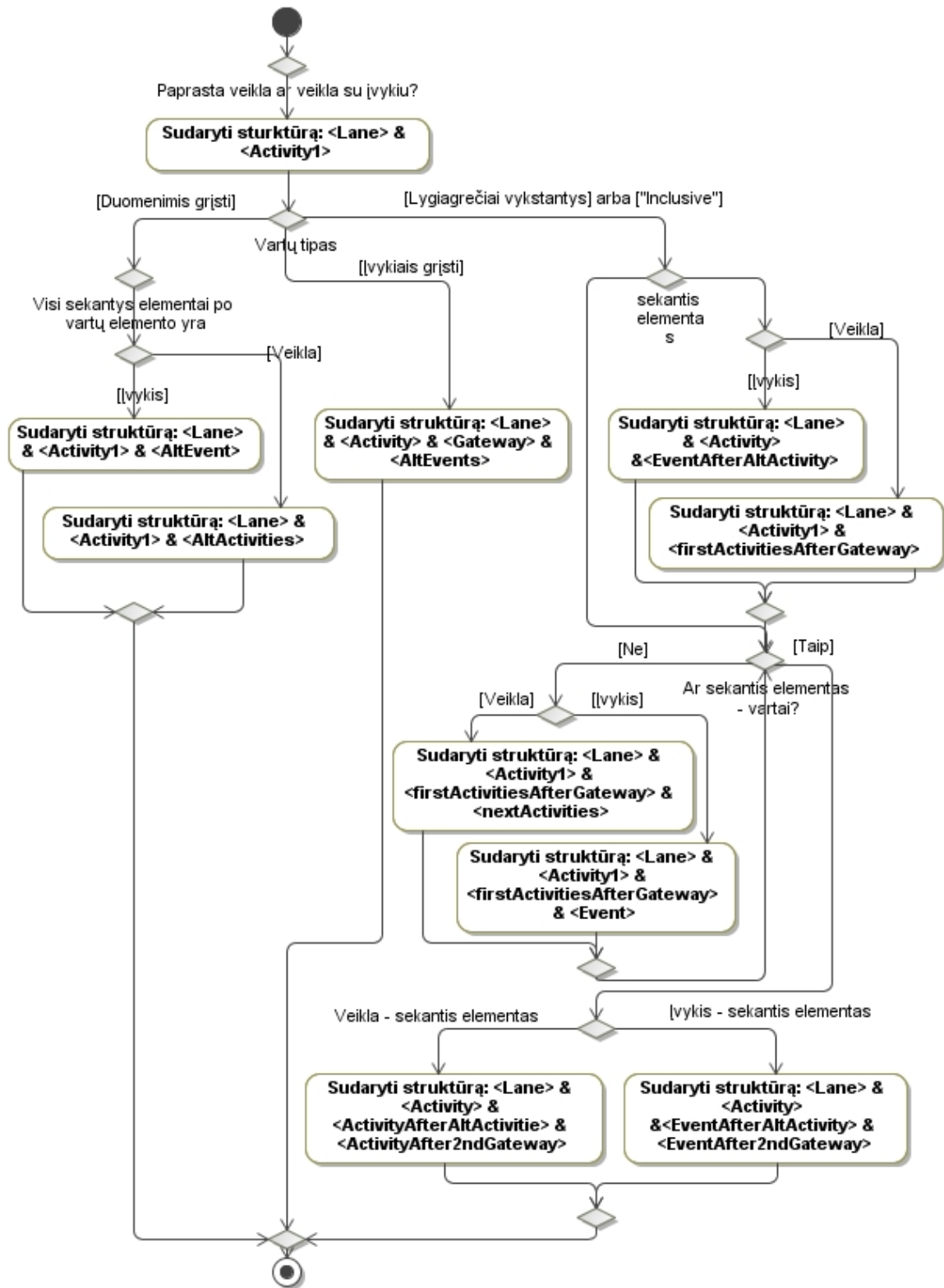
žingsnio metu suformuojama elementų struktūra, pagal kurią galiausiai sudaroma taisyklė. Algoritmų veiklose galima pastebėti tokių žodžių kaip *AltActivitie*, *AltEvents*, *firstActivitiesAfterGateway*, kurie apibūdina elementus, įtraukiamus į struktūros sudarymą. Pavyzdžiui, *AltActivities* reiškia, kad naudojamos visos pirmosios alternatyvios veiklos (angl. *Activities*), esančios po, pavyzdžiui, vartų elemento. Algoritmuose pavaizduotos struktūros, kuriose matoma, kaip gali būti jungiami elementai, tačiau realizacijai nagrinėjami 3 elementai.

3.11 paveiksle pateikiamas algoritmas elementų struktūroms sudaryti, kai pirmas sutiktas elementas yra vartai (angl. *Gateway*).

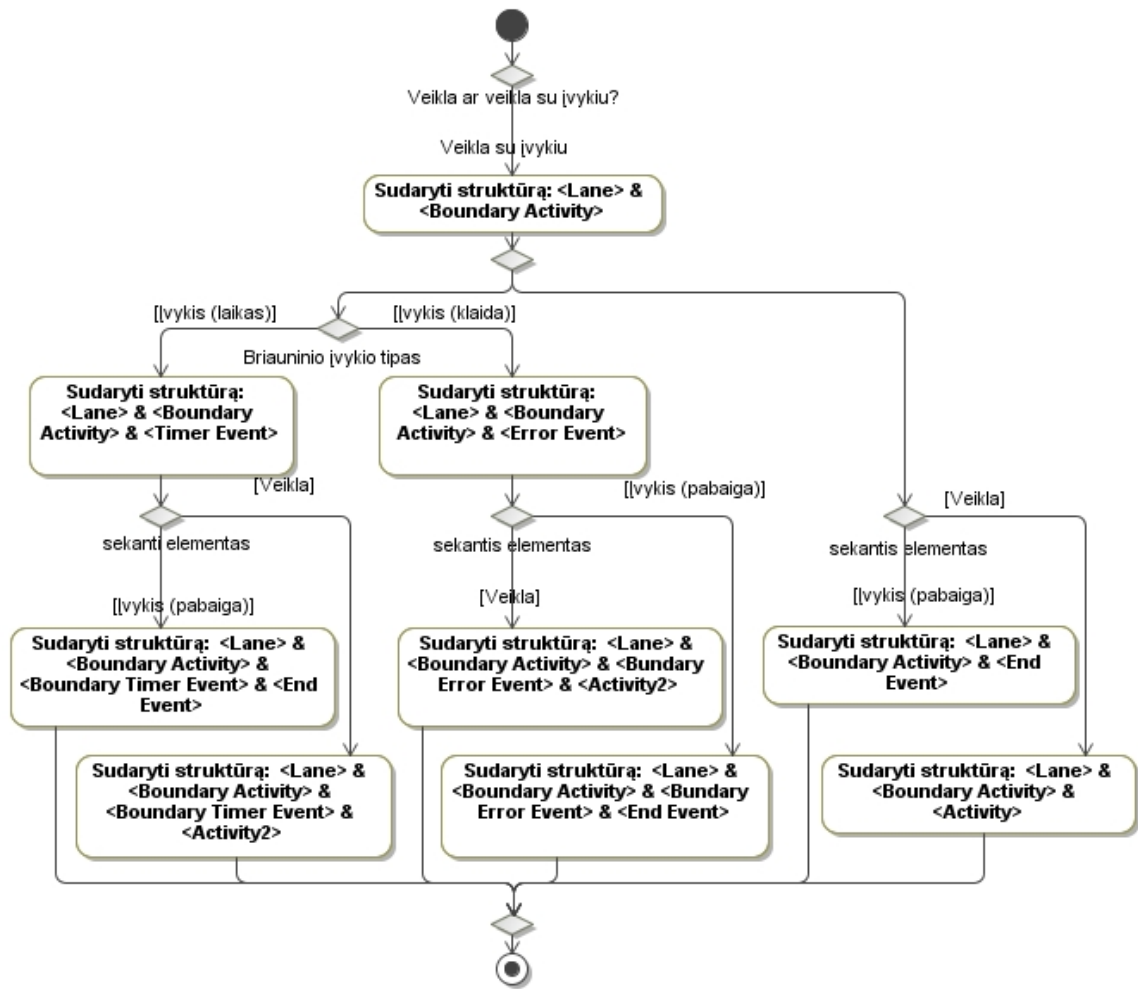


3.11 pav. Detalizuojamos veiklos „Nustatyti įvykio (angl. *Event*) elemento struktūras“ veiklos diagrama

3.12 paveiksle pateikiamas algoritmas elementų struktūroms sudaryti, kai pirmas sutiktas elementas yra veikla, o 3.13 – kai elementas yra veikla su įvykiu (angl. *Boundary Activity*).

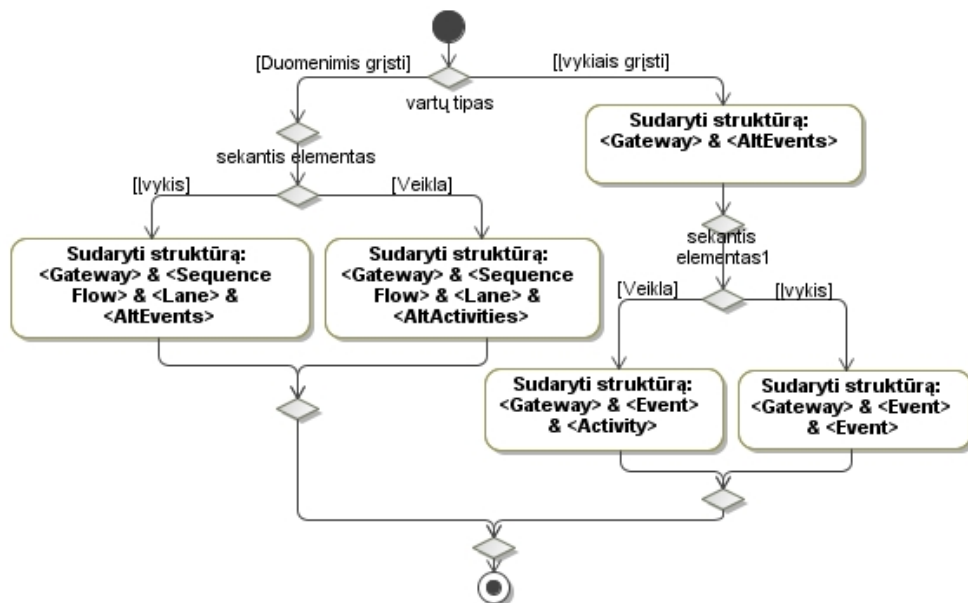


3.12 pav. Detalizuojamos veiklos „Nustatyti veiklos (angl. *Activity*) elemento struktūras“ veiklos diagrama



3.13 pav. Detalizuojamos veiklos „Nustatyti veiklos su įvykiu (angl. *Boundary Activity*) elemento struktūras“ veiklos diagrama

3.14 paveiksle pateikiamas algoritmas elementų struktūroms sudaryti, kai pirmas sutiktas elementas yra vartai.



3.14 pav. Detalizuojamos veiklos „Nustatyti vartų (angl. *Gateway*) elemento struktūras“ veiklos diagrama



Sudarius algoritmą nustatyta, kad:

1. Taisyklės gali būti sudaromos vienam elementui, neskaitant „Pool“, „Lane“ elementų egzistavimo (tokia struktūra atrodo <Pool> & <Activity>). Tuomet VT yra struktūrinė būtinumo (angl. *Necessity*).
2. Vykdomosios taisyklės (pagal pateiktą analizę ir algoritmą) gali būti kelių tipų: įgalinimo/ įpareigojimo (angl. *Obligation*) ir ribojimo (angl. *Prohibition*), tik skiriasi elementų pavadinimų išdėstymas sakinyje, bet ne taisyklės esmė.

### **3.3 Veiklos žodyno ir veiklos taisyklių struktūrų išgavimas iš veiklos procesų elementų**

3.1 ir 3.2 poskyriuose pavaizduotiems algoritmams sudaryti nagrinėjama BPMN2 elementų saugoma informacija, atskiros situacijos (jungiant skirtingus (grupių) elementus, juos skirtingai vardinant), kurios atvaizduotos grafiškai. Kiekvienai iš jų sudarytas VŽ ir VT (remiantis SBVR metodologija).

Žemiau pateiktuose pavyzdžiuose VŽ pateiktas rūšiuojant pagal tipus, o VT – pagal elementų VPD nuoseklumą.

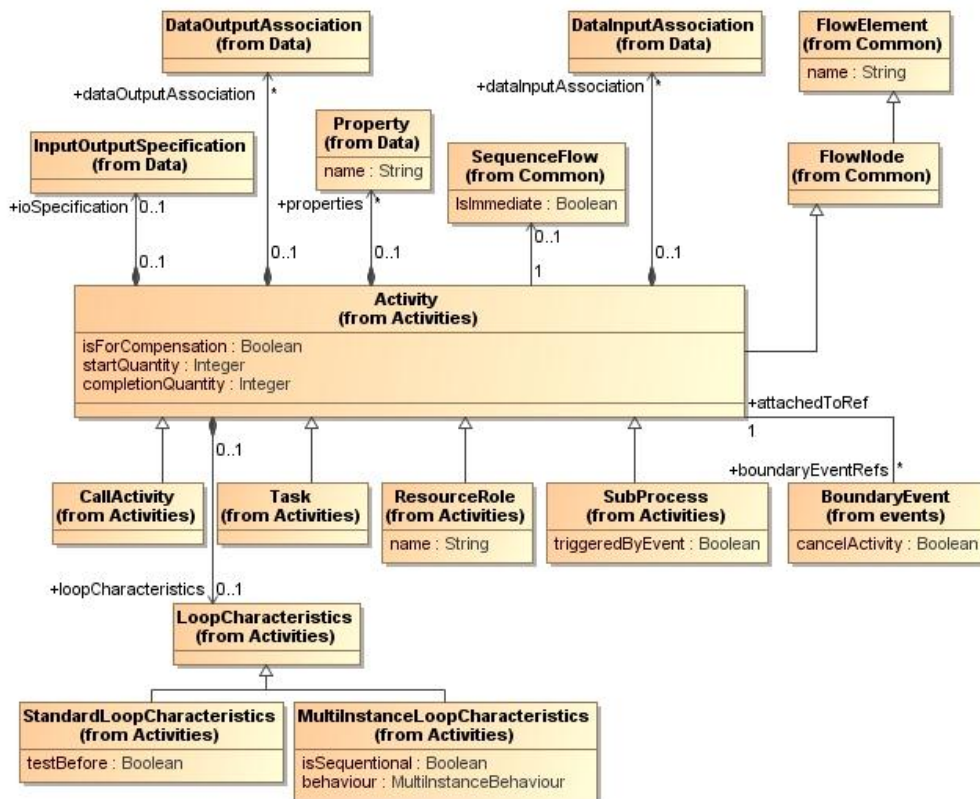
#### **3.3.1 VŽ ir VT išgavimas iš srautų objektų elementų**

##### **3.3.1.1 Veikla (angl. *Activity*)**

Veikla – tai terminas, apibūdinantis darbą, kurį atlieka įmonė ar jos atstovas. Veikla gali būti neskaidoma (atominė) arba skaidoma.

Veiklos elementas paveldi visus „FlowElement“ atributus ir modelio ryšius (angl. *Model association*).

Šis elementas gali turėti įeinantį (-čius) ir/ ar išeinantį (-čius) sekos srautus (angl. *Sequence flow*). Tokie srautai galimi: veikla – subprocesas, veikla – veikla, veikla – vartai, veikla – įvykis.



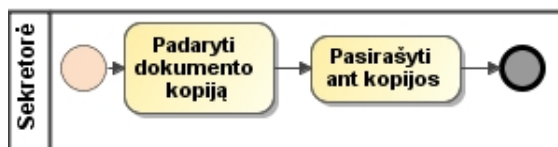
3.15 pav. Veiklos klasės metamodelis

### Grafinis atvaizdavimas – nr. 1

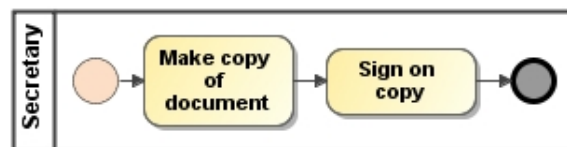
Iš BPMN metamodelio ir 3.15 paveiksle matoma, kad „Task“ klasė iš „Activity“ klasės paveldi unikalų numerį (angl. *Id*), vardą (angl. *Name*) bei aprašą (angl. *Description*).

- *Unikalus numeris* ir *aprašas* paveldimas iš „Object“ klasės, kuri yra „tėvinė“ ir kitiems elementams;
- *Pavadinimas* – „Properties“ (žiūrėti 3.15 pav.) klasės;

3.16 ir 3.17 paveiksluose pateikta situacija, kaip naudojamos veiklos: „Padaryti dokumento kopiją“ (angl. „*Make copy of document*“) ir „Pasirašyti ant kopijos“ (angl. „*Sign on copy*“). Pastaroji vykdoma tik tada, kai atlikta pirmoji veikla.



3.16 pav. Situacijos nr. 1 sprendimas (liet.)



3.17 pav. Situacijos nr. 1 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.1 ir 3.2 lentelėse.

### Veiklos taisyklės ir veiklos žodynas

3.1 lentelė. Situacijos nr. 1 veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
Pool	Daiktavardis	Term	secretary	<a href="#">secretary</a>
Activity	Daiktavardis	Term	copy	<a href="#">copy</a>
Activity	Daiktavardis	Term	document	<a href="#">document</a>
Activity	Veiksmažodis	Verb	make	<i>make</i>
Activity	Veiksmažodis	Verb	sign	<i>sign</i>
Activity	Veiksmažodis	Verb	sign on	<i>sign on</i>
Activity	Raktažodis	Keyword	on	<i>on</i>
Activity	Išraiška	Binary fact type <term><verb><term>	copy of document	<a href="#">copy of document</a>

3.2 lentelė. Situacijos nr. 1 veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Veiklos taisyklės	Atvaizdavimas
<Pool> & <Activity>	Definitional business rule (necessity)	<i>It is necessary that</i> <a href="#">secretary</a> <i>make</i> <a href="#">copy of document</a>	It is necessary that secretary make copy of document
Pool> & <Activity> & <Activity>	Operational business rule (obligation)	<i>It is obligatory that</i> <a href="#">secretary</a> <i>make</i> <a href="#">copy of document</a> <i>to</i> <i>sign on</i> <a href="#">copy</a>	It is obligatory that secretary make copy of document to sign on copy.

#### BPMN saugoma informacija:

##### „Make copy of document“

```
<node xmi:type="uml:StructuredActivityNode"
xmi:id="_17_0_1_8cd027c_1332267855114_569094_2044" name="Make copy of
document" clientDependency="_17_0_1_8cd027c_1332267866374_806455_2064"
inPartition="_17_0_1_8cd027c_1332267801924_596029_2000">
```

Dokumentacija pateikia papildomą informaciją apie elementą (jei ji nurodyta, pvz, elemento savybių lange).

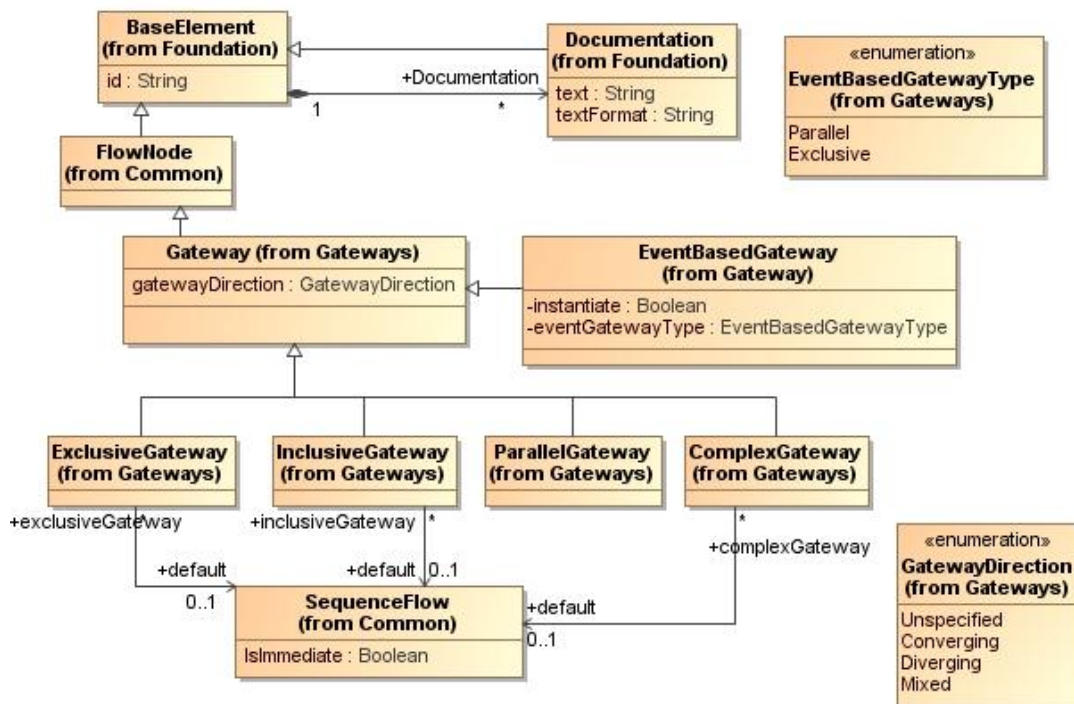
#### 3.3.1.2 Vartai (angl. Gateways)

Vartai yra naudojami tik siekiant kontroliuoti veiksmų seką. Žodis „vartai“ reiškia mechanizmą, kuris leidžia arba ne perėjimą nuo vienos veiklos prie kitos. Be to, vartai kontroliuoja kartu išsišakojančius ir susiliejančius sekos srautus. Tai yra, vienas vartų simbolis gali turėti keletą įėjimo ir išėjimo srautų sekos (angl. *Sequence Flow*).

Vartai yra skirtingų tipų, kurių kiekvienas apibrėžia galimų išėjimų skaičių. Vienintelio sprendimo/suliejimo (angl. *Exclusive*) (duomenimis paremtas (angl. *Data-Based*) ir įvykiu paremtas (angl. *Event-Based*)), daugiau nei vieno sprendimo/suliejimo (angl. *Inclusive*),

sudėtinio sprendimo/suliejimo (angl. *Complex*) ir lygiagretaus išsišakojimo/sujungimo (angl. *Parallel*).

Taip pat vartai priklauso pagrindinių elementų grupei, kurioje saugomas elementų unikalus numeris ir dokumentacija (angl. *Documentation*). Varto vardas – pavadinimas (angl. *Name*) saugomas „Activity“ klasėje. Taigi kiekvienas vartų tipas savyje saugo savo unikalų numerį, vardą (angl. *Name*), aprašą bei dokumentaciją (angl. *Description* ir *Documentation*), tipą. Visa ši informacija pateikiama ir 3.18 paveiksle.

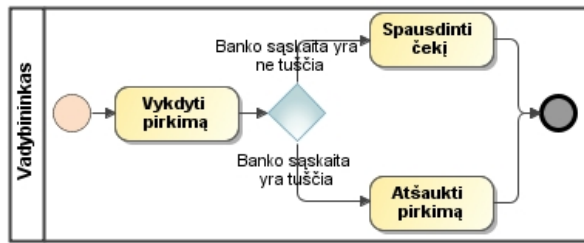


3.18 pav. Vartų (angl. *Gateway*) klasės metamodelis

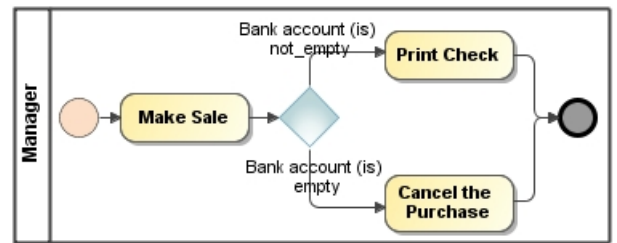
3.19 – 3.26 paveiksluose pateiktos vienintelio sprendimo/suliejimo: duomenimis paremto ir įvykiu paremto, taip pat daugiau nei vieno sprendimo/suliejimo (OR) ir lygiagretaus išsišakojimo/sujungimo (AND) vartų tipų situacijos.

## Grafinis atvaizdavimas – nr.2

3.19 ir 3.20 paveiksluose naudojami duomenimis grindžiami vartai (angl. *Data Based Gateway*). Paveiksluose matoma, kad sąlyga yra paprasta: veiksmai atliekami atsižvelgiant į sekos srauto vardą: „*Banko sąskaita yra tuščia*“ (angl. „*Bank account (is) empty*“) arba „*Banko sąskaita yra netuščia*“ (angl. „*Bank account (is) not empty*“). Pavyzdžiui, „*Spausdinti čekį*“ (angl. „*Print check*“) veiksmas atliekamas tada, kai „*Banko sąskaita yra tuščia*“ (angl. „*Bank account (is) empty*“). Veiksmas „*Atšaukti pirkimą*“ (angl. „*Cancel purchase*“) vykdoma, kai „*Banko sąskaita yra tuščia*“ (angl. „*Bank account (is) empty*“).



3.19 pav. Situacijos nr. 2 sprendimas (liet.)



3.20 pav. Situacijos nr. 2 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.3 ir 3.4 lentelėse.

3.3 lentelė. Situacijos nr. 2 veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Sequence Flow>	Daiktavardis	Term	bank	<a href="#">bank</a>
<Sequence Flow>	Daiktavardis	Term	account	<a href="#">account</a>
<Activity>	Daiktavardis	Term	sale	<a href="#">sale</a>
<Activity>	Daiktavardis	Term	check	<a href="#">check</a>
<Activity>	Daiktavardis	Term	purchase	<a href="#">purchase</a>
<Lane>	Daiktavardis	Term	manager	<a href="#">manager</a>
<Activity>	Veiksmažodis	Verb	make	<a href="#">make</a>
<Activity>	Veiksmažodis	Verb	print	<a href="#">print</a>
<Activity>	Veiksmažodis	Verb	cancel	<a href="#">cancel</a>
<Sequence Flow>	Veiksmažodis	Verb	(is)	<a href="#">(is)</a>
<Sequence Flow>	Veiksmažodis	Verb	empty	<a href="#">empty</a>
<Sequence Flow>	Veiksmažodis	Verb	not_empty	<a href="#">not_empty</a>
<Activity>	Raktažodis	Keyword	the	<a href="#">the</a>

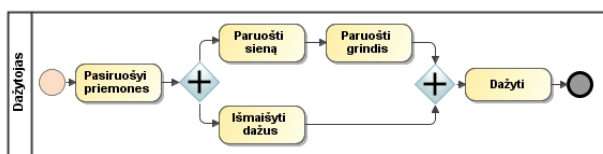
3.4 lentelė. Situacijos nr. 2 veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Lane> & <AltActivity> & <Sequence Flow>	Operational business rule (obligation)	It is obligatory that <a href="#">manager</a> <a href="#">cancel</a> the <a href="#">purchase</a> if <a href="#">bank account</a> <a href="#">(is) empty</a> .	It is obligatory that manager cancel the purchase if bank account (is) empty.
<Lane> & <AltActivity> & <Sequence Flow>	Operational business rule (obligation)	It is obligatory that <a href="#">manager</a> <a href="#">print</a> the <a href="#">check</a> if <a href="#">bank account</a> <a href="#">(is) not_empty</a> .	It is obligatory that manager print the check if bank account (is) not_empty.

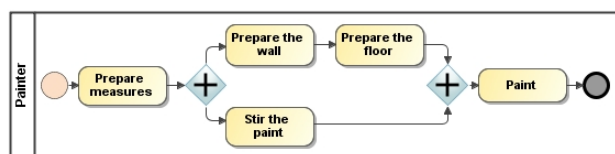
### Grafinis atvaizdavimas – nr. 3

Lygiagretūs vartai (angl. *Parallel Gateway*) (žiūrėti 3.21 ir 3.22 pav.) tinkami naudoti, kai veiksmai turi vykti lygiagrečiai ir turi būti įvykdyti, norint atlikti veiklą, esančią po antrojo vartų simbolio. Šiuo atveju veiksmas „Dažyti“ (angl. „*Paint*“) nevykdomas, jei atliktas tik veiksmas „Paruošti sieną“ (angl. „*Prepare the wall*“). Kad sieną būtų galima dažyti, reikia

sulaukti ir veiksmų „Paruošti grindis“ (angl. „Prepare the floor“) bei „Išmaišyti dažus“ (angl. „Stir the paint“) pabaigos. Kai visi veiksmai yra atlikti, tada galima veikla, kuri seka po antrojo lygiagrečių vartų simbolio.



3.21 pav. Situacijos nr. 3 sprendimas (liet.)



3.22 pav. Situacijos nr. 3 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.5 ir 3.6 lentelėse.

3.5 lentelė. Situacijos nr. 3 veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Pool>	Daiktavardis	Term	painter	<a href="#">painter</a>
<Activity>	Daiktavardis	Term	paint	<a href="#">paint</a>
<Activity>	Daiktavardis	Term	measures	<a href="#">measures</a>
<Activity>	Daiktavardis	Term	wall	<a href="#">wall</a>
<Activity>	Daiktavardis	Term	floor	<a href="#">floor</a>
<Activity>	Veiksmažodis	Verb	prepare	<a href="#">prepare</a>
<Activity>	Veiksmažodis	Verb	stir	<a href="#">stir</a>
<Activity>	Veiksmažodis	Verb	paint	<a href="#">paint</a>
<Activity>	Raktažodis	Keyword	the	<a href="#">the</a>

3.6 lentelė. Situacijos nr. 3 veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Pool> & <ActivityBefore1stGateway> & <AltActivitiesAfter1stGateway>	Operational business rules (obligation)	It is obligatory that <a href="#">painter</a> <a href="#">prepare</a> <a href="#">measures</a> AND <a href="#">prepare</a> the <a href="#">wall</a> AND <a href="#">stir</a> the <a href="#">paint</a> .	It is obligatory that painter prepare measures AND prepare the wall AND stir the paint.
<Pool> & <AltActivitiesBefore2ndGateway> & <ActivityAfter2ndGateway>	Operational business rules (obligation)	It is obligatory that <a href="#">painter</a> <a href="#">prepare</a> the <a href="#">floor</a> AND <a href="#">stir</a> the <a href="#">paint</a> AND <a href="#">paint</a> .	It is obligatory that painter prepare the floor AND stir the paint AND paint.

### BPMN saugoma informacija:

### Lygiagrečių vartų tipo nurodymas programiniame kode:

```
<node xmi:type="uml:DecisionNode"
xmi:id="_17_0_1_8cd027c_1332268790236_662945_2695"
clientDependency="_17_0_1_8cd027c_1332268793836_733980_2709_17_0_1_8cd027c_1332268800306_113042_2722"
inPartition="_17_0_1_8cd027c_1332268766326_441645_2639">
  <eAnnotations xmi:id="_zy9JwI1tEeGMep-p415kfA"
source="appliedStereotypes">
```



```

        <contents xmi:type="BPMNProfile_0:BPMNProfile__and"
xmi:id="_zy9JwY1tEeGMep-p415kfA"/>
        </eAnnotations>
</node>

```

Pagrindinis dalykas, kuris skiriasi nuo ankstesnėje situacijoje naudojamų vartų, yra tas, kad nagrinėjamas elementas saugo papildomą informaciją: „xmi:type="BPMNProfile\_0:BPMNProfile\_\_and“. Elementas priklauso „EventBasedGateway (from Gateway)“ klasei, kurioje ir saugoma tipo informacija.

#### Grafinis atvaizdavimas – nr. 4

Vieno sprendimo/suliejimo tipo vartai (žiūrėti 3.23 ir 3.24 pav.) naudojami, kai siekiama sulaukti bent vienos veiklos įvykdymo. 3.23 ir 3.24 paveikslus galima suprasti taip: veiksmas „Suteikti pirmąją pagalbą“ (angl. „Provide first aid“) vykdomas, kai įvyksta bent vienas veiksmas iš: „Iškviešti policiją“ (angl. „Call the police“), „Iškviešti greitąją“ (angl. „Call the ambulance“) arba „Iškviešti gaisrinę“ (angl. „Call the fire-service“).



3.23 pav. Situacijos nr. 4 sprendimas (liet.)



3.24 pav. Situacijos nr. 4 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.7 ir 3.8 lentelėse.

3.7 lentelė. Situacijos nr. 4 veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Activity>	Daiktavardis	Term	police	<a href="#">police</a>
<Activity>	Daiktavardis	Term	ambulance	<a href="#">ambulance</a>
<Activity>	Daiktavardis	Term	fire-service	<a href="#">fire-service</a>
<Activity>	Daiktavardis	Term	first aid	<a href="#">first-aid</a>
<Activity>	Daiktavardis	Term	help	<a href="#">help</a>
<Activity>	Veiksmažodis	Verb	call	<a href="#">call</a>
<Activity>	Raktažodis	Keyword	the	<a href="#">the</a>

3.8 lentelė. Situacijos nr. 4 veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklė
<ActivityBefore1stgateway> & <AltActivitiesAfter1stGateway>	Definitional business rules (necessity)	It is necessary to <a href="#">call the police</a> OR <a href="#">call the ambulance</a> OR <a href="#">call the fire-service</a> .	It is necessary to <a href="#">call the police</a> OR <a href="#">call the ambulance</a> OR <a href="#">call the fire-service</a> .
<AltActivitiesBefore2ndGateway> & <ActivityAfter2ndGateway>	Operational business rules (obligation)	It is obligatory to <a href="#">call the police</a> OR <a href="#">call the ambulance</a> OR <a href="#">call the fire-service</a> to <a href="#">provide first aid</a> .	It is obligatory to <a href="#">call the police</a> OR <a href="#">call the ambulance</a> OR <a href="#">call the fire-service</a> to <a href="#">provide first aid</a> .

## BPMN saugoma informacija:

### Vieno sprendimo/suliejimo tipo nurodymas programiniame kode:

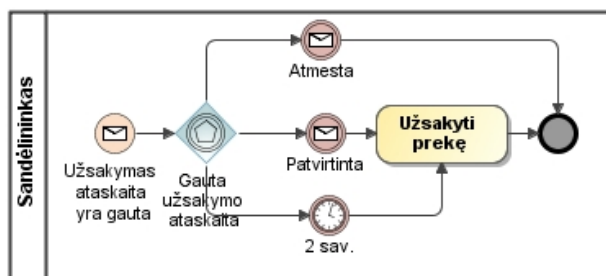
```
<node xmi:type="uml:DecisionNode"
xmi:id="_17_0_1_8cd027c_1332269396776_885435_3106"
clientDependency="_17_0_1_8cd027c_1332269407335_144550_3138
_17_0_1_8cd027c_1332269403043_188492_3126
_17_0_1_8cd027c_1332269410438_865364_3151"
inPartition="_17_0_1_8cd027c_1332269364742_734786_3050">
  <eAnnotations xmi:id="_zywVcI1tEeGMep-p415kfA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__oR"
xmi:id="_zywVcY1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>
```

Šiam elementui taip pat nurodomas tipas. Programiniame kode jis atrodo taip:

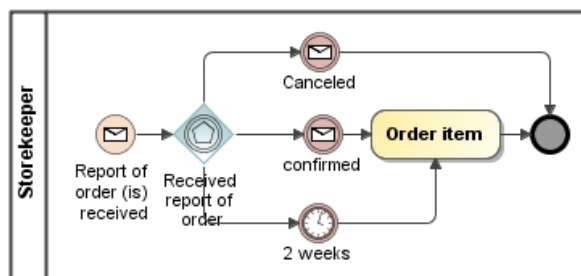
```
„xmi:type="BPMNProfile_0:BPMNProfile__oR“.
```

### Grafinis atvaizdavimas – Nr.5

3.25 ir 3.26 paveiksluose naudojami įvykiu grindžiami (angl. *Event Based*) tipo vartai. Jie taikomi, kai reikalingas informavimas prieš veiksmo atlikimą. Pavyzdžiui, *sandėlininkas* (angl. *Storekeeper*) informuojamas pranešimu. Tai reiškia, kad prieš vykdant veiksmą „Užsakyti prekę“ (angl. „*Order item*“) pirmiausiai yra aktyvuojamas įvykis (vienas iš „Atmesta“ (angl. „*cancelled*“), „Patvirtinta“ (angl. „*confirmed*“), „2 sav.“ (angl. „*2 weeks*“)) ir tada bus vykdomi tie veiksmai, kurie ir seka po sužadinto įvykio. Visi kiti keliai tampa nebegaliojantys.



3.25 pav. Situacijos nr. 5 sprendimas (liet.)



3.26 pav. Situacijos nr. 5 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.9 ir 3.10 lentelėse.

3.9 lentelė. Situacijos nr. 5 veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Intermediate Timer Event>	Pavadinimas	Name	2	<u>2</u>
<Activity>	Daiktavardis	Term	report	<a href="#">report</a>
<Activity>	Daiktavardis	Term	order	<a href="#">order</a>
<Intermediate Message Event>	Daiktavardis	Term	weeks	<a href="#">weeks</a>



BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Activity>	Daiktavardis	Term	item	<u>item</u>
<Start Message Event>	Veiksmažodis	Verb	is	<i>is</i>
<Intermediate Message Event>	Išraiška	Characteristic fact type	cancelled	<i>cancelled</i>
<Intermediate Message Event>	Išraiška	Characteristic fact type	confirmed	<i>confirmed</i>
<Start Message Event> & <Event Based Gateway>	Išraiška	Characteristic fact type	received	<i>received</i>
<Start Message Event>	Raktažodis	Verb	of	<i>of</i>
<Start Message Event> & <Event Based Gateway>	Išraiška	Binary fact type <term><verb><term>	report of order	<u>report of order</u>

Daiktavardžio formoje fakto tipo simbolis <fact type symbol>, kuris yra išreiškiamas per <verb>, paprastai įgauna reikšmę „*of*“. Pvz.: document has copy tampa copy of document.

### 3.10 lentelė. Situacijos nr. 5 veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Pool> & <Start Message Event>	Operational business rules (obligation)	<i>It is obligatory that <u>storekeeper has report of order that (is) received.</u></i>	It is obligatory that storekeeper has report of order that (is) received.
<Start Message Event> & <Event Based Gateway> & <AltEventsAfter Gateway>	Definitional business rules (restricted possibility)	<i>It is possible that <u>report of order that (is) received can have status cancelled OR confirmed OR waited for 2 weeks.</u></i>	It is possible that report of order that (is) received can have status cancelled OR confirmed OR waited for 2 weeks.
<Activity> & <Event Based Gateway> & <Intermediate Message Event>	Operational business rules (obligation)	<i>It is obligatory to <u>order item if received report of order is confirmed.</u></i>	It is obligatory to order item if received report of order is confirmed.
<Activity> & <Event Based Gateway> & <Intermediate Time Event>	Operational business rules (obligation)	<i>It is obligatory to <u>order item if received report of order is done in 2 weeks</u></i>	It is obligatory to order item if received report of order is done in 2 weeks

#### BPMN saugoma informacija:

#### Duomenimis grindžiamų vartų saugoma informacija:

```
<node xmi:type="uml:DecisionNode"
xmi:id="_17_0_1_8cd027c_1332274843122_46750_3589" name="Received report of
order" clientDependency="_17_0_1_8cd027c_1332274872905_724140_3725
_17_0_1_8cd027c_1332274869076_660953_3719"
```

```

_17_0_1_8cd027c_1332274880900_338194_3743"
inPartition="_17_0_1_8cd027c_1332274819215_25762_3533">
  <eAnnotations xmi:id="_zy5fYY1tEeGMep-p415kfA"
source="appliedStereotypes">
  <contents xmi:type="BPMNProfile_0:BPMNProfile__eventBasedXOR"
xmi:id="_zy5fYY1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>

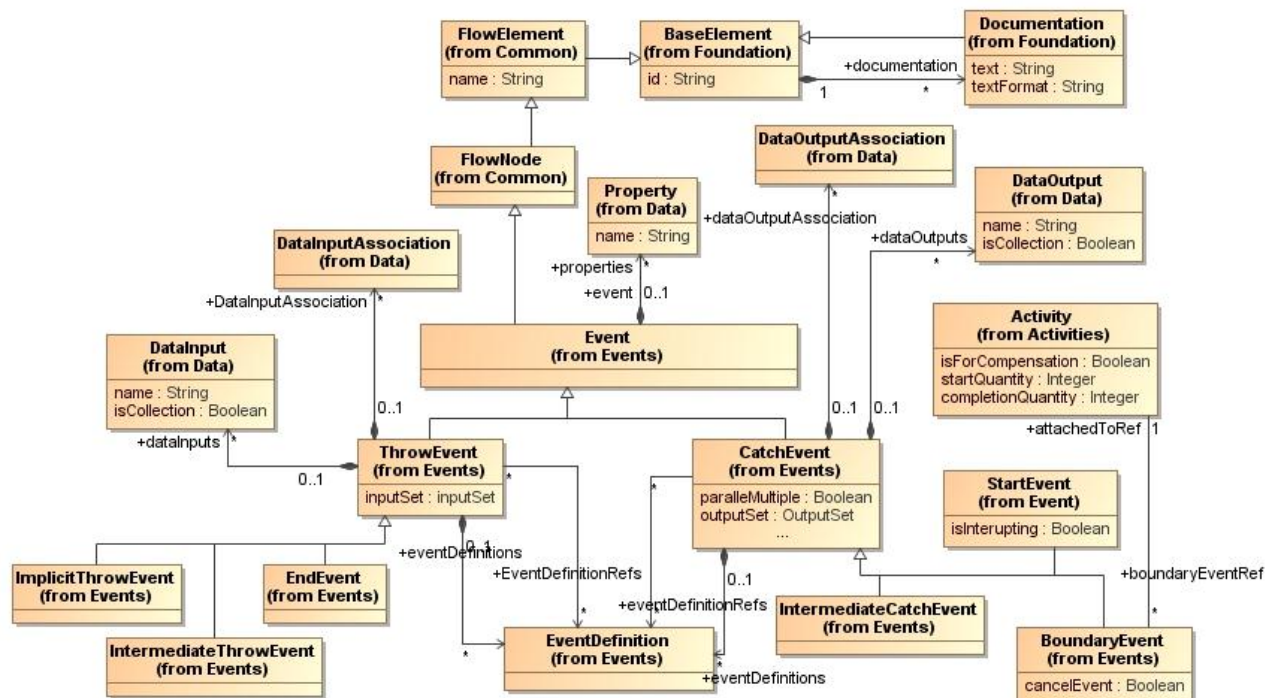
```

Šis elementas, kaip ir anksčiau aptarti vartų tipai, saugo tokią pačią informaciją, t.y., *unikalus numeris, vardas, dokumentacija, vartų tipas (GatewayType)*. Šių vartų tipas aprašomas taip: `type="BPMNProfile_0:BPMNProfile__eventBasedXOR"`.

### 3.3.1.3 Įvykiai (angl. *Events*)

Modeliuojant verslo procesą, modeliuojami įvykiai, kurie vyksta versle. Jais parodoma kaip veikiami procesų srautai. Įvykis arba pradeda proceso srautą, arba įvyksta srauto metu, arba užbaigia jį. BPMN2 notacijoje yra išskirti trijų pagrindinių tipų įvykiai: pradžios (angl. *Start*), tarpiniai (angl. *Intermediate*) ir pabaigos (angl. *End*). Šie įvykiai įtakoja proceso srautą ir dažniausiai turi priežastį.

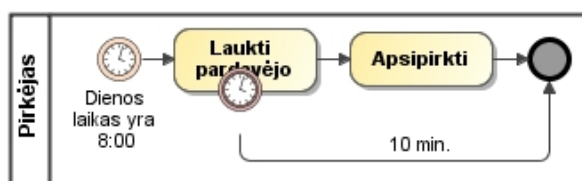
Įvykiai gali būti *throw* (tarpinis ir pabaigos įvykiai) ir *catch* (tarpinis, pradžios ir veiklą įtakojantis (angl. *Boundary*) įvykiai) tipo. Tai atitinka pagrindines įvykio metamodelio klases (žiūrėti 3.27 pav.).



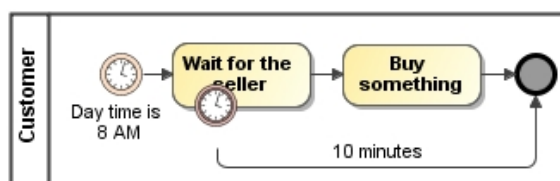
3.27 pav. Įvykio (angl. *Event*) klasės metamodelis

## Grafinis atvaizdavimas – nr. 6

Įtakos procesam turintys įvykiai pavaizduoti 3.28 ir 3.29 paveiksluose. Matomi trijų tipų įvykiai: pradžios, pabaigos ir tarpinis. Pradžios ir tarpinis įvykiai turi laiko (angl. *Timer*) tipą, o pabaigos įvykis yra standartinis. Šiuo atveju tarpinis įvykis gali būti vadinamas briauniniu įvykiu (angl. *Boundary Event*). Tai reiškia, kad pardavėjo laukiama 10 minučių („10 min.“ (angl. „10 minutes“)) ir jei jo nesulaukiama per tiek laiko, tuomet procesas baigiamas. Priešingu atveju galima „Apsipirkti“ (angl. „Buy something“). „Dienos laikas yra 8:00“ (angl. „Day time is 8 AM“) įvykis reiškia, kad 8 valandą ryto pardavėjas turi būti atėjęs; pabaigos įvykiu baigiamas visas procesas.



3.28 pav. Situacijos nr. 6 sprendimas (liet.)



3.29 pav. Situacijos nr. 6 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas pateikti 3.11 ir 3.12 lentelėse.

3.11 lentelė. Situacijos nr. 6 sprendimo veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Start Time Event>	Pavadinimas	Name	8	<u>8</u>
<Intermediate Timer Event>	Pavadinimas	Name	10	<u>10</u>
<Start Time Event>	Pavadinimas	Name	AM	<u>AM</u>
<Pool>	Daiktavardis	Term	customer	<u>customer</u>
<Start Time Event>	Daiktavardis	Term	day	<u>day</u>
<Start Timer Event> & <Intermediate Timer Event>	Daiktavardis	Term	time	<u>time</u>
<Activity>	Daiktavardis	Term	something	<u>something</u>
<Boundary Activity>	Daiktavardis	Term	seller	<u>seller</u>
<Intermediate Timer Event>	Daiktavardis	Term	minutes	<u>minutes</u>
<Start Timer Event> & <Intermediate Timer Event> & <Sequence Flow>	Veiksmažodis	Verb	is	<u>is</u>
<Activity>	Veiksmažodis	Verb	buy	<u>buy</u>
<Boundary Activity>	Veiksmažodis	Verb	wait	<u>wait</u>
<Boundary Activity>	Veiksmažodis	Verb	wait for	<u>wait for</u>
<Boundary Activity>	Veiksmažodis	Verb	for	<u>for</u>
<Boundary Activity>	Raktažodis	Keyword	the	<u>the</u>

3.12 lentelė. Situacijos nr. 6 sprendimo veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Start Timer Event>	Definitional business rule (necessity)	It is necessary that <u>day time is 8 AM</u> .	It is necessary that day time is 8 AM.
<Start Timer Event> & <Pool> & <Activity>	Operational business rule (obligation)	It is obligatory that <u>day time is 8 AM for customer to wait for the seller</u> .	It is obligatory that day time is 8 AM for customer to wait for the seller.
<Pool> & <Boundary Activity> & <Timer Event>	Definitional business rule (necessity)	It is necessary that <u>customer wait for the seller 10 minutes</u> .	It is necessary that customer wait for the seller 10 minutes.
<Pool> & <Boundary Activity> & <Boundary Timer Event> & <End Event>	Operational business rule (obligation)	It is obligatory that <u>customer wait for the seller more than 10 minutes to finish</u> .	It is obligatory that customer wait for the seller more than 10 minutes to finish.
<Start Timer Event> & <Activity>	Operational business rule (prohibition)	It is prohibited that <u>day time is less than 8 AM to buy something</u>	It is prohibited that day time is less than 8AM to buy something
<Pool> & <Boundary Activity> & <Start Timer Event>	Operational business rules (restricted)	It is permitted that <u>customer wait for the seller if day time is 8 AM</u>	It is permitted that customer wait for the seller if day time is 8 AM.
<Pool> & <Activity> & <End Event>	Operational business rule (obligation)	It is obligatory that <u>customer buy something to finish</u> .	It is obligatory that customer buy something to finish.

### BPMN saugoma informacija:

Kiekvienas pradžios įvykis (nesvarbu, ar jo tipas yra laikas (angl. *Timer*), pranešimas (angl. *Message*), sąlyga (angl. *Conditional*), signalas (angl. *Signal*) ar kartotinis (angl. *Multiple*)) saugo tokią pat informaciją kaip ir kiti srautų grupės elementai. Yra vienas skirtumas tarp paprastojo pradžios įvykio ir įvykio, kuriam nurodytas tipas.

Paprastas pradžios įvykis saugo tokią informaciją:

```
<node xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1332276168959_979138_4597"
clientDependency="_17_0_1_8cd027c_1332276171295_289779_4623"
inPartition="_17_0_1_8cd027c_1332276143784_140257_4571">
  <eAnnotations xmi:id="_z0eMsI1tEeGMep-p415kfA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__startEvent"
xmi:id="_z0eMsY1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>
```

*Pavadinimo* informacijos nėra, kadangi paprastam įvykiui jis buvo nenurodytas.

„*Day time is 8 AM*“

Tai yra pradžios įvykis, kuris yra laiko tipo. Tipas nurodomas programiniame kode:  
<StartEvent Trigger="Timer">. Tai yra esminis paprastojo ir įvykio su tipu saugomos informacijos skirtumas.

```
<node xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1332275282640_930854_3912" name="Day time is &#xA;8
AM" clientDependency="_17_0_1_8cd027c_1332275284859_766694_3935"
inPartition="_17_0_1_8cd027c_1332275251175_229148_3886">
  <eAnnotations xmi:id="_zzCpUIltEeGMep-p415kfA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__timerStartEvent"
xmi:id="_zzCpUY1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>
```

### Tarpinis įvykis (šiuo atveju įvykis ant veiklos „Wait for the seller“ briaunos)

Duomenų faile jis aprašomas prie to paties veiklos elemento. Papildomai pateikiama tokia informacija:

```
    <containedNode xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1335436239164_614522_2265"
clientDependency="_17_0_1_8cd027c_1335436241702_47744_2282">
      <eAnnotations xmi:id="_y7jYwI-_EeGd-NCWcLOhuQ"
source="appliedStereotypes">
        <contents
xmi:type="BPMNProfile_0:BPMNProfile__timerIntermediateEvent" xmi:id="_y7jYwY-
_EeGd-NCWcLOhuQ"/>
      </eAnnotations>
    </containedNode>
</node>
```

Įterpiamas „containedNod“ struktūra, kurioje pateikiama, koks elemento tipas yra ant veiklos briaunos. Šiuo atveju:

```
xmi:type="BPMNProfile_0:BPMNProfile__timerIntermediateEvent"
xmi:id="_y7jYwY-_EeGd-NCWcLOhuQ"
```

Visa ši informacija apie veiklą ir įvykį, pateikiama žemiau esančiame programiniame kode:

```
<node xmi:type="uml:StructuredActivityNode"
xmi:id="_17_0_1_8cd027c_1332275284826_145690_3929" name="Wait for the seller"
clientDependency="_17_0_1_8cd027c_1332275289298_646806_3947"
inPartition="_17_0_1_8cd027c_1332275251175_229148_3886">
  <eAnnotations xmi:id="_y8BS0I-_EeGd-NCWcLOhuQ"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__task"
xmi:id="_y8BS0Y-_EeGd-NCWcLOhuQ"/>
  </eAnnotations>
  <containedNode xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1335436239164_614522_2265"
clientDependency="_17_0_1_8cd027c_1335436241702_47744_2282">
    <eAnnotations xmi:id="_y7jYwI-_EeGd-NCWcLOhuQ"
source="appliedStereotypes">
      <contents
xmi:type="BPMNProfile_0:BPMNProfile__timerIntermediateEvent" xmi:id="_y7jYwY-
_EeGd-NCWcLOhuQ"/>
    </contents>
  </containedNode>
</node>
```

```

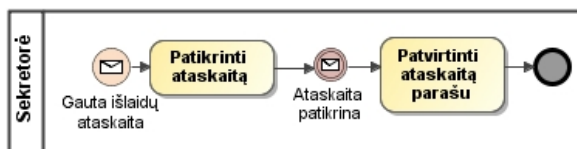
    </eAnnotations>
  </containedNode>
</node>

```

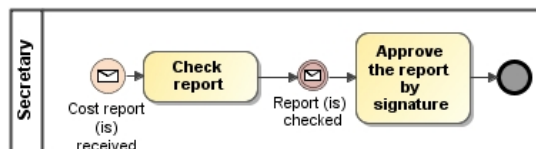
### Grafinis atvaizdavimas – nr. 7

3.30 ir 3.31 paveiksluose pateikta situacija nelabai skiriasi nuo šeštosios. Esminis skirtumas tai, kad tarpinis įvykis „Ataskaita patikrinta“ (angl. „*The report (is) checked*“) nėra briaunos įvykis. Be to, pavyzdyje naudojami pranešimo tipo pradžios ir tarpinis įvykiai, kurie turi įtakos visam procesui, bei pabaigos įvykis.

Kai tarpinis įvykis nepriklauso jokiai veiklai, tuomet jis nesaugo kito elemento unikalus numeris bei taip-ne reikšmės). Toks įvykis taip pat naudojamas kitaip, nei ankstesniame pavyzdyje esantis laiko įvykis. Čia „Ataskaita patikrinta“ (angl. „*Report (is) checked*“) yra kaip pranešimas (rezultatas) ir nėra jokios sąlygos, pavyzdžiui, veiklai „Patikrinti ataskaitą“ (angl. „*Check report*“). Po pranešimo (įvykio) iškart vykdomas veiksmas „Patvirtinti ataskaitą parašu“ (angl. „*Approve the report by signature*“).



3.30 pav. Situacijos nr. 7 sprendimas (liet.)



3.31 pav. Situacijos nr. 7 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.13 ir 3.14 lentelėse.

3.13 lentelė. Situacijos nr. 7 sprendimo veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Start Message Event>	Daiktavardis	Term	cost	<a href="#">cost</a>
<Activity> & <Start Message Event> & <Intermediate Message Event>	Daiktavardis	Term	report	<a href="#">report</a>
<Activity>	Daiktavardis	Term	signature	<a href="#">signature</a>
<Activity>	Veiksmažodis	Verb	check	<a href="#">check</a>
<Activity>	Veiksmažodis	Verb	approve	<a href="#">approve</a>
<Activity> & <Intermediate Message Event>	Veiksmažodis	Verb	checked	<a href="#">checked</a>
<Start Message Event>	Veiksmažodis	Verb	received	<a href="#">received</a>
<Start Message Event> & <Intermediate Message Event>	Veiksmažodis	Verb	is	<a href="#">(is)</a>
<Activity>	Veiksmažodis	Verb	by	<a href="#">by</a>



BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Activity>	Raktažodis	Keyword	the	<i>the</i>
<Intermediate Message Event>	Išraiška	Characteristic fact type <term><verb>	report (is) checked	<u>report</u> (is) <i>checked</i>
<Start Message Event>	Išraiška	Characteristic fact type <term><verb>	report (is) received	<u>report</u> (is) <i>received</i>
<Activity>	Išraiška	Partitive fact type <term><verb><term>	report by signature	<u>report</u> by <u>signature</u>

3.14 lentelė. Situacijos nr. 7 sprendimo veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Pool> & <Start Message Event>	Definitional business rule (necessity)	<i>It is necessary that <u>secretary has cost report that (is) received.</u></i>	It is necessary that secretary has cost report that (is) received.
<Pool> & <Start Message Event> & <Activity>	Operational business rule (obligation)	<i>It is obligatory that <u>secretary has cost report that (is) received to check report.</u></i>	It is obligatory that secretary has cost report that (is) received to check report.
<Pool> & <Activity> & <Intermediate Message Event>	Operational business rule (obligation)	<i>It is obligatory that <u>secretary check report to have report checked.</u></i>	It is obligatory that secretary check report to have report checked.
<Pool> & <Intermediate Message Event> & <Activity>	Operational business rule (obligation)	<i>It is obligatory that <u>secretary has report checked to approve the report by signature.</u></i>	It is obligatory that secretary has report checked to approve the report by signature.
<Pool> & <Activity> & <End Event>	Operational business rule (obligation)	<i>It is obligatory that <u>secretary approve the report by signature to finish.</u></i>	It is obligatory that secretary approve the report by signature to finish.

### BPMN saugoma informacija:

#### „Cost report (is) received“

Tai yra pradžios (pranešimo tipo) įvykis. Tipas nurodomas programiniame kode:

```
type="BPMNProfile_0:BPMNProfile__messageStartEvent".
```

```
<node xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1332275616296_544528_4161" name="Cost
report&#xA;(is)&#xA;received"
clientDependency="_17_0_1_8cd027c_1332275619164_977379_4187"
inPartition="_17_0_1_8cd027c_1332275589607_6529_4133">
  <eAnnotations xmi:id="_zzCCQI1tEeGMep-p415kfA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__messageStartEvent"
xmi:id="_zzCCQY1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>
```

#### „The report (is) checked“

Tai yra tarpinis pranešimo tipo įvykis. Saugomai informacijai įtakos turi tik tai, kad yra nurodoma: `type="BPMNProfile_0:BPMNProfile__messageIntermediateEvent"`, o ne `<contents xmi:type="BPMNProfile_0:BPMNProfile__messageStartEvent" xmi:id="_zzCCQYltEeGMep-p415kfA"/>`

```

<node xmi:type="uml:Action"
xmi:id="_17_0_1_8cd027c_1332275622616_176319_4199" name="Report (is)
&#xA;checked" clientDependency="_17_0_1_8cd027c_1332275632526_715199_4231"
inPartition="_17_0_1_8cd027c_1332275589607_6529_4133">
  <eAnnotations xmi:id="_zzFsoIltEeGMep-p415kfA"
source="appliedStereotypes">
    <contents
xmi:type="BPMNProfile_0:BPMNProfile__messageIntermediateEvent"
xmi:id="_zzFsoYltEeGMep-p415kfA"/>
  </eAnnotations>
</node>

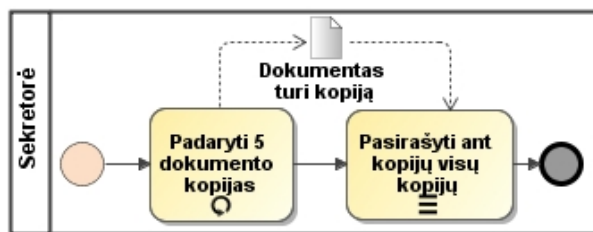
```

### 3.3.2 VŽ ir VT išgavimas iš duomenų elementų

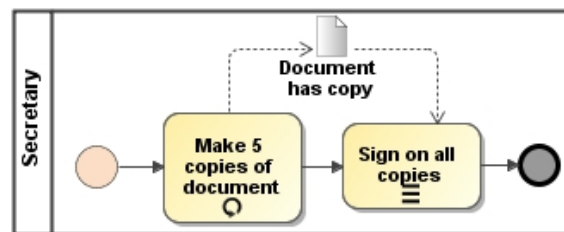
#### 3.3.2.1 Duomenų objektas (angl. *Data Object*)

##### Grafinis atvaizdavimas – nr.8

Situacijoje nr. 8 naudojamos dvi veiklos ir duomenų objektas (žiūrėti 3.32 ir 3.33 pav.). Kitaip nei ankstesniame pirmajame pavyzdyje (Situacija nr. 1), čia naudojamos ciklo tipo (angl. *Loop Type*) veiklos: standartinės (angl. *Standard*) („Padaryti 5 dokumento kopijas“ (angl. „*Make 5 copies of document*“) ir kartotinės (angl. *Multi-Instance*) („Pasirašyti ant visų kopijų“ (angl. „*Sign on all copies*“)). Duomenų objektas – „Dokumentas turi kopiją“ (angl. „*Document has copy*“).



3.32 pav. Situacijos nr. 8 sprendimas (liet.)



3.33 pav. Situacijos nr. 8 sprendimas (angl.)

Pagal sudarytus algoritmus išgautos veiklos taisyklės ir veiklos žodynas, pateikti 3.15 ir 3.16 lentelėse.

3.15 lentelė. Situacijos nr. 8 sprendimo veiklos žodynas

BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
<Activity>	Pavadinimas	Name	5	<u>5</u>
<Activity> & <Data Object>	Daiktavardis	Term	document	<u>document</u>
<Data	Daiktavardis	Term	copy	<u>copy</u>



BPMN elem.	Tipas	SBVR terminai	Žodžiai	Atvaizdavimas
Object>				
<Activity>	Daiktavardis	Term	copies	<a href="#">copies</a>
Pool	Daiktavardis	Term	secretary	<a href="#">secretary</a>
<Activity>	Veiksmažodis	Verb	sign on	<a href="#">sign on</a>
<Activity>	Veiksmažodis	Verb	make	<a href="#">make</a>
<Activity>	Raktažodis	Keyword	all	<a href="#">all</a>
<Activity>	Išraiška	Binary fact type <term><verb><term>	copies of document	<a href="#">copies of document</a>
<Data Object>	Išraiška	Binary fact type <term><verb><term>	document has copy	<a href="#">document has copy</a> <a href="#">copy of document</a>

3.16 lentelė. Situacijos nr. 8 sprendimo veiklos taisyklių sąrašas

BPMN elem.	Taisyklės tipas	Atvaizdavimas	Veiklos taisyklės
<Data Object>	Definitional business rules (necessity)	<i>It is necessary that</i> <a href="#">document has copy.</a>	It is necessary that document has copy.
<Data Object> & <Activity>	Operational business rule (obligation)	<i>It is obligatory that</i> <a href="#">document has copy to sign on all copies.</a>	It is obligatory that document has copy to sign on all copies.

### BPMN saugoma informacija:

*Unikalus numeris, pavadinimas ir dokumentacija* atvaizduojama taip pat, kaip ir ankstesniame pavyzdyje.

### „Make 5 copies of document“

Šis išsiskiria papildoma saugoma informacija, t.y. ciklui apibrėžti – LoopType požymis.

```
<node xmi:type="uml:StructuredActivityNode"
xmi:id="_17_0_1_8cd027c_1332276171265_360266_4616" name="Make 5 copies of
document" clientDependency="_17_0_1_8cd027c_1335125904322_316642_2589
_17_0_1_8cd027c_1332276178067_646233_4636"
inPartition="_17_0_1_8cd027c_1332276143784_140257_4571">
  <eAnnotations xmi:id="_zy4RQI1tEeGMep-p415kFA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__standardLoop"
xmi:id="_zy4RQY1tEeGMep-p415kFA"/>
    <contents xmi:type="BPMNProfile_0:BPMNProfile__task"
xmi:id="_zzc5AI1tEeGMep-p415kFA"/>
  </eAnnotations>
```

Kitose notacijose ciklas XMI faile atvaizduojamas taip:

```
<Loop LoopType="Standard">
<LoopStandard LoopCounter="0" LoopMaximum="0" TestTime="Before">
  <LoopCondition />
</LoopStandard>
</Loop>
```

Šiuo atveju ciklo tipo veikla yra standartinė ( LoopType – Standard). Toks tipas rodo, kad veikla turi būti kartojama nurodytą pakartojimų kiekį (*LoopMaximum*).

### „Sign on all copies“

Šiuo atveju skiriasi tik informacija apie tipą. Kai veikla kartotinė (angl. *Multi-Instance*) reiškia, kad ji vykdoma nuosekliai (šiuo atveju) arba gali būti vykdoma lygiagrečiai nurodytą skaičių kartų. Konkrečiu atveju, tai reiškia, kad komiteto narys pasirašęs ant vieno dokumento kopijos, turi pasirašyti ir ant likusių kopijų.

```
<node xmi:type="uml:StructuredActivityNode"
xmi:id="_17_0_1_8cd027c_1332276178027_575645_4629" name="Sign on all copies"
clientDependency="_17_0_1_8cd027c_1332276189356_376406_4664"
inPartition="_17_0_1_8cd027c_1332276143784_140257_4571">
  <eAnnotations xmi:id="_zzmC8I1tEeGMep-p415kfA"
source="appliedStereotypes">
    <contents xmi:type="BPMNProfile_0:BPMNProfile__task"
xmi:id="_zzmC8Y1tEeGMep-p415kfA"/>
    <contents xmi:type="BPMNProfile_0:BPMNProfile__multiInstanceLoop"
xmi:id="_zz4W0I1tEeGMep-p415kfA"/>
  </eAnnotations>
</node>
```

### **„Document has copy“**

Duomenų objektas saugo tik *unikalaus numerio, dokumentacijos ir pavadinimo* informaciją.

## **3.4 Automatizuoto veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų diagramos algoritmo sudarymo rezultatai**

1. Siekiant išsiaiškinti, kaip turi veikti algoritmas, kad automatiniu būdu būtų kuo tiksliau sudarytas VŽ bei VT, buvo analizuojamos tarpusavyje glaudžiai susiję BPMN2 ir SBVR standartai.
2. Darbo metu buvo analizuojamos BPMN2 elementai ir pagrindinės jų jungimo kombinacijos: <„Start Message/Timer Event“> & <„(Boundary) Activity“>; <„(Boundary) Activity“> & <„Activity“>; <„Activity“> arba <„Start/Intermediate“> & <„Event Based/ Data Based/ Parallel/ Inclusive Gateway“>; <„(Boundary Activity)“> & <„Intermediate/ End Event“>; <„Event Based/ Data Based/ Parallel/ Inclusive Gateway“> & <„(Boundary) Activity“> arba <„Intermediate/ End Event“> (priklausomai nuo vartų tipo). Tai suteikia galimybę lengviau sudaryti pradines elementų vardinimo rekomendacijas, kad tolesniame etape būtų lengvesnis bendradarbiavimas tarp abiejų metodo vartotojų ir kad būtų sudarytas kuo tikslesnis VŽ.
3. Ištyrus BPMN2 elementus pastebėta, kad pagrindinė informacija, kuri naudinga sudarant VŽ ir VT, yra elemento unikalus numeris, jo pavadinimas bei numeriai, kurie nurodo, koks yra prieš tai buvęs arba sekantis elementas.
4. Ištyrus SBVR metodologiją nustatyti žodžių, įeinančių į VŽ, tipai, atrinkti fakto tipai. Išanalizavus atskirų atvejų sudarytus VŽ, nustatyta, kad žodynui sudaryti dažniausiai parenkami unarinis (angl. *Characteristic*), binarinis (angl. *Binary*), dalies-visumos (angl.

*Partitive*) fakto tipai bei veiksmažodžiai (angl. *Verb*), daiktavardžiai (angl. *Term*), raktažodžiai (angl. *Keyword*) ir vardiniai pavadinimai (angl. *Name*). Pirmasis fakto tipas dažniausiai naudojamas su „Event“ tipo elementu rezultatui nurodyti, antrasis – sudedamajai daliai nurodyti (pvz.: document has copy), trečiasis – faktui per veiksmažodį išreikšti (pvz.: car movement is included in rental).

5. Remiantis skirtingais BPMN2 modelių atvejais taip pat buvo analizuojamos sudarytos VT, siekiant apibendrinti jas ir nustatyti bendrą struktūrą, pagal kurią vartotojui sistemoje pateikiamas VT sąrašas.
6. Apibendrinus struktūras nustatyta, kad sistemoje bus naudojamos dviejų tipų taisyklės: struktūrinė būtinumo (angl. *Necessity*) ir vykdomoji įpareigojimo/inicijavimo (angl. *Obligation*). Taip nuspręsta todėl, kad taisyklės automatinio būdu sudaromos, kai modelio elementai tikrinami (analizuojami) nuosekliai. Priešingu atveju tinka ir struktūrinės (negalėjimo (angl. *Impossibility*), riboto galėjimo (angl. *Restricted possibility*)) bei vykdomosios (draudimo (angl. *Prohibition*), ribojimo (angl. *Restricted*)) taisyklės.
7. Įvertinus visus BPMN2 elementų jungimo atvejus bei elementų pavadinimus leidžiama lengviau nustatyti VŽ ir VT būsimoje sistemoje sudarymo taisyklės, pagal kurias suformuotas veiklos žodyno bei veiklos taisyklių formavimo algoritmas.
8. Gauti rezultatai naudingi projektuojant panaudojimo atvejų, duomenų bazės klasių (modelis, kuriame saugoma nuskaityti duomenys bei gauti rezultatai susiję su VP, VŽ bei VT), komponentų modelių metodo realizacijai.

## 4 Apribojimai veiklos procesų diagramos elementų pavadinimų užrašymui

### 4.1 Baseino (angl. *Pool*) ir juostų (angl. *Lane*) elementai

Šie elementai VP modelyje yra naudojami skyriams, pareigoms aprašyti, todėl pavadinimai turi būti sudaryti tik iš daiktavardžių: *secretary, bank, sales manager, department of health, etc.* Žodžių junginyje gali būti naudojami raktažodžiai (jungiamieji žodžiai), tokie kaip „*of*“.

### 4.2 Veiklos (angl. *Activity*) elementai

Pagrindinė elemento užrašymo taisyklė yra:

1. Pavadinimas pradedamas veiksmažodžiu, tačiau nenaudojamas žodelis (nurodantis bendratį) *to*.

4.1 lentelė. Veiklos elementų pavadinimų pavyzdžiai

Pavadinimas		Pavadinimas	
prepare measures	✓	to check if extra insurance is necessary	✗
copy document	✓	to sign on copies	✗
package goods	✓		

2. Pavadinimai aiškūs, trumpi. Rekomenduojama pavadinimą skaidyti į dvi ar daugiau veiklų, jei pavadinimas yra ilgas ir išskaidžius nepasikeičia proceso esmė (didžiausias pavadinime naudojamų žodžių skaičius – 4 arba 5).

Pavyzdžiui, veikla „*Check report and make 5 copies*“ gali būti išskaidyta į dvi: „*Check report*“ ir „*Make 5 copies of document*“.

3. Veiklos elemento pavadinime vengti asmens/ skyriaus/ pareigų įvardinimo (tam naudojami baseinai (angl. *Pools*) arba juostos (angl. *Lanes*)).
4. Jei elemento pavadinimas sudaromas iš 2 žodžių, patartina naudoti struktūrą: <Veiksmažodis> <Daiktavardis>

### 4.3 Duomenimis grįsti vartai (angl. *Data – Based Gateway*)

Sudarant VPD, sutinkami kelių tipų vartai, tačiau tik vieno tipo vartams yra suteikiamas pavadinimas. Tai yra, vardinami tik duomenimis grįsti tipo vartai.

Kadangi taisyklės pateikiamos anglų kalba, todėl žemiau pateikiami apribojimai angliškiems pavadinimams:

1. Pavadinimai **nepradedami** veiksmažodžiu *is, do, does, are, have*, siekiant išvengti tokių žodžių pasikartojimų ar taisyklių nekorektiškumo.

4.2 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 1

Pavadinimas	Taisyklė	
customer has an account	It is obligatory that <u>customer</u> <i>has</i> an <u>account</u> <i>is yes to transfer salary to bank account.</i>	✓
has customer an account	It is obligatory that <i>has</i> <u>customer</u> an <u>account</u> <i>is yes to transfer salary to bank account.</i>	✗

2. Nenaudojami įvardžiai, apibūdinantys „mano“, „tavo“, „savo“ ir pan. Nebent „*itself*“, kuris apibūdina asmenį, o ne daiktą.

4.3 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 2

Pavadinimas	Taisyklė	
can handle problem itself	It is obligatory that <u>key account manager</u> <i>can handle</i> <u>problem itself</u> <i>is yes to explain solution</i>	✓
can handle problem myself	It is obligatory that <u>key account manager</u> <i>can handle</i> <u>problem myself</u> <i>is yes to explain solution</i>	✗

3. Jei ant išsišakojimų naudojami ne „Yes“, „No“, o kitokie žodžių junginiai, tai **rekomenduojama** tame junginyje (bet neprivaloma) naudoti žodelį „būdas“ (angl. *Mode of*), „būsena“ (angl. *Status*) ir pan.

4.4 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 3

Pavadinimas	Taisyklė	
mode of delivery	It is obligatory that <u>clerk</u> <i>decide if post or shipment</i> and that <u>mode of delivery</u> <i>is post to check if extra insurance is necessary.</i>	✓
delivery	It is obligatory that <u>clerk</u> <i>decide if post or shipment</i> and that <u>delivery</u> <i>is post to check if extra insurance is necessary.</i>	✓

4. Pavadinimui tinkama (rekomenduojama) .. *of* .. , .. *has* ... kombinacija.

4.5 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 5

Pavadinimas	Taisyklė	
Necessity of insurance	It is obligatory that <u>necessity of insurance</u> <i>is yes to take out insurance.</i>	✓
Extra insurance is necessary?	It is obligatory that <u>insurance</u> <i>is necessary is yes to take out insurance.</i>	✗

5. Pavadinime nenaudojami būdvardžiai (o jei naudojami, tai būdvardis sujungiamas simboliu su žodžiu, kurį jis (būdvardis) apibūdina), kadangi BPMN specifikacijoje nėra apibrėžtas toks žodžių tipas (žiūrėti 4.5 lentelėje).
6. Pavadinime rekomenduojama vengti pirmoje vietoje naudoti veiksmazodį, jei galima panaudoti daiktavardį. Tai yra:

4.6 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 6

Pavadinimas	Taisyklė	
Document needs check?	It is obligatory that <u>document</u> <i>needs</i> <u>check</u> <i>is yes to sign on.</i>	✓
Need document check?	It is obligatory that <i>need</i> <u>document</u> <u>check</u> <i>is yes to sign on.</i>	✗
Client can wait?	It is obligatory that <u>client</u> <i>can</i> <u>wait</u> <i>is yes to sign on.</i>	✓
Can client wait?	It is obligatory that <i>can</i> <u>client</u> <u>wait</u> <i>is yes to sign on.</i>	✗

7. Jei naudojamas būtasis laikas, tai kombinacija turėtų būti pasyvios formos:

4.7 lentelė. Duomenimis grįstų vartų pavadinimų pavyzdžiai 7

Pavadinimas	Taisyklė	
Check printed	It is obligatory that <u>document</u> <i>needs</i> <u>check</u> <i>is yes to sign on.</i>	✓
Print check	It is obligatory that <i>need</i> <u>document</u> <u>check</u> <i>is yes to sign on.</i>	✗

#### 4.4 Sekos srautų (angl. *Sequence Flow*) elementai

Pavadinimai ant sekos srautų elementų dažniausiai rašomi po vartų simbolio (tuomet nevardinamas pastarasis elementas). Šis elementas apibrėžia faktą (dažniau būseną) ir svarbus formuojant veiklos taisykles, todėl jam sudaryti reikalingas veiksmažodis *is, was, are* ir pan. šis elementas taip pat atitinka ir 4.3 poskyryje aprašytas 1 ir 2 taisykles.

Tinkamiausia elemento vardinimo struktūra pateikiama 4.8 lentelėje.

4.8 lentelė. Sekos srautų pavadinimų pavyzdys

Pavadinimas	Taisyklė	
changes (are) received	It is obligatory to inform <u>bank</u> if the <u>changes</u> <i>are</i> <u>received</u>	✓
received_changes	It is obligatory to inform <u>bank</u> if <u>received</u> <u>changes</u>	✗

#### 4.5 Įvykių (angl. *Event*) elementai

Dažniausiai įvykiai sutinkami proceso pradžioje, proceso eigoje po veiklos elemento ar įvykiu grįstais vartais.

1. Pradžios įvykis:

- a. faktas. Pavyzdžiui, „*Day time is 8 AM*“, „*Goods to ship*“. Junginio pradžioje rašomas daiktavardis.

2. Tarpinis įvykis:
- a. pavadinimui nusakyti naudojama neveikiamoji rūšis pavadinimo pradžioje rašant daiktavardį, o po jo skliausteliuose veiksmažodį *is*, *has*, *was* ir pan., kuris vartotojo gali būti skaitomas arba ne (atsižvelgiama į žodžių junginio prasmę). Taip labiau užtikrinamas suformuotos taisyklės korektiškumas bei lengviau nustatomas žodžių tipas veiklos žodyne.

4.9 lentelė. Tarpinių įvykių pavadinimų pavyzdžiai 1

Pavadinimas	
Shop (is) registered	✓
Copy (is) signed	✓
Received report	✗

- b. jei įvykio elementas seka po veiklos elemento, tuomet įvykio pavadinimas turi nusakyti atlikto veiksmo rezultatą.

4.10 lentelė. Tarpinių įvykių pavadinimų pavyzdžiai 2

Veiklos elemento pavadinimas	Įvykio elemento pavadinimas	Taisyklė	
Register shop	Shop (is) registered	It is obligatory to register <u>shop</u> to have <u>shop (is) registered</u> .	✓
Register shop	opened	It is obligatory to register <u>shop</u> to have <u>opened</u> .	✗

- c. laiko įvykiui nurodomas skaičiukas ir žodis, nusakantis, ar tai yra minutės, sekundės, dienos ir pan. Prieš skaičiuką nenurodomas joks papildomas žodis kaip, pavyzdžiui, *wait*, *delay* (tokie žodeliai įterpiami taisyklėje, kaip „*waited*“, „*delayed*“).
- d. Tarpinių įvykių pavadinimui taikytinos ir 4.6 skyrelyje aprašytos rekomendacijos.
3. Pabaigos įvykis, kaip ir tarpinis, naudojamas rezultatui apibrėžti. Pavadinimas vardinamas pagal tokias pat taisykles, kaip ir tarpiniai įvykiai.

#### 4.6 Duomenų (angl. *Data*) elementai

- Kaip ir pradžios įvykis, taip ir duomenų objektas yra faktas/ rezultatas. Šis elementas gali būti aprašomas taip: „*Document has copy*“, „*Issue list*“, etc.
- Jei aprašant duomenų objektą neišvengiama veiksmažodžių, tai rekomenduojama naudoti pasyviają veiksmažodžio formą. Tuomet duomenų objekto pavadinimas turi atrodyti taip: „*List (is) issued*“, „*Report (is) checked*“ arba „*Issued\_list*“, „*Checked\_Report*“, etc.

## 4.7 Kiti svarbūs apribojimai

1. Norint apibrėžti daikto savybes, nerekomenduojama naudoti būdvardžių. Pavadinimo struktūra turi būti daiktavardis – veiksmažodis/ raktažodis – daiktavardis (jei naudojami trys žodžiai), daiktavardis – veiksmažodis (konstatuojant faktą ir kai dėl tam tikrų priežasčių neįmanoma panaudoti kombinacijos „daiktavardis – veiksmažodis/ raktažodis – daiktavardis“). Pavyzdžiai pateikiami 4.11 lentelėje.

4.11 lentelė. Būdvardžių naudojimo pavyzdžiai

Pavadinimas		Pavadinimas	
Necessity of document	✔	Document is necessary	✘
Issued_list is received	✔	Issued list is received	✘

Jei naudojami ✘ simboliu pažymėti variantai, tuomet suformuotame žodyne, žodis „*necessary*“ arba „*Issued*“ neįgyja reikšmės arba ji nustatoma neteisingai.

2. Jei terminus (du daiktavardžius) jungia raktažodžiai, kurie gali įgyti kitą reikšmę, bet dėl tam tikrų priežasčių ji negali būti panaudota, taip pat reikia įterpti sujungimo simbolį („\_“). Pavyzdžiui, terminas „*loan of the bank*“ gali įgyti ir reikšmę „*bank gives loan*“. Todėl, norint suformuotame VŽ matyti teisingą reikšmę, reikia naudoti minėtą simbolį. Tada terminas atrodo taip: „*loan of\_the bank*“. Ši taisyklė labiau naudojama formuojant terminus.
3. Simbolis „\_“ taip pat naudojamas veiksmažodžiams ir raktažodžiams, kurie yra vienas nuo kito priklausomi, aprašyti. Pavyzdžiai pateikti 4.12 lentelėje.

4.12 lentelė. „Jungtinių veiksmažodžių“ ir raktažodžių atvaizdavimo pavyzdžiai

Pavadinimas		Pavadinimas	
Sign_on	✔	Sign on	✘
Turn_off	✔	Turn off	✘
Send_back	✔	Send back	✘
Is_the_category_of	✔	Is the category of	✘

VŽ ir žodžių junginiai, apibrėžiantys matematinius veiksmus, sudaromi netiksliai, nes algoritmas labiau pritaikytas veiksmams tarp asmenų, organizacijų, skyrių ir pan., o ne matematiniams veiksmams grafiškai atvaizduoti.



## 5 Automatizuoto veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų diagramos algoritmo kūrimas, testavimas ir rezultatai

Analizuojant kuriamos sistemos funkcionalumą, apsvarstyta vartotojų grupė, kuri labiausiai suinteresuota VŽ ir VT išgavimo iš VPD algoritmo realizavimu. Taigi prieš pateikiant detalią sistemos reikalavimų specifikaciją, analizuojami sistemos vartotojai.

### 5.1. Vartotojų analizė

Vartotojų analizei pasirinkti vertinimo kriterijai, kurie turėjo didžiausią įtaką teisingai išskiriant vartotojų grupes. Taigi siekiant išanalizuoti jų tipus, reikia atkreipti dėmesį į atsakomybes (tikslus).

#### Vartotojų aibė, tipai ir savybės

Šio – veiklos taisyklių ir veiklos žodyno iš VPD – metodo vartotojų aibė nėra didelė, kadangi metodas yra orientuota į įmones. Dėl to pagrindiniais vartotojais laikomi:

- IT analitikas ir
- Įmonės atstovas

Šie vartotojai suinteresuoti metodo realizavimu, kadangi jis leidžia vartotojams peržvelgti sumodeliuotas VP taisykles, t.y. apribojimus bei žodyną ir leidžia juos redaguoti.

Kiti, mažiau įtakos turintys asmenys, yra įmonės darbuotojai, kuriems VŽ reikalingas kaip komunikavimo priemonė. Šie asmenys yra VŽ bei VT galutiniai vartotojai.

**IT analitikas** yra asmuo, kuris specializuojasi VP modeliavimo srityje ir modeliuoja minėtus procesus, atsižvelgdamas į įmonės atstovo norus bei poreikius.

**Įmonės atstovas** – tai asmuo, kuris atsakingas už VP bei jų modelius, jų teisingą pateikimą (t.y. VP diagramos atitikimą organizacijos VP).

#### Vartotojų tikslai ir problemos

Kuriamas metodas yra naudingas tuo, kad juo galima išgauti VT ir VŽ iš VPD, kuris labiausiai aktualus įmonės atstovui. Jo tikslai:

- patikrinti sumodeliuoto produkto teisingumą;
- suprasti išgautą VT ir VŽ sąrašą, pastebėti netikslumus ir gebėti juos pakeisti.

IT analitiko tikslai:

- sumodeliuoti VP diagramą ir pateikti užsakovui;

- naudojantis sukurtu metodu išgauti VT ir VŽ;
- peržiūrėti išgautą informaciją ir esant poreikiui atlikti pakeitimus.

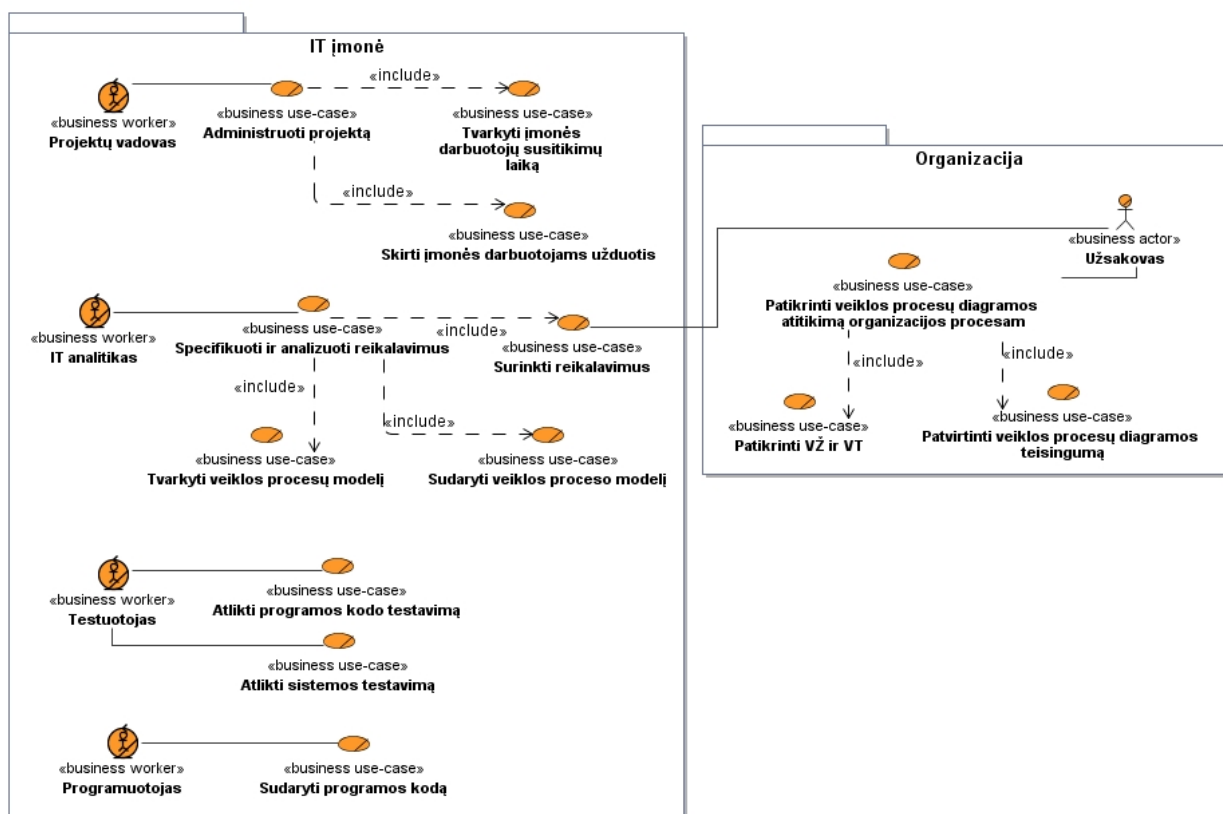
Kadangi be šio metodo daugumą funkcijų rankiniu būdu sudėtinga atlikti (taip yra todėl, kad įmonės atstovas nėra suinteresuotas suprasti modelį/diagramą), todėl dalį išvardintų tikslų galima laikyti problemomis. T.y., nesant šiam metodui, IT analitikas, negali aiškiai žinoti, ko tiksliai nori įmonės vartotojas, kokią vietą reikia keisti ir pan., nes pastarasis vartotojas neturi VT ir VŽ sąrašo.

## **5.2. Reikalavimų automatizuotam veiklos žodyno ir veiklos taisyklių išgavimo iš veiklos procesų diagramos algoritmui specifikacija**

### **5.2.1. Veiklos modeliai**

Kuriamas metodas yra teorinis, tačiau jo įgyvendinimui (t.y. VT ir VŽ iš VPD automatizuotam išgavimui) būtini 5.1 poskyryje minėti vartotojai (t.y., IT analitikas ir įmonės atstovas, kuris žemiau pateiktuose modeliuose dar vadinamas „Užsakovu“).

Taigi veiklos panaudojimo atvejų modelyje išskirti du skyriai, kuriems priklauso minėtieji, su metodo įgyvendinimu susiję, asmenys. IT analitikas priklauso IT įmonei, o užsakovas – kokiai nors organizacijai. Pastarasis yra atsakingas už VPD patikrinimą, o IT analitikas – reikalavimų specifikavimą ir analizavimą. Organizacijos tarpusavyje yra susiję viena, bendrai atliekama, veikla – reikalavimų surinkimu (žiūrėti 5.1 pav.).



5.1 pav. Veiklos panaudojimo atvejų modelis

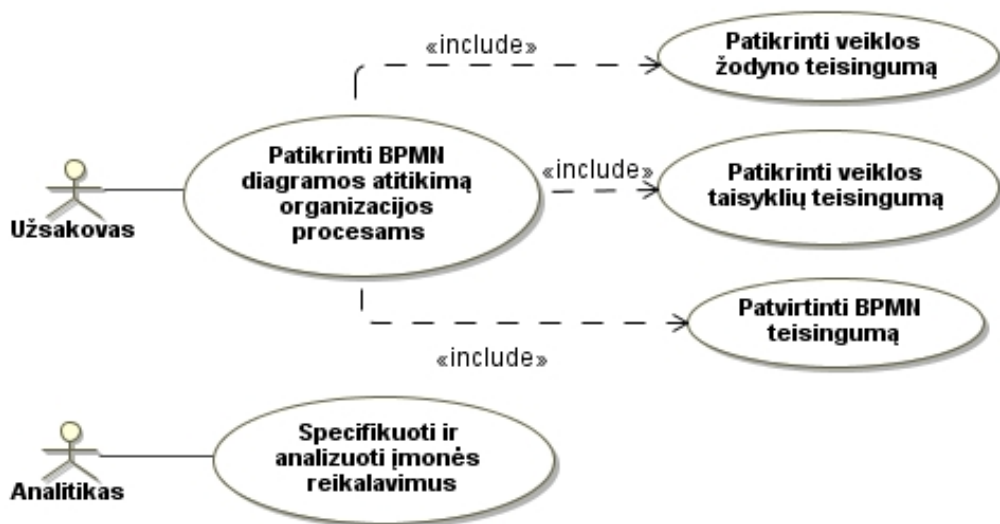
Pastarajame pavaizduoti panaudojimo atvejai realizuoja tam tikrus darbo tikslus. 5.2 paveiksle galima pastebėti, kad norint padidinti veiklos proceso diagramos atitikimą organizacijos VP, reikia patikrinti VŽ ir VT atitikimą. Minėtiems tikslams pasiekti, reikia kad būtų atlikti iškelti uždaviniai. Vieni (pagrindiniai) iš jų – sudaryti vartotojo sąsają (sistemos prototipą) bei programinį kodą, kuriuo būtų galima iš VPD išgauti kuo tikslesnį VŽ bei VT sąrašus.



5.2 pav. Tiriamojo darbo tikslų modelis

### 5.2.2. Reikalavimų modeliai

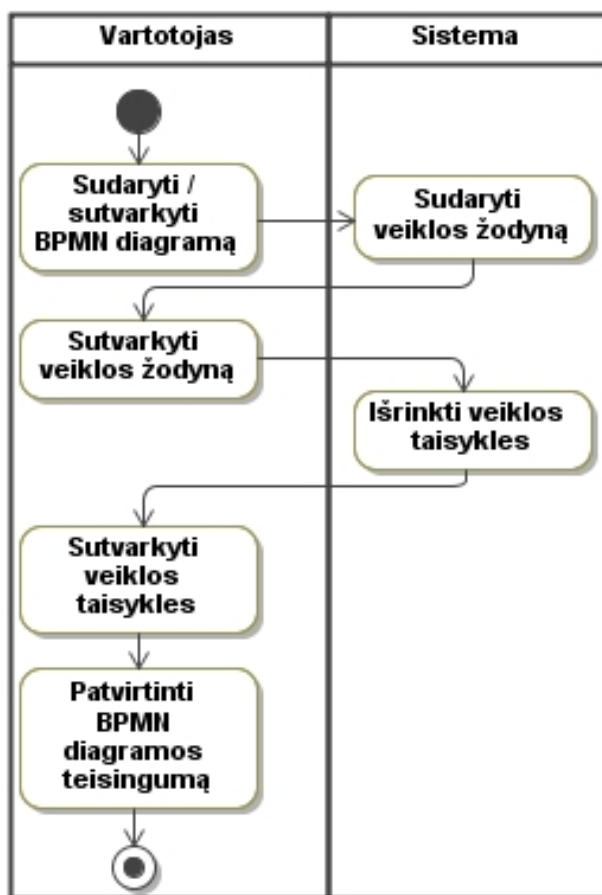
5.3 ir 5.4 paveiksluose pateikiami modeliai, kuriuose matoma, ką kiekvienas vartotojas turi padaryti, kad būtų pasiektas projekto tikslas. 5.3 pav. pateikti užsakovo ir analitiko veiksmai (panaudojimo atvejai), kurie turi būti kompiuterizuojami.



5.3 pav. Kompiuterizuojamų panaudojimo atvejų modelis

Kiekvieno kompiuterizuojamo panaudojimo atvejo specifikacija pateikiama 8.4 priede.

VP diagrama reikalavimų etape (žiūrėti 5.4 pav.) atitinka magistrinio darbo siekiamą sprendimą. Ji apibendrintai parodo, kokius veiksmus vartotojas turės atlikti, kad sistema išgautų VŽ ir VT.

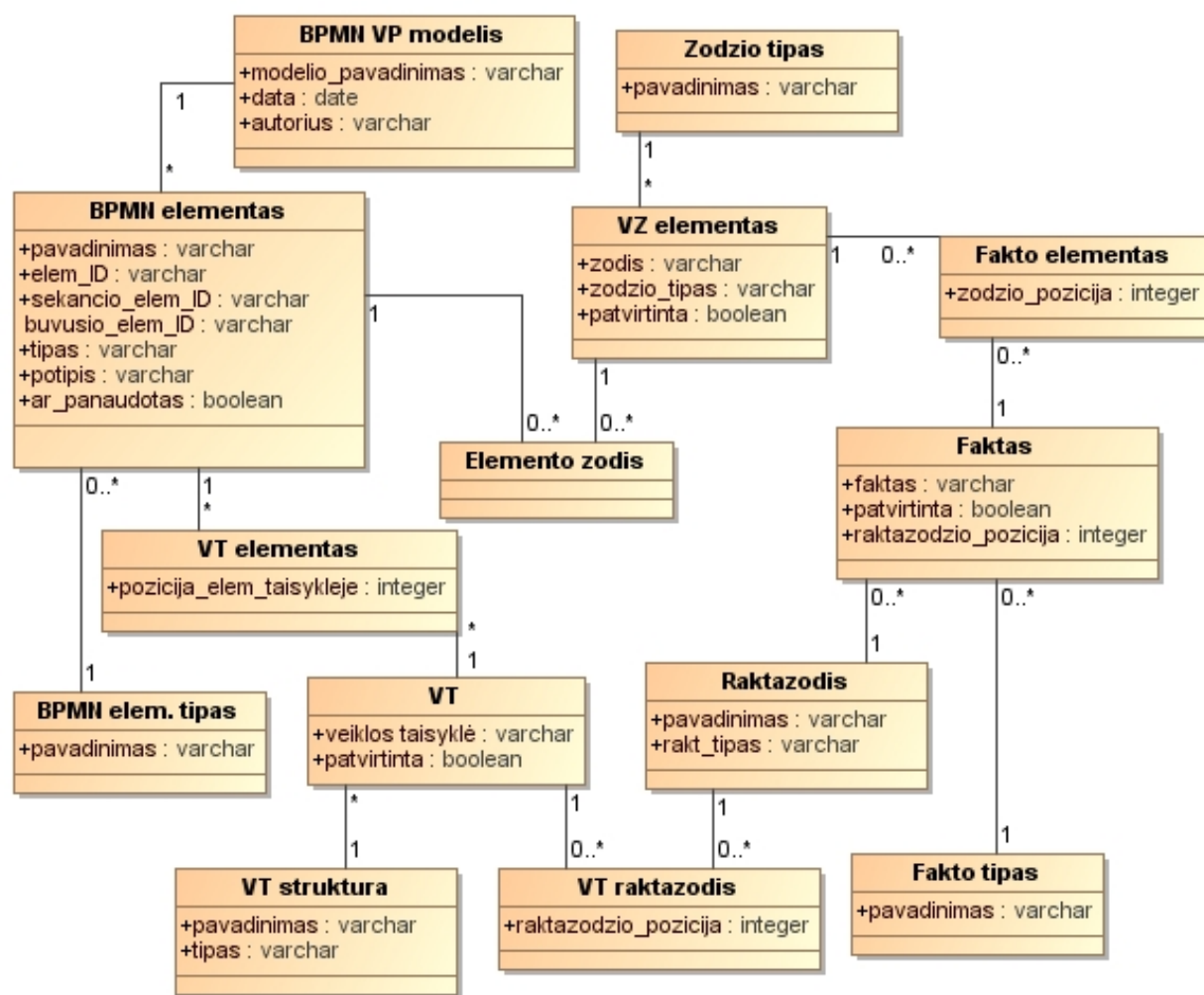


5.4 pav. Veiklos proceso diagrama

### 5.2.3. Dalykinės srities esybių klasių modelis

Dalykinės srities esybių klasių modelis sudarytas remiantis išanalizuota BPMN2 bei SBVR standartų informacija, taip pat informacija, kuri saugoma VPD elementuose, bei sudarytais panaudojimo atvejų ir VP modeliais (žiūrėti 5.3 ir 5.4 pav.).

Jame pavaizduotos pagrindinės klasės ir atributai, kurie reikalingi, kad būtų išsaugoma informacija apie modelį. Be to saugomi nagrinėjamo veiklos proceso elementai bei informacija, naudinga VŽ ir VT sudarymui, VT struktūra bei pati taisyklė, VŽ priklausantys žodžiai bei jų tipai, raktažodžiais, žodžio ar elemento pozicija elemento pavadinime, taisyklėje.



5.5 pav. Duomenų bazės klasių modelis

Kiekvienos 5.5 paveiksle pateiktos klasės paskirtis bendrai aprašyta 5.1 lentelėje, o DB lentelių specifikacijos – 8.5 priede.

5.1 lentelė. DB klasių modelio lentelių specifikacija

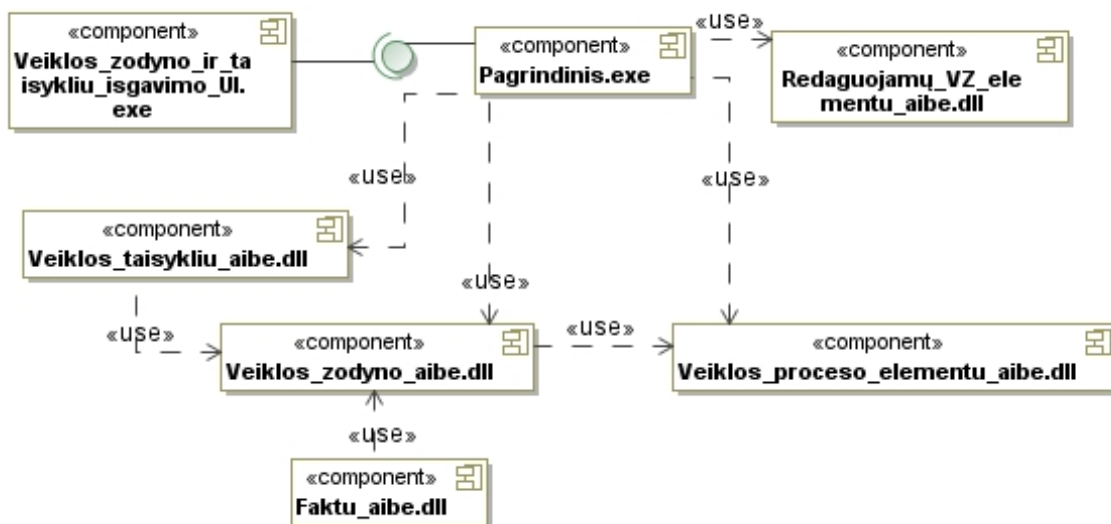
Klasės pavadinimas	Aprašymas
<b>INFORMACIJAI APIE BPMN VP MODELĮ IR JO ELEMENTUS SAUGOTI</b>	
BPMN VP modelis	Saugoma informacija apie modelį. Klasės specifikacija pateikta 8.10 lentelėje.

<b>Klasės pavadinimas</b>	<b>Aprašymas</b>
BPMN elementas	Saugoma pagrindinė informacija apie kiekvieną modelyje rastą elementą. Klasės specifikacija pateikta 8.11 lentelėje.
BPMN elem. tipas	Saugomi visi įmanomi BPMN elementų tipų pavadinimai anglų kalba (Activity, Event, Gateway ir kt.). Klasės specifikacija pateikta 8.12 lentelėje.
<b>INFORMACIJAI APIE VŽ SAUGOTI</b>	
VŽ elementas	Saugoma informacija reikalinga galutiniam VŽ formavimui ir pateikimui vartotojams. Klasės specifikacija pateikta 8.13 lentelėje.
Elemento zodis	Nurodo ryšį tarp BPMN VP elemento ir VŽ elemento.
Zodžio tipas	Saugomi visi įmanomi žodžio tipai, atsižvelgiant į SBVR standarto terminologiją. Klasės specifikacija pateikta 8.14 lentelėje.
Fakto elementas	Nurodo ryšį tarp Fakto ir VŽ elemento.
Faktas	Saugomas suformuotų faktų sąrašas, patvirtinimo požymis bei raktažodžio, esančio fakte, pozicija. Klasės specifikacija pateikta 8.15 lentelėje.
Fakto tipas	Saugomas visų įmanomų iš BPMN VP modelio automatizuotu būdu išgauti fakto tipų sąrašas. Klasės specifikacija pateikta 8.16 lentelėje.
<b>INFORMACIJAI APIE VT SAUGOTI</b>	
VT elementas	Nurodo ryšį tarp BPMN elemento ir VT
VT	Saugoma suformuota VT bei patvirtinimo požymis. Klasės specifikacija pateikta 8.17 lentelėje.
VT struktūra	Saugomi įmanomi VT tipai ir jų pavadinimai (pagal SBVR standarto terminologiją). Klasės specifikacija pateikta 8.18 lentelėje.
VT raktazodis	Nurodo ryšį tarp VT ir galimų raktažodžių.
<b>BENDRAI INFORMACIJAI (APIE VT IR VŽ RAKTOŽODŽIUS) SAUGOTI</b>	
Raktazodis	Saugomi visi įmanomi raktažodžiai, naudojami tiek VŽ tiek VT: <i>has, it is obligatory that</i> ir pan. Klasės specifikacija pateikta 8.19 lentelėje.

Duomenų bazėje, kuri gaunama realizavus duomenų bazės klasių modelį, saugoma informacija pateikiama vartotojui, kuris ją galis valdyti: redaguoti, patvirtinti – saugoti.

#### **5.2.4. Realizacijos modelis. Komponentų diagrama.**

Vartotojo sąsajos, kuri skirta VŽ (pavieniams žodžiais ir jų tipams, faktams) ir VT pateikti, realizacijai sudarytas komponentų modelis, kuris pavaizduotas 5.6 paveiksle. Sukūrus sudarytų VŽ ir VT išgavimo iš VPD algoritmo prototipą, kiekvienas 5.6 paveiksle pavaizduotas komponentas atitinka vartotojo sąsajos langą, kuriame pateikiama komponento pavadinimą atitinkanti informacija. Pavyzdžiui, lange, kurį atitinka komponentas „Veiklos\_zodyno\_aibe“, vaizduojami visi VPD sudarantys žodžiai, jų tipai (veiksmažodis, daiktavardis, vardinis pavadinimas), elementai, kuriems priskiriamas žodis, elemento unikalus numeris. Sąrašas pateikiamas lentelės forma.



5.6 pav. Komponentų modelis

Kiekvieno sistemą sudarančio komponento aprašymas pateikiamas 5.2 lentelėje.

5.2 lentelė. Komponentų specifikacija

Komponento pavadinimas	Aprašymas
Veiklos_zodyno_ir_taisykliu_iscavimo_UI.exe	Sistemos langas
Pagrindinis.dll	Pagrindinis sistemos langas
Redaguojamu_VZ_elementu_aibe.dll	Išskaidytų elementų aibė, su galimybe peržiūrėti žodžių tipus pagal elemento tipą bei pavadinimą
Veiklos_taisykliu_aibe.dll	VT sąrašas
Veiklos_zodyno_aibe.dll	Visų elementų pavadinimų žodžių su tipais pateikimo komponentas
Veiklos_proceso_elementu_aibe.dll	Visų elementų su tipais pateikimo komponentas
Faktu_aibe.dll	Sudarytų faktų pateikimo komponentas.

### 5.3. Algoritmo testavimo etapai

Įvertinus reikalavimus sistemai, sudarytos vartotojo sąsajos bei realizuoto VŽ bei VT iš VPD išgavimo algoritmo įvertinimui numatyti 4 pagrindiniai etapai. Tai yra:

1. **Duomenų nuskaitymas**
2. **Duomenų atvaizdavimas** (elementai ir jų pavadinimai). Pavienu elementų pavadinimų su tipais atvaizdavimas lentelės bei pasirinkimo forma.
3. **VŽ sudarymas**. Elementų pavadinimų skaidymas pavieniais žodžiais, pagal ankstesniame etape sudarytą VŽ išgavimo algoritmą.
4. **VT sudarymas**. VT sudarymas, pagal ankstesniame etape nustatytus VT šablonus ir sudarytą VT sudarymo algoritmą.

Kiekvienas etapas testuojamas realizavus bent vieną funkciją. Tai yra, VŽ sudaryti kuriamos atskiros funkcijos (VŽ iš veiklos elementų, vartų, įvykių, juostų bei baseinų). Importuojame faile ties atskiru tipu keičiami pavadinimai, taip siekiant paspartinti sudaryto VŽ algoritmo testavimo etapą.

Jei programavimo darbai atlikti teisingai, tuomet seka kitas etapas. Sekančiame testavimo etape kartu testuojama ir ankstesnis ar ankstesni etapai.

Taigi realizacijos metu numatyti 4 pagrindiniai (atitinkamai kiekvienai daliai) bei detalūs (atskirai kiekvienam elemento tipui) testavimo žingsniai:

### **1. Duomenų nuskaitymo testavimas**

Šio testavimo metu pasirenkami keli \*.bpmn failai, kurie sukurtoje sistemoje yra nuskaityti. Nuskaityta informacija pateikiama sąrašo forma, todėl pagal tai galima patikrinti, ar visi elementai nuskaityti (informacijos lyginimas ekrane, .bpmn bei VP modelio faile).

### **2. Duomenų atvaizdavimas**

Tikrinama, ar teisingai nuskaityti elementai. Elemento informacija: pavadinimas, tipas, unikalus numeris. Duomenys taip pat pateikiami sistemoje lentelės forma, todėl lengva patikrinti su BPMN failu.

### **3. Veiklos žodyno testavimas**

Testavimo metu tikrinami pavieniai žodžiai bei jų priklausomybė elemento tipui, t.y., kokio elemento pavadinime yra naudojamas tikrinamas žodis. Tai atliekama remiantis sudarytu algoritmu bei sistemoje atvaizduotais duomenimis.

VŽ testavimui redaguojamas importuojamas failas. Šio etapo metu keičiami tikrinamo elemento pavadinimai. Vartotojo sąsajos lange (žiūrėti 5.13 pav.) galima aiškiai matyti išskirstytus žodžius ir nustatyti, ar teisingai priskirtas žodžio tipas. Tokiu būdu tikrinimas žodyno teisingumas.

### **4. Veiklos taisyklių testavimas**

Šio etapo metu tikrinamos VT. Testuojant remiamasi sudarytu algoritmu, VPD.

Paminėtus 4 testavimo etapus apjungia **galutinis sistemos**, kaip vientiso objekto, testavimas. Tai reikalinga tam, kad būtų galima įsitikinti sistemos veikimo teisingumu: ar sistema veikia be trukdžių, ar sudaromas VŽ iš visų nuskaitytų bei atvaizduotų duomenų, kaip tiksliai sudaromas VŽ bei VT.



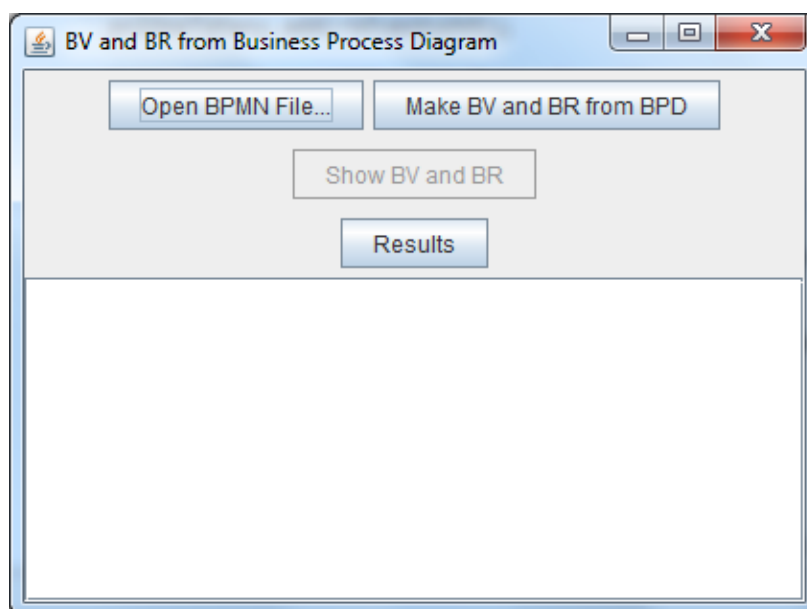
## 5.4. Prototipo testavimas ir rezultatų analizė

5.3 skyriuje apibrėžtiems testavimo etapams įgyvendinti sukurta vartotojo sąsaja, kuria naudojantis išgaunamas VŽ su galimybe jį peržiūrėti. Tam vartotojui pateikiami tokie būtini įvykdyti veiksmai, tokie kaip: „Open BPMN file“, „Make BV and BR from BPD“, „Show BV and BR“, „Results“.

Pagrindinių etapų testavimas pateikiamas 5.4.1, 5.4.2 ir 5.4.3 poskyriuose.

### 5.4.1. Duomenų nuskaitymas

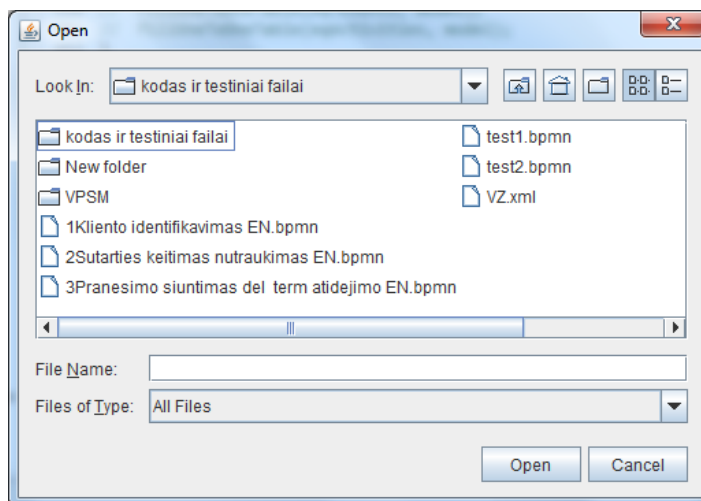
Prototipe realizuotas duomenų nuskaitymas. Duomenys, gauti iš eksportuotų modelių, pateiktų 8.6 - 8.8 prieduose, nuskaitymi naudojančios forma, tokia kaip pateikta 5.7 bei 5.8 paveiksluose.



5.7 pav. Pagrindinis vartotojo sąsajos langas

Formos komponentai ir funkcijos, susiję su duomenų iš \*.bpmn failo nuskaitymu:

- 1) „Open BPMN File..“ atidaro failo pasirinkimo formą (žiūrėti 5.8 pav.):

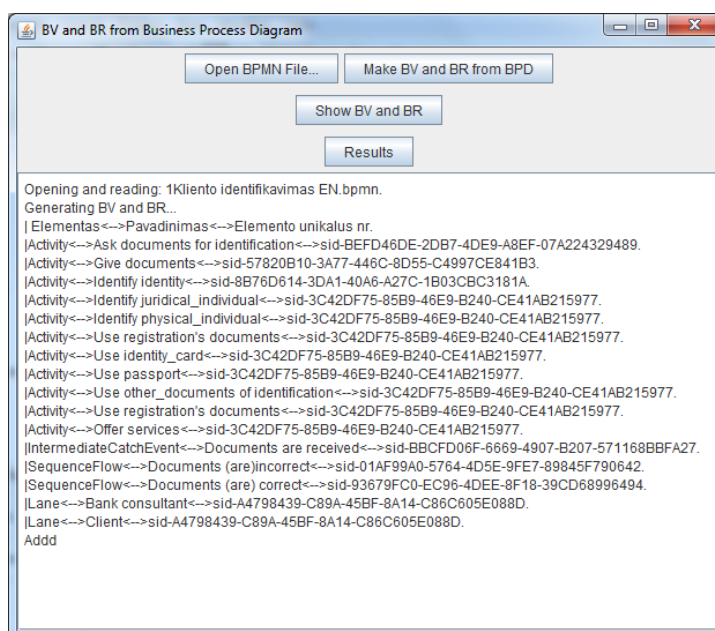


5.8 pav. Failo pasirinkimo forma

### 5.4.2. Duomenų atvaizdavimas

Formos komponentai ir funkcijos, susiję su nuskaitytų duomenų atvaizdavimu pagrindiniame vartotojo sąsajos lange:

- 1) Mygtukas „*Make BV and BR from BPD*“. Naudojamas sudaryti VŽ bei VT iš nuskaitytų duomenų.
- 2) Mygtukas „*Show BV and BR*“. Pateikiami veiklos diagramą sudarančių elementų tipai, naudojamų elementų pavadinimai (pateikiami atsižvelgiant į pasirinktą elemento tipą, t.y. veikla, įvykis, vartai ir kt.).
- 3) Mygtukas „*Results*“. Pateikiamas atliktų veiksmų sąrašas ir rezultatai (nuskaityta informacija). Nuskaitytų duomenų ir atliktų veiksmų sąrašo atvaizdavimo forma pavaizduota 5.9 bei 5.10 paveiksluose.



5.9 pav. Atliktų veiksmų sąrašas ir rezultatai

5.9 paveiksle matom faile esantys duomenys, kurie reikalingi sudarant VŽ bei VT sudarymui. Pateiktame sąrašė matoma: BPMN elemento tipo pavadinimas (pvz.: Activity), elemento vardas (pvz.: „Ask documents for identification“) bei unikalus to elemento numeris (pvz.: sid-BEFD46DE-2DB7-4DE9-A8EF-07A224329489).

5.9 paveiksle pavaizduota informacija lentelės forma pateikiama skiltyje „All Elements“ (žiūrėti 5.10 pav.). Pastarajame lange esančioje kortelėje „Vocabulary“ (žiūrėti 5.11 pav.) vartotojas taip pat gali matyti VŽ, tik papildomai su elemento, iš kurio pavadinimo „paimtas“ žodis, pavadinimu, tipo pavadinimu ir žodžio tipu (veiksmažodis, daiktavardis ar vardinis pavadinimas). Taip vartotojas gali peržiūrėti ir nustatyti, ar teisingai automatizuotu būdu priskirtas žodžio tipas.

Element Type	Name
Activity	Ask documents for identification
Activity	Give documents
Activity	Identify identity
Activity	Identify juridical_individual
Activity	Identify physical_individual
Activity	Use registration's documents
Activity	Use identity_card
Activity	Use passport
Activity	Use other_documents of identification
Activity	Use registration's documents
Activity	Offer services
Intermediate Event	Documents are received
SequenceFlow	Documents (are) incorrect
SequenceFlow	Documents (are) correct
Lane	Bank consultant
Lane	Client

5.10 pav. Nuskaitytų duomenų pateikimo vartotojams langas

Word	Type of Word	Name of Element	Type of Element
identity	Term	Identify identity	Activity
Identify	Verb	Identify juridical_individual	Activity
juridical_individual	Term	Identify juridical_individual	Activity
Identify	Verb	Identify physical_individual	Activity
physical_individual	Term	Identify physical_individual	Activity
Use	Verb	Use registration's documents	Activity
registration's	Term	Use registration's documents	Activity
documents	Term	Use registration's documents	Activity
Use	Verb	Use identity_card	Activity
identity_card	Term	Use identity_card	Activity
Use	Verb	Use passport	Activity
passport	Term	Use passport	Activity
Use	Verb	Use other_documents of identification	Activity
other_documents	Term	Use other_documents of identification	Activity
of	Keyword	Use other_documents of identification	Activity
identification	Term	Use other_documents of identification	Activity
Use	Verb	Use registration's documents	Activity
registration's	Term	Use registration's documents	Activity
documents	Term	Use registration's documents	Activity
Offer	Verb	Offer services	Activity
services	Term	Offer services	Activity
Documents	Term	Documents are received	Intermediate Event
are	Keyword	Documents are received	Intermediate Event
received	Verb	Documents are received	Intermediate Event
Documents	Term	Documents (are) incorrect	Sequence Flow
(are)	Keyword	Documents (are) incorrect	Sequence Flow

Approve Business Vocabulary

5.11 pav. Pavienių žodžių pateikimo langas

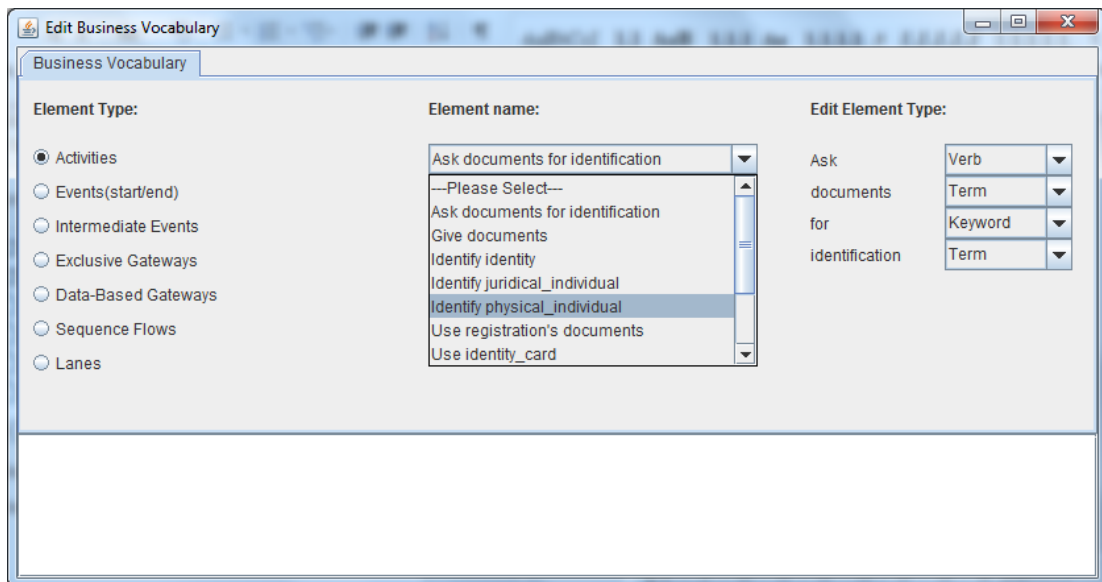
Be VŽ vartotojui pateikiamas ir faktų sąrašas su struktūra (pvz.: Term & Verb), automatinio būdu nustatyto tipu ir elemento tipo, kurio pavadinime buvo rastas faktas, pavadinimu. Toks vartotojo sąsajos langas pateikiamas 5.12 paveiksle.

Fact	Structure of Fact	Fact Type	Element Name	Type of Element
other_documents of identific...	Term & Verb & Term	binary fact type	Use other_documents of identification	Activity
Documents are received	Term & Verb	characteristic fact type	Documents are received	Intermediate Event
Documents (are) incorrect	Term & Verb	characteristic fact type	Documents (are) incorrect	Sequence Flow
Documents (are) correct	Term & Verb	characteristic fact type	Documents (are) correct	Sequence Flow
Bank consultant	Term & Verb & Term	binary fact type	Bank consultant	Lane

Approve Fact types

5.12 pav. Faktų sąrašo pateikimo langas

Aptikus netikslumų sudaryto žodyno sąrašė, vartotojas gali redaguoti tipus (žiūrėti 5.13 pav.). Ši funkcija galima, kai pagrindiniame lange pasirenkama funkcija „Show BV and BR“. Taip pat jis gali peržiūrėti žodžius, pasirinkdamas konkretų elemento tipą ir pavadinimą. Vartotojui pateikiamas išskaidytas pavadinimas su tipais šalia.



5.13 pav. Veiklos žodyno redagavimo langas

Suformuotas veiklos taisyklių sąrašas vartotojui pateikiamas forma, kaip pavaizduota 5.14 paveiksle. Sąrašė vartotojas gali matyti taisyklę, jos struktūrą, t.y. iš kokių elementų išgauta taisyklė, bei veiklos taisyklės tipu.

Rule	Structure of Rule	Type of Rule
It is necessary that Bank Consultant Ask documents for identification to Give documents	Lane & Activity & Activity	Definitional business rules (necessity)
It is obligatory that client give documents to have documents (are) received	Lane & Activity & Intermediate Event	Operational business rules (obligation)
It is necessary to identify juridical_individual or if documents (are) correct	AltActivity & Sequence Flow	Definitional business rules (necessity)
It is obligatory to Use identity_card or use passport or use other_documents of identification to o...	AltActivity & Activity	Operational business rules (obligation)
It is obligatory that bank consultant has documents (are) received to identify identity	Lane & Intermediate Event & Activity	Operational business rules (obligation)

Approve Business Rules

5.14 pav. Veiklos taisyklių pateikimo langas

### 5.4.3. Veiklos žodyno testavimas

Siekiant kuo tiksliau realizuoti sudarytą algoritmą, testavimas buvo vykdomas modeliuojant įvairias VPD (žiūrėti 8.6 – 8.8 prieduose) bei keičiant nuskaitomame faile esančią informaciją. Prieš keičiant faile pavadinimus buvo įvertinamas pavadinimų ilgis, galimi „neteisingi“ elementų pavadinimai (t.y., nesilaikant 4 skyriuje apibrėžtų reikalavimų). Taip siekiama įvertinti sukurto VŽ sudarymo prototipo kokybę. Testavimo analizė ir rezultatai pateikiami 5.4.4 poskyryje, kur galima pastebėti, kad sistemos automatizuotu būdu suformuotame VŽ yra neatitikimų. Pagrindinės to **priežastys** yra:

1. duomenys pateikiami nesilaikant 4 skyriuje pateiktų rekomendacijų;
2. analitikas prie raktažodžių, artikelių aibių nenurodo papildomų žodžių. T.y., gali būti, kad vartotojas nori, jog tokie žodžiai kaip „Empty“, „Enough“ ar net keletas būdvardžių būtų raktažodžio, veiksmazodžio ar kito tipo;

3. dėl žodžių junginių įvairovės nepavyko nustatyti žodžių tipo.

Testavimo metu taip pat siekiama įvertinti ir suformuotų faktų bei jų tipų (faktų tipai: unarinis (charakteristikos), binarinis, dalies-visumos, yra-savybė bei kategorizavimo tikslumą. Faktai išrinkti remiantis tais pačiais pavyzdžiais (žiūrėti 8.6 – 8.8 prieduose), o gauti rezultatai vartotojui pateikiami lentelės forma, kaip pavaizduota 5.12 paveiksle, o jų analizė ir testavimo atvejų analizė taip pat pateikiama 5.4.4 poskyryje. Kaip ir pavieniai žodžiai su tipais, taip ir faktai gali neatitikti siekiamų rezultatų.

**Priežastys**, dėl kurių 5.12 paveiksle gali būti matomi klaidingi rezultatai, yra:

1. neteisingai priskirti žodžių tipai pirmajame testavimo etape;
2. netinkami pavadinimai.

#### **5.4.4. Prototipo vertinimas kiekybiniais ir kokybiniais aspektais**

Kadangi pagrindinis metodo ir prototipo tikslas yra patikrinti sumodeliuotos VPD atitikimą realioms organizacijos procesams, todėl darbo metu išskirti kokybiniai ir kiekybiniai kriterijai, pagal kuriuos galima nustatyti, kiek sukurtas prototipas naudingas vartotojui. Taigi vienas iš svarbiausių sprendimo kokybės kriterijų yra funkcionalumas, t.y. vertinamas:

1. automatizuoto VŽ ir VT formavimo išbaigtumas,
2. VŽ ir VT sąrašo pilnumas bei tikslumas,
3. VŽ ir VT funkcionalumo patogumas vartotojams (t.y., ar vartotojui, neturinčiam techninių žinių, patogiu naudotis prototipu ir ar jam naudingas galutinis rezultatas).

Kokybei įvertinti naudojami teisingi ir neteisingi (tokie, kurie neatitinka 4 skyriuje apibrėžtų reikalavimų modeliuojamai VP diagramai) duomenys. Gauti rezultatai – išsamus VT aprašymas ir VT pilnumas (kadangi vartotojui pateikiamas sąrašas su pavieniais žodžiais, elemento pavadinimais ir tipais, kuriame naudojamas konkretus žodis). Kokybei užtikrinti patvirtinamas VP modelio atitikimas realaus pasaulio procesams. Visa tai leidžia vartotojams atlikti modelio redagavimo ir papildymo darbus, kas pagerina organizacijoje bendravimą ir vykstančius VP.

VP modelio kokybei užtikrinti, vartotojas privalo sistemoje peržiūrėti suformuotą VŽ ir, esant poreikiui, rankiniu būdu pakeisti žodžio tipus, taip užtikrinant, kad analitikas greičiau sutvarkys modeliuojamą VP modelį.

Kokybės vertinimui atliekamas eksperimentas. Šiam tikslui sukurti 3 vienos dalykinės srities testavimo atvejai. Kompiuterizuoti tokie veiklos procesai kaip:

- kliento identifikavimas (žiūrėti 5.4.4.1 skyrelyje),
- sutarties keitimas, nutraukimas (žiūrėti 5.4.4.2 skyrelyje) bei
- pranešimo siuntimas dėl mokėjimo terminų atidėjimo (žiūrėti 5.4.4.3 skyrelyje).

Sumodeliuotose veiklos procesų diagramose, kurių grafinis atvaizdavimas pateikiamas 8.6 – 8.8 prieduose, naudojami pagrindiniai elementai (veikla, pradžios/ pabaigos/ tarpiniai (žinutės ir laiko) įvykiai, vartai).

Eksperimento metu siekiama įvertinti metodo kokybę, atsižvelgiant į:

- elementų tipus bei jų kiekį viename veiklos proceso modelyje;
- jungimo kombinacijas (elementų sekas);
- kiekvieno elemento pavadinimą;
- žodžių kiekį pavadinime;
- papildomų žodžių naudojimą pavadinime;
- 4 skyriuje apibrėžtų taisyklių pritaikymą.

Modeliuojant veiklos procesų diagramas, įvertinti pavadinimų vardinimo apribojimai (žiūrėti 4 skyriuje) bei laikoma, kad vartotojui pakanka anglų kalbos žinių kuo tiksliau aprašyti elementus.

Naudojantis prototipe realizuotomis funkcijomis, minėtuose pavyzdžiuose pateiktų testuojamų atvejų elementai nuskaityti ir atitinkamai 5.3, 5.5 bei 5.7 lentelėse pateikti suskirstyti pagal elementų tipus skliausteliuose pažymint, kiek tokio tipo elementų panaudota veiklos procesų modelyje. Naudojantis prototipe pateiktais duomenimis, lentelės stulpelyje „Atitikimas“ pateikti rezultatai, parodantys, kaip tiksliai automatizuotu būdu VPD elementų pavadinimų pavieniams žodžiams nustatyti tipai. Stulpelyje „Siūlomas pavadinimas“ reikšmė užpildoma, kai formuojant žodyną neteisingai nustatytas bent vienas žodžio tipas, o stulpelyje „Išgautos/Tikėtinos tipų reikšmių sk.“, kurį apima 5 galimos reikšmės, atvaizduojama, kiek pavadinime yra daiktavardžių ir kt. (kiek išgauta prototipo pagalba), bei kiek tikėtasi ruošiant veiklos proceso modelį.

Tikėtini ir automatinio būdu išrinkti faktai taip pat atitinkamai pateikti 5.4, 5.6 bei 5.8 lentelėse. Jose taip pat atvaizduojamas nustatytas tipas, planuotų („Tikėtinas faktų sk.“), gautų su priskirtais („Suformuotų faktų sk. su priskirtais tipais“) ir nepriskirtais („Suformuotų faktų be tipo sk.“) tipais faktų skaičius ir bendras suformuotų faktų skaičius, esantis faktų sąrašė.

#### 5.4.4.1. Veiklos proceso „Kliento identifikavimas“ analizė

Rezultatai 5.3 ir 5.4 lentelėse gauti iš 8.6 priede pateikto veiklos proceso „Kliento identifikavimas“.

5.3 lentelė. VŽ iš VP „Kliento identifikavimas“ analizė

Elemento pavadinimas	Atitikimas	Išgautas/Tikėtinas tipų <sup>4</sup> reikšmių sk.				Siūlomas pavadinimas
		T P	V	R	?	
Veikla (10)						
Ask documents for identification	4/4	2/2(T)	1/1	1/1		
Give documents	2/2	1/1(T)	1/1			
Identify identity	2/2	1/1(T)	1/1			
Identify juridical_individual	2/2	1/1(T)	1/1			
Identify physical_individual	2/2	1/1(T)	1/1			
Use registration's documents	3/3	2/2(T)	1/1			
Use identity_card	2/2	1/1(T)	1/1			
Use passport	2/2	1/1(T)	1/1			
Use other_documents of identification	4/4	2/2(T)	1/1	1/1		
Offer services	2/2	1/1(T)	1/1			
<b>Atitikimas</b>	<b>25/25</b>	<b>13/13(T)</b>	<b>10/10</b>	<b>2/2</b>		
Tarpiniai įvykiai (1)						
Documents (are) received	3/3	1/1(T)	1/1	1/1		
<b>Atitikimas</b>	<b>3/3</b>	<b>1/1(T)</b>	<b>1/1</b>	<b>1/1</b>		
Jungimo srautas (2)						
Documents (are) incorrect	3/3	1/1(T)	1/1	1/1		
Documents (are) correct	3/3	1/1(T)	1/1	1/1		
<b>Atitikimas</b>	<b>6/6</b>	<b>2/2(T)</b>	<b>2/2</b>	<b>2/2</b>		
Juosta (2)						
Bank consultant	2/2	2/2(T)				
Client	1/1	1/1(T)				
<b>Atitikimas</b>	<b>3/3</b>	<b>3/3(T)</b>				
<b>Iš viso:</b>	<b>37/37</b>	<b>18/18(T)</b>	<b>12/13</b>	<b>5/5</b>		

4 T- terminas, V – veiksmazodis, R – raktažodis, P – pavadinimas, ? – „tučia“ reikšmė



5.4 lentelė. Faktų iš VP „Kliento identifikavimas“ analizė

Pavadinimas	Gautas faktas	Tikėtinas faktas	Fakto tipas (FT)
Use other_documents of identification	other_documents of identification	other_documents of identification	Binarinis FT
Documents (are) received	documents (are) received	documents (are) received	Unarinis FT
Documents (are) incorrect	documents (are) incorrect	documents (are) incorrect	Unarinis FT
Documents (are) correct	documents (are) correct	documents (are) correct	Unarinis FT
Bank consultant	bank consultant	bank consultant	Binarinis FT
Statistika	Tikėtinas faktų sk.	Suformuotų faktų sk. su priskirtais tipais	Suformuotų faktų be tipo sk.
<b>Unarinis FT</b>	3	3	
<b>Binarinis FT</b>	2	2	
<b>Dalies-visumos FT</b>			
<b>Yra-savybė FT</b>			
<b>Kategorizavimo FT</b>			
<b>? FT</b>			
<b>IŠ VISO:</b>	<b>5</b>	<b>5</b>	

5.3 lentelėje matoma, kad automatinio būdu suformuotas VŽ atitiko tikėtinus žodžių tipus. Taip yra todėl, kad šiame pavyzdyje naudojama daug veiklos elementų, kuriems apibrėžti (taip kad būtų aišku visiems žmonėms) užtenka 4 žodžių. Taip pat tai paprastas veiklos procesų modelis (tai galima pastebėti dar ir iš to, kad modelyje naudojami 10 veiklos elementų iš 15), kuriam pateikti remtasi 4 skyriuje nurodytais apribojimais. 5.4 lentelėje pateiktų faktų skaičiai sutampa, t.y. suformuota tiek konkrečios rūšies faktų, kiek tikėtasi.

#### 5.4.4.2. Veiklos proceso „Sutarties keitimas, nutraukimas“ analizė

Rezultatai 5.5 bei 5.6 lentelėse gauti iš 8.7 priede pateikto veiklos proceso „Sutarties keitimas, nutraukimas“.

5.5 lentelė. VŽ iš VP „Sutarties keitimas, nutraukimas“ analizė

Elemento pavadinimas	Atitikimas	Išgautas/Tikėtinas tipų <sup>5</sup> reikšmių sk.				Siūlomas pavadinimas
		T P	V	R	?	
Įvykiai (pradžios/pabaigos) (1)						
Changed contract (is) received	3/4	2/1(T)	1/1	1/1	0/1	Changed_contract (is) received
<b>Atitikimas</b>	<b>3/4</b>	<b>2/1(T)</b>	<b>1/1</b>	<b>1/1</b>	<b>0/1</b>	
Veikla (6)						
Change agreement	2/2	1/1(T)	1/1			
Declare new_conditions of agreement	4/4	2/2(T)	1/1	1/1		
Approve new_conditions	2/2	1/1(T)	1/1			
Inform bank	2/2	1/1(T)	1/1			
Sign_on for changes	3/3	1/1(T)	1/1	1/1		
Apply <b>approved</b> conditions	3/3	1/1(T)	1/1		1/1	Apply approved_conditions
<b>Atitikimas</b>	<b>16/16</b>	<b>7/7(T)</b>	<b>6/6</b>	<b>2/2</b>	<b>1/1</b>	
Tarpiniai įvykiai (3)						
approved	1/1		1/1			
rejected	1/1		1/1			
60 days	2/2	1/1(T) 1/1(P)				
<b>Atitikimas</b>	<b>4/4</b>	<b>1/1(T) 1/1(P)</b>	<b>2/2</b>			
Vartai (1)						
Received changes of contract	4/4	2/2(T)		1/1	1/1	Received_changes of contract
<b>Atitikimas</b>	<b>4/4</b>	<b>2/2(T)</b>		<b>1/1</b>	<b>1/1</b>	
Jungimo srautas (2)						
Changes (are) approved	3/3	1/1(T)		1/1	1/1	
Changes (are) rejected	3/3	1/1(T)	1/1	1/1		
<b>Atitikimas</b>	<b>6/6</b>	<b>2/2(T)</b>	<b>2/2</b>	<b>2/2</b>		
Juosta (2)						
Client	1/1	1/1(T)				
Lawyer of bank	3/3	2/2(T)		1/1		
<b>Atitikimas</b>	<b>4/4</b>	<b>3/3(T)</b>		<b>1/1</b>		
<b>Iš viso:</b>	<b>38/68</b>	<b>17/16(T) 1/1(P)</b>	<b>11/11</b>	<b>7/7</b>	<b>2/3</b>	

5 T- terminas, V – veiksmazodis, R – raktažodis, P – pavadinimas, ? – „tučia“ reikšmė

5.6 lentelė. Faktų išVP „Sutarties keitimas, nutraukimas“ analizė

Pavadinimas	Gautas faktas	Tikėtinas faktas	Fakto tipas (FT)
Declare new_conditions of agreement	new_conditions of agreement	new_conditions of agreement	Binarinis FT
Changed contract (is) received	contract (is) received	contract (is) received	Unarinis FT
Received changes of contract	changes of contract	changes of contract	Binarinis FT
Changes (are) rejected	changes (are) rejected	changes (are) rejected	Unarinis FT
Changes (are) approved	changes (are) approved	changes (are) approved	Unarinis FT
Lawyer of bank	lawyer of bank	lawyer of bank	Binarinis FT
<b>Statistika</b>	<b>Tikėtinas faktų sk.</b>	<b>Suformuotų faktų sk. su priskirtais tipais</b>	<b>Suformuotų faktų be tipo sk.</b>
<b>Unarinis FT</b>	3	3	
<b>Binarinis FT</b>	3	3	
<b>Dalies-visumos FT</b>			
<b>Yra-savybė FT</b>			
<b>Kategorizavimo FT</b>			
<b>? FT</b>			
<b>IŠ VISO:</b>	<b>6</b>	<b>6</b>	

Iš 5.5 lentelėje pateiktų rezultatų galima pastebėti, kad tarp išgautų ir tikėtinų reikšmių atsiranda neatitikimų. Taip dažniausiai atsitinka dėl to, kad modeliuojant veiklos procesą analitikas nesilaiko pavadinimų vardinimo taisyklių. Pavyzdžiui, įvykio elemento „Changed contract (is) received“ pavadinimo žodžių tipai (išgauti automatizuotu būdu) atitinka 3 tipus iš 4. Žodis „Changed“ pavadinime yra naudojamas kaip būdvardis, o tokio tipo SBVR terminologijoje nėra. Šiam žodžiui sistema turi priskirti tuščią reikšmę (t.y. ? – 0/1), tačiau priskirtas daiktavardis (t.y. T – 2/1). Kadangi tipo būdvardžiui pagal SBVR terminologiją negalima nustatyti, todėl siūlomas variantas pavadinimui vardinti yra „Changed\_contract (is) received“, kuomet T – 1/1, V – 1/1, R – 1/1, o ? – 0/0. Nors prototipas ir teisingai nustato tuščią tipą, tačiau, tai nėra geriausias variantas veiklos žodynui, nes žodis lieka neapibrėžtas.

Faktų rezultatų lentelėje (žiūrėti 5.6 lentelėje) teisingai pateikti visi numatyti faktai bei jų tipai. Teisingi duomenys gauti dėl to, kad veiklos žodyne teisingai nustatyti raktažodžiai ir jų pozicijos pavadinimuose, o išgautos klaidos neįtakoja faktų formavimo.

### 5.4.4.3. Veiklos proceso „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ analizė

8.8 priede veiklos proceso „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ duomenys ir rezultatai pateikti 5.7 bei 5.8 lentelėse.

5.7 lentelė. VŽ iš VP „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ analizė

Elemento pavadinimas	Atitikimas	Išgautas/Tikėtinas tipų <sup>6</sup> reikšmių sk.				Siūlomas pavadinimas
		T P	V	R	?	
Įvykiai (pradžios/pabaigos) (1)						
Monday	1/1	1/1(P)				
<b>Atitikimas</b>	<b>1/1</b>	<b>1/1(P)</b>				
Veikla (10)						
Send message	2/2	1/1(T)	1/1			
Use fax number	3/3	2/2(T)	1/1			
Send <b>using</b> snail-mail	<b>2/3</b>	<b>2/1(T)</b>	<b>1/2</b>			Send letter arba Post document
Call the courier	3/3	1/1(T)	1/1	1/1		
Inform in <b>other</b> way	<b>3/4</b>	2/1(T)	1/1	1/1	0/1	Inform in other_way
Sign_on copies of document	4/4	2/2(T)	1/1	1/1		
Send_back 1 copy	3/3	1/1(T) 1/1(P)	1/1			
Wait for the answer	4/4	1/1(T)	1/1	2/2		
Check filled_information	2/2	1/1(T)	1/1			
Inform client <b>about</b> mistakes	<b>3/4</b>	3/2(T)	1/1		0/1	Show client's mistakes
<b>Atitikimas</b>	<b>29/32</b>	<b>16/13(T)</b> <b>1/1(P)</b>	<b>10/11</b>	<b>5/5</b>	<b>0/2</b>	
Tarpiniai įvykiai (2)						
Answer (is) received	3/3	1/1(T)	1/1	1/1		
5 days	2/2	1/1(T) 1/1(P)				
<b>Atitikimas</b>	<b>5/5</b>	<b>2/2(T)</b> <b>1/1(P)</b>	<b>1/1</b>	<b>1/1</b>		
Jungimo srutai (2)						
Information (is) correct	3/3	1/1(T)	1/1	1/1		
Information (is) wrong	3/3	1/1(T)	1/1	1/1		
<b>Atitikimas</b>	<b>6/6</b>	<b>2/2(T)</b>	<b>2/2</b>	<b>2/2</b>		
Juosta (2)						
Bank	1/1	1/1(T)				
Bank client	2/2	2/2(T)				
<b>Atitikimas</b>	<b>3/3</b>	<b>2/2(T)</b>				
<b>Iš viso:</b>	<b>44/47</b>	<b>22/19(T)</b> <b>3/3(P)</b>	<b>12/13</b>	<b>6/6</b>	<b>3/3</b>	

<sup>6</sup> T- terminas, V – veiksmazodis, R – raktažodis, P – pavadinimas, ? – „tučia“ reikšmė

5.8 lentelė. Faktų iš VP „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ analizė

Pavadinimas	Gautas faktas	Tikėtinas faktas	Fakto tipas (FT)
Sign_on copies of document	copies of document	copies of document	Binarinis FT
Information (is) correct	information (is) correct	information (is) correct	Unarinis FT
Information (is) wrong	information (is) wrong	information (is) wrong	Unarinis FT
Answer (is) received	answer (is) received	answer (is) received	Unarinis FT
Bank client	bank client	bank client	Binarinis FT
Statistika	Tikėtinas faktų sk.	Suformuotų faktų sk. su priskirtais tipais	Suformuotų faktų be tipo sk.
<b>Unarinis FT</b>	3	3	
<b>Binarinis FT</b>	2	2	
<b>Dalies-visumos FT</b>			
<b>Yra-savybė FT</b>			
<b>Kategorizavimo FT</b>			
<b>? FT</b>			
<b>IŠ VISO:</b>	<b>5</b>	<b>5</b>	

5.7 lentelėje pateikti rezultatai panašūs į esančius 5.5 lentelėje. Nesutapimai atsiranda dėl papildomų žodžių (other, about), kurie nėra apibrėžti prototipe. To galima išvengti keičiant pavadinimą, atsižvelgiant į vardinimo taisyklės, pavadinimo struktūrą ar visą pavadinimą.

5.8 lentelėje pateikti rezultatai atitinka 5.4.4.3 poskyryje pateiktam faktų lentelės aprašymui. Gautų rezultatų analizė, atsižvelgiant į keletą vertinimo kriterijų, pateikiama 5.9 lentelėje. Joje matoma, kokiems duomenims esant veiklos žodynas išgaunamas teisingai, o kada fiksuojamos klaidos.

5.9 lentelė. Apibendrinti eksperimento rezultatai

	VPD	1. „Kliento identifikavimas“	2. „Sutarties keitimas, nutraukimas“	3. „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“
	KRITERIJAI			
<b>VEIKLOS ŽODYNAS</b>	<b>Elementų kiekis VPD:</b>			
	Veiklos Jungimo sratai	10	6	10
	Tarpiniai įvykiai	2	2	2
	Įvykiai (pradžia, pabaiga)	1	3	2
	Juostos	-	1	1
	Vartai	2	2	2
		-	1	-
	<b>Žodžių sk. pavadinime, kuriame aptikta klaidų</b>	-	4 (pradžios įvykis: „Changed contract (is) received“)	3 (veikla: „Send using snail-mail“) 4 (veikla: „Inform client about mistakes“)
	<b>Kitos pastabos/paiškinimai</b>	-	Veiklos „Apply approved conditions“ žodžiui <b>approved</b>	Veiklos „Send using snail-mail“ žodžiams priskirtos netinkamos

	VPD KRITERIJAI	1. „Kliento identifikavimas“	2. „Sutarties keitimas, nutraukimas“	3. „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“
			priskirta tuščia reikšmė; Vartų „Received changes of contract“ žodžiui priskirta tuščia reikšmė	reikšmės: using – daiktavardis, nors turėjo būti veiksmažodis; „Inform in other way“ – other priskirta termino reikšmė, o ne ?.
	<b>Atitikimas</b> <sup>7</sup>			
	Veiklos	25/25	16/16	<b>29/32</b>
	Jungimo srautai	6/6	6/6	6/6
	Tarpiniai įvykiai	3/3	4/4	5/5
	Įvykiai (pradžiai, pabaiga)	-	<b>3/4</b>	1/1
	Juostos	3/3	4/4	3/3
<b>FAKTAI</b>	<b>Faktų, su tipais, kiekis</b>			
	Unarinis	3	3	3
	Binarinis	2	3	2
	Dalies-visumos	-	-	-
	Yra-savybė	-	-	-
	Kategorizavimo ?	-	-	-
	<b>Atitikimas</b>			
	Tikėtina/suformuota	5/5	6/6	5/5

Iš 5.9 lentelės rezultatų matoma, kiek ir kokių elementų sudaro VPD, elementų pavadinimai, kuriuose nesutampa tikėtinos ir gautos reikšmės (antra ir trečia VPD), taip pat pastabos, kurios nusako, kas elemento pavadinime negerai, bei žodžių atitikimas.

Taigi lentelėje matoma, kad teisingiausiai VŽ suformuotas pirmajai veiklai. Joje naudojama mažai elementų tipų: 10 veiklos, 2 jungimo srautų, 1 tarpinio įvykio bei 2 juostų elementai, kuriems priskirtas pavadinimas. Lyginant su likusiais procesais, nepanaudoti dar dviejų elementų tipai, dėl ko pirmąjį procesą būtų galima vadinti vienu iš paprastesnių (tai pabrėžia ir 5.3 lentelėje pateikti rezultatai – duomenys formuoti laikantis vardinimo taisyklių, dažniausiai sutinkamas žodžių skaičius yra 3). Šiame procese automatinio būdu suformuotas žodynas atitinka vartotojo lūkesčius (rezultatai pateikiami „Atitikimas“ skiltyje). Kitų procesų rezultatai skiriasi nuo pirmojo tuo, kad abejuose procesuose rasta klaidų, kurios atsiranda, kai elemente pradama naudoti 4 ir daugiau žodžių, įtraukiami būdvardžiai (**Changed** contract; **approved** conditions) – nesilaikoma apibrėžtų vardinimo taisyklių (pvz.: būdvardis ir

<sup>7</sup> Reikšmės stulpeliuose vaizduoja, kiek iš viso yra atitinkamo elementų žodžių ir kiek iš jų vartotojas patvirtina kaip teisingus (vartotojo patvirtinta/ kiek iš viso žodžių)

daiktavardis jungiami „\_“ simboliu) ar naudojami žodžiai, kurie nėra apibrėžti kaip raktažodžiai ar kt., pavyzdžiui, **about**. Neaiškiems žodžiams priskiriamos tuščios reikšmės, tačiau tai nėra informatyvu vartotojams.

Faktų skiltyje pateikiamas apibendrintas galimų suformuoti faktų tipų sąrašas ir kiek jų iš viso suformuota pagal turimus eksperimento atvejus. Modeliuojant veiklos procesų diagramas dažniausiai pasitaikantys yra unarinis ir binarinis faktų tipai. Teisingas jų formavimas priklauso nuo teisingo VŽ suformavimo, o labiausiai nuo to, ar teisingai nustatytas raktažodis ir jo pozicija pavadinime. Teisingas raktažodžių priskyrimas matomas ir lentelės „Atitikimų“ skiltyje.

### 5.5. Prototipo kūrimo ir testavimo rezultatai

1. Reikalavimų surinkimo etape nustatyti vartotojai bei išanalizuoti jų poreikiai. Tokiu būdu sudaryti vartotojų panaudojimo atvejų bei VP modeliai, apibrėžiantys funkcinis reikalavimus kuriamam prototipui. Atsižvelgus į vartotojo poreikius, sudaryta dalykinės srities klasių modelis bei komponentų diagrama, pagal kurią sukurta vartotojo sąsaja, kuria realizuojamas VŽ ir VT iš VPD išgavimo algoritmas.
2. Kuriamam prototipui apibrėžti išskirti 4 pagrindiniai kūrimo bei testavimo etapai, susiję su pagrindinėmis prototipo funkcijomis: duomenų nuskaitymu iš failo, duomenų atvaizdavimu, VŽ bei VT formavimu.
3. Testavimo metu vertinama realizuoto VŽ ir VT išgavimo iš VPD siūlomo metodo prototipo kokybė, parenkant įvairius (teisingus bei neteisingus – netenkinančius apibrėžtų reikalavimų) duomenis, taip pat sukurti VP modeliai. Kadangi žodynui sudaryti nereikia tikslios elementų sekos, o tik pavadinimų, todėl pradiniam prototipo testavimui buvo redaguojamas nuskaitymas failas, jame keičiant esamus pavadinimus ir pridėdant naujus (failas papildomas tam tikro elemento išskarpomis).
4. Eksperimento rezultatai parodė, kad žodžių tipų teisingumas labiausiai priklauso nuo elemento pavadinimų aibės (didelė elemento pavadinimų įvairovė), žodžių elemento pavadinime skaičiaus (optimalus elementus aprašyti naudojant 4 žodžius), papildomų žodžių įtraukimo, būdvardžių naudojimo/ užrašymo, nustatytų reikalavimų laikymosi bei vartotojo anglų kalbos žinių (laikydamosis reikalavimų anglų kalbos žinių neturintis vartotojas visgi gali neteisingai sudaryti pavadinimą).
5. Faktų tipų priskyrimas labiausiai priklauso nuo raktažodžio pozicijos pavadinime ir raktažodžio reikšmės (jos gali būti „has“, „is“, „is\_property\_of“ ir kt.). Pagal apibrėžtas reikšmes suformuojama fakto struktūra ir nustatomas fakto tipas (unarinis, binarinis, yra kategorija ir kt.). Išanalizavus galimus faktų variantus, nustatyta, kad įvykių, duomenų objektų, vartų ir sekų srautų elementai laikomi kaip faktai.

## 6 Išvados

1. Išanalizavus vartotojų poreikius, žinias, organizacijos struktūrą, nustatyta, kad vienas iš svarbiausių faktorių automatizuoto veiklos žodyno bei veiklos taisyklių išgavimo metodo realizacijai yra galimybė išlaikyti vieningumą, modeliuojant skirtingus veiklos proceso aspektus, bei aiškumą, bendradarbiaujant organizacijoje.
2. Išanalizavus veiklos procesų modeliavimo metodus nustatyta, kad tinkamiausia yra BPMN2 modeliavimo kalba, kadangi veiklos procesų modeliai, priešingai nei naudojant darbų sekų ar UML veiklos diagramų modeliavimo standartus, vartotojui pateikiami ribota natūralia kalba, kai perskaityti modelį nereikia papildomų techninių žinių. BPMN2 notacija tinkamai atvaizduoja realaus pasaulio objektus, kadangi joje egzistuoja plati elementų aibė įvairaus sudėtingumo situacijoms atvaizduoti.
3. Išanalizavus veiklos taisyklių modeliavimo standartus (SBVR, PRR, SRML) nustatyta, kad tinkamiausias yra SBVR - į IT žinių neturinčius vartotojus orientuotas - standartas. Jame saugoma informacija vartotojams pateikiama formalia struktūra bei ribota natūralia kalba. Parinktų standartų kombinacija užtikrina tinkamą veiklos žodyno, veiklos taisyklių ir veiklos procesų sinchronizaciją, eliminuojant galimus neatitikimus.
4. Atlikta literatūros šaltinių analizė parodė, kad šiuo metu dar nėra metodo, kuris automatizuotu būdu suformuotų veiklos žodyną bei veiklos taisykles. Dėl šios priežasties darbo metu papildomai nustatyti apribojimai BPMN2 elementų pavadinimų struktūroms, pagal kuriuos galima automatizuotu būdu suformuoti kuo tikslesnį veiklos žodyno ir veiklos taisyklių sąrašą anglų kalba. Remiantis apribojimais realizuojamas ir pats metodas.
5. Sudarytas metodas veiklos žodyno bei veiklos taisyklių automatizuotam formavimui, kurį sudaro detalios veiklos diagramos skirtingų elementų tipų analizavimui, iš BPMN sumodeliuotų veiklos procesų modelių suformuoti veiklos žodyno ir veiklos taisyklių sąrašą bei, naudojantis SBVR terminologija ir atvaizdavimo savybėmis, vizualia forma ir natūralia kalba pakeikti jį vartotojui. Pateikti rezultatai lengvai suprantami techninių žinių neturintiems asmenims. Tokiu būdu IT vartotojas gali bendrauti su analitiku tuo momentu, kai lygina sumodeliuotus veiklos procesus su realiais veiklos procesais ir taip tobulinti diagramas bei įsigilinti į organizacijos veiklą. Metodas taip pat suteikia galimybę organizacijai tobulėti, apibrėžti ir valdyti darbuotojų atsakomybes, tikslus, viziją.
6. Prototipo kūrimo metu realizuotas veiklos žodyno bei veiklos taisyklių automatinio formavimo algoritmas, kurio pagrindas yra veiklos proceso modelio elementų ir su jais susijusios informacijos nuskaitymas, elementų bei jų pavadinimų ir pavienių žodžių



nagrinėjimas pagal algoritmus, suformuotus atskiriems elementų tipams atsižvelgiant į anglų kalbos taisykles, pavadinimų rašymo rekomendacijas bei elementų seką veiklos procesų modelyje. Vartotojas išgautą informaciją mato grafinėje sąsajoje lentelių pavidalu, tokiu būdu jis gali valdyti turimą informaciją, ją keisti ir užtikrinti jos atitikimą realiems procesams. Atlikus eksperimentinį įvertinimą pagal numatytus kiekybinius ir kokybinius parametrus, nustatyta, kad automatizuotas VŽ ir VT išgavimas pagreitina VŽ formavimą ir sinchronizavimą su esamais veiklos procesais, taip pat VT valdymas užtikrina klaidų eliminavimą iš veiklos procesų modelio, tokiu būdu didinant jo atitikimą realiems procesams.

7. Pagrindinis metodo rezultatas yra suformuoti VŽ elementai: terminai, faktai, veiksmazodžiai, raktažodžiai ir vardiniai pavadinimai bei VT (su elementų jungimo struktūromis ir tipais). Gautas veiklos žodynas gali būti sėkmingai panaudotas tolimesniuose IS kūrimo etapuose ar organizacijos veikloje (pvz.: bendradarbiaujant tarp skirtingų organizacijos skyrių), o veiklos taisyklės - apibrėžti organizacijos struktūrą, susieti organizacijos struktūrinius vienetus. Taip pat apibrėžti organizacijos atstovo pozicijas, leisti vartotojui ir analitikui greičiau valdyti organizacijos procesus (kurti naujas bei modifikuoti esamas veiklos taisykles).

## 7 Literatūra





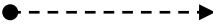





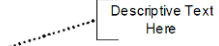
1. Amit Raj, T. V. Prabhakar, Stan Hendryx. Transformation of SBVR Business Design to UML Models. *India Software Engineering Conference. Proceedings of the 1st India software engineering conference*. Hyderabad, India, 2008, p. 29 – 30.
2. Martin Owen and Jog Raj, Popkin Software BPMN and Business Process Management: Introduction to the New Business Process Modeling Standard. Popkin Software, 2003, p. 3, 4, 7-26.
3. Ryan K. L. Ko; Stephen S G Lee; Eng Wah Lee. Business Process Management (BPM) standards: A Survey. *Business Process Management Journal*, Vol. 15, No. 5, 2009, p. 11-15.
4. Roberto Paiano, Anna Lisa Guido, Andrea. Evolution of Business Process Notation. *Designing Complex Web Information Systems: Integrating Evolutionary Process Engineering*. 2009, p. 1-23
5. Michael zurMuehlen, MartaIndulska. Modeling languages for business processes and business rules: A representational analysis. *Information systems*, 2009
6. Using UML Part Two – Behavioral Modeling Diagrams. Sparx Systems, 2007, p. 5-15.
7. Wil M. P. Van der Aalst. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. *Lecture Notes in Computer Science*, 2004.
8. Business Process Model and Notation (BPMN). FTF Beta 1 for Version 2.0. OMG, 2009 – 08, p. 27–41. - [paskutinį kartą žiūrėta 2010-11-14]. Prieiga per internetą: <http://www.omg.org/spec/BPMN/2.0>
9. Hans-Erik Eriksson, Magnus Penker. Business Modeling with UML. [paskutinį kartą žiūrėta 2010-11-14]. Prieiga per internetą: [http://www.imamu.edu.sa/DContent/IT\\_topics/Business%20Modeling%20with%20UML.pdf](http://www.imamu.edu.sa/DContent/IT_topics/Business%20Modeling%20with%20UML.pdf)
10. N. Russell, Wil M.P. van der Aalst, P. Wohed, Arthur H.M. ter Hofstede. On the Suitability of UML 2.0 Activity Diagrams for Business Process Modelling. *APCCM '06 Proceedings of the 3rd Asia-Pacific conference on Conceptual modelling - Volume 53*, 2006, p. 1-10. Prieiga per internetą: <http://www.workflowpatterns.com/documentation/documents/UMLEvalAPCCM.pdf>
11. Semantics of Business Vocabulary and Business Rules (SBVR), v1.0. 2008-01-02, p. 85 – 86. – [paskutinį kartą žiūrėta 2010-11-14]. Prieiga per internetą: <http://www.omg.org/spec/SBVR/1.0/PDF/>
12. F. Rosenberg, S. Dustdar. Design and Implementation of a Service-Oriented Business Rules Broker. *Seventh IEEE International Conference. E-Commerce Technology Workshops*, Austria, 2005, p. 3. Prieiga per internetą: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.67.6889&rep=rep1&type=pdf>
13. Andy CHUN, Hon Wai, Dennis YEUNG, Wai Ming. Rule-based Approach to the Validation of Subway Engineering Work Allocation Plans. *International Conference on Computing, Communications and Control Technologie*. Austin, Texas, USA, 2004-08-(14-17). – [paskutinį kartą žiūrėta 2010-11-14]. Prieiga per internetą: <http://www.cs.cityu.edu.hk/~hwchun/research/PDF/MTRC%20-%20CCCT%202004%20a.pdf>

14. J. W. Schmidt, F. Matthes. Integration of a Rule Engine Component With a Portal Platform. Germany, 2004-01-05, p. 19, – [paskutinį kartą žiūrėta 2010-11-13]. Prieiga per internetą: < <http://www.sts.tu-harburg.de/pw-and-m-theses/2004/chan04.pdf>>
15. OMG Production Rule Representation - Context and Current Status. OMG RFP, 2003-09-03. – [paskutinė kartą žiūrėta 2010-11-16]. Prieiga per internetą: <<http://www.w3.org/2004/12/rules-ws/paper/53/>>
16. Production Rule Representation (PRR), v1.0. OMG, 2009 – 12, p. 1, 6 – 8, 16, 53 – 54. – [paskutinį kartą žiūrėtas 2011-01-25]
17. P. Vincent. OCEB White Paper on Business Rules, Decisions, and PRR, v1.1. TIBCO Software, 2008 – 12, p. 4, 11 – 13. – [paskutinį kartą žiūrėta 2011-01-25]. Prieiga per internetą: <<http://www.omg.org/oceb/OCEB-WhitePaperforPRR1.1.pdf>>
18. D. Miers. The OMG Business Process Related Standards An emerging set of standards that enable Model Driven businesses. BPM Focus, 2007, p. 11. – [paskutinį kartą žiūrėta 2011-01-25]
19. Simple Rule Markup Language (SRML): A General XML Rule Representation for Forward – chaining Rules. In Cover Pages, 2001 – 05 – 17. – [paskutinį kartą žiūrėta 2011 – 01 – 25]. Prieiga per internetą: < <http://xml.coverpages.org/SRML-examples20010517.txt>> SRML
20. S. Haseloff. Context Awareness in Information Logistics. Berlin, 2005 – 04 – 12, p. 132. – [paskutinį kartą žiūrėta 2011 – 01 - 25]
21. Simple Rule Markup Language (SRML). In Cover Pages, 2001 – 05 – 17. – [paskutinį kartą žiūrėta 2011 – 01 – 25]. Prieiga per internetą: < <http://xml.coverpages.org/srml.html>>
22. David Hollingsworth. Workflow Management Coalition The Workflow Reference Model. UK, 1995-01-19, p. 6. – [paskutinį kartą žiūrėtas 2011 – 01 – 25]. Prieiga per internetą: < <http://www.wfmc.org/standards/docs/tc003v11.pdf> >.
23. W M P van der Aalst, S. Jablonski. Dealing with workflow change identification of issues and solutions. International Journal of Computer Systems, Vol. 15, No. 5, 2000-09-05, p. 267 – 268. – [paskutinį kartą žiūrėta 2011 – 02 - 21]. Prieiga per internetą: < <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.4413&rep=rep1&type=pdf>>
24. Business process Improvement. Manage process change using the provision model suite. Proforma Corporation, 2002 – 2003, p. 2-3. Prieiga per internetą: <<http://www.itoamerica.com/index.php?section=pdf&id=522&name=Business+Process+Improvement:+Managing+Process+Change+Using+Metastorm+ProVision>>. (\_\_\_Provision Workbench.php)
25. A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss and A. J. Forster. Business process management with the user requirements notation. Canada, 2009-08-06, p. 273-275. – [paskutinį kartą žiūrėta 2012 – 01 – 25]. Prieiga per internetą: < <http://www.sce.carleton.ca/faculty/weiss/papers/2009/pourshahid-ecr-2009.pdf>>
26. B. Lublinsky, Didier Le Tien. Implementation of business rules and business processes in SOA. SOA, 2007 – 03 -12. – [paskutinį kartą žiūrėtas 2011 – 01 – 27]. Prieiga per internetą: <<http://www.infoq.com/articles/business-rules-processes;jsessionid=15F77F58D05FF037299E3169C7ACC629>>
27. A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss, A.J. Forster. "Toward an Integrated User Requirements Notation Framework and Tool for Business Process Management. 2008 International MCETECH Conference on e-Technologies, 2008, p. 3 – 4.

## 8 PRIEDAI

### 8.1 1 priedas. BPMN elementų aprašymas

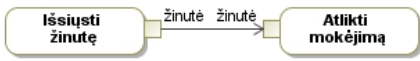

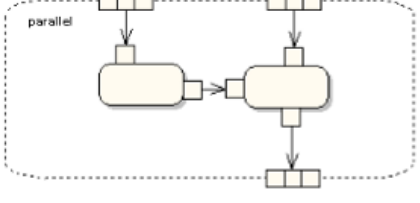
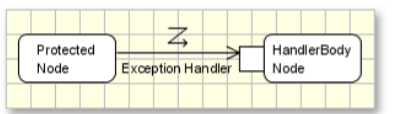

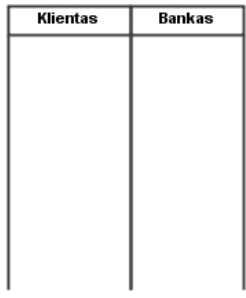
8.1 lentelė. BPMN elementų aprašymas

Elementas	Aprašymas	Žymėjimas
<b>SRAUTŲ OBJEKTAI</b>		
Įvykiai (angl. <i>Events</i> )	Tai yra kažkas, kas nutinka vykstant VP. Šie įvykiai įtakoja proceso eigą ir dažniausiai turi priežastį arba rezultatą. Yra trys įvykių tipai, priklausomai nuo to, kada jie įtakoja veiklą: pradžia (angl. <i>Start</i> ), tarpinis įvykis (angl. <i>Intermediate</i> ) ir pabaiga (angl. <i>End</i> ).	
Veiklos (angl. <i>Actions</i> )	Tai bendras terminas, apibūdinantis darbą, kurį atlieka įmonė. Veikla gali būti neskaidoma (atominė) arba skaidoma. Veiklų tipai, kurie yra proceso modelio dalis, yra: procesas, subprocesas arba užduotis (veikla).	
Vartai (angl. <i>Gateway</i> )	Vartai yra naudojami kontroliuoti sekos srautų susijungimus ir išsiskyrimus.	
<b>JUNGIMO OBJEKTAI</b>		
Sekos srautas (angl. <i>Sequence Flow</i> )	Sekos srautas rodo veiklą, atliekamą proceso metu, seką.	
Žinučių srautas (angl. <i>Message Flow</i> )	Tai rodo informacijos perdavimo kryptis tarp vidinių ir išorinių dalyvių (pvz.: įmonės ir kliento)	
Ryšiai (angl. <i>Association</i> )	Šie elementai naudojami, siekiant sujungti informaciją ir srautų objektus.	
<b>PLAUKIMO TAKELIAI</b>		
Konteineriai (angl. <i>Pools</i> )	Vaizduojami proceso dalyviai (organizacijos).	
Juostos (angl. <i>Lanes</i> )	Juosta yra konteinerio dalis. Jos naudojamos veikloms organizuoti ir klasifikuoti, bei rolėms vaizduoti. Juostos taip pat vaizduoja dalyvius, kurie priklauso, pavyzdžiui, tam tikrai organizacijai.	
<b>DUOMENYS</b>		
Duomenų objektai (angl. <i>Data Objects</i> )	Duomenų objektai yra laikomi artefaktais, nes jie neturi jokio tikslaus efekto sekos ir žinučių srautams, bet jie teikia informaciją apie tai, kokios veiklos turi būti atliktos.	
<b>ARTEFAKTAI</b>		
Grupavimo elementai (dėžė aplink grupę objektų)	Veiklų grupavimas, kuris neveikia sekos srautų. Grupavimas naudojamas dokumentavimui ar analizei. Grupės gali būti naudojamos identifikuoti transakcines veiklas (t.y. veiklų rinkinius), kurios vaizduojamos už konteinerio ribų.	
Komentavimo elementai	Šie elementai BPMN diagramos skaitytojui teikia papildomą informaciją.	

## 8.2 2 priedas. UML elementų aprašymas

8.2 lentelė. UML veiklos diagramos elementai



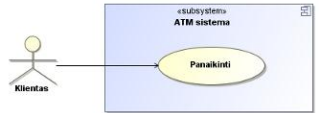
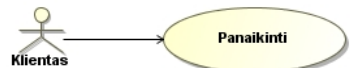

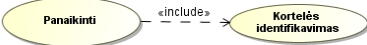
Elementas	Aprašymas	Žymėjimas
<b>VEIKSMAI (angl. Actions)</b>		
Veiklos (angl. <i>Activities</i> )	Aprašo darbo sraute esančių veiklų ar žingsnių atlikimą.	
Veiksmas (angl. <i>Action</i> )	Aprašo vieną veiksmą veikloje.	
Veiklos apribojimai (angl. <i>Action Constraints</i> )	Apribojimas gali būti priskirtas kokiam nors veiksmui.	
<b>KONTROLĖS MAZGAI/TAŠKAI (angl. Control Nodes)</b>		
Pradžios mazgas (angl. <i>Initial Node</i> )	Įtakoja veiklos pradžios laiką ir žymimas dideliu juodu apskritimu.	
Pabaigos mazgas (angl. <i>Activity Final Node</i> )	Pažymi visų kontrolės srautų (angl. <i>Control flows</i> ) pabaigą.	
Srauto pabaigos mazgas (angl. <i>Flow Final</i> )	Žymi vieno kontrolės srauto pabaigą.	
Sprendimo mazgas (angl. <i>Decision Node</i> )	Veikla, sekanti po sprendimo mazgo, turės pradinės (angl. <i>Guard</i> ) sąlygas, kurios leidžia veiksmui vykti (angl. <i>Control to Flow</i> ) toliau, jei ta pradinė sąlyga yra įvykdoma.	
Suliejimo mazgas (angl. <i>Merge node</i> )		
Išsišakojimas / susijungimas (angl. <i>Fork/Join</i> )	Sujungimas (angl. <i>Join</i> ) skiriasi nuo suliejimo (angl. <i>Merge</i> ), nes sujungimas sinchronizuoja dvi įeinančias veiklas ir išveda vieną išeinančią veiklą. Ši veikla negali būti vykdoma, kol visos įeinančios veiklos nebuvo įvykdytos.	
Kontrolės srautas (angl. <i>Control flow</i> )	Parodo kontrolės srautą iš vieno veiksmo į kitą.	
<b>OBJEKTAI IR OBJEKTŲ SRAUTAI (angl. Objects and Object Flows)</b>		
Objektai (angl. <i>Object</i> )	Perduodamas objektas.	
Objekto srautas (angl. <i>Object Flow</i> )	Objekto srautas yra kelias, kuriuo gali būti perduoti objektai arba duomenys. Šis srautas yra vaizduojamas rodykle, nurodančia kryptį, kuria objektas yra perduodamas.	

Elementas	Aprašymas	Žymėjimas
Įvesties ir išvesties jungtys (angl. <i>Input and output pins</i> )	Objekto srautas privalo turėti objektą bent viename jo gale. Sutrumpintam žymėjimui naudojamos įvesties ir išvesties jungtys.	
Duomenų saugykla (angl. <i>Data store</i> )	Duomenų saugykla vaizduojama kaip objektas, tik su specialios paskirties žodžiu <<datastore>>.	
Išplėtimo sritis (angl. <i>Expansion Region</i> )	Struktūrizuota veiklos sritis, kuri vykdoma daug kartų. Įvesties ir išvesties išplėtimo mazgai yra pavaizduoti kaip trijų dėžučių grupė, simbolizuojanti daugelio elementų pasirinkimą. Specialios paskirties žodžiai „iterative“ (kartotinis), „parallel“ (lygiagrečius) arba „stream“ (duomenų srautas) yra rodomi kairiajame srities kampe.	
Išimties vedliai (angl. <i>Exception Handlers</i> )	Gali būti modeliuojami veiklos diagramose.	
Pertraukiamos veiklos sritis (angl. <i>Interruptible Activity Region</i> )	Apgaubia veiksmų, kurie gali būti pertraukiami, grupę.	
Skirstymas (angl. <i>Partition</i> )	Veiklų skirstymas yra vaizduojamas kaip horizontalus arba vertikalus plaukimo takelis. Žymėjimo pavyzdyje, skyriai yra naudojami atskirti veiklos veiksmus, atliekamus banko ir kliento.	

### UML panaudojimo atvejų diagramos (angl. *UML Use Case diagram*) elementai [4]

Svarbiausia UML verslo modeliavimo diagrama yra veiklos diagrama. Ji yra plačiai naudojama, todėl taip ji gali parodyti veiklas (vykstančias nuosekliai ir lygiagrečiai), sunaudotas (angl. *Consumed*), panaudotas ar sukurtus objektus, kurie yra atsakingi už veiklas, taip pat ryšius ir priklausomybes tarp kelių veiklų. Visa tai svarbu verslo modeliavime (angl. *In business modelling*). [8]

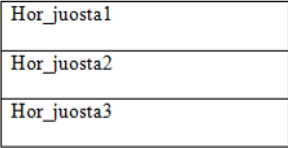
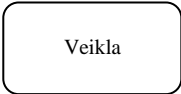
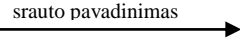
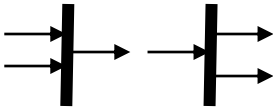
8.3 lentelė. UML panaudojimo atvejų diagramos elementai

Elementas	Aprašymas	Žymėjimas
Aktorius (angl. <i>Actors</i> )	Panaudojimo atvejų diagrama vaizduoja sąveiką tarp sistemos ir išorės esybų. Minėtos esybės vadinamos aktoriais, kurie simbolizuoja vaidmenis (roles), kurios įtraukia žmones, įrangą ar kitas sistemas.	
Panaudojimo atvejis (angl. <i>Use Case</i> )	Šis elementas žymi reikšmingą darbą/veiklą. Jis teikia aukšto lygmens elgsenos vaizdą, matomą kažkam ar kažko, kas yra už sistemos.	
Sistemos riba (angl. <i>System Boundary</i> )	Naudojama dažniausiai pavaizduoti panaudojimo atvejus, esančius sistemoje (sistemos viduje), ir aktorius, esančius už sistemos ribų.	
Jungimo linija (angl. <i>Connecting Line</i> )	Žymėjimas, nurodantis kryptį, t.y., koks atvejis (veikla) yra naudojamas. Pavyzdžiui, „Klientas“ naudoja „Panaikinti“ panaudojimo atvejį (veiklą).	
Veiklos panaudojimo linija (angl. <i>Uses Connector</i> )	Kiekvienos jungimo linijos pabaigoje, pasirinktinai gali būti įvairių reikšmių, kurios nurodo priklausomybes tarp dalyvių ir veiklų. Pavyzdžiui, „Klientas“ vienu metu gali turėti tik vieną kartą „Panaikinti“ veiklai atlikti, tačiau „Bankas“ tuo pačiu metu gali turėti bet kokį skaičių klientų, atliekančių minėtąją veiklą.	
Scenarijai (angl. <i>Scenarios</i> )	Tai yra įvykių srauto, kuris atsiranda vykstant veikloms, formalus aprašymas. Jis apibūdina tam tikrą įvykių seką tarp sistemos ir išorinių veikėjų (angl. <i>Actors</i> ).	
(angl. <i>Including Use Cases</i> )	Viena veikla gali įtakoti kitos veiklos funkcionalumą, kaip jų normalų apdorojimą (angl. <i>As part of their normal processing</i> ). Bendrai, yra manoma (tariama), kad bet kokia įtraukta veikla bus vykdoma kiekvieną kartą, kai pagrindinis kelias yra aktyvuojamas (angl. <i>basic path is run</i> ). Pavyzdžiui, veiklos <Kortelės identifikavimas> vykdymas turi būti paleistas kaip veiklos <Panaikinti> dalis.	

Ši diagrama apibūdina VP, kurie yra aprašomi kaip veiksmų seka, teikianti pastebimą vertę verslo aktoriui (angl. *Provide observable value to a business actor*).

### 8.3 3 priedas. Workflow elementų aprašymas

8.4 lentelė. Workflow diagramos elementai

Pavadinimas	Aprašymas	Žymėjimas
Horizontalios juostos (angl. <i>Horizontal Bands</i> )	Tai darbų sekų juostos (kartais dėl savo išvaizdos vadinamos plaukimo takeliais), kurios atvaizduoja organizacijas arba asmenis, atsakingus už proceso dalis. Darbų sekų juostos gali būti klientai, tiekėjai arba vidaus organizacijos vienetai. Labiau detalizuotame modelyje, juostos gali reikšti ir tam tikras roles (pvz.: darbo funkcijas) ar sistemas	
Veiklos (angl. <i>Activities</i> )	Jos žymimos kaip dėžės. Veikla yra atliekama, siekiant įgyvendinti VP. Jos yra talpinamos į dalyvio (ar organizacijos ir pan.) juostą (plaukimo takelius), kuri atsakinga už vykdomą veiklą. Išsami informacija apie veiklą sąnaudas ir laiką yra įrašoma „dėžėse“ vėlesnei analizei.	
Darbų sekos (angl. <i>Workflows</i> )	Šiomis rodyklėmis yra jungiamos veiklos. Jos atvaizduoja seką, kuria bus atliekamos veiklos ir duomenų perdavimas iš vienos veiklos į kitą.	
Įvykiai (angl. <i>Event</i> )	Tai yra pradžios (angl. <i>Arrival</i> ) arba sąlygos simbolis, kuris sukelia veiklos vykdymą.	
Rezultatas (angl. <i>Deliverable</i> )	Tai darbo produktas, pagamintas vienos veiklos ir perduotas kitai.	
Talpykla (angl. <i>Store</i> )	Elementas modelyje naudojamas kaip mechanizmas, kuriuo siekiama pristatyti ar sulaikyti rezultatus.	
Jungtys (angl. <i>Junctions</i> )	Jos naudojamos siekiant pavaizduoti kur dauguma veiklų susijungia (angl. <i>Join</i> ) arba išsiskiria (angl. <i>Split</i> )	



#### 8.4 4 priedas. Kompiuterizuojamų panaudojimo atvejų specifikacijos

Panaudojimo atvejų, pavaizduotų 3.4 pav., specifikacijos pateiktos 8.5 – 8.9 lentelėse.

8.5 lentelė. PA „Patikrinti BPMN diagramos atitikimą realiems organizacijos procesams“ specifikacija

<b>PA „Patikrinti BPMN diagramos atitikimą organizacijos procesams“</b>		
<b>Tikslas.</b> Patikrinti, ar IT analitikas teisingai nubraižė VPD.		
<b>Aprašymas.</b> Šis PA apima VPD modelio sudarymą bei patvirtinimą, VT ir VŽ išgavimą bei patvirtinimą.		
<b>Prieš sąlyga</b>	Nubraižytas VP modelis Išgautas VŽ ir VT rinkinys	
<b>Aktorius</b>	Užsakovas	
<b>Sužadinimo sąlyga</b>	Vartotojas nori patvirtinti VPD	
<b>Susiję panaudojimo atvejai</b>	<b>Išplečia PA</b>	
	<b>Apima PA</b>	Patikrinti VŽ teisingumą Patikrinti VT teisingumą Patvirtinti BPMN teisingumą
	<b>Specializuoja PA</b>	
<b>Pagrindinis įvykių srautas</b>		
1. Užsakovas nori peržiūrėti VŽ		
2. Užsakovas nori peržiūrėti VT		
<b>Po sąlyga</b>	VPD patvirtinta, kaip atitinkanti arba neatitinkanti, lyginant su organizacijoje vykstančiais procesais.	
<b>Alternatyvūs scenarijai</b>		

8.6 lentelė. PA „Specifikuoti ir analizuoti įmonės reikalavimus“ specifikacija

<b>PA „Specifikuoti ir analizuoti įmonės reikalavimus“</b>	
<b>Tikslas.</b> Nustatyti, kokie reikalavimai, apribojimai egzistuoja tam tikriems organizacijoje vykstantiems procesams. Atsižvelgiant į tai sudaryti teisingą VPD.	
<b>Aprašymas.</b> Šis PA apima reikalavimų surinkimą, taip išsiaiškinant apribojimus, taisykles.	
<b>Prieš sąlyga</b>	
<b>Aktorius</b>	Analitikas
<b>Sužadinimo sąlyga</b>	Vartotojas nori nubraižyti teisingą VPD
<b>Susiję panaudojimo atvejai</b>	<b>Išplečia PA</b>
	<b>Apima PA</b>
	<b>Specializuoja PA</b>
<b>Pagrindinis įvykių srautas</b>	
<b>Po sąlyga</b>	Sudaryta VP diagrama, kuri atitinka surinktus ir išanalizuotus reikalavimus, taisykles ir apribojimus.
<b>Alternatyvūs scenarijai</b>	

8.7 lentelė. PA „Patikrinti veiklos žodyno teisingumą“ specifikacija

<b>PA „Patikrinti veiklos žodyno teisingumą“</b>		
<b>Tikslas.</b> Patikrinti, ar iš VPD išgauti žodžiai atitinka realius objektus.		
<b>Aprašymas.</b> Šis panaudojimo atvejis yra PA „Patikrinti BPMN diagramos atitikimą organizacijos procesams“ specializacija ir paveldi visas jo savybes, bei turi papildomų savybių, kurios aprašytos šioje lentelėje.		
<b>Prieš sąlyga</b>		
<b>Aktorius</b>	Užsakovas	
<b>Sužadinimo sąlyga</b>		
<b>Susiję panaudojimo atvejai</b>	<b>Išplečia PA</b>	
	<b>Apima PA</b>	
	<b>Specializuoja PA</b>	Patikrinti BPMN diagramos atitikimą organizacijos procesams
<b>Pagrindinis įvykių srautas</b>		
1. Užsakovas nori peržiūrėti VŽ		Sistema pateikia pasirinktą informaciją
2. Užsakovas nori priskirti naują žodžio tipą		Sistema išsaugo naują tipą
3. Sistema baigia PA.		
<b>Po sąlyga</b>		
<b>Alternatyvūs scenarijai</b>		

8.8 lentelė. PA „Patikrinti veiklos taisyklių teisingumą“ specifikacija

<b>PA „Patikrinti veiklos taisyklių teisingumą“</b>		
<b>Tikslas.</b> Patikrinti, ar iš VPD išgautos VT atitinka realius apribojimus ir taisykles.		
<b>Aprašymas.</b> Šis panaudojimo atvejis yra PA „Patikrinti BPMN diagramos atitikimą organizacijos procesams“ specializacija ir paveldi visas jo savybes, bei turi papildomų savybių, kurios aprašytos šioje lentelėje.		
<b>Prieš sąlyga</b>		
<b>Aktorius</b>	Užsakovas	
<b>Sužadinimo sąlyga</b>		
<b>Susiję panaudojimo atvejai</b>	<b>Išplečia PA</b>	
	<b>Apima PA</b>	
	<b>Specializuoja PA</b>	Patikrinti BPMN diagramos atitikimą realiams organizacijos procesams
<b>Pagrindinis įvykių srautas</b>		
1. Užsakovas nori peržiūrėti VT		Sistema pateikia pasirinktą informaciją
2. Užsakovas nori priskirti naują taisyklės tipą		Sistema išsaugo naują tipą
3. Sistema baigia PA.		
<b>Alternatyvūs scenarijai</b>		

8.9 lentelė. PA „Patvirtinti BPMN teisingumą“ specifikacija

<b>PA „Patvirtinti BPMN teisingumą“</b>	
<b>Tikslas.</b> Patvirtinti, kad sudarytas VP modelis yra teisingas lyginant su organizacijoje vykstančiais procesais.	
<b>Aprašymas.</b> Šis panaudojimo atvejis yra PA „Patikrinti BPMN diagramos atitikimą realioms organizacijos procesams“ specializacija ir paveldi visas jo savybes, bei turi papildomų savybių, kurios aprašytos šioje lentelėje.	
<b>Prieš sąlyga</b>	
<b>Aktorius</b>	Užsakovas
<b>Sužadinimo sąlyga</b>	Vartotojas nori patvirtinti VP diagramos teisingumą ir atitikimą realioms organizacijos procesams.
<b>Susiję panaudojimo atvejai</b>	<b>Išplečia PA</b>
	<b>Apima PA</b>
	<b>Specializuoja PA</b>
<b>Pagrindinis įvykių srautas</b>	
1. Užsakovas nori patvirtinti redaguotą VŽ	Sistema išsaugo redaguotą informaciją
2. Užsakovas nori patvirtinti redaguotą VT	Sistema išsaugo redaguotas VT
3. Užsakovas nori patvirtinti BPMN diagramą.	
4. Sistema baigia PA.	
<b>Po sąlyga</b>	
<b>Alternatyvūs scenarijai</b>	

## 8.5 5 priedas. Dalykinės srities esybių klasių modelio specifikacija

8.10 lentelė. DB klasių modelio klasės „BPMN VP modelis“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
modelio_pavadinimas	varchar	BPMN VP pavadinimas
data	date	Modelio sukūrimo data
autorius	varchar	BPMN VP modelio kūrėjo informacija (vardas ir pavardė)

8.11 lentelė. DB klasių modelio klasės „BPMN elementas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	BPMN VP pavadinimas
elem_ID	varchar	BPMN VP modelio nagrinėjamo elemento unikalus numeris
sekancio_elem_ID	varchar	BPMN VP modelio sekancio nagrinėjamo elemento unikalus numeris
buvusio_elem_ID	varchar	Anksčiau nagrinėto BPMN elemento unikalus numeris
tipas	varchar	BPMN VP elemento tipas, pavyzdžiui „Event“
potipis	varchar	BPMN VP elemento potipio tipo pavadinimas, pavyzdžiui „Intermediate Timer Event“
ar_panaudotas	boolean	Požymis, naudojamas tikrinant, ar visi elementai nuskaityti.

8.12 lentelė. DB klasių modelio klasės „Elemento tipas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	BPMN VP elementų sąrašas anglų kalba: activity, event, gateway ir kt.

8.13 lentelė. DB klasių modelio klasės „VZ elementas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
zodis	varchar	Pavienis žodis, gautas iš BPMN VP modelio elemento pavadinimo, arba faktas, sudarytas sujungus terminus ir raktažodį.
zodio_tipas (pakeisti modelyje – priskirti tipą)	varchar	Sistemos arba vartotojo nustatytas tipas. Pagal SBVR terminologiją tipai gali būti: terminas, vardas, veiksmazodis, raktažodis, fakto tipas.
patvirtinta	boolean	Požymis, kuris pažymimas, kai vartotojas patikrina VP atitikimą realioms organizacijos procesams ir patvirtina VŽ.

8.14 lentelė. DB klasių modelio klasės „Zodio tipas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	Tipų sąrašas, kuriame gali būti terminas, vardas, veiksmazodis, raktažodis, fakto tipas.

8.15 lentelė. DB klasių modelio klasės „Faktas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
faktas	varchar	Suformuotas faktas iš VP modelio elemento.
patvirtinta	boolean	Požymis, kuris pažymimas, kai vartotojas patikrina VP atitikimą realioms organizacijos procesams ir patvirtina VŽ.

8.16 lentelė. DB klasių modelio klasės „Fakto tipas“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	Saugomas visų įmanomų iš BPMN VP modelio automatizuotu būdu išgauti fakto tipų sąrašas. Paprastai fakto tipai gali būti: <b>unarinis</b> , <b>charakteristika</b> (angl. <i>Characteristic</i> ), <b>binarinis</b> (angl. <i>Binary</i> ), <b>dalies-visumos</b> (angl. <i>Partitive</i> ), <b>yra-savybė</b> (angl. <i>Is-property-of</i> ), <b>kategorizavimo</b> (angl. <i>Categorization</i> ), <b>yra-rolė</b> (angl. <i>Is-role-of</i> ) arba <b>rūšiavimo</b> (angl. <i>Assortment</i> ).
patvirtinta	boolean	Požymis, kuris pažymimas, kai vartotojas patikrina VT atitikimą realiems organizacijos procesams ir patvirtina VT sąrašą.

8.17 lentelė. DB klasių modelio klasės „VT“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
veiklos_taisykle	varchar	Suformuota VT, kuri pateikiama galutiniam vartotojui, kuris gali panaudoti ją tikrinant VP modelio atitikimą realiems procesams arba naudoti atliekant konkrečias veiklas organizacijoje.
patvirtinta	boolean	Požymis, kuris pažymimas, kai vartotojas patikrina VT atitikimą realiems organizacijos procesams ir patvirtina VT sąrašą.

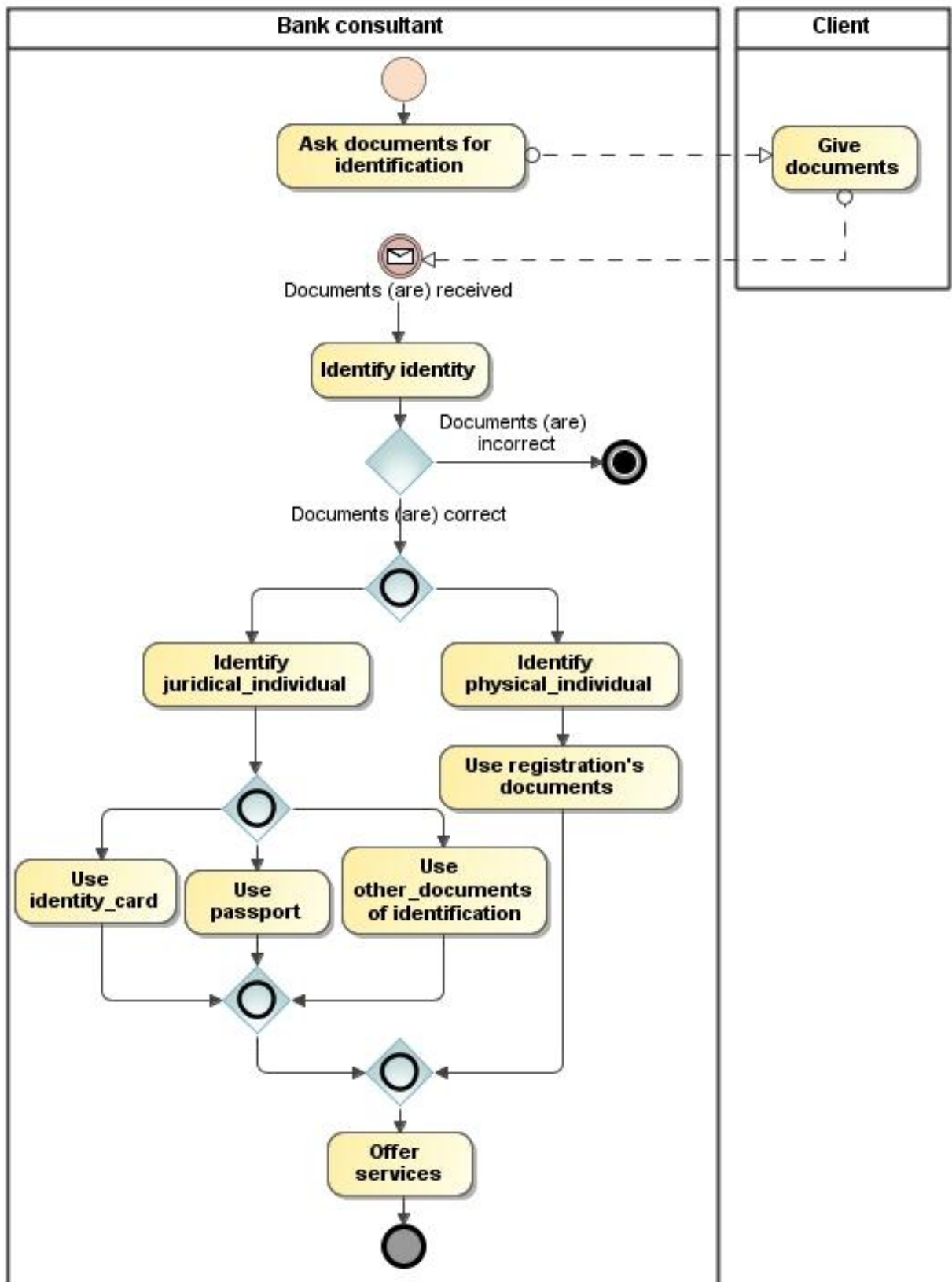
8.18 lentelė. DB klasių modelio klasės „VT struktūra“ specifikacija

Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	VT pavadinimai gali būti: būtinumo (angl. <i>Necessity</i> ), negalėjimo (angl. <i>Impossibility</i> ), ribojimo (angl. <i>Restricted possibility</i> ), įpareigojimo/iniciavimo (angl. <i>Obligation</i> ), draudimo (angl. <i>Prohibition</i> ), ribojimo (angl. <i>Restricted</i> ).
tipas	varchar	VT gali būti dviejų tipų: vykdomoji arba struktūrinė.

8.19 lentelė. DB klasių modelio klasės „Raktazodis“ specifikacija

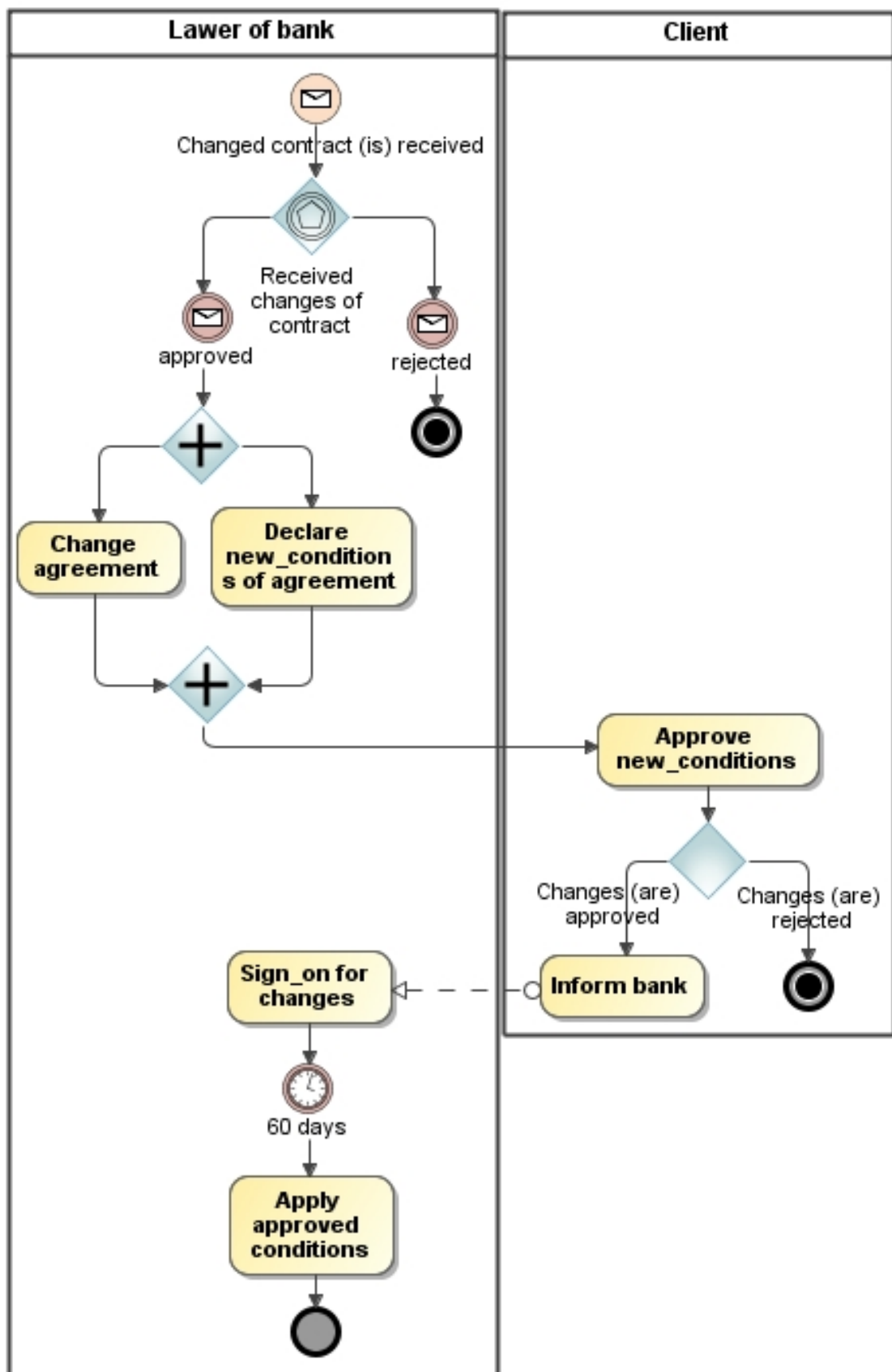
Lauko pavadinimas	Tipas	Saugoma informacija:
pavadinimas	varchar	Žodžiai skirti sudarant faktus (has, is ir kt.), VT (it is obligatory that, must, must not, may ir kt.).

8.6 6 priedas. Veiklos „Kliento identifikavimas“ proceso testavimo atvejis



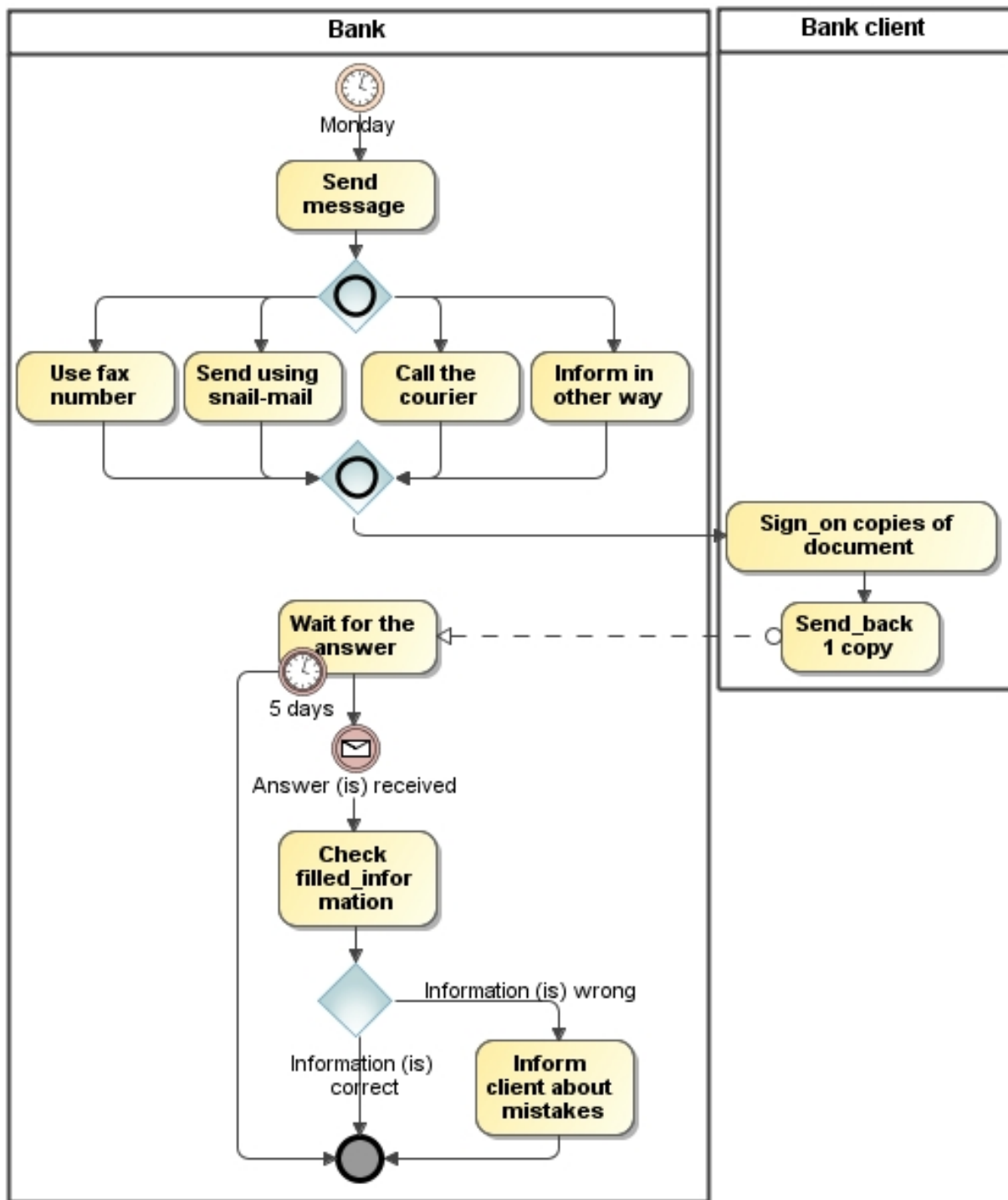
8.1 pav. Grafinis banko veiklos proceso „Kliento identifikavimas“ pateikimas

8.7 7 priedas. Veiklos „Sutarties keitimas, nutraukimas“ proceso testavimo atvejis



8.2 pav. Grafinis banko veiklos proceso „Sutarties keitimas, nutraukimas“ pateikimas

8.8 8 priedas. Veiklos „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ proceso testavimo atvejis



8.3 pav. Grafinis banko veiklos proceso „Pranešimo siuntimas dėl mokėjimo terminų atidėjimo“ pateikimas



## 8.9 9 Priedas. Straipsnis XVII tarpuniversitetinėje magistrantų ir doktorantų mokslinėje konferencijoje „Informacinės technologijos 2012“

### AUTOMATIZUOTAS VEIKLOS ŽODYNO IŠGAVIMO IŠ VEIKLOS PROCESŲ DIAGRAMOS METODAS

Milda Mickevičiūtė, Lina Tutkutė

*Kauno technologijos universitetas, Informacijos sistemų katedra, Studentų 50-309a, Kaunas, Lietuva, milda.mickeviciute@gmail.com, lina.tutkute@ktu.lt*

**Santrauka.** Straipsnyje nagrinėjamas veiklos procesų modeliavimas (BPM) bei identifikuojama veiklos žodyno svarba organizacijoje. Taip pat pateiktas automatiniu būdu iš veiklos procesų modelių (VP) sudaromo VŽ metodas. Jo rezultatas yra suformuotas žodžių sąrašas, leidžiantis palyginti VP diagramos atitikimą realioms organizacijoje vykdomais procesams, efektyviau bendrauti organizacijoje, sėkmingai diegti IS.

**Raktiniai žodžiai:** veiklos procesai, BPMN, veiklos žodynas, SBVR.

#### 1 Įvadas

Šiandienos [1] įmonės privalo įvertinti ne tik produktus ar teikiamas paslaugas, tačiau ir naudojamas informacines sistemas, kurios tampa verslo sudėtine dalimi. Verslas yra tai, kas galiausiai kuria informacinių sistemų reikalavimus, o projektuoti programinę įrangą tinkamai nesuvokiant, kokia bus sistemos paskirtis, yra pavojinga. Siekiant tai suprasti, svarbu sukurti ir suprasti veiklos procesų modelį. Modelis yra supaprastintas tikrovės vaizdas, kuriame pašalinamos nereikšmingos detalės ir sutelkiamas dėmesys į vieną ar daugiau svarbių aspektų. Tinkamai suprojektuoti modeliai palengvina derybas tarp skirtingų veikloje dalyvaujančių aktorių, taip suteikiant jiems galimybę sutarti dėl to, kas svarbiausia, ir toliau dirbti siekiant pagrindinių tikslų. Veiklos modelis pateikia informaciją organizacijos nariams, kuri padeda suprasti veiklos procesus ir juos valdyti.

Kadangi organizacijoje egzistuoja ne vienas veiklos procesas, todėl norint išlaikyti vieningumą, modeliuojant veiklos modelius, ar aiškumą bendraujant organizacijoje, reikalingas veiklos žodynas, apibrėžiantis organizacijoje naudojamus pagrindinius terminus, faktus bei sinonimus. Dėl veiklos procesų pokyčių ir didelės apimties, veiklos žodyno sudarymui turi būti naudojamas automatinis VŽ formavimo metodas, kurio pagalba verslo atstovas sutaupo laiko ir užtikrina jo atitikimą veiklos procesams.

#### 2 Veiklos proceso modeliavimas

Veiklos proceso modeliavimas [8] (angl. *Business Process Modeling*) (toliau BPM) yra veiklos procesų grafiškas atvaizdavimas ir dokumentavimas. Tai yra struktūrizuotas metodas, kuris padeda organizacijos veikloje dalyvaujantiems aktoriams analizuoti procesus ir nustatyti galimus patobulinimus. Proceso, kuris paprastai apibrėžia veiklas, modeliavimas yra atliekamas skirtingų vykdytojų, kurie yra įtraukiami į kuriamą procesą.

Veiklos procesų gyvavimo ciklą sudaro 5 pasikartojantys etapai, užtikrinantys nuolatinį veiklos procesų kokybės gerinimą. BPM gyvavimo ciklas [7] prasideda verslo procesų *tyrinėjimu ir modeliavimu*, po kurių seka veiklos procesų *validavimas* bei *simuliacija* (angl. *Validation & Simulation*), *išdėstymas ir vykdymas* (angl. *Deployment and Execution*), valdymo etapų *stebėjimas* (angl. *Monitoring*) ir *vertinimas* (angl. *Performance*), o baigiasi *tobulinimo* (angl. *Improvement*) etapu, kuriame naudojami veiklos vertinimo rezultatai.

Norint atvaizduoti veiklos procesus ir jų galimus srautus ir perėjimus grafiškai, yra naudojami grafiniai standartai [2]: UML veiklos diagramos (angl. *UML Activity Diagrams*), BPMN (angl. *Business Process Modelling Notations*), EPC (angl. *Event-driven Process Chains*), RADs (angl. *Role-Activity Diagrams*) ir darbų sekų diagramos (angl. *Workflow*). BPMN [3] (angl. *The Business Process Model and Notation*) – veiklos procesų modeliavimo notacija – yra naujas veiklos procesų srautų ir tinklo paslaugų modeliavimo standartas. Tai lengvai verslo vartotojams suprantama notacija, su galimybe grafiškai atvaizduoti veiklos procesus. Taip pat, BPMN apibrėžia veiklos proceso diagramą (angl. *Business Process Diagram (BPD)*), kuri suteikia galimybę aiškiai sumodeliuoti sudėtingus veiklos procesus ir susieti juos su veiklos vykdymo kalbomis. Modeliavimo kalbos elementai yra suskirstyti į 5 kategorijas: 1) Srautų objektai (angl. *Flow Objects*), 2) Duomenys (angl. *Data*), 3) Jungimo objektai (angl. *Connecting Objects*), 4) Konteineriai ir juostos (angl. *Pool, Lane*) ir 5) Artefaktai. UML [4] (angl. *The Unified Modeling Language*) yra specifikavimo kalba, naudojama apibrėžti programinės įrangos sistemas – detalizuoti artefaktus sistemoje, aprašyti ir struktūrizuoti loginę struktūrą. UML siūlo standartinį būdą, skirtą aprašyti sistemos projektus, įtraukiant abstrakčius dalykus, tokius kaip veiklos procesai ir sistemos funkcijos, taip pat realius dalykus, tokius kaip programavimo kalbos teiginiai, duomenų bazės schemas ir daugkartinio naudojimo programinės įrangos komponentai. Šiuo atveju notacijos skirstomos į 1) Elgsenos, 2) Sąveikos bei 3) Struktūrinės diagramas. WfMC [5] (angl. *The Workflow Management Coalition*) darbų seką (angl. *Workflow*) apibrėžia taip: darbų seka – tai veiklos procesų automatizavimas, kurio metu vienas dalyvis perduoda dokumentus, informaciją ar užduotis kitam dalyviui, atsižvelgiant į procesinių taisyklių rinkinį.

Taip siekiama galutinio organizacijos tikslo. Pagrindinis darbų sekų proceso tikslas yra valdyti (vienas į kitą) panašius atvejus. Darbų sekų proceso modeliu aprašoma, kokios veiklos turi būti vykdomos ir kokia tvarka.

MDA [6] architektūra (angl. *The Model-Driven Architecture*) apibrėžia modeliais grindžiamą programinės įrangos kūrimą. MDA specifikacija sudaryta iš trijų sluoksnių: CIM, PIM ir PSM. CIM (nuo skaičiavimų nepriklausome) sluoksnyje atvaizduojami veiklos procesai, aprašyti natūralia kalba. Tokie modeliai (tarp jų ir modeliai, nubraižyti naudojant BPMN) lengvai suprantami techninių žinių neturintiems atstovams, todėl šiuo atveju yra tinkami veiklos žodynui sudaryti.

## 2.1 Veiklos žodyno samprata

Veiklos žodynas [8] (VŽ) yra veiklos esybių, jų egzempliorių ir ryšių tarp jų rinkinys, kurį savo raštuose ar kalboje veiklos kurso metu gali naudoti bet kokia organizacija. SBVR standartas apima ir veiklos žodyną, ir veiklos taisykles. Jis yra orientuotas į verslo atstovus, kurie ir yra pagrindiniai VŽ naudotojai. Pagal MDA architektūrą, jis yra priskiriamas CIM lygmeniui.

VŽ aprašyti naudojami specialios paskirties žodžių ir terminologijos rinkinys:

- **Terminas** (angl. *Term*) – daiktavardis arba žodžių grupė, kuri bendrai gali būti naudojama veiklos esybės nurodyti, pvz. „bankas“ arba „investicinis bankas“.
- **Vardas** (angl. *Name*) – žodis, naudojamas pristatant tam tikro termino (esybės) egzempliorių, pvz. „SEB“ yra esybės **bankas** egzempliorius.
- **Veiksmažodis** (angl. *Verb*) naudojamas fakto tipui – dažniausiai veiksmažodžiui, prielinksniui ar jų kombinacijai pažymėti, pvz. *turi, naudoja, dažo*.
- **Raktažodis** (angl. *Keyword*) naudojamas kitiems žodžiams, iš kurių sudaromas faktas arba taisyklė, pažymėti, pvz. *bent, jei, ar, kai, nors* (angliški: *the, an, a, in, into, etc.*) ir t.t.
- **Fakto tipas** (angl. *Fact Type*). Tai yra sakiniai, kuriais aprašomi ryšiai tarp terminų.

Nustatant ryšį tarp dviejų terminų, naudojamas šablonas „terminas – veiksmažodis – terminas“. Taip yra visiškai aišku, kad bendri ryšiai tarp 3 esybių gali būti lengvai išskirstyti į daugiausiai 3 dvejetainius ryšius (angl. *3 binary entities*). Pavyzdžiui, fakto tipas „klientas turi sąskaitą yra narys“ teigia, kad klientas yra susijęs su sąskaita ir sąskaita yra susijusi su nariu, o asmuo, kuriam priklauso sąskaita, bus nariu. Šis ryšys gali būti išskaidytas į du ryšius, aprašytus dviem fakto tipais: „klientas turi sąskaitą“ ir „klientas yra narys“.

## 3 Veiklos žodyno iš veiklos procesų diagramos išgavimo metodas

Šis metodas skirtas automatiniam veiklos žodyno išgavimui iš veiklos procesų diagramos. Rezultatas gali būti plačiai panaudotas organizacijoje, pvz. kuriant kitus modelius, diegiant informacines sistemas, bendraujant su kitais organizacijos skyriais. Taip būtų išvengiama ne tik modeliavimo klaidų, bet ir palengvinamas bendravimas tarp analitiko ir įmonės atstovo, įvairių organizacijos skyrių. Šiam tikslui pasiekti vartotojas pirmiausia turi sukurti BPMN diagramą, iš kurios automatizuotu būdu bus sudarytas veiklos žodynas. Jį vartotojas galės formuoti ir, esant poreikiui, atnaujinti.

### 3.1 Veiklos žodyno automatizuoto išgavimo iš veiklos procesų diagramos algoritmas

Pagrindiniai veiklos žodyno išgavimo algoritmo žingsniai yra:

1. Sudaryti veiklos procesų diagramą (VPD);
2. Automatizuotu būdu formuoti VŽ (algoritmas);
3. Rankiniu būdu sutvarkyti VŽ.

#### 1 žingsnis. Sudaryti veiklos procesų diagramą (VPD)

Kadangi analitikai braižo ne vieną organizacijos modelį, todėl vartotojui patogiau pasirinkti veiklos procesų diagramą, kurią jis nori naudoti veiklos žodyno formavimui. Sistema automatiškai patikrina, ar diagramoje yra tinkama žodyno išgavimui (ar ji turi elementų, ar tai BPMN diagrama).

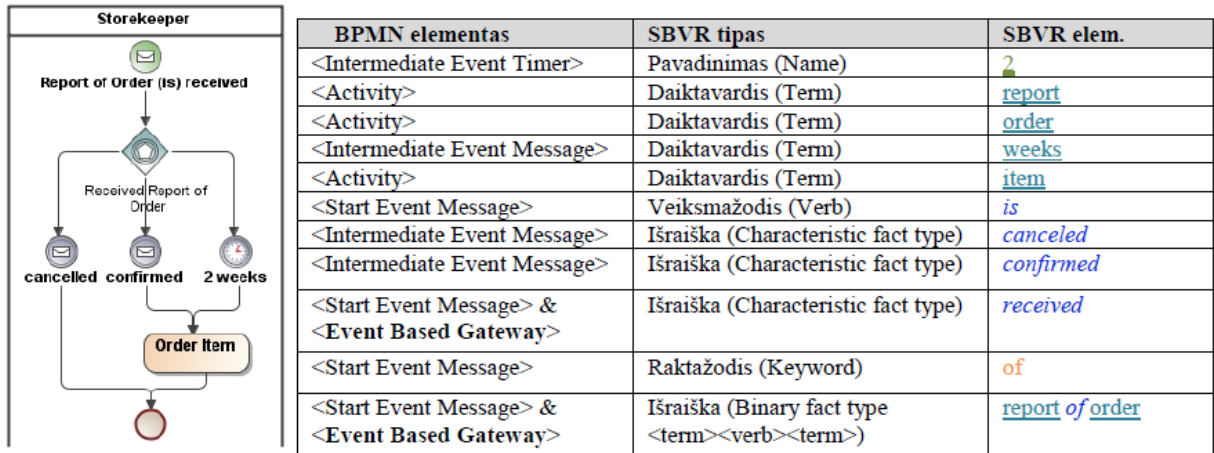
#### 2 žingsnis. Automatizuotu būdu formuoti VŽ (algoritmas)

Algoritmą sudaro 2 etapai: „Atrinkti terminus“ ir „Nustatyti tipus“. Pirmame etape nagrinėjamas kiekvieno elemento pavadinimo atributas, o antrame – tikrinamas kiekvienas elementas atskirai. Pagrindiniai algoritmo žingsniai, žodžių tipų nustatymas:

- analizuojama, iš kokio tipo elemento išrenkami žodžiai. Pagal tai sudaryti skirtingi algoritmai atskiriems elementų tipams;
- analizuojama, koks pirmas elemento pavadinimo žodis. Šio etapo metu analizuojama:
  - koks yra pirmas žodis (skaičius, artikelis ar veiksmažodis „Can“, „Need“ ir pan.);
  - žodžio struktūra;
- nagrinėjama žodžio vieta elemento pavadinime;
- analizuojama, kiek žodžių sudaro elemento pavadinimą.



1 pav. pateikta situacija, kaip VŽ sudarymui yra panaudojami įvairūs BPMN elementai. Šiuo atveju: pradžios („Report of order (is) received“), tarpiniai („canceled“, „confirmed“, „2 weeks“) ir pabaigos įvykiai, įvykiais – grįsti vartai („Received report of order“) ir veikla „Order Item“ (tai elementas, kuris yra dažniausiai naudojamas modeliuojant veiklos procesus).



1 pav. BPMN diagramos pavyzdys ir pagal jį suformuotas veiklos žodynas

Veiklos žodyno sudarymo eiga:

- nustatomas elemento tipas (veiklos, vartai ir jų tipai, baseinas bei juosta, įvykiai ir jų tipai);
- nagrinėjamas elemento pavadinimas. Pavyzdžiui, „Activity“ elemento pavadinimas „Order Item“.
  - Tikrinama, ar rastas žodis yra pirmas. Jei taip, tuomet žodžio tipas – „Verb“ (1 pav.).
  - Jei yra dar „Activity“ elemento žodžių, tikrinamas sekantis.
  - Jei jis nėra pirmas (No. < 1), tada žodžio tipas „Term“, išskyrus tuos atvejus, kai sekantis žodis po pirmojo yra skaičius. Numatoma, kad struktūroje po veiksmažodžio nebūna artikelio ar kito raktažodžio.
  - Sekančiame etape analizuojami likusių žodžių tipai (kaip ir a punkte). Atitinkamai priskiriami tipai (pvz.: visada po raktažodžio „of“ naudojamas daiktavardis, ir pan.).

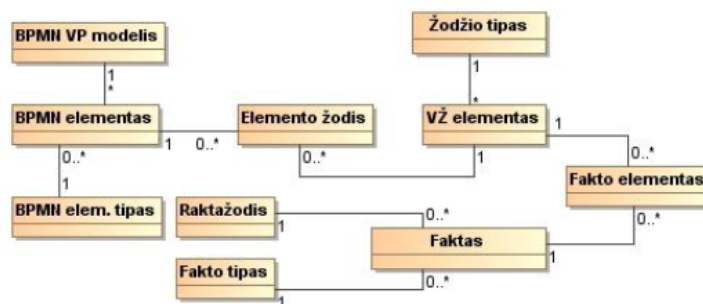
### 3 žingsnis. Rankiniu būdu sutvarkyti VŽ

Automatizuotu būdu sudarant žodyną gali atsirasti klaidų ar neatitikimų. Tokiu atveju vartotojui pateikiamas žodynas redagavimui, kuriame jis gali nustatyti kitokią reikšmę (pvz.: veiksmažodis → daiktavardis). Visa nustatyta informacija išsaugoma duomenų bazėje. Pagrindiniai etapai yra:

- Parenkamas tikrinamas VŽ elementas. Jei vartotojui atrodo, kad neteisingai priskirta reikšmė, tai nustatomas kitas tipas.
- Iš elemento pavadinimo sudaromas/ tikrinamas (sudarytas) faktas. Taip pat kaip ir su žodžių tipais, galima sudaryti arba patvirtinti jau esantį faktą.
- Iš žodyne pateiktų žodžių sudaromas sinonimų sąrašas.

## 3.2 Dalykinės srities esybių klasių modelis

Algoritmo realizacijai sudarytas dalykinės srities esybių klasių modelis (2 pav.), remiantis BPMN2 bei SBVR standartais.



2 pav. Dalykinės srities esybių klasių modelis žodynas

Klasių modelyje pavaizduotos pagrindinės klasės, kurios reikalingos informacijai apie modelį ir VŽ išgavimą išsaugoti. Jame saugoma informacija apie veiklos proceso modelį („BPMN VP modelis“). Informacija apie BPMN elementus yra saugoma „BPMN elementas“ ir „BPMN elem. tipas“ klasėse, t.y. yra pagrindinė

informacija reikalinga VŽ sudarymui. Anksčiau aptartiems punktams reikalinga informacija talpinama likusiose klasėse, užpildant jas požymiu, kad elemento pavadinimo žodis jau yra panaudotas arba patvirtintas kaip teisingai nustatytas, nurodoma žodžio pozicija pavadinime. Pagrindiniai sudaryto VŽ elementai (žodžiai ir jų tipai) saugomi klasėse „VŽ elementas“ ir „Faktas“.

Visa išsaugota informacija pateikiama vartotojui lentelės forma (3 pav.), kurioje matomi žodžiai, jų tipai, elemento tipas bei jo pavadinimas. Prototipas realizuotas Java technologija, nuskaitant \*.xml failą, kuriame saugoma pagrindinė informacija apie elementus (pavadinimas, unikalus numeris, buvusio bei sekančio elemento unikalus numeris ir kiti parametrai). Kiekvienas pavadinimas yra skaidomas į pavienius žodžius ir naudojantis funkcijomis (atsižvelgiant į tai, kokio elemento pavadinimas nagrinėjamas), pagal žodžio poziciją sakinyje, priskiriami tipai. Taip galutinis vartotojas dar kartą pasitikrina, ar sudarytas VŽ yra teisingas.

Word	Type of Word	Name of Element	Type of Element
Identify	Verb	Identify Client's Issues	Activity
Client's	Term	Identify Client's Issues	Activity
Issues	Term	Identify Client's Issues	Activity
Enter	Verb	Enter the order	Activity
the	Keyword	Enter the order	Activity
order	Term	Enter the order	Activity
Sell	Verb	Sell the order	Activity
the	Keyword	Sell the order	Activity
order	Term	Sell the order	Activity

3 pav. Automatinio būdu suformuotas veiklos žodynas

#### 4 Išvados

Nors BPMN2 ir SBVR yra natūralia kalba išreikštos priemonės organizacijoje vykstantiems procesams atvaizduoti, visgi viena svarbiausių tarpusavio bendravimo priemonių tarp skirtingų organizacijos asmenų, analizuojant veiklos procesų modelius, yra veiklos žodynas. Automatinis veiklos žodyno iš veiklos procesų diagramos sudarymo algoritmas nagrinėja pavienius elementus, jų tipus, žodžius ir jų pozicijas pavadinime. Verslo atstovas pats gali sudaryti veiklos žodyną automatinio būdu ir esant poreikiui jį keisti. Pagrindinis rezultatas yra suformuoti VŽ elementai: terminai, faktai („characteristic fact type“, „binary fact type“, „partitive fact type“), veiksmazodžiai, raktažodžiai ir vardiniai pavadinimai. Gautas veiklos žodynas gali būti sėkmingai panaudotas tolimesniuose IS kūrimo etapuose ar organizacijos veikloje.

#### 5 Literatūra

- Hans-Erik Eriksson, Magnus Penker. Business Modeling with UML. [paskutinį kartą žiūrėta 2010-11-14]. Prieiga per internetą: <[http://www.imamu.edu.sa/DContent/IT\\_topics/Business%20Modeling%20with%20UML.pdf](http://www.imamu.edu.sa/DContent/IT_topics/Business%20Modeling%20with%20UML.pdf)>
- Ryan K. L. Ko; Stephen S G Lee; Eng Wah Lee. Business Process Management (BPM) standards: A Survey. *Business Process Management Journal*, Vol. 15, No. 5, 2009, p. 11-15.
- Martin Owen and Jog Raj, Popkin Software BPMN and Business Process Management: Introduction to the New Business Process Modeling Standard. Popkin Software, 2003, p. 3, 4, 7-26.
- Business Process Model and Notation (BPMN). FTF Beta 1 for Version 2.0. OMG, 2009 – 08, p. 27–41. - [paskutinį kartą žiūrėta 2012-03-15]. Prieiga per internetą: <<http://www.omg.org/spec/BPMN/2.0/>>
- Wil M. P. Vad der Aalst. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. *Lecture Notes in Computer Science*, 2004.
- OMG Production Rule Representation - Context and Current Status. OMG RFP, 2003-09-03. – [paskutinį kartą žiūrėta 2010-11-16]. Prieiga per internetą: <<http://www.w3.org/2004/12/rules-ws/paper/53/>>
- A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss, A.J. Forster. "Toward an Integrated User Requirements Notation Framework and Tool for Business Process Management. *2008 International MCETECH Conference on e-Technologies*, 2008, p. 3 – 4.
- Amit Raj, T. V. Prabhakar, Stan Hendryx. Transformation of SBVR Business Design to UML Models. *India Software Engineering Conference. Proceedings of the 1st India software engineering conference*. Hyderabad, India, 2008, p. 29 – 30.

### AUTOMATED EXTRACTION OF BUSINESS VOCABULARY FROM BUSINESS PROCESS DIAGRAMS

This article analyzes business process modeling, its notation and its synchronization's possibilities with business vocabulary. The main issue is the automated extraction of business vocabulary from business process diagrams which is used not only to compare business model to real organization processes, but also is useful to have business vocabulary in organization for more effective collaboration. The created prototype of the method represents words list with its name, element, word type using table for mapping and enables user to manage BV.