

ქ.ი. ყაჭიაშვილი

ბიზნეს-პროცესების მოდელირება

2013

სახელმძღვანელო მომზადებულია OMG-ის (Object Management Group) მიერ გამოქვეყნებული დოკუმენტების Business Process Model and Notation (BPMN), *Version 1.2* და BPMN Modeler for Visio მისევეთ. მასში მოცემულია BPMN-ის ტერმინები, აღნიშვნები, პირობები და მათი გამოყენების და აღქმის წესები ბიზნეს პროცესების აღწერის პროცესის ავტომატიზაციისათვის. სახელმძღვანელოში აგრეთვე მოცემულია ამ მიზნით შექმნილი სპეციალური პროგრამული პროდუქტის Interfacing BPMN Modeler for Visio გამოყენების წესები, შესაძლებლობები და შესრულების მიმდევრობა. წარმოდგენილი მასალის ადვილად გაგების მიზნით, მას დართული აქვს დიდი რაოდენობის გრაფიკული მასალა.

შინაარსი

შესავალი.....	8
თავი 1. ზოგადი მიმოხილვა.....	10
1.1. BPMN-ის შესაძლებლობები (კომპეტენცია)	10
1.1.1. BPMN-ის გამოყენება.....	10
1.1.2. თვალსაზრისის დიაგრამა.....	14
1.1.3. BPMN-ის გაფართოებადობა და ვერდიკალური დომენები.....	15
თავი 2. ბიზნეს პროცესის დიაგრამები.....	16
2.1. ბიზნეს პროცესების დიაგრამების (ბპდ) ძირითადი ელემენტების.....	16
სიმრავლე	
2.2. ბიზნეს პროცესების დიაგრამების (ბპდ) გაფართოებული სიმრავლე.....	21
2.3. ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში.....	33
2.4. ნაკადის ობიექტის შეერთების წესები.....	33
2.4.1. შესრულების ნაკადის წესები.....	34
2.4.2. შეტყობინების ნაკადის წესები.....	35
2.5. ბიზნეს პროცესის დიაგრამის ატრიბუტები.....	36
2.6. პროცესები.....	37
2.6.1. ატრიბუტები.....	38
თავი 3. ბიზნეს პროცესის დიაგრამის გრაფიკული ობიექტები.....	41
3.1. BPMN-ის ელემენტების საერთო ატრიბუტები.....	41
3.2. ნაკადის ობიექტის საერთო ატრიბუტები.....	41
3.3. ხდომილებები.....	42
3.3.1. ხდომილების საერთო ატრიბუტები.....	43
3.3.2. დაწყება.....	43
3.3.2.1. დაწყების ხდომილების გადამრთველები.....	46
3.3.2.2. ატრიბუტები.....	47
3.3.2.3. შესრულების ნაკადის შეერთებები.....	48
3.3.2.4. შეტყობინების ნაკადის შეერთებები.....	49
3.3.3. დამთავრება.....	49
3.3.3.1. დამთავრების ხდომილების შედეგები.....	51
3.3.3.2. ატრიბუტები.....	54
3.3.3.3. შესრულების ნაკადის შეერთებები.....	55
3.3.3.4. შეტყობინების ნაკადის შეერთებები.....	55
3.3.4. შუალედური.....	56
3.3.4.1. შუალედური ხდომილების გადამრთველები.....	57
3.3.4.2. ატრიბუტები.....	63

3.3.4.3. მოქმედების საზღვრის შეერთებები.....	64
3.3.4.4. შესრულების ნაკადის შეერთებები.....	65
3.3.4.5. შეტყობინების ნაკადის შეერთებები.....	66
3.3.5. ხდომილების დეტალები.....	67
3.3.5.1. ხდომილება დეტალის საერთო ატრიბუტები.....	68
3.3.5.2. პირობითი ხდომილების დეტალები.....	68
3.3.5.3. კომპენსაციის ხდომილების დეტალები.....	69
3.3.5.4. შეცდომა ხდომილების დეტალები.....	70
3.3.5.5. რგოლი ხდომილების დეტალები.....	70
3.3.5.6. შეტყობინების ხდომილების დეტალები.....	71
3.3.5.7. სიგნალის ხდომილების დეტალები.....	71
3.3.5.8. ტაიმერის ხდომილების დეტალები.....	72
3.4. მოქმედებები.....	72
3.4.1. საერთო მოქმედების ატრიბუტები.....	72
3.4.1.1. მარყუჟის სტანდარტული ატრიბუტები.....	74
3.4.1.2. მარყუჟის მრავალ-შემთხვევიანი ატრიბუტები.....	75
3.4.2. ქვე-პროცესი.....	77
3.4.2.1. ატრიბუტები.....	80
3.4.2.2. ჩადგმული ქვე-პროცესი.....	81
3.4.2.3. ახლიდან (განმეორებით) გამოყენებული ქვე-პროცესი.....	82
3.4.2.4. მითითების ქვე-პროცესი.....	86
3.4.2.5. ქვე-პროცესის ქვევა როგორც შესრულება.....	86
3.4.2.6. შესრულების ნაკადის შეერთებები.....	88
3.4.2.7. შეტყობინების ნაკადის შეერთებები.....	90
3.4.3. ამოცანა.....	90
3.4.3.1. ატრიბუტები.....	91
3.4.3.2. მომსახურების ამოცანა.....	92
3.4.3.3. მიღების ამოცანა.....	93
3.4.3.4. გაგზავნის ამოცანა.....	95
3.4.3.5. მომხმარებლის ამოცანა.....	96
3.4.3.6. ხელნაწერის ამოცანა.....	97
3.4.3.7. სახელმძღვანელოს ამოცანა.....	97
3.4.3.8. მითითების ამოცანა.....	97
3.4.3.9. შესრულების ნაკადის კავშირები.....	98
3.4.3.10. შეტყობინების ნაკადის შეერთებები.....	99
3.5. გასასვლელები.....	99
3.5.1. გასასვლელის საერთო თავისებურებები.....	101
3.5.1.1. გასასვლელის საერთო ატრიბუტები.....	101
3.5.1.2. გასასვლელის შესრულების ნაკადის საერთო შეერთებები.....	102
3.5.1.3. შეტყობინების ნაკადის შეერთებები.....	103
3.5.1.4. კარებები.....	103
3.5.2. განსაკუთრებული გასასვლელები.....	104
3.5.2.1. მონაცემებზე დაფუძნებული.....	105

3.5.2.2. ატრიბუტები.....	108
3.5.2.3. შესრულების ნაკადის შეერთებები.....	109
3.5.2.4. ხლომილებაზე დაფუძნებული.....	110
3.5.2.5. ატრიბუტები.....	112
3.5.2.6. შესრულების ნაკადის შეერთებები.....	113
3.5.3. შემცველი გასასვლელები.....	114
3.5.3.1. ატრიბუტები.....	117
3.5.3.2. შესრულების ნაკადის შეერთებები.....	117
3.5.4. რთული გასასვლელები.....	118
3.5.4.1. ატრიბუტები.....	120
3.5.4.2. შესრულების ნაკადის შეერთებები.....	120
3.5.5. პარალელური გასასვლელები.....	121
3.5.5.1. ატრიბუტები.....	122
3.5.5.2. შესრულების ნაკადის შეერთებები.....	122
3.6. პროცესის მონაწილენი (გუბეები და ბილიკები).....	123
3.6.1. პროცესის მონაწილის საერთო ატრიბუტები.....	124
3.6.2. გუბე.....	124
3.6.2.1. ატრიბუტები.....	127
3.6.3. ბილიკი (Lane).....	128
3.6.3.1. ატრიბუტები.....	130
3.7. არტიფაქტები.....	130
3.7.1. არტიფაქტების საერთო განსაზღვრებები.....	130
3.7.1.1. არტიფაქტების საერთო ატრიბუტები.....	130
3.7.1.2. არტიფაქტის და შესრულების ნაკადის შეერთებები.....	131
3.7.1.3. არტიფაქტის და შეტყობინების ნაკადის შეერთებები.....	131
3.7.2. მონაცემების ობიექტი.....	131
3.7.2.1. ატრიბუტები.....	133
3.7.3. ტექსტური ანოტაცია.....	134
3.7.3.1. ატრიბუტები.....	135
3.7.4. ჯგუფი.....	135
3.7.4.1. ატრიბუტები.....	136
თავი 4. ბიზნეს პროცესის დიაგრამის შემაერთებელი ობიექტები.....	138
4.1. გრაფიკული შემაერთებელი ობიექტები.....	138
4.1.1. შემაერთებელი ობიექტების საერთო ატრიბუტები.....	138
4.1.2. შესრულების ნაკადი.....	139
4.1.2.1. ატრიბუტები.....	141
4.1.3. შეტყობინების ნაკადი.....	142
4.1.3.1. ატრიბუტები.....	144
4.1.4. გაერთიანება.....	144
4.1.4.1. ატრიბუტები.....	146
4.2. შესრულების ნაკადის მექანიზმები.....	146

4.2.1. ნორმალური ნაკადი.....	147
4.2.1.1. განშტოება ნაკადი.....	151
4.2.1.2. ნაკადის შეერთება.....	154
4.2.1.3. ნაკადის გახლეჩვა.....	156
4.2.1.4. ნაკადის შერწყმა.....	159
4.2.1.5. გადაგრეხვა (მარყუჟის გაკეთება).....	163
4.2.1.6. მოქმედება გადაგრეხვა (მარყუჟის გაკეთება).....	163
4.2.1.7. შესრულების ნაკადის გადაგრეხვა.....	165
4.2.1.8. შესრულების ნაკადის გადახტომა (გვერდის გარეთ შემავრთებლები და ობიექტებზე გადასვლა).....	167
4.2.1.9. ნაკადის გატარება ქვე-პროცესისაკენ და ქვე-პროცესიდან.....	170
4.2.1.10. ნაკადის მართვა პროცესების შიგნით.....	171
4.2.1.11. არალეგალური (უკანონო) მოდელების და მოულოდნელი ქცევის თავიდან აცილება.....	172
4.2.2 გამონაკლისი ნაკადი.....	174
4.2.3. სპეციალური.....	175
4.3. კომპენსაციის გაერთიანება.....	177
ლაბორატორიული: პროცესის მოდელის აგება.....	180
ლაბორატორიული სამუშაო №1.....	180
მოდელის მიმოხილვა.....	180
ახალი ნახაზის გახსნა და სახელის დარქმევა.....	180
ნახაზის შენახვა.....	181
ორგანიზაციული სტრუქტურის აგება.....	181
ორგანიზაციის გვერდის შექმნა.....	182
ორგანიზაციისა და სცენარის გვერდებს შორის მოძრაობა.....	183
ლაბორატორიული სამუშაო №2.....	184
ორგანიზაციული სტრუქტურის შექმნა.....	184
ორგანიზაციის ერთეულების შეერთება.....	184
რესურსების გვერდის გახსნა.....	185
როლის შექმნა.....	185
როლების დამატება.....	186
რესურსების დამატება.....	186
ტელეფონის ნომრის და ი-მეილის დამატება რესურსზე.....	189
ლაბორატორიული სამუშაო №3.....	190
პროცესის იერარქიის აგება.....	190
პროცესის იერარქიის მიმოხილვა.....	190
პროცესის ნაკადის უმაღლესი დონის შექმნა.....	191
უმაღლესი დონის ფორმების შექმნა.....	191
პროცესის გვერდებს შორის გადაადგილება.....	192

ლაბორატორიული სამუშაო №4.....	193
უმაღლესი დონის შეერთებების ჩასმა.....	193
პროცესის ნაკადის გაწვევტა – სამუშაოს აღწერის დამუშავება.....	195
BPMN–ს ფორმების შეერთება.....	197
ლაბორატორიული სამუშაო №5.....	199
პროცესის ნაკადის გაწვევტა – გადაბირების დაწვევა.....	199
პროცესის ნაკადის გაწვევტა – ინტერვიუს აღება კანდიდატებისაგან.....	201
ლაბორატორიული სამუშაო №6.....	204
შეტანის და გამოტანის პროცესების მიმოხილვა.....	204
მასალების გვერდის შექმნა.....	205
მასალების გვერდზე მასალების დამატება.....	205
ლაბორატორიული სამუშაო №7.....	207
პროცესებს შორის შეერთებების (კავშირების) განსაზღვრა.....	207
დარჩენილი შეერთებების (კავშირების) განსაზღვრა.....	210
ლაბორატორიული სამუშაო №8.....	213
დარჩენილი მასალების შექმნა და მინიჭება.....	213
ლაბორატორიული სამუშაო №9.....	215
სიბრტითი სქემის (ნახაზის) შექმნა.....	215
გასასვლელების ფორმების შედგენისათვის ალბათობების მინიჭება.....	215
ლაბორატორიული სამუშაო №10.....	218
პროცესის დამთავრებული სქემის შემოწმება.....	218
პროცესისათვის თვისებების მინიჭება.....	220
ლაბორატორიული სამუშაო №11.....	222
BPMN–ს ფორმებისათვის შემსრულებლების დანიშვნა.....	222
ლაბორატორიული სამუშაო №12.....	225
ხილული თვისებების შექმნა.....	225
ლაბორატორიული სამუშაო №13.....	228
პროცესისათვის მუშაობის ძირითადი ინდიკატორის მინიჭება.....	228
განცალკევებულ ტექსტურ ყუთში თვისებების ჩვენება.....	230
ფუნქციის ნაკადის გვერდი შექმნა (წარმოება)	231
ლაბორატორიული სამუშაო №14.....	233
ლაბორატორიული სამუშაო №15.....	235
ლიტერატურა.....	237

შესავალი

თუ თქვენ საქმე გაქვთ ბიზნეს-პროცესებთან, საერთოდ პროცესების მართვასთან და კერძოდ, ბიზნეს პროცესების მართვასთან (**Business Process Management (BPM)**), მაშინ თქვენთვის საჭიროა **BPMN (Business Process Management Notation-ბიზნეს პროცესის მართვის აღნიშვნები)**. ბიზნეს პროცესების ნოტაცია (აღნიშვნები) დიდი ხანია არსებობს: **IDEF, DFD, eEPC, UML**. მაგრამ **BPMN** მათგან განსხვავდება უპირველესად იმით, რომ ის შექმნილია არა მარტო ბიზნეს-პროცესების დოკუმენტირებისათვის, არამედ მათი შესრულებისთვისაც. მეორეს მხრივ, **BPMN**-მა მოიპოვა საერთო აღიარება: დღეს ვერ იპოვით ბიზნეს-პროცესების მოდელირების სპეციალისტს ან ავტომატიზაციის საშუალებების გამაგრებლებს, რომელიც არ აღიარებდეს ამ სტანდარტის მნიშვნელობას. ამიტომ, თუ თქვენ ხართ:

- ტექნოკრატიული საწყობის ხელმძღვანელი;
- ხარისხის ან ბიზნეს-პროცესების სპეციალისტი;
- ბიზნეს ანალიტიკოსი;
- ინჟინერ-ტექნიკური ხელმძღვანელი ან სპეციალისტი,

მაშინ **BPMN** სტანდარტი თქვენთვის გახდება რთული პრობლემების გასაღები, განსაკუთრებით იმ პრობლემებისათვის, რომლებიც წარმოიშობიან სხვადასხვა ბიზნეს-ქვედანაყოფების ერთმანეთთან და სხვა ინჟინერ-ტექნიკურ პრობლემებთან გადაკვეთაზე.

ინფორმაციული ტექნოლოგიების თვალსაზრისით, **BPMN**-ის როლი ბიზნეს-პროცესების მართვაში დაახლოებით იგივეა, რაც **SQL**-ის როლი მონაცემთა ბაზების მართვაში. ორივე სტანდარტი არის პროგრესის ერთ-ერთი მამოძრავებელი ძალა თავის მიმართულებაში.

ბიზნეს პროცესების მართვის ინიციატივის (**Business Process Management Initiative (BPMI)**) ჯგუფმა შექმნა სტანდარტი ბიზნეს პროცესების მართვის ნოტაციები (აღნიშვნები) (**Business Process Modeling Notation (BPMN)**). **BPMN**-ის მთავარი მიზანი არის განსაზღვროს ნოტაციები (აღნიშვნები) (**Notations**), რომლებიც ადვილად გასაგები იქნება ყველა ბიზნეს-გამომყენებლისათვის, ბიზნეს-ანალიტიკოსიდან დაწყებული (რომელიც ქმნის პროცესების საწყის გეგმებს), ტექნოლოგიების შემსრულებელ ტექნიკურ პერსონალამდე, რომლებიც შეასრულებენ ამ პროცესებს და ბოლოს, ბიზნეს-პროცესებამდე, რომლებიც მართავენ და აკონტროლებენ ამ პროცესებს. ამრიგად, **BPMN** არის სტანდარტიზირებულ ხიდი ბიზნეს-პროცესების დამუშავებასა და ამ პროცესების შესრულებას შორის არსებულ განსხვავებას შორის.

სხვა, მაგრამ არა ნაკლებ მნიშვნელოვანი, მიზანი არის, რომ მოდელირების ენები, რომლებიც დამუშავებულია ბიზნეს-პროცესების შესრულებისათვის (ისეთი როგორცაა **BPEL4WS (Business Process Execution Language for Web Services)** (ბიზნეს პროცესის შესრულების ენა ვებ-სერვისისათვის)), იყვნენ ვიზუალიზირებული ანუ შესაძლებელი იყოს მათი თვალსაჩინოდ წარმოდგენა ბიზნესზე ორიენტირებული აღნიშვნებით.

ეს სპეციფიკაცია განსაზღვრავს ბიზნეს პროცესების დიაგრამების (**Business Process Diagram (BPD)**) აღნიშვნებს და სემანტიკებს და წარმოადგენს საუკეთესო საშუალებას ბიზნეს მოდელირების საზოგადოებისათვის. **BPMN**-ის მიზანი არის

მოახდინოს ბიზნეს პროცესის მოდელირების აღნიშვნების სტანდარტიზირება მოდელირების სხვა მრავალი აღნიშვნების და შეხედულებების გარემოში. აკეთებს რა ამას, BPMN იძლევა ბიზნესის სხვა მომხმარებლებთან, პროცესის შემსრულებლებთან, მომხმარებლებთან და მომმარაგებლებთან ურთიერთობის პროცესის ინფორმაციით უზრუნველყოფის მარტივ საშუალებას.

ბიზნესმენებისათვის ძალზე მოხერხებულია ბიზნეს-პროცესების ვიზუალიზაცია ნაკადი-გრაფიკის ფორმატში. ათასობით ბიზნეს ანალიტიკოსი შეისწავლის კომპანიების მუშაობის გზებს და განსაზღვრავენ ბიზნეს პროცესებს მარტივი ნაკადების გრაფიკებით. ეს წარმოშობს ტექნიკურ განსხვავებას ბიზნეს პროცესების საწყისი დამუშავების ფორმატსა და ენების (ისეთი როგორცაა BPEL4WS), რომლების შეასრულებენ ამ პროცესებს, ფორმატს შორის. განსხვავება უნდა იქნას დაძლეული ფორმალური მექანიზმის საშუალებით, რომელიც წარმოადგენს ბიზნეს პროცესების შესაბამის ვიზუალიზაციას შესრულების შესაბამისი ფორმატისაკენ.

ბიზნეს-პროცესების ურთიერთ ზემოქმედება ადამიანის დონეზე შეიძლება გადაწყვეტილი იყოს ბიზნეს პროცესის მოდელირების აღნიშვნების (BPMN) სტანდარტიზირებით. BPMN-ის გამოყენებით იქმნება ბიზნეს პროცესის დიაგრამები (BPD), რომლებიც არიან ბიზნეს პროცესების დამუშავებელი და ხელმძღვანელი პიროვნებების გამოსაყენებლად დამუშავებული დიაგრამები.

BPMN ბიზნესს უზრუნველყოფს შესაძლებლობით გასაგები გახადოს მათი გარე ბიზნესს პროცედურები გრაფიკული აღნიშვნებით და ორგანიზაციებს აძლევს შესაძლებლობას ურთიერთობა დაამყაროს ამ პროცედურებთან სტანდარტული ფორმით. ჩვეულებრივ, პროცესის მოდელირების საშუალებებსა და მეთოდოლოგიებს შორის არსებობს კავშირი. ცნობილია, რომ პიროვნებები გადადიან ერთი კომპანიიდან მეორეში და კომპანიები ერთიანდებიან და იყოფიან. ბიზნეს ანალიტიკოსებს მოეთხოვებათ, რომ მათთვის გასაგები იყოს ერთი და იგივე ბიზნეს პროცესების სხვადასხვა წარმოდგენები. ამიტომ სტანდარტული გრაფიკული აღნიშვნები ხელს შეუწყობს გაგებას და თანამშრომლობის განხორციელებას ორგანიზაციებს შორის. BPMN ასევე ავითარებს ტრადიციული ბიზნეს პროცესების აღნიშვნების შესაძლებლობებს მისთვის დამახასიათებელი ბიზნეს პროცესების კონცეფციების (ისეთების როგორებიცაა საზოგადოებრივი და კერძო პროცესები), ასევე მოდელირების თანამედროვე კონცეფციების გამოყენებით.

თავი 1. ზოგადი მიმოხილვა (Overview)

1.1. BPMN-ის შესაძლებლობები (კომპეტენცია)

BPMN უზრუნველყოფს მხოლოდ მოდელირების კონცეფციას რაც გამოსაყენებელია ბიზნეს პროცესებში. ეს ნიშნავს, რომ სხვა ტიპის მოდელირება, რომლებსაც ასრულებენ ბიზნეს პროცესების ორგანიზაციები ბიზნესის მიზნებიდან გამომდინარე, არიან BPMN-ის კომპეტენციის გარეთ. მაგალითად, შემდეგი ტიპის მოდელირება არ არის BPMN-ის შესაძლებლობის ფარგლებში:

- ორგანიზაციული სტრუქტურები და რესურსები;
- ფუნქციონალური დარღვევები;
- მონაცემების და ინფორმაციული მოდელები;
- სტრატეგია;
- ბიზნესის წესები.

რამდენადაც ასეთი ტიპის მაღალი დონის მოდელირება პირდაპირ ან არაპირდაპირ მოქმედებს ბიზნეს პროცესზე, BPMN-სა და სხვა მაღალი დონის ბიზნეს მოდელირებას შორის ურთიერთ კავშირი განსაზღვრული იქნება უფრო მეტად ფორმალურად, ვიდრე ეს არის BPMN-ში და სხვა სპეციფიკაციები იქნებიან წამოწეული წინ.

გარდა ამისა, რამდენადაც BPMN აჩვენებს მონაცემთა ნაკადებისა (შეტყობინებების) და მონაცემთა არტიფაქტების გაერთიანებას მოქმედებებთან, ის არ არის მონაცემთა ნაკადების დიაგრამა.

1.1.1. BPMN-ის გამოყენება

ბიზნეს პროცესის მოდელირება გამოიყენება მრავალფეროვანი ინფორმაციის დასაკავშირებლად მრავალფეროვან აუდიტორიასთან. BPMN არის შექმნილი მრავალი ტიპის მოდელირების მოსაცვლად და საშუალებას იძლევა შეიქმნას ბიზნეს პროცესის დიაგრამა (მოდელი) თავიდან ბოლომდე. BPMN-ის სტრუქტურული ელემენტები საშუალებას იძლევიან ადვილად შევნიშნოთ განსხვავება BPMN-ის დიაგრამის სექციებს შორის.

BPMN-ის მთლიან მოდელში შესაძლებელია სამი ძირითადი ტიპის ქვე-მოდელი:

1. კერძო (შიდა) ბიზნეს პროცესები;
2. აბსტრაქტული (საზოგადოებრივი) პროცესები;
3. თანამშრომლობის (გლობალური) პროცესები.

შენიშვნა: სხვადასხვა ტიპის პროცესების აღსაწერად გამოყენებული ტერმინოლოგია არ არის სტანდარტიზირებული. მიმდინარეობს სამუშაოები W3C (the World Wide Web Consortium)-ში და OASIS (the Organization for the Advancement of Structured Information Standards)-ში რომლებიც, იმედია, რომ მოახდენენ ამ ტერმინოლოგიის უნიფიკაციას.

რადგან BPMN-ის ზოგიერთი სპეციფიკური ტერმინის ზუსტი თარგმანი ვერ ასახავს ამ ტერმინების არსს (მაგალითად, ისეთი, როგორცაა პროცესის მონაწილენი (Swimlanes) (ანუ გუბეები და ბილიკები) (e.g., Pools and Lanes)), ქვემოთ ამ ტერმინების თარგმანს ხშირად თან ახლავს ფრჩხილებში მითითებული ინგლისური დასახელებები და უმჯობესია ისინი დავიმახსოვროთ როგორც ტერმინები.

კერძო (შიდა) ბიზნეს პროცესები

კერძო ბიზნეს პროცესები არიან შიდა, სპეციფიკური ორგანიზაციებისათვის და არიან ისეთი პროცესის ტიპები, რომლებსაც ზოგადად უწოდებენ სამუშაო ნაკადების (workflow) ან ბიზნეს პროცესების მოდელირების (BPM) პროცესებს (იხ. ნახ. 1.1).

თუ Swimlanes (პროცესის მონაწილენი) არიან გამოყენებული, მაშინ კერძო ბიზნეს პროცესი განხილული იქნება მარტივი გუბის (Pool) შიგნით (გარემოში). პროცესის ნაკადების მიმდევრობა ამის გამო არის მოთავსებული გუბის (Pool) შიგნით და არ შეუძლია გადაკვეთოს გუბის (Pool) საზღვრები. შეტყობინებების ნაკადს შეუძლია გადაკვეთოს გუბის (Pool) საზღვრები და აჩვენებს ურთიერთქმედებებს, რომლებიც არსებობენ ცალკეულ კერძო ბიზნეს პროცესებს შორის. ამრიგად, მარტივი ბიზნეს პროცესის დიაგრამას შეუძლია აჩვენოს მრავალი კერძო ბიზნეს პროცესი, რომელთაგან ყოველი განცალკევებულია BPEL4WS წარმოდგენით.



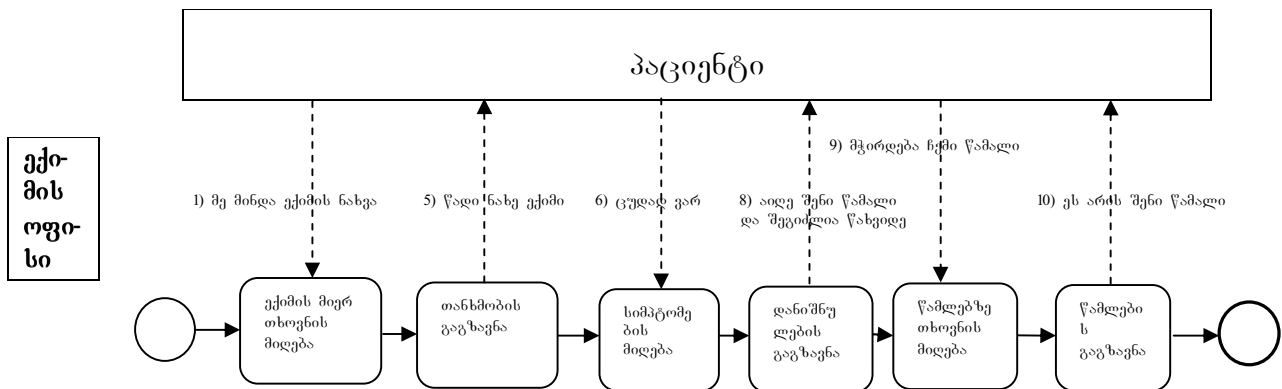
ნახ. 1.1. კერძო ბიზნეს პროცესის მაგალითი

აბსტრაქტული (საზოგადოებრივი) პროცესები

აბსტრაქტული (საზოგადოებრივი) პროცესები წარმოადგენენ კერძო ბიზნეს პროცესსა და სხვა პროცესებს ან მონაწილეს შორის ურთიერთქმედებებს (იხ. ნახ. 1.2). მხოლოდ ის მოქმედებები, რომლებიც არიან გამოყენებული კერძო ბიზნეს პროცესების გარეთ შეტყობინებისათვის, პლიუს შესაბამისი ნაკადების მართვის მექანიზმები, არიან ჩართული აბსტრაქტულ პროცესებში. კერძო ბიზნეს პროცესების ყველა სხვა “შიდა” მოქმედებები არ არიან ნაჩვენები აბსტრაქტულ პროცესებში. ამრიგად, აბსტრაქტული პროცესი მიუთითებს გარე სამყაროში შეტყობინებების მიმდევრობაზე, რომლებიც ურთიერთმოქმედებენ ბიზნეს პროცესზე.

აბსტრაქტული პროცესები არიან შეცვლილი გუბეში (Pool) და შეიძლება მოდელირებული იქნან დამოუკიდებლად (განცალკევებით) ან BPMN-ის დიდი დიაგრამის შიგნით, აბსტრაქტული პროცესების მოქმედებებს და სხვა

ორგანიზაციებს შორის შეტყობინებების ნაკადების საჩვენებლად. თუ აბსტრაქტული პროცესი არის იგივე დიაგრამაში, რომელშიც მისი შესაბამისი კერძო ბიზნეს პროცესია, მაშინ მოქმედებები, რომლებიც საერთო არიან ორივე პროცესისათვის შეიძლება იყვნენ გაერთიანებული.

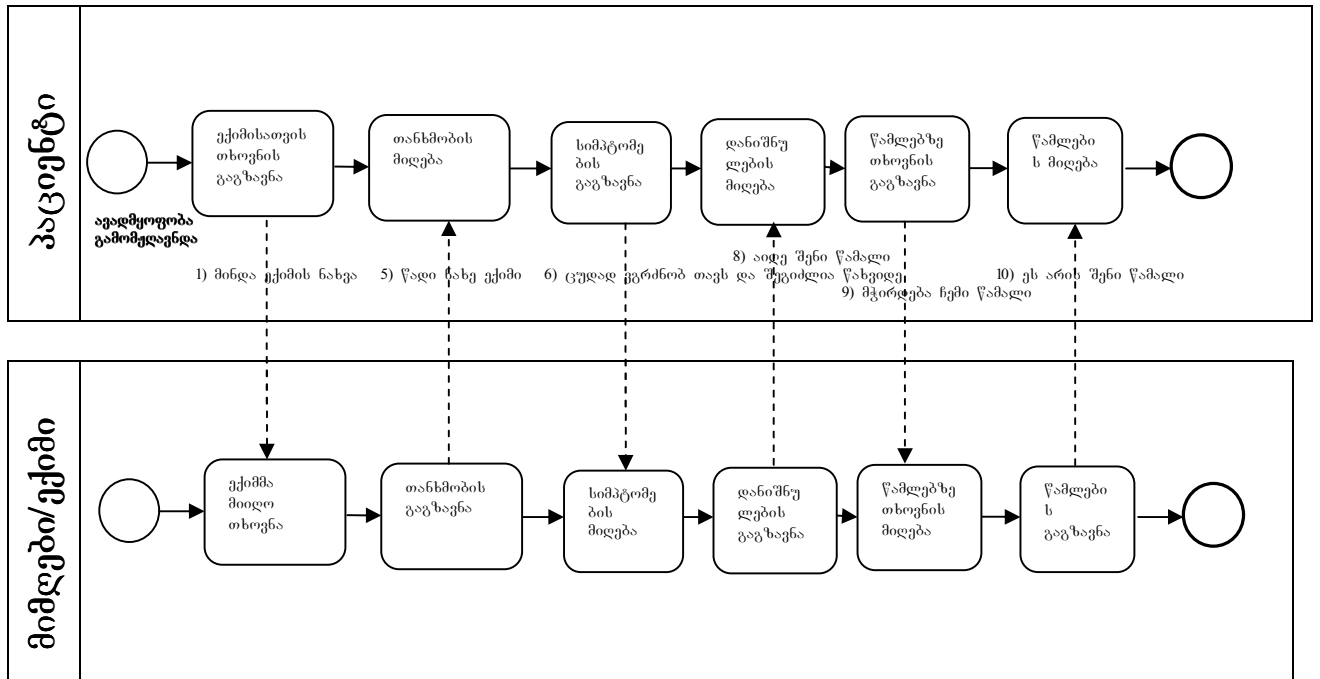


ნახ. 1.2. აბსტრაქტული ბიზნეს პროცესის მაგალითი

თანამშრომლობის (გლობალური) პროცესები

თანამშრომლობის პროცესი გამოსახავს ორ ან მეტ ბიზნეს ორგანიზაციას შორის ურთიერთქმედებას. ეს ურთიერთქმედებები განსაზღვრული არიან მოქმედებების მიმდევრობის სახით, რაც წარმოადგენს შეტყობინების გაცვლის მაგალითს საქმეში ჩართულ ორგანიზაციებს შორის.

თანამშრომლობის პროცესი შეიძლება ნაჩვენები იქნას როგორც ორი ან მეტი აბსტრაქტული პროცესი, რომლებიც ურთიერთქმედებენ ერთმანეთზე (იხ. ნახ. 1.3). მოქმედებები თანამონაწილეების თანამშრომლობისათვის აბსტრაქტულ პროცესთან შეიძლება განხილული იქნას როგორც “შესება-წერტილები” მონაწილეებს შორის. ფაქტიურ (შესრულებად) პროცესებს შესაძლებლობა აქვთ ჰქონდეთ უფრო მეტი მოქმედებები და დეტალები ვიდრე ნაჩვენებია აბსტრაქტულ პროცესებში.



ნახ. 1.3. თანამშრომლობის ბიზნეს პროცესის მაგალითი

ბიზნეს პროცესის დიაგრამების (ბპდ) (BPD) ტიპები

BPMN -ის ამ სამ ქვე-მოდელში და მათ შორის დიაგრამების ბევრი ტიპი შეიძლება იყოს შექმნილი. ქვემოთ ჩამოთვლილი არიან ბიზნეს პროცესების ტიპები, რომლებიც შეიძლება იყვნენ მოდელირებული BPMN-ით (ვარსკვლავით აღნიშნულების გამოხატვა შეიძლება არ იყოს შესაძლებელი შესრულებადი ენით, ანუ არ იყოს რეალიზებული პროგრამის კონკრეტულ ვერსიაში):

- მაღალდონიანი კერძო პროცეს მოქმედებები*;
- დაწვრილებითი კერძო ბიზნეს პროცესი;
- დაწვრილებითი კერძო ბიზნეს პროცესი ერთ ან მეტ გარე ორგანიზაციებთან ურთიერთქმედებით (ან “Black Box” (შავი ყუთი) პროცესები);
- ორი ან მეტი დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთქმედება;
- დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთდამოკიდებულება აბსტრაქტულ პროცესთან;
- დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთდამოკიდებულება თანამშრომლობის პროცესთან;
- ორი ან მეტი აბსტრაქტული პროცესი*;
- აბსტრაქტული პროცესის ურთიერთდამოკიდებულება თანამშრომლობის პროცესთან;
- თანამშრომლობის პროცესი მხოლოდ*;

- ორი ან მეტი დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთქმედება (მათი აბსტრაქტული პროცესების საშუალებით);
- ორი ან მეტი დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთქმედება (თანამშრომლობის პროცესის საშუალებით);
- ორი ან მეტი დაწვრილებითი კერძო ბიზნეს პროცესების ურთიერთქმედება (მათი აბსტრაქტული პროცესების და თანამშრომლობის პროცესის საშუალებებით).

BPMN-ში რეალიზებულია ყველა ზემოთ ჩამოთვლილი დიაგრამების ტიპები. თუმცა, ყურადსაღებია ის გარემოება, რომ თუ ქვე-მოდულების ძალზე ბევრი ტიპია გაერთიანებული (ისეთები როგორცაა სამი ან მეტი კერძო პროცესები შეტყობინებების ნაკადებით ყოველ მათგანს შორის), მაშინ დიაგრამა შეიძლება გახდეს გასაგებად ძალზე რთული ზოგიერთისათვის. ამრიგად, რეკომენდირებულია, რომ მოდელის შემქმნელმა ჩამოაყალიბოს კონკრეტული მიზანი BPD-თვის (ბიზნეს პროცესის განვითარებისათვის); მაგალითად, ისეთი როგორცაა კერძო პროცესი ან თანამშრომლობის პროცესი.

1.1.2. თვალსაზრისის (Point of View) დიაგრამა

რამდენადაც BPMN დიაგრამას შეუძლია წარმოადგინოს სხვადასხვა მონაწილეების პროცესები, ყოველ მონაწილეს შეუძლია განიხილოს დიაგრამა სხვადასხვა ნაირად. ეს ნიშნავს, რომ მონაწილეებს აქვთ სხვადასხვა თვალსაზრისი იმის შესახებ, თუ რამდენად შეესაბამება (გამოადგება) პროცესები მათ. მონაწილისათვის ზოგიერთი მოქმედება იქნება შიდა (მონაწილის მიერ ან მისი კონტროლის ქვეშ შესრულებული მოქმედების მნიშვნელობა), ხოლო სხვა მოქმედებები იქნებიან გარე. ყოველ მონაწილეს ექნება სხვადასხვა თვალსაზრისი თუ რომელი იქნება შიდა და რომელი - გარე. შესრულების დროს შიდა და გარე მოქმედებებს შორის განსხვავება არის მნიშვნელოვანი იმისათვის, თუ როგორ შეუძლია მონაწილეს განიხილოს მოქმედებების სტატუსი ან გამოიკვილოს რაიმე პრობლემა. მიუხედავად ამისა, დიაგრამა თავის თავად რჩება უცვლელი. ზამთო მოყვნილი ნახ. 1.3 გვიჩვენებს ბიზნეს პროცესს, რომელსაც გააჩნია ორი თვალსაზრისი (points of view). ერთი თვალსაზრისი ეკუთვნის პაციენტს, მეორე არის ექიმის ოფისის. დიაგრამა აჩვენებს ორივე მონაწილის მოქმედებებს პროცესში მაგრამ, როცა პროცესი ნამდვილად დასრულდება, ყოველ მონაწილეს ექნება კონტროლი მხოლოდ მათ საკუთარ მოქმედებებზე.

თუმცა თვალსაზრისის დიაგრამა მნიშვნელოვანია დიაგრამის მიმომხილველისათვის რათა გაიგოს თუ როგორ არის დაკავშირებული პროცესის ქცევა (მოქმედება) ასეთ მიმომხილველთან, BPMN ჩვეულებრივ ვერ განსაზღვრავს რაიმე გრაფიკულ მექანიზმს თვალსაზრისის თვალსაჩინოებისათვის. მოდელირების შემსრულებლისათვის ან მოდელირების საშუალებების (ინსტრუმენტების) გამოიყენებისათვის გახსნილია თვალსაჩინოების ყველა რეპლიკა დიაგრამის ამ მახასიათებლისათვის ხაზის გასასმელად.

1.1.3. BPMN-ის გაფართოებადობა და ვერდიკალური დომენები

შესაძლებელია, რომ BPMN იყოს გაფართოებული მოდელირების შემსრულებლების მიერ და მოდელირების ინსტრუმენტებით. ეს გაფართოებადობა საშუალებას აძლევს მოდელირების შემსრულებელს დაამატოს არა სტანდარტული ელემენტები ან არტიფაქტები (ნიმუშები) (Artifacts) სპეციფიკური საჭიროებების დასაკმაყოფილებლად, ისეთების როგორებიცაა ვერდიკალური არეების უნიკალური მოთხოვნები. გაფართოებისას BPMN-ის დიაგრამებს უნდა შეუნარჩუნდეთ ძირითადი ფორმა (გარეგნობა) და ნებისმიერი მოდელირების მიერ გაკეთებული დიაგრამა უნდა იყოს ადვილად გასაგები დიაგრამის ნებისმიერი მიმომხილველისათვის. ამრიგად, ძირითადი ნაკადის ელემენტების (ხდომილებების, მოქმედებების და გასასვლელების) (Events, activities, and Gateways) კვალი (ქვედა წარწერა) (footprint) არ უნდა იყოს შეცვლილი. აგრეთვე ბიზნეს პროცესების დიაგრამაში (BPD) არ უნდა იყოს დამატებული რომელიმე ახალი ნაკადის ელემენტი, რამდენადაც არ არსებობს სპეციფიკაცია თუ როგორი შესრულების ან შეტყობინების ნაკადი დაუკავშირდება რომელიმე ახალ ნაკადის ობიექტს (Flow Object). დამატებითი მოდელირების კონცეფციების დასაკმაყოფილებლად, რომლებიც არ არიან ძირითადი ნაკადის ელემენტების ნაწილი, BPMN უზრუნველყოფს არტიფაქტების (ნიმუშების) (Artifacts) კონცეფციას რაც შეიძლება დაკავშირებული იყოს არსებულ ნაკადის ობიექტებთან ასოციაციების (გაერთიანებების) (Associations) საშუალებით. ამრიგად, არტიფაქტები არ მოქმედებენ ძირითად შესრულების ან შეტყობინების ნაკადებზე, ისინი ასევე არ მოქმედებენ შესრულების ენების შედეგებზე.

BPMN-ის გრაფიკული ელემენტები არიან დამუშავებული ისე, რომ იყვნენ ღია და საშუალებას აძლევდნენ სპეციალიზირებულ მარკერებს გადაიტანონ სპეციალიზირებული ინფორმაცია. მაგალითად, ყველა ხის (განშტოების) ტიპის მოვლენებს მარკერებისათვის აქვთ ღია ცენტრები, რასაც BPMN უკეთებს სტანდარტიზირებას, როგორც გამომყენებლის მიერ განსაზღვრულ მარკერებს.

თავი 2. ბიზნეს პროცესის დიაგრამები (BPD)

ამ თავში მოცემულია BPMN-ის გრაფიკული ობიექტები და მათი ურთიერთ კავშირები. ცნებების (კონცეფციების) შესახებ უფრო დაწვრილებითი ინფორმაცია მოცემული იქნება 3-ე თავში (ბიზნეს პროცესის დიაგრამის გრაფიკული ობიექტები) და 4-ე თავში (ბიზნეს პროცესის დიაგრამის შემაერთებელი (დამაკავშირებელი) ობიექტები).

BPMN-ის განვითარების მიზანი არის, რომ გამოყენებული აღნიშვნები იყოს მარტივი და მისაღები ბიზნეს ანალიტიკოსებისათვის. აგრეთვე, არსებობს პოტენციურად კონფლიქტური მოთხოვნა, რომ BPMN-ს ჰქონდეს საშუალება გამოსახოს რთული ბიზნეს პროცესები და ისინი წარმოადგინოს ბიზნეს პროცესების მოდელის (BPM) შესრულებად ენაში. იმის გასაგებად, თუ როგორ შეუძლია BPMN-ს მართოს ორივე მოთხოვნა, BPMN-ის გრაფიკული ელემენტების ჩამონათვალი მოცემულია ორ ჯგუფში.

პირველ ჯგუფში მოცემულია ცენტრალური (ძირითადი) ელემენტების ჩამონათვალი, რომლებიც აკმაყოფილებენ მარტივი აღნიშვნების მოთხოვნებს. ესენი არიან ელემენტები, რომლებიც განსაზღვრავენ BPMN-ის ძირითად სტრუქტურას და სახეს. ბიზნეს პროცესების უმრავლესობა იქნება მოდელირებული ადეკვატურად ამ ელემენტებით. *მეორე ჯგუფში* მოცემულია ელემენტების მთლიანი ჩამონათვალი, ცენტრალური ელემენტების ჩათვლით, რომლებიც დაგვეხმარებიან უზრუნველყოთ უნივერსალური აღნიშვნების მოთხოვნები, რათა გავეუმკლავდეთ მოდელირების უფრო რთულ სიტუაციებს. შემდეგ, აღნიშვნების გრაფიკული ელემენტები მხარდაჭერილი იქნება არა გრაფიკული ატრიბუტებით, რაც უზრუნველყოფს დარჩენილი ინფორმაციის წარმოდგენას, რომელიც აუცილებელია შესასრულებელ ენაზე (ან სხვა ბიზნეს მოდელირების მიზნებისათვის) წარმოსადგენად. ატრიბუტები იძლევიან დამატებით ინფორმაციას, საჭიროს პროცესის BPEL4WS (Business Process Execution Language for Web Services)-ში წარმოსადგენად.

2.1. ბიზნეს პროცესების დიაგრამების (ბპდ) (BPD) ძირითადი ელემენტების სიმრავლე

ხაზი უნდა გაესვას იმ გარემოებას, რომ BPMN-ის განვითარებისათვის ერთ-ერთი მამოძრავებელია ბიზნეს პროცესების მოდელის შექმნისათვის მარტივი მექანიზმის შექმნა ისე, რომ იმავედროულად შესაძლებელი იყოს ბიზნეს პროცესების სირთულეებთან გამკლავება. მიდგომა, რომელმაც საშუალება მისცა გამკლავებოდა ამ ორ კონფლიქტურ მოთხოვნილებას, იყო აღნიშვნების გრაფიკული ასპექტების ორგანიზება სპეციფიკურ კატეგორიებში. ეს უზრუნველყოფს აღნიშვნების კატეგორიის მცირე სიმრავლეს ისე, რომ BPMN-ის დიაგრამების მკითხველს შეეძლოს ადვილად გაარჩიოს ელემენტების ძირითადი ტიპები და გაიგოს დიაგრამა. ელემენტების ძირითადი კატეგორიების შიგნით შეიძლება იყოს დამატებული დამატებითი ცვალებადობა და ინფორმაცია კომპლექსურობის მოთხოვნების მხარდასაჭერად, დიაგრამის ძირითადი ფორმის (გარეგნობის) და

არსის არსებითი ცვლილების გარეშე. ელემენტების ოთხი ძირითადი კატეგორია არსებობს:

1. ნაკადის ობიექტები (Flow Objects);
2. შემაერთებელი (დამაკავშირებელი) ობიექტები (Connecting Objects);
3. პროცესის მონაწილენი (Swimlanes);
4. არტიფაქტები (ნიმუშები) (Artifacts).

ნაკადის ობიექტები (Flow Objects) არიან ძირითადი გრაფიკული ელემენტები, რომლებიც განსაზღვრავენ ბიზნეს პროცესების ყოფა-ქცევას. არსებობენ სამი სახის ნაკადის ობიექტები (Flow Objects):

1. ხდომილებები (Events);
2. მოქმედებები (Activities);
3. გასასვლელები (Gateways).

ნაკადის ობიექტების (Flow Objects) ერთმანეთთან ან სხვა ინფორმაციასთან დაკავშირების (შეერთების) სამი გზა არსებობს. არსებობენ სამი სახის დამაკავშირებელი (შემაერთებელი) ობიექტები (Connecting Objects):

1. შესრულების ნაკადი (Sequence Flow);
2. შეტყობინების ნაკადი (Message Flow);
3. გაერთიანება (Association).

“პროცესის მონაწილენის” (“Swimlanes”) საშუალებით მოდელირების ძირითადი ელემენტების დაჯგუფების ორი გზა არსებობს:


1. გუბეები (Pools);
2. ბილიკები (Lanes).


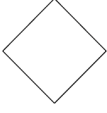
არტიფაქტები (ნიმუშები) (Artifacts) გამოიყენება პროცესის შესახებ დამატებითი ინფორმაციის უზრუნველსაყოფად. არსებობს სამი სტანდარტიზირებული არტიფაქტი, მაგრამ მოდელირებს (modelers) ან მოდელირების ინსტრუმენტებს (modeling tools) შეუძლიათ დაამატონ რამდენიც საჭიროა იმდენი არტიფაქტი. BPMN-ს აქვს დამატებითი შესაძლებლობები სტანდარტიზირება გაუკეთოს არტიფაქტების დიდ სიმრავლეს საერთო გამოყენებისათვის. არტიფაქტების არსებული სიმრავლე მოიცავს:

1. მონაცემების ობიექტი (Data Object);
2. ჯგუფი (Group);
3. ანოტაცია (შენიშვნა) (Annotation).


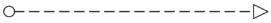
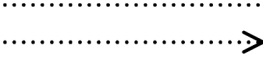
ცხრილ 2.1-ში მოცემულია მოდელირების ძირითადი ელემენტების ჩამონათვალი, რომლებიც წარმოდგენილი არიან შესაბამისი აღნიშვნებით.





ცხრილი 2.1. მოდელირების ძირითადი ელემენტები

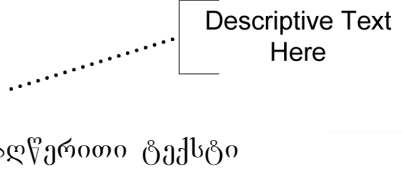
ელემენტი Element	აღწერა Description	ნოტაცია (აღნიშვნა) Notation
ხდომილება Event	ხდომილება არის რაღაც, რაც ხდება ბიზნეს პროცესის მიმდინარეობისას (“ხდომილებები”, იხ. § 3.3., გვ. 42). ეს ხდომილებები მოქმედებენ	

	<p>პროცესის ნაკადზე და ჩვეულებრივ აქვთ მიზეზი (გადამრთველი) ან ზემოქმედება (შედეგი). ხლომილებები არიან წრეები ღია ცენტრებით, რომლებიც შიდა მარკერებს საშუალებას აძლევენ განასხვავონ სხვადასვა გადამრთველები ან შედეგები. არსებობს <i>ხლომილებების სამი ტიპი</i> იმის და მიხედვით თუ როგორ მოქმედებენ ისინი ნაკადზე: დაწყება, შუალედური და დამთავრება (Start, Intermediate and End).</p>	
<p>მოქმედებები Activities</p>	<p>მოქმედება არის საერთო აღნიშვნა სამუშაოსათვის, რომელსაც კომპანია ასრულებს ("მოქმედებები", იხ. § 3.4., გვ. 72). მოქმედება შეიძლება იყოს ელემენტარული (atomic) და არა ელემენტარული (შედგენილი) (non-atomic (compound)). <i>მოქმედებების ტიპები</i>, რომლებიც არიან პროცესის მოდელის ნაწილი, არიან: პროცესი, ქვეპროცესი და ამოცანა (Process, Sub-Process, and Task). ამოცანები და ქვეპროცესები არიან მომრგვალებულ კუთხეებიანი სწორკუთხედები. პროცესებს შეიცავენ გუბები (Pool).</p>	
<p>გასასვლელი Gateway</p>	<p>გასასვლელი გამოიყენება შესრულების ნაკადის განშლის და შერწყმის სამართავად ("გასასვლელი", იხ. § 3.5., გვ. 99). ამრიგად, განსახდურული იქნება ბილიკების განშტოება, გაშლა, შერწყმა და შეერთება (branching, forking, merging, and joining). შიდა მარკერები მიუთითებენ ქცევის მართვის ტიპს.</p>	

ცხრილი 2.2. ბიზნეს პროცესების დიაგრამების (BPD) ძირითადი ელემენტების სიმრავლე

ელემენტი Element	აღწერა Description	ნოტაცია (აღნიშვნა) Notation
შესრულების ნაკადი (Sequence Flow)	შესრულების ნაკადი გამოიყენება იმ მიმდევრობის საჩვენებლად რომლითაც მოქმედებები იქნებიან შესრულებული პროცესში (“შესრულების ნაკადი” იხ. 4.1.2., 139 გვერდზე).	
შეტყობინების ნაკადი (Message Flow)	შეტყობინების ნაკადი გამოიყენება ორ მონაწილეს შორის შეტყობინებების ნაკადის საჩვენებლად, რომლებიც მომზადებული არიან, რომ გაგზავნონ და მიიღონ შეტყობინება (“შეტყობინების ნაკადი” იხ. 4.1.3., 142 გვერდზე). BPMN-ს დიაგრამაში ორი სხვადასხვა გუბე წარმოადგენს ორ მონაწილეს (ანუ ბიზნეს ორგანიზაციებს ან ბიზნეს როლებს (business roles)).	
გაერთიანება (Association)	გაერთიანება გამოიყენება ინფორმაციის გასაერთიანებლად ნაკადის ობიექტებთან (Flow Objects). ტექსტური და გრაფიკული არა ნაკადის ობიექტები (non-Flow Objects) შეიძლება გაერთიანებული იყვნენ ნაკადის ობიექტებთან (Flow Objects). ისრის მიმართულება გაერთიანებაზე მიუთითებს შესაბამისი ნაკადის (ანუ მონაცემების) მიმართულებაზე (“გაერთიანება” იხ. 4.1.4., 144 გვერდზე).	
გუბე (Pool)	გუბე წარმოადგენს პროცესის მონაწილეს (“გუბე” იხ. 3.6.2., 124 გვერდზე), აგრეთ-	





	<p>ვე მოქმედებს როგორც “პროცესის მონაწილე” (“swimlane”) და გრაფიკული კონტეინერი (შემცველი) მოქმედებების სიმრავლის გამოყოფისათვის სხვა გუბებისაგან.</p>	
<p>ბილიკი (Lane)</p>	<p>ბილიკი არის ქვე-დაყოფა გუბის შიგნით და აფართოვებს გუბის მთლიან სიგრძეს როგორც ვერდიკალურად ასევე ჰორიზონტალურად (“ბილიკი” იხ. 3.6.3., 128 გვერდზე).</p>	
<p>მონაცემების ობიექტი (Data Object)</p>	<p>მონაცემების ობიექტები არიან არტიფაქტები (ნიმუშები) (Artifacts), რადგან მათ არა აქვთ რაიმე პირდაპირი გავლენა პროცესის შესრულების ნაკადზე ან შეტყობინების ნაკადზე, მაგრამ ისინი უზრუნველყოფენ ინფორმაციას იმის შესახებ თუ რა მოქმედებების შესრულება არის საჭირო და/ან რას აწარმოებენ ისინი (“მონაცემების ობიექტი” იხ. 3.7.2., 131 გვერდზე).</p>	
<p>ჯგუფი (Group) (ყუთი ობიექტების ჯგუფის გარშემო იგივე კატეგორიის შიგნით)</p>	<p>მოქმედებების დაჯგუფება რომლებიც არიან იგივე კატეგორიის შიგნით (“ჯგუფი” იხ. 3.7.4., გვ. 135). დაჯგუფებების ეს ტიპი არ მოქმედებს მოქმედებების შესრულების ნაკადზე ჯგუფში. კატეგორიის სახელი ნაჩვენებია დიაგრამაზე ჯგუფის ჭდის (label) სახით. კატეგორიები შეიძლება იქნენ გამოყენებული დოკუმენტებისათვის ან ანალიზის მიზნით. ჯგუფები იძლევიან კატეგორიების დიაგრამაზე ობიექტების ვიზუალურად წარმოდგენის ერთ-ერთ შესაძლებლობას.</p>	

<p>ტექსტური ანოტაცია (Text Annotation) (მიდგმულია (მიმაგრებულია) გაერთიანებასთან (Association))</p>	<p>ტექსტური ანოტაცია არის მექანიზმი მოდელირებისათვის რათა მკითხველი უზრუნველყოს BPMN-ის დიაგრამის დამატებითი ინფორმაციით (“ტექსტური ანოტაცია” იხ. 3.7.3, გვ. 134).</p>	 <p>აღწერილობითი ტექსტი</p>
---	--	--


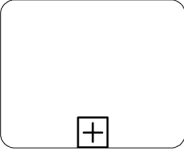
2.2. ბიზნეს პროცესების დიაგრამების (ბპდ) (Business Process Diagram (BPD)) გაფართოებული სიმრავლე

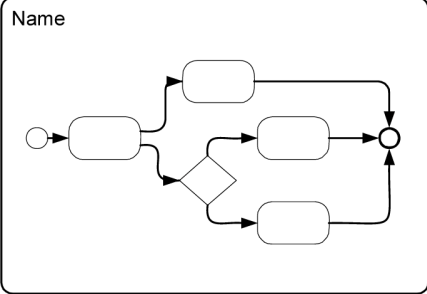
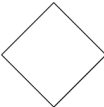






ცხრილ 2.3-ში ნაჩვენებია ბიზნეს პროცესის კონცეფციების (ცნებების) გაფართოებული ჩამონათვალი, რომლებიც შეიძლება წარმოდგენილი იყვნენ ბიზნეს პროცესის მოდელირების აღნიშვნების საშუალებით.

ცხრილი 2.3. ბიზნეს პროცესების დიაგრამების (BPD) ელემენტების გაფართოებული სიმრავლე


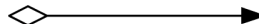
ელემენტი	აღწერა	აღნიშვნა	
<p>ხდომილება (Event)</p>	<p>ხდომილება არის ის, რაც “ხდება” ბიზნეს პროცესის მიმდინარეობის დროს. ეს ხდომილებები მოქმედებენ პროცესების ნაკადებზე და ჩვეულებრივ აქვთ რაიმე მიზეზი (გადამრთველი) ან რაიმე ზემოქმედება (შედეგი). ხდომილებების სამი ტიპი არსებობს, რომლებიც მოქმედებენ ნაკადზე: დაწყება (Start), შუალედური (Intermediate) და დამთავრება (End).</p>		<p>სახელი ან წყარო</p>
<p>ნაკადის განხორციელება (ანუ დაწყება, შუალედური, დამთავრება)</p> <p>დაწყება (Start) (ცარიელი, შეტყობინება, ტაიმერი, პირობითი, სიგნალი, მრავლობითი)</p> <p>შუალედური (Intermediate) (ცარიელი, შეტყობინება, ტაიმერი, შეცდომა, გაუქმება, კომპენსაცია, პირობითი, რგოლი, სიგნალი, მრავლობითი)</p>	<p>როგორც სახელიდან გამომდინარეობს, დაწყების ხდომილება მიუთითებს სად დაიწყება მოცემული პროცესი (“დაწყება” იხ. 3.3.2., გვ. 43)</p> <p>შუალედური ხდომილებები გეხვდება დაწყების ხდომილებასა და დამთავრების ხდომილებებს შორის (“შუალედური” იხ. 3.3.4., გვ. 56). ისინი მოქმედებენ პროცესების ნაკადებზე, მაგრამ არ იწყებენ ან (პირდაპირ) არ ამთავრებენ პროცესს.</p>	<p>Start </p> <p>Intermediate </p> <p>End </p>	<p>დაწყება</p> <p>შუალედური</p> <p>დამთავრება</p>

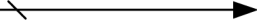
<p>დამთავრება (End) (ცარიელი, შეტყობინება, შეცდომა, გაუქმება, კომპენსაცია, სიგნალი, დამთავრება, მრავლობითი)</p>	<p>როგორც სახელიდან გამომდინარეობს, დამთავრების ხდომილება მიუთითებს სადმთავრდება პროცესი (“დამთავრება” იხ. 3.3.3., გვ. 49).</p>																																														
<p>ტიპის განზომილება (ანუ ცარიელი, შეტყობინება, ტაიმერი, შეცდომა, გაუქმება, კომპენსაცია, პირობითი, რგოლი, სიგნალი, მრავლობითი, დამთავრება.) (Type Dimension (e.g., None, Message, Timer, Error, Cancel, Compensation, Conditional, Link, Signal, Multiple, Terminate.))</p>	<p>დაწყების და უმრავლეს შუალედურ ხდომილებებს აქვთ “გადამრთველები” (“Triggers”), რომლებიც განსაზღვრავენ ხდომილებისათვის მიზეზს. არსებობს მრავალი გზა ამ ხდომილებების გამოსაძახებლად (დასაწყებად). დამთავრების ხდომილებებმა შეიძლება განსაზღვრონ “შედგეი”, რაც არის შესრულების ნაკადის დამთავრების შედეგი. დაწყების ხდომილებას შეუძლია რეაგირება (“დაჭერა”) მხოლოდ გადამრთველზე (trigger). დამთავრების ხდომილებებს შეუძლია შექმნან (“გადააგდონ”) მხოლოდ შედეგი. შუალედურ ხდომილებებს შეუძლიათ “დაიჭირონ” (მიიღონ) ან “გადააგდონ” (გადართონ) გადამრთველები (Triggers). ხდომილებებისათვის, რომლებიც გადამრთველებს “უჭირავენ”, მარკერები არიან შეუვსებელი (ფერით) და გადამრთველები-</p>	<p style="text-align: center;">“დაჭერა” “გადაგდება”</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th></th> <th>“Catching”</th> <th colspan="2">“Throwing”</th> </tr> </thead> <tbody> <tr> <td>Message</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Timer</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Error</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Cancel</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Compensation</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Conditional</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Link</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Signal</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Terminate</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Multiple</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		“Catching”	“Throwing”		Message				Timer				Error				Cancel				Compensation				Conditional				Link				Signal				Terminate				Multiple				<p>შეტყობინება ტაიმერი შეცდომა გაუქმება კომპენსაცია პირობითი რგოლი (ЗВЕНО) სიგნალი დამთავრება მრავლობითი</p>
	“Catching”	“Throwing”																																													
Message																																															
Timer																																															
Error																																															
Cancel																																															
Compensation																																															
Conditional																																															
Link																																															
Signal																																															
Terminate																																															
Multiple																																															

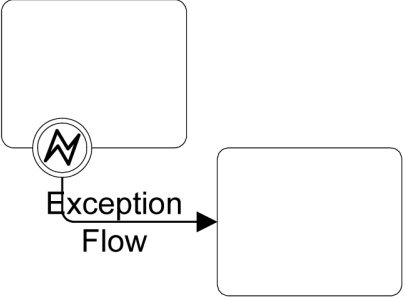
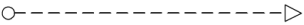
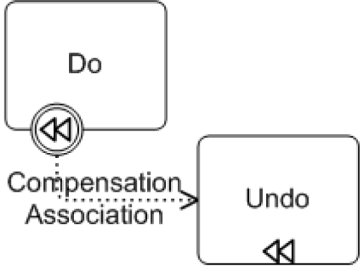
	<p>სათვის და შედეგებისათვის, რომლებიც არიან “გადაგდებული”, მარკერები არიან შევსებული (ფერით).</p>		
<p>ამოცანა (ელემენტარული) (Task (Atomic))</p>	<p>ამოცანა არის ელემენტარული (ატომური) აქტიურობა, რომელიც არის ჩართული პროცესში (“ამოცანა” იხ. 3.4.3., გვ. 90). ამოცანა გამოიყენება, როცა პროცესში მუშაობა არ არის გაწყვეტილი პროცესის დეტალური მოდელის საბოლოო დონეზე.</p>		
<p>პროცესი/ქვე-პროცესი (არა ატომური) (Process/Sub-Process (nonatomic))</p>	<p>ქვე-პროცესი არის რთული (შედგენილი) აქტიურობა, რომელიც არის ჩართული პროცესში (“ქვე-პროცესი” იხ. 3.4.2., გვ. 77). ის შედგენილია ისე, რომ ის შეიძლება იყოს შეწყვეტილი დაწვრილებითი პროცესის ბოლო დონეზე, ქვე-აქტიურობების სიმრავლის საშუალებით.</p>	<p>ნახე შემდეგი ორი ნახაზი.</p>	
<p>დანგრეული ქვეპროცესი (Collapsed Sub-Process)</p>	<p>ქვე-პროცესის დეტალები არ ჩანან დიაგრამაზე. “პლიუს” ნიშანი ფორმის ქვედა ცენტრში მიუთითებს, რომ მოქმედება არის ქვე-პროცესი და აქვს დაწვრილების (დეტალირების) უმდაბლესი დონე.</p>		

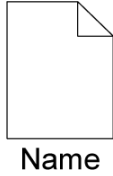
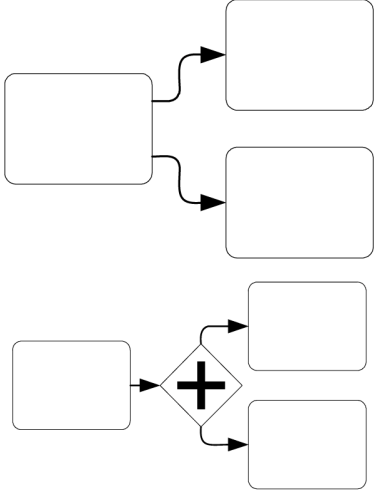
<p>გაფართოებული ქვე-პროცესი (Expanded Sub-Process)</p>	<p>ქვე-პროცესის საზღვარი არის გაფართოებული და პროცესის დეტალები ჩანს მის საზღვრებში. შევნიშნავთ, რომ შესრულების ნაკადს არ შეუძლია გადაკვეთოს ქვე-პროცესის საზღვარი.</p>	<p>Name</p> 	<p>სახელი</p>
<p>გასასვლელი (პორტი) (Gateway)</p>	<p>გასასვლელი გამოიყენება მრავალი შესრულების ნაკადის განშტოებისა და შეკრების (ერთ წერტილში) მართვისათვის (“გასასვლელები” იხ. 3.5, გვ. 99). ამრიგად, ის განსაზღვრავს ბილიკების (გზების) განშტოებებს, გაყრას, შერწყმას და შეერთებებს.</p>		
<p>გასასვლელის (პორტის) მართვის ტიპები (Gateway Control Types)</p>	<p>პიქტოგრამები (Icons) რომლის ფორმაში მიუთითებენ ნაკადის მართვის ქცევის ტიპებზე. მართვის ტიპები შეიცავენ:</p> <ul style="list-style-type: none"> განსაკუთრებულ (ერთადერთ) გადაწყვეტილებას და შერწყმას. ორივეს მონაცემებზე დაფუძნებულს (“Data-Based” იხ. 3.5.2.1, გვ. 105) და ხდომილებაზე დაფუძნებულს (“Event-Based” იხ. 3.5.24, გვ. 110). მონაცემებზე დაფუძნებული 	<p>Exclusive Data-Based  or </p> <p>Event-Based </p> <p>Inclusive </p> <p>Complex </p> <p>Parallel </p>	<p>განსაკუთრებული (ერთადერთი) მონაცემებზე დაფუძნებული</p> <p>ხდომილებაზე დაფუძნებული</p> <p>თავის თავში შემცველი</p> <p>როული</p> <p>პარალელური</p>

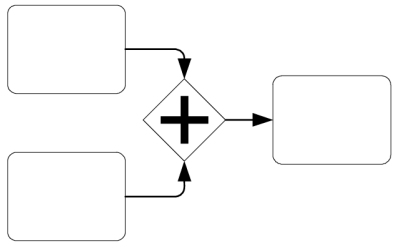
	<p>შეიძლება იყოს ნაჩვენები “X” ნიშნით ან უიშისოდ.</p> <ul style="list-style-type: none"> • თავის თავში შემცველი გადაწყვეტილება (Inclusive decision) და შერწყმა (merging) (“შემცველი გასასვლელები” იხ. 3.5.3, გვ. 114) • რთული – რთული პირობები და სიტუაციები (“რთული გასასვლელები” იხ. 3.5.4, გვ. 118). • პარალელური განშტოებები და შეერთებები (“პარალელური გასასვლელები”, იხ. 3.5.5, გვ. 121). <p>მართვის ყოველი ტიპი მოქმედებს ორივე შემთხვევაზე და გამავალ ნაკადებზე.</p>		
<p>შესრულების ნაკადი (Sequence Flow)</p>	<p>შესრულების ნაკადი გამოიყენება იმ მიმდევრობის საჩვენებლად, რომლითაც მოქმედებები შესრულდება პროცესში (“შესრულების ნაკადი” იხ. 4.1.2, გვ. 139).</p>	<p>ნახე შემდეგი შვიდი ნახაზი.</p>	
<p>ნორმალური ნაკადი (Normal Flow)</p>	<p>ნორმალური შესრულების ნაკადი მიუთითებს ნაკადზე, რომელიც იწყება დაწყების ხდომილებიდან და მოქმედებების საშუალებით გრძელდება ალტერნატი-</p>	<p style="text-align: center;">→</p>	




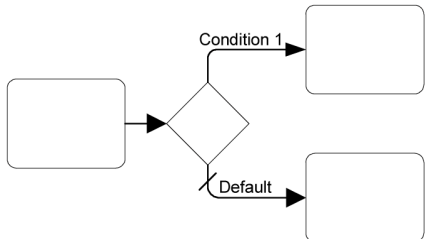
	ული და პარალელური ბილიკების გავლით, სანამ ის დამთავრდება დამთავრების ხდომილებასთან (“ნორმალური ნაკადი” იხ. 4.2.1, გვ. 147).		
არაკონტროლებადი ნაკადი (Uncontrolled flow)	არაკონტროლებადი ნაკადი მიეკუთვნება ნაკადს, რომელზედაც არ მოქმედებს რაიმე პირობა ან არ მიედინება რომელიმე გასასვლელის გავლით. ამის უმარტივესი მაგალითი არის ორი მოქმედების შემაერთებელი მარტივი შესრულების ნაკადი. ეს შეიძლება აგრეთვე იყოს მრავლობითი შესრულების ნაკადი, რომელიც იკრიბება მოქმედებისაკენ ან იშლება მისგან. ყოველი არაკონტროლებადი შესრულების ნაკადისათვის “სიმბოლო” წავა წაყრო ობიექტიდან მიზანი ობიექტისაკენ.		
პირობითი ნაკადი (Conditional flow)	შესრულების ნაკადს შეიძლება ჰქონდეს პირობის გამოსახულებები, რომლებიც ფასდებიან შესრულების მომენტში, რომ განსაზღვრონ იქნება თუ არა ნაკადი გამოყენებული (“შესრულების ნაკადი” იხ. 4.1.2, გვ. 139). <ul style="list-style-type: none"> • თუ პირობითი ნაკადი მოქმედებიდან 		

	<p>არის გამომდინარე, მაშინ შესრულების ნაკადს ხაზის დასაწყისში ექნება პატარა რომბი (იხ. ნახაზი მარჯვნივ).</p> <ul style="list-style-type: none"> • თუ პირობითი ნაკადი გამოდის გასასვლელიდან, მაშინ ხაზს არ ექნება მინი რომბი (იხ. ნახაზი ზედა რიგში). 		
<p>გამოუცხადებლად გამოყენებული ნაკადი (Default flow)</p>	<p>მონაცემებზე დაფუძნებული განსაკუთრებული გადაწყვეტილებებისათვის (Exclusive Decisions) ან თავის თავში შემცველი გადაწყვეტილებებისათვის (Inclusive Decisions) ნაკადის ერთი ტიპი არის გამოუცხადებლად გამოყენებული პირობიანი ნაკადი (“შესრულების ნაკადი” იხ. 4.12, გვ. 139). ეს ნაკადი გამოყენებული იქნება მხოლოდ მაშინ, თუ ყველა სხვა გამომდინარე პირობითი ნაკადი არ არის ჭეშმარიტი შესრულების დროს. ამ შესრულების ნაკადებს ექნებათ დახრილი ხაზი, რომლებიც იქნებიან დამატებული ხაზის დასაწყისში (იხ. ნახაზი მარჯვნივ).</p>		

<p>გამონაკლისი ნაკადი (Exception Flow)</p>	<p>გამონაკლისი ნაკადი წარმოიქმნება პროცესის ნორმალური ნაკადის გარეთ და განთავსებულია შუალედური ხდომილების ახლოს, რომელიც წარმოიქმნება პროცესის შესრულების დროს (“გამონაკლისი ნაკადი” იხ. 4.2.2, გვ. 174)</p>		<p>გამონაკლისი ნაკადი</p>
<p>შეტყობინების ნაკადი (Message Flow)</p>	<p>შეტყობინების ნაკადი გამოიყენება ორ ორგანიზაციას შორის შეტყობინებების ნაკადის საჩვენებლად (გამოსასახავად), რომლებიც მზად არიან მათ გასაგზავნად და მისაღებად (“შეტყობინების ნაკადი” იხ. 4.1.3, გვ. 142). BPMN-ის დიაგრამაში ორი განცალკევებული გუბე წარმოადგენს ორ ორგანიზაციას.</p>		
<p>კომპენსაციის გაერთიანება (Compensation Association)</p>	<p>კომპენსაციის გაერთიანება ჩნდება (გამოიყენება) პროცესის ნორმალური ნაკადის გარეთ და განთავსებულია ხდომილებასთან ახლოს (კომპენსაციის შუალედური ხდომილება), რომელიც გამოიძახება გარიგებაზე უარის თქმით ან კომპენსირების ხდომილებით. (“კომპენსირება გაერთიანება” იხ. 4.3, გვ. 177). გაერთიანების მიზანი უნდა იყოს აღნიშნული როგორც</p>		<p>გაკეთება</p> <p>კომპენსაციის გაერთიანების განადგურება (მოსპობა)</p>

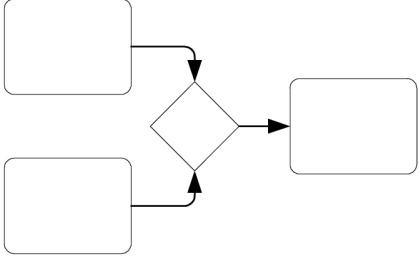
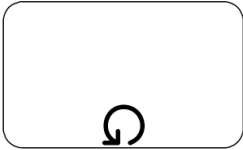
	კომპენსირების მოქმედება.		
მონაცემების ობიექტი (Data Object)	<p>მონაცემების ობიექტები არიან არტიფაქტები, რადგან მათ არა აქვთ რაიმე პირდაპირი ეფექტი პროცესის შესრულების ნაკადზე ან შეტყობინების ნაკადზე, მაგრამ ისინი უზრუნველყოფენ ინფორმაციას თუ რომელი მოქმედებები უნდა იყვნენ შესრულებული და/ან რას აწარმოებენ ისინი (“მონაცემების ობიექტი” იხ. 3.7.2, გვ. 131).</p>		სახელი
განშტოება (Fork)	<p>BPMN გამოიყენებს ტერმინს “განშტოება” (“fork”), რომ წარმოადგინოს ბილიკის გაყოფა ორ ან მეტ პარალელურ ბილიკად (აგრეთვე ცნობილია, როგორც დაგახლეჩვა (AND-Split); “განშტოების ნაკადი” იხ. 4.2.1.1, გვ. 151).</p> <p>ის არის განთავსებული პროცესში, სადაც მოქმედებები შეიძლება იყვნენ შესრულებული ერთდროულად, და არა მიმდევრობით. ისინი არიან ორნაირი:</p> <ul style="list-style-type: none"> • შეიძლება გამოყენებული იყოს მრავლობითი გამომდინარე შესრულების ნაკადი (იხ. მარჯვნივ ზედა ნახაზი). უმეტეს სი- 		

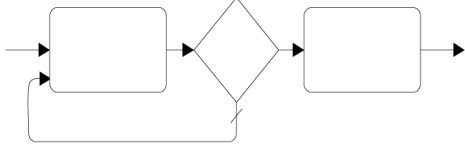

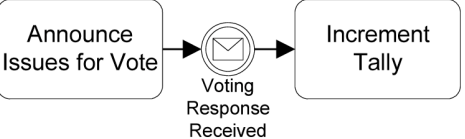
	<p>ტუაციებში “არაკონტროლირებული” ნაკადის ეს წარმოდგენა არის უმჯობესი მეთოდი.</p> <ul style="list-style-type: none"> • შეიძლება გამოყენებული იყოს პარალელური გასასვლელი (Gateway) (იხ. მარჯვნივ ქვედა ნახაზი). ეს გამოიყენება იშვიათად, ჩვეულებრივ, სხვა გასასვლელებთან კომბინაციებში. 		
<p>შეერთება (Join)</p>	<p>BPMN იყენებს ტერმინს “შეერთება” (“join”) ორი ან მეტი პარალელური ბილიკის ერთ ბილიკში გაერთიანების აღსანიშნავად (კიდევ ცნობილია, როგორც დაგაერთიანება (AND-Join) ან სინქრონიზაცია; “ნაკადის შეერთება” იხ. 4.2.12, გვ. 154). პარალელური გასასვლელი გამოიყენება მრავლობითი ნაკადის გაერთიანების საჩვენებლად.</p>	 <p>The diagram illustrates an AND-Join Gateway. It consists of two rectangular boxes on the left, each with an arrow pointing towards a central diamond-shaped gateway containing a plus sign (+). From the right side of the gateway, a single arrow points to a rectangular box on the right, representing the continuation of the process flow after the parallel paths have merged.</p>	
<p>გადაწყვეტილება, განშტოების წერტილი (Decision, Branching Point)</p>	<p>გადაწყვეტილებები არიან გასასვლელები (Gateways) ბიზნეს პროცესში, სადაც მართვის ნაკადს შეუძლია აირჩიოს ერთი ან მეტი ალტერნატიული ბილიკი (“კარებები” (“Gates”) იხ. 3.5.14, გვ. 103).</p>	<p>იხ. შემდეგი ხუთი რიგი.</p>	
<p>განსაკუთრებული</p>	<p>განსაკუთრებული გასას-</p>		

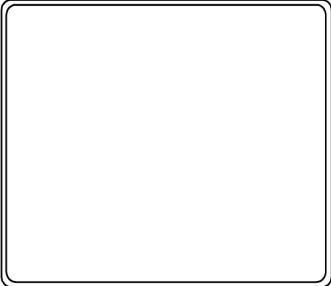
<p>(Exclusive)</p>	<p>ვლელი შეზღუდავს ნაკადს ისე, რომ შესრულების დროს ალტერნატივების სიმრავლიდან მხოლოდ ერთი შეიძლება იყოს ამორჩეული. არსებობს ორი ტიპის განსაკუთრებული გასასვლელი: მონაცემებზე დაფუძნებული (Data-based) და ხდომილებებზე დაფუძნებული (Event-based).</p>	<p>Exclusive Data-Based  or </p> <p>Event-Based </p>	<p>განსაკუთრებული მონაცემებზე დაფუძნებული ან ხდომილებებზე დაფუძნებული</p>
<p>მონაცემებზე დაფუძნებული (Data-based)</p>	<p>ეს გადაწყვეტილება წარმოადგენს განშტოების წერტილს, სადაც ალტერნატივები არიან დაფუძნებული პირობით გამოსახულებებზე, რომლებიც შედიან გამომავალ შესრულების ნაკადში (“მონაცემებზე დაფუძნებული” იხ. 3.5.2.1, გვ. 105). ალტერნატივებიდან მხოლოდ ერთი იქნება ამორჩეული.</p>		<p>პირობა 1 (გამოყენებული) გამოუცხადებლად</p>
<p>ხდომილებებზე დაფუძნებული (Event-based)</p>	<p>ეს გადაწყვეტილება წარმოადგენს განშტოების წერტილს, სადაც ალტერნატივები არიან დაფუძნებული ხდომილებაზე, რომელიც წარმოიშობა პროცესის ამ წერტილში (“ხდომილებაზე დაფუძნებული” იხ. 3.5.2.4, გვ. 110). ეს სპეციფიური ხდომილება, ჩვეულებრივ არის შეტყობინების მიმღები და განსაზღვრავს თუ რომელი ბი-</p>		



	<p>ლიკი იქნება შერჩეული. შეიძლება იყვნენ გამოყენებული სხვა ტიპის ხდომილებები, ისეთი როგორც არის ტაიმერი. მხოლოდ ერთი ალტერნატივა იქნება ამორჩეული. არსებობს შეტყობინების მიღების ორი შესაძლებლობა:</p> <ul style="list-style-type: none"> • შეიძლება იყვნენ გამოყენებული ტიპის მიღების ამოცანები (იხ. მარჯვენა ზედა ნახაზი). • შეიძლება იყვნენ გამოყენებული შეტყობინების ტიპის შუალედური ხდომილებები (იხ. მარჯვენა ქვედა ნახაზი). 		<p>[ტიპის მიღება]</p> <p>[ტიპის მიღება]</p>
<p>თავის თავში შემცველი (Inclusive)</p>	<p>ეს გადაწყვეტილება წარმოადგენს განშტოების წერტილს, სადაც ალტერნატივები არიან დაფუძნებული პირობით გამოსახულებებზე, რომლებიც შედიან გამომავალ შესრულების ნაკადში (“შემცველი გასასვლელები” (“Inclusive Gateways”) იხ. 3.5.3, გვ. 114). რაღაც გაგებით ეს არის ურთიერთ დაკავშირებული დამოუკიდებელი ბინარული (და/არა) გადაწყვეტილებების დაჯგუფება.</p> <p>რამდენადაც ყოველი</p>		

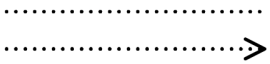
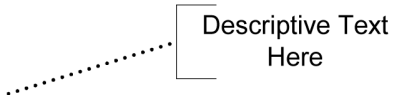

	<p>ბილიკი არის დამოუკიდებელი, ბილიკების ყველა კომბინაცია, ნოლიდან ყველამდე, შეიძლება იყოს აღებული. თუმცა ის შეიძლება ისე იყოს შექმნილი, რომ სულ მცირე ერთი ბილიკი იყოს არჩეული. გამოუცხადებელი (Default) პირობა შეიძლება იყოს გამოყენებული, რომ დაურწმუნდეთ, რომ სულ მცირე ერთი ბილიკი არის არჩეული.</p> <p>არსებობს ამ ტიპის გადაწყვეტილების ორი ვარიანტი:</p> <ul style="list-style-type: none"> • პირველი გამოიყენებს პირობით შესრულების ნაკადის კოლექციას (ნაკრებს). აღნიშნულია მინი რომბით (იხ. მარჯვენა ზედა ნახაზი). • მეორე იყენებს თავის თავში შემცველ გასასვლელს (იხ. მარჯვენა ქვედა ნახაზი). 		<p>პირობა 1</p> <p>პირობა 2</p> <p>პირობა 1</p> <p>პირობა 2</p>
<p>შერწყმა (Merging)</p>	<p>BPMN იყენებს ტერმინს “შერწყმა” (“merge”) ორი ან მეტი ბილიკის ერთ ბილიკში ცალსახად გაერთიანების აღსანიშნავად (აგრეთვე ცნობილია, როგორც ანგაერთიანება (OR-Join); “ნაკადის შერწყმა” იხ. 4.2.1.4,</p>		

	<p>გვ. 159). გასასვლელის ცალსახა შერწყმა გამოიყენება მრავლობითი ნაკადის შერწყმის საჩვენებლად. თუ ყველა შემომავალი ნაკადი არის ალტერნატიული, მაშინ გასასვლელი არ არის საჭირო. ეს ნიშნავს, რომ არა კონტროლირებადი ნაკადი უზრუნველყოფს იგივე ქცევას.</p>		
<p>მარყუჟის კეთება (Looping)</p>	<p>BPMN უზრუნველყოფს პროცესში მარყუჟის კეთების ორ მექანიზმს.</p>	<p>იხ. შემდეგი ორი ნახაზი.</p>	
<p>მოქმედებისათვის მარყუჟის გაკეთება (Activity Looping)</p>	<p>ამოცანების და ქვე-პროცესების ატრიბუტები განსაზღვრავენ არიან თუ არა ისინი გამეორებული ან შესრულებული ერთხელ (“მარყუჟის გაკეთება” იხ. 4.2.1.5, გვ. 163). მარყუჟების ორი ტიპი არსებობს: სტანდარტული (Standard) და მრავალ შემთხვევიანი (Multi-Instance). მარყუჟის კეთების პატარა ინდიკატორი ნაჩვენებია იქნება მოქმედების ქვედა ცენტრში.</p>		
<p>შესრულების ნაკადისათვის მარყუჟის კეთება (Sequence Flow Looping)</p>	<p>მარყუჟები შეიძლება შექმნილი იყოს შესრულების ნაკადის “დინების მიმართულები ზევით” ობიექტებთან შეერთებით. ობიექტი არის “დინების</p>		

	<p>მიმართულები ზევით” თუ ამ ობიექტს აქვს გამომავალი შესრულების ნაკადი, რომელსაც მივყავართ სხვა შესრულების ნაკადების რიგთან, რომელთაგან ბოლოც არის ორიგინალური (პირველსაწყისი) ობიექტის შემომავალი შესრულების ნაკადი.</p>		
<p>მრავალჯერადი მაგალითები (Multiple Instances)</p>	<p>ამოცანების და ქვე-პროცესების ატრიბუტები განსაზღვრავენ არიან თუ არა ისინი განმეორებადი ან შესრულებული ერთხელ (“მარყუჟის გაკეთება” იხ. 4.2.1.5, გვ. 163). პატარა პარალელური ინდიკატორი ნაჩვენებია იქნება მოქმედების ქვედა ცენტრში.</p>		
<p>პროცესის გაწყვეტა (Process Break) (პროცესის მართვის გარეთ რაღაცა ასრულებს პროცესის შეწყვეტას)</p>	<p>პროცესის გაწყვეტა არის ადგილი პროცესში, რომელიც აჩვენებს თუ სად მოხდება მოსალოდნელი დაყოვნება პროცესში (“შუალედური” იხ. 3.3.4, გვ. 56). შუალედური ხდომილება არის გამოყენებული ფაქტიური ქცევის საჩვენებლად (იხ. მარჯვენა ზედა ნახაზი). გარდა ამისა, პროცესის შეწყვეტის არტიფაქტი, მოდელიორის ან მოდელირების ინსტრუმენტის მიერ დაპროექტების შესაბამისად,</p>		<p>გამოაცხადე საკითხები ხმის მისაცემად ხმის მიცემა მიღებული პასუხი ზრდა (ნაზრდი) შეესაბამება</p>

	<p>შეიძლება გაერთიანებული იყოს ხდომილებასთან ნაკადში დაყოვნების ადგილის საჩვენებლად.</p>		
<p>შესრულება (Transaction)</p>	<p>შესრულება (Transaction) არის ქვე-პროცესი, რომელიც მხარდაჭერილია სპეციალრი პროტოკოლით, რაც უზრუნველყოფს ყველა ჩართული ჯგუფის ყოფნას სრულ თანხმობაში და იმას, რომ მოქმედება იყოს დასრულებული ან გაუქმებული ("ქვე-პროცესის ქცევა როგორც შესრულება" იხ. 3.4.2.5, გვ. 86). მოქმედების ატრიბუტები განსაზღვრავენ არის თუ არა მოქმედება შესრულება. ორ-ხაზიანი საზღვრები მიუთითებენ, რომ ქვე-პროცესი არის შესრულება.</p>		
<p>დაბუდებული/ჩადგმული ქვე-პროცესი (ბლოკი ხაზში) (Nested/Embedded Sub-Process (Inline Block))</p>	<p>დაბუდებული (ან ჩადგმული) ქვე-პროცესი არის მოქმედება, რომელიც ყოფს მონაცემების იგივე სიმრავლეს მისი მშობლიური პროცესის მსგავსად ("ჩადგმული ქვე-პროცესი" იხ. 3.4.2.2, გვ. 81). ეს არის დამოუკიდებელი, ახლიდან გამოყენებადი და მშობლიური პროცესიდან მითითებული ქვე-პროცესის საწინააღმდეგო.</p>	<p>დაბუდებული ქვე-პროცესებისათვის არ არსებობს სპეციალური ინდიკატორები.</p>	

	<p>მითითებულ ქვე-პროცესთან საჭიროა მონაცემების გამოტოვება, მაგრამ არა დაბუდებულ ქვე-პროცესთან.</p>		
<p>ჯგუფი (ყუთი ობიექტების ჯგუფის გარშემო იმავე კატეგორიაში) (Group (a box around a group of objects within the same category))</p>	<p>მოქმედებების დაჯგუფება, რომლებიც არიან იმავე კატეგორიაში (“ჯგუფი” იხ. 3.7.4, გვ. 135). დაჯგუფების ეს ტიპი არ მოქმედებს ჯგუფში მოქმედებების შესრულების ნაკადზე. კატეგორიის სახელი დიაგრამაზე გამონიშნულია როგორც ჯგუფის ჭდე. კატეგორიები შეიძლება გამოყენებული იყვნენ დოკუმენტებისათვის ან ანალიზის მიზნით. ჯგუფები არიან ერთი შესაძლო გზა იმის საჩვენებლად, თუ რომელ კატეგორიებში შეიძლება იყვნენ ობიექტები ვიზუალურად ნაჩვენები დიაგრამაზე.</p>		
<p>გვერდის გარეთ შემართებელი (Off-Page Connector)</p>	<p>ძირითადად გამოიყენება ბეჭდვისათვის. ეს ობიექტი აჩვენებს სად ტოვებს შესრულების ნაკადი ერთ გვერდს და შემდეგ იწყება მომდევნო გვერდზე (“შესრულების ნაკადის გადახტომა (გვერდის გარეთ შემართებლები და ობიექტებზე გადასვლა)”) იხ. 4.2.1.8, გვ. 167). შუალედური ხდომილების რგოლი შეიძლება გამოყე-</p>		

	<p>ნებული იყოს როგორც გვერდის გარეთ შემაერთებული.</p>		
<p>გაერთიანება (Association)</p>	<p>გაერთიანება გამოიყენება ინფორმაციის ნაკადის ობიექტებთან გასაერთიანებლად (“გაერთიანება” იხ. 4.1.4, გვ. 144). ტექსტური და გრაფიკული არა ნაკადის ობიექტები შეიძლება იყვნენ გაერთიანებული ნაკადის ობიექტებთან.</p>		
<p>ტექსტური ანოტაცია (მიერთებული გაერთიანებასთან) (Text Annotation (attached with an Association))</p>	<p>ტექსტური ანოტაციები არიან მექანიზმები მოდელირებისათვის, რათა უზრუნველყოს დამატებითი ინფორმაცია BPMN-ს დიაგრამის მკითხველისათვის (“ტექსტური ანოტაცია” იხ. 3.7.3, გვ. 134).</p>		<p>აღწერილობითი ტექსტი აქ</p>
<p>გუბე (Pool)</p>	<p>გუბე წარმოადგენს პროცესში მონაწილეს (“გუბე” იხ. 3.6.2, გვ. 124). ის აგრეთვე არის მოქმედებები, როგორც “პროცესის მონაწილე” (“swimlane”) და გრაფიკული შემცველი სხვა გუბეებიდან მოქმედებების სიმრავლის გამოსაყოფად.</p>		<p>სახელი</p>
<p>ბილიკები (Lanes)</p>	<p>ბილიკი არის ქვე-დაყოფა გუბეში და აფართოებს გუბის მთლიან სიგრძეს როგორც ვერტიკალურად, ასევე ჰორიზონტალურად</p>		

	<p>(“ბილიკი” ის. 3.6.3, გვ. 128). ბილიკები გამოიყენება გუ- ბეში მოქმედებების ორგანი- ზებისა და კატეგორიებში გადასანაწილებლად.</p>	<table border="1"><tr><td data-bbox="1055 233 1088 323">Name</td><td data-bbox="1088 233 1485 276"></td></tr><tr><td data-bbox="1055 276 1088 323">Name</td><td data-bbox="1088 276 1485 323"></td></tr></table>	Name		Name		<p>სახელი</p>
Name							
Name							

2.3. ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში

ტექსტური აღნიშვნის ობიექტები შეიძლება გამოყენებული იყვნენ პროცესის ან პროცესის შიგნით ობიექტების ატრიბუტების შესახებ დამატებითი ინფორმაციის საჩვენებლად.

- ნაკადის ობიექტებს და ნაკადს შეიძლება ჰქონდეს ჭდეები (ანუ მისი სახელი და/ან სხვა ატრიბუტები) განთავსებული ფორმის შიგნით ან მის ზევით ან მის ქვევით, ნებისმიერი მიმართულებით ან განლაგებით, მოდელიორის ან მოდელირების ინსტრუმენტების მწარმოებლის შეხედულებისამებრ.
- ფონები (შეფერებები), რომლებიც არიან გამოყენებული გრაფიკული ელემენტებისათვის შეიძლება იყვნენ თეთრი ან ნათელი.
 - აღნიშვნები შეიძლება იყვნენ გაფართოებული სხვა შემავსებელი ფერების გამოყენებით მოდელიორის ან ინსტრუმენტის მიზნის შესაბამისად (ანუ იმისათვის, რომ ნათელი გახადოს ობიექტის ატრიბუტის მნიშვნელობა).
- ნაკადის ობიექტები და მარკერები შეიძლება იყვნენ ნებისმიერი ზომის, მოდელიორის ან მოდელირების ინსტრუმენტის მიზნებიდან გამომდინარე.
- ხაზები, რომლებიც გამოყენებულია გრაფიკული ელემენტების ხაზისათვის შეიძლება იყვნენ შავი ფერის.
 - აღნიშვნები შეიძლება იყვნენ გაფართოებული სხვა ხაზის ფერების გამოყენებით მოდელიორის ან მოდელირების ინსტრუმენტის მიზნებიდან გამომდინარე (ანუ იმისათვის, რომ ნათელი გახადოს ობიექტის ატრიბუტის მნიშვნელობა).
 - აღნიშვნები შეიძლება იყვნენ გაფართოებული სხვა ხაზების სტილების გამოყენებით მოდელიორის ან მოდელირების ინსტრუმენტის მიზნებიდან გამომდინარე (ანუ იმისათვის, რომ ნათელი გახადოს ობიექტის ატრიბუტის მნიშვნელობა) იმ პირობით, რომ ხაზის სტილი არ უნდა იყოს კონფლიქტში BPMN-ით განსაზღვრულ არც ერთი ხაზის სტილთან. ამრიგად, შესრულების ნაკადის (Sequence Flow), შეტყობინების ნაკადის (Message Flow) და გაერთიანებების (Associations) ხაზების სტილები არ უნდა იყვნენ მოდიფიცირებული.

2.4. ნაკადის ობიექტის შეერთების წესები

შემომავალი შესრულების ნაკადი შეიძლება დაუკავშირდეს ნაკადის ობიექტს ნებისმიერ ადგილას (მარცხივ, მარჯვნივ, ზემოდან ან ქვემოდან). ასევე, გამომავალი შესრულების ნაკადი შეიძლება დაუკავშირდეს ნაკადის ობიექტს ნებისმიერ ადგილას (მარცხივ, მარჯვნივ, ზემოდან ან ქვემოდან). შეტყობინების ნაკადსაც აქვს ეს შესაძლებლობა. BPMN-ს გააჩნის ეს მოქნილობა, მაგრამ ასევე რეკომენდირებულია, რომ მოდელირებმა გამოიყენონ არსებული გამოცდილება და ცოდნა რათა ნაკადის ობიექტები ერთმანეთს დაუკავშირონ ისე, რომ დიაგრამების მკითხველებისათვის ყველაფერი იყოს ნათელი და ადვილი თვალის მისაღწევბ-










ლად. ეს უფრო მეტად მნიშვნელოვანია მაშინ, როცა დიაგრამა შეიცავს შესრულების ნაკადს და შეტყობინების ნაკადს. ამ სიტუაციებში საუკეთესოა აირჩიო შესრულების ნაკადის მიმართულება ან მარცხნიდან მარჯვნივ, ან ზემოდან ქვემოთ და შემდეგ შეტყობინების ნაკადი მიმართო 90°-ნი კუთხით შესრულების ნაკადისაკენ. ასე მიღებული დიაგრამები იქნებიან გასაგებად უფრო ადვილი.




2.4.1. შესრულების ნაკადის წესები

ცხრილ 2.4-ში მოცემულია BPMN-ის ნაკადის ობიექტები და ნახვენებია როგორ შეუძლიათ ამ ობიექტებს ერთიმეორესთან შეერთება შესრულების ნაკადის საშუალებით. სიმბოლო **↗** მიუთითებს, რომ სტრიქონში მოცემულ ობიექტს შეუძლია შეუერთდეს სვეტში მოცემულ ობიექტს. შეერთებების რაოდენობა ობიექტში და ობიექტის გარეთ, რომელიც არის სხვადასხვა კონფიგურაციული დამოკიდებულებების არსი, არ განიხილება აქ. ყოველი ცალკეული ობიექტისათვის შესაბამისი შეერთების წესების შესახებ უფრო მეტი დაწვრილებითი ინფორმაცია მოცემულია შემდეგ თავში.

***შენიშვნა:** თუ ქვე-პროცესი იყო გაფართოებული დიაგრამის შიგნით, ობიექტები ქვე-პროცესის შიგნით შეუძლებელია იყვნენ შეერთებული ამ ქვე-პროცესის გარეთ მყოფ ობიექტებთან. ასევე არ შეუძლია შესრულების ნაკადს გადაკვეთოს გუბის საზღვრები.*


ცხრილი 2.4. შესრულების ნაკადის შეერთების წესები

დან/კენ (From/To)	 დაწყება	 დანერგული ქვე-პროცესი	 ამოცანა (ელემენტარული)	 გასას- ვლელი	 შუალე- დური ხლომილება	 დამთავ- რების ხლომილება
 დაწყება		↗	↗	↗	↗	↗
 დანერგული ქვეპროცესი		↗	↗	↗	↗	↗
 ამოცანა (ელემენტარული)		↗	↗	↗	↗	↗

 გასასვლელი		↗	↗	↗	↗	↗
 შუალედური ხდომილება		↗	↗	↗	↗	↗
 დამთავრების ხდომილება						



შენიშვნა: მხოლოდ ის ობიექტები, რომლებსაც შეუძლიათ ჰქონდეთ შემომავალი და/ან გამომავალი შესრულების ნაკადი არიან ნახვენები ცხრილში. ამრიგად, გუბე, ბილიკი, მონაცემების ობიექტი და ტექსტური აღნიშვნა არ არიან მოცემული ცხრილში.







2.4.2. შეტყობინების ნაკადის წესები

ცხრილ 2.5-ში მოცემულია BPMN-ის მოდელირების ობიექტები და ნახვენებია თუ როგორ შეიძლება იყვნენ შეერთებული (დაკავშირებული) ეს ობიექტები ერთმანეთთან შეტყობინების ნაკადის საშუალებით. სიმბოლო  მიუთითებს, რომ სტრიქონში მითითებული ობიექტი შეიძლება იყოს დაკავშირებული სვეტში მითითებულ ობიექტთან. შეერთებების რაოდენობა ობიექტში და ობიექტის გარეთ, რომელიც არის სხვადასხვა კონფიგურაციული დამოკიდებულებების არსი, არ განიხილება აქ. ყოველი ცალკეული ობიექტისათვის შესაბამისი შეერთების წესების შესახებ უფრო მეტი დაწვრილებითი ინფორმაცია მოცემულია შემდეგ თავში.

შენიშვნა: შეტყობინების ნაკადი არ შეიძლება დაუკავშირდეს ობიექტებს, რომლებიც არიან იმავე გუბის შიგნით.

ცხრილი 2.5. შეტყობინების ნაკადის შეერთების წესები

დან/კენ (From/To)	 შეტყობინება	 გუბე	 დანგრეული ქვეპროცესი	 ამოცანა (ელემენტარული)	 შეტყობინება	 შეტყობინება

 შეტყობინება						
 გუბე	↗	↗	↗	↗	↗	
 დანგრეული ქვეპროცესი	↗	↗	↗	↗	↗	
 ამოცანა (ელემენტარული)	↗	↗	↗	↗	↗	
 შეტყობინება	↗	↗	↗	↗	↗	
 შეტყობინება	↗	↗	↗	↗	↗	

შენიშვნა: მხოლოდ ის ობიექტები, რომლებსაც შეუძლიათ ჰქონდეთ შემომავალი და/ან გამომავალი შეტყობინების ნაკადი არიან ნახვენები ცხრილში. ამრიგად, ბილიკი, გასასვლელი, მონაცემების ობიექტი და ტექსტური აღნიშვნა არ არიან მოცემული ცხრილში.

2.5. ბიზნეს პროცესის დიაგრამის ატრიბუტები

შემდეგ ცხრილში მოცემულია ბიზნეს პროცესის დიაგრამის ატრიბუტების სიმრავლე. **შენიშვნა:** BPMN-ს გააჩნია ატრიბუტების დიდი და მრავალფეროვანი სიმრავლე. გასაგები მიზეზების გამო, წინამდებარე სახელმძღვანელოში წარმოდგენილია მათი ძირითადი რაოდენობა და ფუნქციები.

ცხრილი 2.6. ბიზნეს პროცესის დიაგრამის ატრიბუტები

ატრიბუტები (Attributes)	აღწერა (Description)
იდენტიფიკატორი: ობიექტი (Id: Object)	ეს არის უნიკელური Id, რომელიც განასხვავებს დიაგრამას სხვა დიაგრამებისაგან.

სახელი: სტრიქონი (Name: String)	სახელი არის ატრიბუტი, რომელიც არის დიაგრამის ტექსტური აღწერა.
ვერსია (0-1) : სტრიქონი (Version (0-1) : String)	ეს განსაზღვრავს დიაგრამის ვერსიის ნომერს.
ავტორი (0-1) : სტრიქონი (Author (0-1) : String)	ეს შეიცავს დიაგრამის ავტორის სახელს.
ენა (0-1) : სტრიქონი (Language (0-1) : String)	ეს შეიცავს ენის სახელს, რომელზედაც ტექსტი არის დაწერილი. გამოტოვება ნიშნავს ინგლისურს.
კითხვაენა (0-1) : სტრიქონი (QueryLanguage (0-1) : String)	ენა შეიძლება იყოს უზრუნველყოფილი ისე, რომ დიაგრამაში გამოყენებული კითხვების სინტაქსის გაგება შეიძლებოდეს.
შექმნათარიღი (0-1) : თარიღი (CreationDate (0-1) : Date)	ეს განსაზღვრავს თარიღს, როდესაც დიაგრამა იყო შექმნილი.
მოდულიკაციათარიღი (0-1) : თარიღი (ModificationDate (0-1) : Date)	ეს განსაზღვრავს თარიღს, როდესაც დიაგრამა იყო ბოლოს მოდიფიცირებული.
გუბეები (1-n) : გუბე (Pools (1-n) : Pool)	ბიზნეს პროცესების დიაგრამა (BPD) შეიცავს ერთ ან მეტ გუბეს. ერთ-ერთი გუბის საზღვარი შეიძლება იყოს უხილავი (განსაკუთრებით თუ დიაგრამაში არის მხოლოდ ერთი გუბე).
დოკუმენტაცია (0-1) : სტრიქონი (Documentation (0-1) : String)	მოდელიორს შეუძლია დაამატოს დამატებითი ტექსტური დოკუმენტი დიაგრამის შესახებ.

2.6. პროცესები

პროცესი არის მოქმედება შესრულებული კომპანიების ან ორგანიზაციების შიგნით ან გარეთ. BPMN-ში პროცესი არის გამოსახული როგორც ნაკადის ობიექტების გრაფიკი, რომლებიც წარმოადგენენ სხვადასხვა მოქმედებების სიმრავლეს და მართვები, რომლებიც მოსდევნ მათ. პროცესის კონცეფცია (ცნება) თვისობრივად არის იერარქიული. პროცესები შეიძლება განსაზღვრული იყვნენ ნებისმიერ დონეზე, ფართო საწარმოო პროცესებიდან, ცალკეული პიროვნებების მიერ შესრულებულ პროცესებამდე. დაბალი დონის პროცესები შეიძლება დაჯგუფებული იქნან ერთად ბიზნესის საერთო მიზნის მისაღწევად.

შეგნიშნოთ, რომ BPMN განსაზღვრავს ტერმინს “პროცესი” სპეციფიურად და განსაზღვრავს “ბიზნეს პროცესს” უფრო ზოგადად, როგორც მოქმედებების სიმრავლეს, რომლებიც არიან შესრულებული ორგანიზაციაში ან ორგანიზაციებს შორის. ყოველ პროცესს შეიძლება ჰქონდეს მისი საკუთარი ქვე-პროცესი და შეიძლება შეცული იყოს გუბის შიგნით. ინდივიდუალური პროცესები შეიძლება იყვნენ დამოუკიდებელი შესრულების ნაკადის ტერმინებში, მაგრამ შეიძლება ჰქონდეთ მათი დამაკავშირებელი შეტყობინების ნაკადი.

2.6.1. ატრიბუტები

შემდეგი ცხრილი შეიცავს პროცესის ატრიბუტების სიმრავლეს, რომელიც აფართოებს BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს.

ცხრილი 2.7. პროცესის ატრიბუტები

ატრიბუტები (Attributes)	აღწერა (Description)
სახელი : სტრიქონი (Name : String)	სახელი არის ატრიბუტი, რომელიც არის ობიექტის ტექსტური აღწერა.
პროცესის ტიპი (ცარიელი კერძო აბსტრაქტული თანამშრომლობა) ცარიელი : სტრიქონი (ProcessType (None Private Abstract Collaboration) None : String)	პროცესის ტიპი არის ატრიბუტი, რომელიც უზრუნველყოფს ინფორმაციას თუ რომელი დაბალი დონის ენა არის გამოყენებული გუბეში ნახაზების გასაკეთებლად. გამოუცხადებლად პროცესის ტიპი არის ცარიელი (ან განუსაზღვრელი).
სტატუსი (ცარიელი მზა აქტიური გაუქმებული გადაგდება გადაგდებული დასრულება დასრულებული) ცარიელი : სტრიქონი (Status (None Ready Active Cancelled Aborting Aborted Completing Completed) None : String)	პროცესის სტატუსი არის განსაზღვრული, როდესაც პროცესი სრულდება პროცესის ძრავით (ამძრავით). პროცესის სტატუსი შეიძლება გამოყენებული იყოს დანიშნულების გამოსახულებების შიგნით.
გრაფიკული ელემენტები (0-n) : ობიექტი (GraphicalElements (0-n) : Object)	ატრიბუტი “გრაფიკული ელემენტები” იდენტიფიცირებას უკეთებს ყველა ობიექტს (ანუ ხლომელებს, მოქმედებებს, გასასვლელებს და არტიფაქტებს), რომლებიც შეცვლილი არიან ბიზნეს პროცესში.
შემსრულებლები (0-n) : სტრიქონი (Performers (0-n) : String)	შეიძლება შემოტანილი იყოს ერთი ან მეტი შემსრულებელი. ატრიბუტი “შემსრულებლები” განსაზღვრავს რესურსს, რომელიც იქნება პასუხისმგებელი პროცესისათვის. შემსრულებლების ჩანაწერი შეიძლება იყოს განსაზღვრული პიროვნების, ჯგუფის, ორგანიზაციის როლის ან ადგილის, ან ორგანიზაციის ფორმაში.
დანიშნულებები (0-n) : დავალება (Assignments (0-n) : Assignment)	ობიექტისათვის შეიძლება იყოს შექმნილი ერთი ან მეტი დანიშნულების გამოსახულება. დანიშნულება შესრულება ისე, როგორც არის განსაზღვრული ატრიბუტით “მიანიჭედრო” (“AssignTime”).
თვისებები (0-n) : თვისება (Properties (0-n) : Property)	მოდელიორის მიერ განსაზღვრული თვისებები შეიძლება იყოს დამატებული პროცესისათვის. ეს თვისებები პროცესის მიმართ არიან “ადგილობრივი” (“local”). ყველა ამოცანას, ქვე-პროცესის ობიექტებს და ქვე-პროცესებს, რომლებიც არიან შენიღბულები, ექნებათ ამ თვისებების წოდება. ამ თვისებების სრულად ჩაწერილი

	<p>სახელი არის “<პროცესის სახელი>.<თვისების სახელი>” (მაგალითად, “დამატე მყიდველი.მყიდველის სახელი”). თუ პროცესი არის შენიღბული სხვა პროცესის შიგნით, მაშინ სრულად წარმოდგენილი სახელის წინ იქნება აგრეთვე მშობელი პროცესის სახელი.</p>
<p>შემავალისიმრავლეები (0-n) : შემავალისიმრავლე (InputSets (0-n) : InputSet)</p>	<p>ატრიბუტი “შემავალისიმრავლეები” განსაზღვრავს პროცესში შესატან მონაცემებზე მოთხოვნებს. შეიძლება იყოს განსაზღვრული ნოლი ან მეტი “შემავალისიმრავლეები”. ყოველი შემავალი სიმრავლე საკმარისია პროცესის შესრულებისათვის (თუ ის პირველად იყო მყისიერად წარმოდგენილი შესაბამისი სიგნალით, მოსული შემომავალი შესრულების ნაკადიდან).</p>
<p>გამომავალისიმრავლეები (0-n) : გამომავალისიმრავლე (OutputSets (0-n) : OutputSet)</p>	<p>ატრიბუტი “გამომავალისიმრავლეები” განსაზღვრავს მონაცემებზე მოთხოვნებს პროცესიდან გამოსასვლელისათვის. შეიძლება განსაზღვრული იყოს ნოლი ან მეტი “გამომავალისიმრავლეები”. პროცესის დამთავრებისას მხოლოდ ერთი “გამომავალისიმრავლე” შეიძლება იყოს გამომუშავებული. ის პროცესის შესრულებამდე განსაზღვრავს სიმრავლეს რომელიც იქნება გამომუშავებული. თუმცა, ატრიბუტი “IORules” შეიძლება მიუთითებდეს “OutputSet” და “InputSet” შორის ურთიერთკავშირზე.</p>
<p>სპეციალური არაჰეშმარიტი : ბულის (AdHoc False : Boolean)</p>	<p>სპეციალური (AdHoc) არის ბულის ატრიბუტი, რომელიც არის გამოტოვებული თუ არა ჰეშმარიტია. ეს მიუთითებს იმაზე არის თუ არა პროცესი სპეციალური. სპეციალურ პროცესში მოქმედებები არ არიან კონტროლირებადი ან მოწესრიგებული სპეციალური მიმდევრობით. მათი შესრულება განსაზღვრულია მოქმედებების შესრულებით. თუ დაყენებულია ჰეშმარიტზე, მაშინ AdHoc მარკერი განლაგდება პროცესის ქვედა ცენტრში ან ქვე-პროცესის ფორმაში სპეციალური პროცესისათვის.</p>
<p>[AdHoc = მხოლოდ ჰეშმარიტი] AdHocმოწესრიგება (0-1) (მიმდევრობითი პარალელური) პარალელური : სტრიქონი ([AdHoc = True only] AdHocOrdering (0-1) (Sequential Parallel) Parallel : String)</p>	<p>თუ პროცესი არის სპეციალური (AdHoc ატრიბუტი არის ჰეშმარიტი), მაშინ ატრიბუტი AdHocმოწესრიგება იქნება ჩართული. ეს ატრიბუტი განსაზღვრავს პროცესის შიგნით მოქმედებები შეიძლება იყოს შესრულებული პარალელურად თუ შეიძლება იყოს შესრულებული მიმდევრობით. <i>გამოუცხადებელზე დაყენება არის პარალელური და მიმდევრობითის დაყენება</i></p>

	არის შეზღუდვა შესრულებაზე, რაც შეიძლება მოთხოვნილი იყოს დანაწილებული რესურსების გამო.
[AdHoc = მხოლოდ ჭეშმარიტი] AdHoc დამთავრებაპირობა (0-1) : გამოსახულება ([AdHoc = True only] AdHocCompletionCondition (0-1) : Expression)	თუ პროცესი არის სპეციალური (AdHoc ატრიბუტი არის ჭეშმარიტი), მაშინ ატრიბუტი AdHocდამთავრებაპირობა იქნება ჩართული. ეს ატრიბუტი განსაზღვრავს პროცესის დამთავრების პირობებს.

თავი 3. ბიზნეს პროცესის დიაგრამის გრაფიკული ობიექტები

ეს თავი აღწერს გრაფიკულ გამოსახულებებს და ბიზნეს პროცესის დიაგრამის გრაფიკული ელემენტების ქცევის სემანტიკებს.

3.1. BPMN-ის ელემენტების საერთო ატრიბუტები

შემდეგ ცხრილში მოცემულია BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლე (გრაფიკული ელემენტები და დამხმარე ელემენტები).

ცხრილი 3.1. BPMN-ის ელემენტების საერთო ატრიბუტები

ატრიბუტები	აღწერა
იდენტიფიკატორი: ობიექტი (Id: Object)	ეს არის უნიკალური იდენტიფიკატორი, რომელიც გამოარჩევა ობიექტს სხვა ობიექტებიდან დიაგრამის შიგნით.
კატეგორიები (0-n) : კატეგორია (Categories (0-n) : Category)	მოდელიორს შეუძლია დაამატოს ერთი ან მეტი განსაზღვრული კატეგორია, რომლებსაც აქვთ გამოყენების მიერ განსაზღვრული სემანტიკები და რომლებიც შეიძლება გამოყენებული იქნენ მოხსენების და ანალიზის მიზნით.
დოკუმენტაცია (0-1) : სტრიქონი (Documentation (0-1) : String)	მოდელიორს შეუძლია დაამატოს ობიექტის შესახებ ტექსტ დოკუმენტი.

ეს ატრიბუტები გამოყენებული არიან გრაფიკული ელემენტებისათვის [ნაკადი ობიექტები (პარაგრაფი 3.2, “ნაკადი ობიექტის საერთო ატრიბუტები” (“Common Flow Object Attributes”) გვ. 41), შემაერთებული (დამაკავშირებელი) ობიექტები (პარაგრაფი 4.1, “გრაფიკული შემაერთებული ობიექტები”, (“Graphical Connecting Objects”), გვ. 138), პროცესის მონაწილენი (Swimlanes) (პარაგრაფი 3.6, “პროცესის მონაწილენი (გუბები და ბილიკები)”, (“Swimlanes (Pools and Lanes)”), გვ. 123) და არტიფაქტები (პარაგრაფი 3.7, “არტიფაქტები” (“ნიმუშები”) (“Artifacts”), გვ. 130)] და დამხმარე ელემენტებისათვის.

3.2. ნაკადის ობიექტის საერთო ატრიბუტები

შემდეგი ცხრილში მოცემულია BPMN-ის ნაკადის ობიექტების (ხდომილებები, მოქმედებები და გასასვლელები) საერთო ატრიბუტების სიმრავლე, რომლებიც აფართოებენ BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.1).

ცხრილი 3.2. ნაკადის ობიექტის საერთო ატრიბუტები

ატრიბუტები	აღწერა
სახელი : სტრიქონი (Name : String)	სახელი არის ატრიბუტი, რომელიც არის ობიექტის ტექსტური აღწერა.
დანიშნულებები (0-n) : დავალება (Assignments (0-n) : Assignment)	ობიექტისათვის შეიძლება იყოს შექმნილი ერთი ან მეტი აღწერის გამოსახულება. მოქმედებებისათვის (ამოცანა, ქვე-პროცესი და პროცესი) დანიშნულებები შეიძლება შესრულდნენ ისე, როგორც განსაზღვრულია “მიანიჭედრო” (“AssignTime”) ატრიბუტით.

3.3. ხდომილებები

ხდომილება არის რაღაც, რაც “ხდება” ბიზნეს პროცესის მიმდინარეობისას. ხდომილებები მოქმედებენ პროცესის ნაკადზე და როგორც წესი აქვთ გამომწვევი მიზანი ან ზემოქმედება (ბიძგი). ტერმინი “ხდომილება” არის საკმაოდ ზოგადი და მოიცავენ მრავალ საგანს ბიზნეს პროცესში. მოქმედების დასაწყისი, მოქმედების დამთავრება, დოკუმენტის მდებარეობის ცვლილება, შეტყობინება, რომელიც მოდის, და ა.შ. ყველა შეიძლება იყოს განხილული ხდომილებებად. თუმცა, BPMN-ი ზღუდავს ხდომილებების გამოიყენებას, შეიცავს მხოლოდ იმ ტიპის ხდომილებებს, რომლებიც მოქმედებენ მიმდევრობაზე ან პროცესის მოქმედებებისათვის შესაბამისი დროის შერჩევაზე. BPMN-ი ახდენს ხდომილებების განაწილებას კატეგორიების სამ მთავარ ტიპად: დაწყება, შუალედური და დამთავრება (Start, Intermediate, and End).

დაწყების და უმრავლეს შუალედურ ხდომილებებს აქვთ “გადამრთველები” (“Triggers”), რომლებიც განსაზღვრავენ ხდომილების მიზეზს. არსებობს ხდომილებების გადართვის მრავალი გზა (“დაწყება ხდომილების გადამრთველები” (“Start Event Triggers”) იხ. 3.3.2.1, გვ. 46 და “შუალედური ხდომილების გადამრთველები” (“Intermediate Event Triggers”) იხ. 3.3.4.1, გვ. 57). დამთავრების ხდომილებებს შეუძლიათ განსაზღვრონ “შედეგი”, რომელიც არის შესრულების ნაკადის დამთავრების მიმდევრობა. არსებობს შედეგების მრავალი ტიპი, რომლებიც შეიძლება იყვნენ განსაზღვრული (“დამთავრების ხდომილების შედეგები” (“End Event Results”) იხ. 3.3.3.1, გვ. 51).

ყველა ხდომილებას აქვს ერთი და იგივე ფორმის გრაფიკული წარმოდგენა - პატარა წრე. ხაზის სხვადასხვა სტილი, როგორც ნაჩვენებია ქვემოთ, განასხვავებს ნაკადის ხდომილების სამ ტიპს. ყველა ხდომილებას აგრეთვე აქვს გახნილი ცენტრი, ასე რომ BPMN-ით ან მოდელიორის მიერ განსაზღვრული პიქტოგრამები (icons) შეიძლება იყვნენ ჩართული ფორმაში გადამრთველის (Trigger) ან ხდომილების შედეგის იდენტიფიცირების გასაადვილებლად.

3.3.1. ხდომილების საერთო ატრიბუტები

შემდეგ ცხრილში მოცემულია სამივე ტიპის ხდომილებებისათვის საერთო ატრიბუტების სიმრავლე, რომლებიც აფართოვებენ ნაკადის ობიექტის საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.2).

ცხრილი 3.3. ხდომილების საერთო ატრიბუტები

ატრიბუტები	აღწერა
ხდომილებათი (დაწყება დამთავრება შუალედური) დაწყება : სტრიქონი (EventType (Start End Intermediate) Start : String)	ხდომილებათი უნდა იყოს დაწყების, დამთავრების ან შუალედური ტიპის.

3.3.2. დაწყება

როგორც გამომდინარეობს სახელწოდებიდან, დაწყების ხდომილება მიუთითებს სად დაიწყება მოცემული პროცესი. შესრულების ნაკადის ტერმინებში, დაწყების ხდომილება იწვევს პროცესის ნაკადს და ამდენად არა აქვს რაიმე შემომავალი შესრულების ნაკადი – არც ერთ შესრულების ნაკადს არ შეუძლია დაუკავშირდეს დაწყების ხდომილებას.

დაწყების ხდომილებას აქვს იგივე ძირითადი ფორმა რაც აქვს შუალედურ ხდომილებას და დამთავრების ხდომილებას. ეს არის გახსნილი წრე ცენტრით ისე, რომ მარკერებს შეეძლოთ განლაგება წრის შიგნით ხდომილების ცვალებადობის მისათითებლად.

- დაწყების ხდომილება არის წრე, რომელიც შეიძლება იყოს დახაზული ერთი წვრილი ხაზით (იხ. ნახ. 3.1).
 - დაწყების ხდომილებისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების მიხედვით (“ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში” გვ. 33) შემდეგი გამონაკლისით:
 - ხაზის სისქე უნდა დარჩეს წვრილი ისე, რომ შესაძლებელი იყოს დაწყების ხდომილების გარჩევა შუალედური და დამთავრების ხდომილებებისაგან.



ნახ. 3.1. დაწყების ხდომილება.

წინამდებარე ტექსტში ჩვენ განვიხილავთ შესრულების ნაკადის მიმდინარეობას (შესრულებას) პროცესში. მსჯელობის გასაადვილებლად გამოვიყენებთ “სიმბოლოს” (“Token”) კონცეფციას (ცნებას), რომელიც გადაკვეთს (ანუ გაივლის)

შესრულების ნაკადს და გაივლის პროცესის ნაკადის ობიექტებში. პროცესის ქცევა შეიძლება აღწერილი იყოს სიმბოლოს ბილიკის(ების) თვალის მიდევნებით მთელ პროცესში. სიმბოლოს ექნება უნიკალური იდენტიფიკატორი, რომელსაც ეწოდება “სიმბოლოId” (“TokenId”) სიმრავლე, რომელიც შეიძლება იყოს გამოყენებული მრავლობითი სიმბოლოების გამოსარჩევად, რომლებიც შეიძლება არსებობდნენ, მაგალითად, ანალოგიური (ასეთივე) პროცესის გამო ან სიმბოლოს გაყოფის გამო პარალელური დამუშავებისათვის, მაგალითად, მარტივი პროცესის შიგნით. სიმბოლოს პარალელური გაყოფა ქმნის “სიმბოლოId” (“TokenId”) სიმრავლის დაბალ დონეს. “სიმბოლოId”-ს (“TokenId”) ყველა დონეების სიმრავლე გამოარჩევს (იდენტიფიცირებას უკეთებს) სიმბოლოს.

დაწყების ხდომილება წარმოქმნის სიმბოლოს, რომელიც საბოლოო ჯამში უნდა იყოს გამოყენებული (შთანთქმული) დამთავრების ხდომილებასთან (რომელიც შეიძლება იყოს არა ცხადი, თუ გრაფიკულად არ არის ნაჩვენები). პროცესის შესრულების ნაკადის ქსელში, გასასვლელებში და მოქმედებებში უნდა იყოს სიმბოლოების ბილიკებისათვის თვალის მიდევნების შესაძლებლობა. ნორმალური შესრულების ნაკადის მიმდინარეობის პროცესში არ უნდა არსებობდეს რაიმე ნაგულისხმევი ნაკადი (ე.ი. ყოველთვის უნდა იყოს ერთ-ერთი შესრულების ნაკადი ან გრაფიკული ინდიკატორი, ისეთი, როგორც შუალედური ხდომილებაა, სიმბოლოების ყველა პოტენციური ბილიკების საჩვენებლად). ჩადგმული ნაკადის მაგალითი არის როდესაც სიმბოლო მიდის გასასვლელთან (Gateway), მაგრამ არც ერთი გასასვლელი არ არის მოქმედი, მაშინ სიმბოლო (დაფარულად) მივა პროცესის ბოლოში, რომელიც შეგვხვდება მოდელირების რაიმე აღნიშვნებით. სიმბოლოები შეიძლება აგრეთვე მიმართული იყოს გამონაკლისი შუალედური ხდომილებების საშუალებით, რომლებიც სრულდებიან მოქმედების იძულებითი დამთავრების მაგვარად.

შენიშვნა: სიმბოლო არ კვეთს შეტყობინების ნაკადს, რადგან ის არის შეტყობინება, რომელიც არის გატარებული ამ ნაკადის ქვემოთ (როგორც სახელი გულისხმობს).

დაწყების ხდომილების სემანტიკები შეიცავენ:

- დაწყების ხდომილება არის ფაკულტატიური (არა სავალდებულო): ზედა დონის პროცესს ან გაფართოვებულ ქვე-პროცესს შეიძლება (არ არის აუცილებელი) ჰქონდეს დაწყება ხდომილება.

შენიშვნა: ბიზნეს პროცესების დიაგრამას (BPD) შეიძლება ჰქონდეს ერთზე მეტი პროცესის დონე (ე.ი. ის შეიძლება შეიცავდეს გაფართოვებულ ქვე-პროცესს). დაწყების და დამთავრების ხდომილებები გამოიყენებიან დამოუკიდებლად დიაგრამის ყოველი დონისათვის.

- თუ პროცესი არის რთული და/ან საწყისი პირობები არ არის გასაგები, მაშინ რეკომენდირებულია, რომ დაწყების ხდომილება იყოს გამოყენებული.
- თუ დაწყების ხდომილება არ არის გამოყენებული, მაშინ ჩადგმული პროცესის დაწყების ხდომილებას არ ექნება გადამრთველი (Trigger).

- თუ გვაქვს დამთავრების ხდომილება, მაშინ უნდა გვქონდეს მინიმუმ ერთი დაწვევის ხდომილება.
- თუ გამოყენებულია დაწვევის ხდომილება, მაშინ არ უნდა გვქონდეს სხვა ნაკადის ელემენტები, რომლებსაც არა აქვთ შემომავალი შესრულების ნაკადი. ყველა სხვა ნაკადის ობიექტები უნდა იყვნენ სულ ცოტა ერთი შესრულების ნაკადის მიზანი.
 - ამის გამონაკლისია მოქმედებები, რომლებიც განსაზღვრულია როგორც მაკომპენსირებელი მოქმედებები (აქვთ მაკომპენსირებელი მარკერები). მაკომპენსირებელ მოქმედებებს არ უნდა ჰქონდეთ რაიმე შემომავალი შესრულების ნაკადი, მაშინაც კი თუ არსებობს დაწვევის ხდომილება პროცესის დონეზე. იხ. პარაგრაფი 4.3, “კომპენსაციის გაერთიანება” (“Compensation Association”), გვ. 177, მეტი ინფორმაციისათვის კომპენსაციის მოქმედებებზე.
 - ამის გამონაკლისია შუალედური ხდომილება, რომელიც შეიძლება იყოს შემომავალი შესრულების ნაკადის გარეშე (როცა დაუკავშირდება (მიედგმება) (attached) მოქმედების საზღვარს).
- თუ არ არის გამოყენებული დაწვევის ხდომილება, მაშინ ყველა ნაკადის ობიექტები, რომლებსაც არა აქვთ შემავალი შესრულების ნაკადი (ე.ი. არ არიან შესრულების ნაკადის მიზანი) იქნება მყისიერად შესრულებული, როდესაც პროცესი არის გამყისიერებული. აქ არის ერთი დაშვება, რომ არსებობს მხოლოდ ერთი ნაგულისხმევი დაწვევის ხდომილება, რაც ნიშნავს, რომ დაწვევის ყველა ნაკადის ობიექტები დაიწვება ერთი და იმავე დროს.
 - ამის გამონაკლისი არიან მოქმედებები, რომლებიც არიან განსაზღვრული, როგორც არსებული კომპენსაციის მოქმედებები (აქვს კომპენსაციის მარკერი). კომპენსაციის მოქმედებები არ არიან განიხილული ნორმალური ნაკადების ნაწილად და არ უნდა იყვნენ შესრულებული მყისიერად, როდესაც პროცესი არის გამყისიერებული.
- პროცესის მოცემული დონისათვის შეიძლება არსებობდნენ მრავლობითი დაწვევის ხდომილებები.
 - ყოველი დაწვევა ხდომილება არის დამოუკიდებელი ხდომილება. ეს ნიშნავს, რომ პროცესის მაგალითი იქნება გენერირებული, როდესაც დაწვევის ხდომილება იქნება გადართული (Triggered).
 - თუ პროცესი გამოყენებულია როგორც ქვე-პროცესი და არსებობენ მრავლობითი ცარიელი დაწვევის ხდომილებები, მაშინ როცა ნაკადი არის გადაცემული მშობელი პროცესიდან ქვე-პროცესისათვის, ქვე-პროცესის დაწვევის ხდომილებებიდან მხოლოდ ერთი იქნება გადართული (Triggered). შესრულების ნაკადის მიზანი Ref (TargetRef) ატრიბუტი შემავალი ქვე-პროცესის ობიექტში შეიძლება იყოს გაფართოებული შესაბამისი დაწვევის ხდომილების იდენტიფიკაციისათვის (როგორც არის განსაზღვრული ქვე-პრო-

ცესების “შესრულების ნაკადის შეერთებები” (“Sequence Flow Connections”) იხ. 3.4.2.6, გვ. 88).

შენიშვნა: პროცესის ქცევა შეიძლება იყოს გასაგებად ძნელი თუ არსებობენ მრავლობითი დაწყების ხდომილებები. რეკომენდირებულია, რომ ეს თავისებურება გამოყენებული იყოს ზომიერად და რომ მოდელიორი აცნობიერებდეს, რომ დიაგრამის სხვა მკითხველებისათვის არსებობს დიაგრამის მიზნის გაგების სიძნელე.

როდესაც დაწყების ხდომილებისათვის გეხვდება გადამრთველი (Trigger), მაშინ ხდება ახალი პროცესის დაწყება და სიმბოლო (Token) იქნება გენერირებული ამ ხდომილებიდან გამომავალი ყოველი შესრულების ნაკადისათვის. ყოველი სიმბოლოსათვის იქნება შექმნილი სიმბოლოID (TokenId) სიმრავლე ისე, რომ შესაძლო იყოს ჯგუფში სიმბოლოების რაოდენობის და იმ ფაქტის იდენტიფიცირება, რომ ყველა სიმბოლო არის ერთი და იგივე პარალელური განშტოებიდან (Fork). ეს სიმბოლოები დაიწყებენ თავიანთ ნაკადს და არ დაელოდებიან, რომ სხვა რომელიმე დაწყების ხდომილება იყოს გადართული.

პროცესის დაწყების წინ თუ არსებობს ერთზე მეტი ხდომილების მოხდენის შესაძლებლობა (ანუ ორი შეტყობინების დაწყება არის მოთხოვნილი), მაშინ დაწყების ხდომილებები უნდა მიედინებოდნენ ერთი და იგივე მოქმედებისაკენ პროცესის შიგნით. მოქმედების ატრიბუტებმა უნდა დააზუსტონ, როდის უნდა დაიწყოს მოქმედება. თუ ატრიბუტები აზუსტებენ, რომ მოქმედება უნდა დაელოდოს ყველა შემომავალს (inputs), მაშინ ყველა დაწყების ხდომილება იქნება გადართული პროცესის დაწყების წინ (მოქმედების ატრიბუტების შესახებ მეტი ინფორმაციისათვის იხ. 3.3.2.2, “ატრიბუტები” (“Attributes”), გვ. 47 (ქვე-პროცესებისათვის) და 3.4.3.1, “ატრიბუტები” (“Attributes”) გვ. 91 (ამოცანებისათვის)). გარდა ამისა, კორელაციის მაქანიზმი იქნება მოთხოვნილი ისე, რომ სხვადასხვა გადართული დაწყების ხდომილებები იქნებიან გამოყენებული იგივე პროცესისათვის.







3.3.2.1. დაწყების ხდომილების გადამრთველები

ბიზნეს პროცესების დაწყების ბევრი გზა არსებობს. BPMN-ში არსებობს დაწყების ხდომილების ექვსი ტიპი: ცარიელი, შეტყობინება, ტაიმერი, პირობითი, სივნალი და მრავლობითი.

ცხრილი 3.4-ში მოცემულია გადამრთველის ტიპები და გრაფიკული მარკერები (აღნიშვნები), რომლებიც გამოიყენებიან ყოველი მათგანისათვის.

ცხრილი 3.4. დაწყების ხდომილების ტიპები

გადამრთველი (Trigger)	აღწერა	აღნიშვნა (Marker)
ცარიელი (None)	მოდელიორი არ აჩვენებს ხდომილების ტიპს. ის აგრეთვე გამოიყენება ქვე-პროცესისათვის, რომელიც იწყება	

	როდესაც ნაკადი არის გადართული მისი მშობელი პროცესის მიერ.	
შეტყობინება (Message)	შეტყობინება მოდის მონაწილისაგან და იწვევს პროცესის დაწყებას.	
ტაიმერი (Timer)	შეიძლება იყოს დაყენებული სპეციფიური დრო-თარიღი ან სპეციფიური ციკლი (მაგალითად, ყოველ ორშაბათს დილის 9 საათზე), რომელიც იწვევს პროცესის დაწყებას.	
პირობითი (Conditional)	ამ ტიპის ხდომილება გამოიყენება, როდესაც პირობა, მაგალითად ისეთი, როგორიცაა “ტემპერატურა 330C ზევით” ხდება ჭეშმარიტი. ხდომილებისათვის პირობაგამოსახულება (Condition Expression) უნდა გახდეს ყალბი (არა ჭეშმარიტი) (false) და შემდეგ ჭეშმარიტი (true) მანამ სანამ ხდომილება შეიძლება იყოს კვლავ გადართული (triggered).	
სიგნალი (Signal)	სხვა პროცესისაგან გადაცემული სიგნალი მოდის და იწვევს პროცესის დაწყებას. შევნიშნოთ, რომ სიგნალი არ არის შეტყობინება, რომელსაც აქვს სპეციალური მიზანი შეტყობინებისათვის. მრავლობით პროცესებს შეიძლება ჰქონდეთ დაწყების ხდომილებები, რომლებიც არიან გადართული ერთი და იგივე გადაცემული სიგნალიდან.	
მრავლობითი (Multiple)	ეს ნიშნავს, რომ არსებობს პროცესის გადართვის (triggering) მრავალი გზა. მხოლოდ ერთი მათგანი იქნება მოთხოვნილი პროცესის დასაწყებად. დაწყების ხდომილების ატრიბუტები განსაზღვრვენ გადამრთვლების სხვადასხვა ტიპებიდან რომელია გამოყენებული.	

3.3.2.2. ატრიბუტები

ცხრილში 3.5 მოცემულია დაწყების ხდომილების ატრიბუტების სიმრავლე, რომელიც აფართოვებს ხდომილების საერთო ატრიბუტების სიმრავლეს.

ცხრილი 3.5. დაწყების ხდომილების ატრიბუტები.

ატრიბუტები	აღწერა
<p>გადამრთველი (0-n) : ხდომილებადეტალი (Trigger (0-n) : EventDetail)</p>	<p>გადამრთველი (ხდომილებადეტალი) (Event-Detail) არის ატრიბუტი, რომელიც განსაზღვრავს დაწყების ხდომილების მოსალოდნელი გადამრთველის ტიპს. ხდომილებადეტალის ტიპების სიმრავლიდან მხოლოდ ოთხი შეიძლება იყოს გამოყენებული დაწყების ხდომილებისათვის: <i>შეტყობინება, ტაიმერი, პირობითი და სიგნალი</i> (Message, Timer, Conditional, and Signal) (იხ. ცხრილი 3.4).</p> <p>თუ არ არის განსაზღვრული ხდომილებადეტალი (EventDetail), მაშინ ეს განიხილება როგორც ცარიელი (None) დაწყების ხდომილება და ხდომილებას არ ექნება შიდა მარკერი (იხ. ცხრილი 3.4).</p> <p>თუ განსაზღვრულია ერთზე მეტი ხდომილებადეტალი (EventDetail), ეს განიხილება როგორც მრავლობითი დაწყების ხდომილება და ხდომილებას ექნება ხუთკუთხა შიდა მარკერი (იხ. ცხრილი 3.4).</p>

3.3.2.3. შესრულების ნაკადის შეერთებები (Sequence Flow Connections)

ობიექტების მთელი სიმრავლე მოცემულია პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34. იქვე ნაჩვენებია თუ როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

- დაწყების ხდომილება არ უნდა იყოს შესრულების ნაკადის მიზანი; მას არ უნდა ჰქონდეს შემომავალი შესრულების ნაკადი.
 - ამისგან გამონაკლისია, როდესაც დაწყების ხდომილება არის გამოყენებული გაფართოვებულ ქვე-პროცესში და მიდგმულია (is attached) ამ ქვე-პროცესის საზღვარზე. ამ შემთხვევაში, შესრულების ნაკადი მაღალი დონის პროცესიდან შეიძლება დაუკავშირდეს ამ დაწყების ხდომილებას ქვე-პროცესის ფაქტიური საზღვრის ნაცვლად (იხ. ნახ. 4.15).
- დაწყების ხდომილება უნდა იყოს წყარო შესრულების ნაკადისათვის.
- მრავლობითი შესრულების ნაკადი შეიძლება დაიწყოს დაწყების ხდომილებისაგან. ყოველი შესრულების ნაკადისათვის, რომელსაც აქვს დაწყების ხდომილება წყაროს სახით, ახალი პარალელური ბილიკი უნდა იყოს გენერირებული.
 - ყოველი შემომავალი შესრულების ნაკადისათვის პირობის ატრიბუტი უნდა იყოს დაყენებული ცარიელზე (None).
 - როდესაც დაწყების ხდომილება არ არის გამოყენებული, მაშინ ყველა ნაკადის ობიექტები, რომლებსაც არა აქვთ შემავალი შესრულების

ნაკადი, უნდა იყოს განცალკევებული პარალელური ბილიკის დასაწყისი.

ყოველ ბილიკს ექნება განცალკევებული უნიკალური სიმბოლო (Token), რომელიც **გადაკვეთს** (ანუ გაივლის) შესრულების ნაკადს.

3.3.2.4. შეტყობინების ნაკადის შეერთებები (Message Flow Connections)

ობიექტების მთელი სიმრავლე მოცემულია პარაგრაფი 2.4.2, “შეტყობინების ნაკადის წესები”, გვ. 35. იქვე ნაჩვენებია თუ როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

შენიშვნა: ყველა შეტყობინების ნაკადმა უნდა შეაერთოს ორი განცალკევებული გუბე. მათ შეუძლიათ მიუერთდნენ გუბის საზღვარს ან ნაკადის ობიექტებს გუბის საზღვრების შიგნით. მათ არ შეუძლიათ დააკავშირონ (შეაერთონ) ორი ობიექტი ერთი და იმავე გუბის შიგნით.

- *დაწყების ხდომილება შეიძლება იყოს შეტყობინების ნაკადის მიზანი.* მას შეუძლია ჰქონდეს *ნოლი ან მეტი* შემომავალი შეტყობინების ნაკადი. ყოველი შეტყობინების ნაკადი მისული დაწყების ხდომილებასთან წარმოადგენს გამშვებ (გადამრთველ) (Trigger) მექანიზმს პროცესისათვის. მხოლოდ ერთ-ერთი გადამრთველთაგანი (Trigger) არის საჭირო ახალი პროცესის დასაწყებად.
 - დაწყების ხდომილების გადამრთველის ატრიბუტი (Trigger attribute) უნდა იყოს დაყენებული “შეტყობინებაზე” ან “მრავლობითზე” (to “Message” or “Multiple”) თუ არსებობს რაიმე შემომავალი შეტყობინების ნაკადი.
 - დაწყების ხდომილების გადამრთველის ატრიბუტი (Trigger attribute) უნდა იყოს დაყენებული “მრავლობითზე” თუ არსებობს ერთზე მეტი შემომავალი შეტყობინების ნაკადი.
- დაწყების ხდომილება არ უნდა იყოს შეტყობინების ნაკადის წყარო. მას არ უნდა ჰქონდეს გამომავალი შეტყობინების ნაკადი.

3.3.3. დამთავრება

როგორც სახელი გვიჩვენებს, დამთავრების ხდომილება გვიჩვენებს სად დამთავრდება პროცესი. შესრულების ნაკადის ტერმინებში, დამთავრების ხდომილება ამთავრებს პროცესის ნაკადს და, ამრიგად, მას არ ექნება რაიმე გამომავალი შესრულების ნაკადი. არავითარი შესრულების ნაკადი არ შეიძლება გამოდიოდეს დამთავრების ხდომილებიდან.

დამთავრების ხდომილებას აქვს იგივე ძირითად ფორმა რაც დაწყების ხდომილებას და შუალედურ ხდომილებას აქვთ. ის არის წრე გახსნილი ცენტრით ისე, რომ მარკერების განლაგება შესაძლო იყოს წრის შიგნით. მარკერები მიუთითებენ ხდომილების ცვლილებებს (ვარიაციებს) (variations).

- დამთავრების ხდომილება არის წრე, რომელიც უნდა იყოს დახაზული ერთი სქელი შავი ხაზით (იხ. ნახ. 3.2).
 - დამთავრების ხდომილებისათვის ტექსტი, ფერი, ზომა და ხაზები უნდა იყვნენ გამოყენებული პარაგრაფ 2.3-ში მოცემული წესების თანახმად, “ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში”, გვ. 33, იმის გამოკლებით, რომ:
 - ხაზის სისქე უნდა დარჩეს სქელი ისე, რომ დამთავრების ხდომილება გამოირჩეოდეს შუალედური და დაწყების ხდომილებებისაგან.



ნახ. 3.2. დამთავრების ხდომილება.

მსჯელობის გასაგრძელებლად იმის შესახებ თუ როგორ გრძელდება ნაკადი პროცესის გავლით, შევნიშნოთ, რომ დამთავრების ხდომილება შთანთქავს (სპობს) სიმბოლოს (Token), რომელიც იყო გენერირებული დაწყების ხდომილებიდან პროცესის იგივე დონის შიგნით. თუ პარალელური შესრულების ნაკადი მიისწრაფის დამთავრების ხდომილებისაკენ, მაშინ სიმბოლოები (Tokens) იქნებიან შთანთქმული (მოხმარებული) იმ მიმდევრობით როგორც ისინი მოდიან დამთავრების ხდომილებასთან. ყველა სიმბოლო, რომელიც იყვნენ გენერირებული პროცესის შიგნით უნდა იქნენ მოხმარებული (შთანთქმული) დამთავრების ხდომილების მიერ პროცესის დამთავრებამდე. სხვა სიტყვებით, თუ პროცესი არის ქვე-პროცესი, ის შეიძლება იყოს გაჩერებული ნორმალურად დამთავრებამდე, გამწყვეტი შუალედური ხდომილებების საშუალებით (მეტი დეტალებისათვის იხ. პარაგრაფი 4.2.2, “გამონაკლისი ნაკადი” (“Exception Flow”), გვ. 174). ამ სიტუაციაში სიმბოლოები (Tokens) იქნებიან შთანთქმული (მოხმარებული) შუალედური ხდომილების მიერ, რომელიც მიდგმულია (მიმაგრებულია) (is attached) ქვე-პროცესის საზღვართან.

დამთავრების ხდომილების სემანტიკა შეიცავს:

- პროცესის ერთი დონის შიგნით შეიძლება იყოს მრავალი დამთავრების ხდომილება.
- დამთავრების ხდომილება არის არა აუცილებელი. პროცესის მოცემულ დონეს, პროცესის ზედა დონეს ან გაფართოვებულ ქვე-პროცესს შეიძლება (არ არის აუცილებელი) ჰქონდეს ეს ფორმა:
 - თუ დამთავრების ხდომილება არ არის გამოყენებული, მაშინ პროცესის არაცხად (ნაგულისხმევ) დამთავრების ხდომილებას არ უნდა ჰქონდეს შედეგი.
 - თუ არსებობს დაწყების ხდომილება, მაშინ უნდა იყოს სულ ცოტა ერთი დამთავრების ხდომილება.
 - თუ დამთავრების ხდომილება არის გამოყენებული, მაშინ არ უნდა იყვნენ სხვა ნაკადის ელემენტები, რომლებსაც არა აქვთ რაიმე

გამომავალი შესრულების ნაკადი. ყველა სხვა ნაკადის ობიექტები უნდა იყვნენ სულ ცოტა ერთი შესრულების ნაკადის წყარო.

- აქედან გამონაკლისი არიან მოქმედებები, რომლებიც არიან განსაზღვრული როგორც კომპენსაციის მოქმედებები (აქვს კომპენსაციის მარკერი). კომპენსირების მოქმედებებს არ უნდა ჰქონდეთ რაიმე გამომავალი შესრულების ნაკადი, მაშინაც კი თუ არსებობს დამთავრების ხდომილება პროცესის დონეში. კომპენსაციის მოქმედებების შესახებ უფრო მეტი ინფორმაციისათვის იხ. პარაგრაფი 4.3, “კომპენსაციის გაერთიანება”, გვ. 177.
- თუ დამთავრების ხდომილება არ არის გამოყენებული, მაშინ ყველა ნაკადის ობიექტები, რომლებსაც არა აქვთ რაიმე გამომავალი შესრულების ნაკადი (ე.ი. არ არიან შესრულების ნაკადის წყარო), აღნიშნავენ ბილიკის დამთავრებას პროცესში. თუმცა, პროცესი არ უნდა დამთავრდეს მანამ, სანამ ყველა პარალელური ბილიკი არ დამთავრდება.
 - გამონაკლისს წარმოადგენენ მოქმედებები, რომლებიც არიან განსაზღვრული როგორც კომპენსაციის მოქმედებები (აქვს კომპენსაციის მარკერი). კომპენსაციის მოქმედებები არ არიან განხილული როგორც ნორმალური ნაკადის ნაწილი და არ უნდა აღნიშნონ პროცესის დამთავრება.

შენიშვნა: ბიზნეს პროცესების დიაგრამებს (BPD) შეიძლება ჰქონდეთ ერთზე მეტი პროცესის დონე (ე.ი. ის შეიძლება შეიცავდეს გაფართოებულ ქვე-პროცესებს). დაწყების და დამთავრების ხდომილებების გამოყენება დამოუკიდებელია დიაგრამის ყოველი დონისათვის.




დამთავრების ხდომილების არ მქონე პროცესებისათვის, ბილიკი-დამთავრება ნაკადის ობიექტის შემავალი სიმბოლო (Token) იქნება შთანთქმული, როდესაც ობიექტის მიერ განხორციელებული (შესრულებული) დამუშავება დამთავრდება (ე.ი. როდესაც ბილიკი დამთავრდება), ისე თითქოს სიმბოლო წავიდა და მიაღწია დამთავრების ხდომილებას. როდესაც ყველა სიმბოლოები მოცემული პროცესისათვის არიან შთანთქმული, მაშინ პროცესი მიაღწევს დამთავრების მდგომარეობას.




3.3.3.1. დამთავრების ხდომილების შედეგები



BPMN –ში არსებობს დამთავრების ხდომილების რვა (8) ტიპი: *ცარიელი (None)*, *შეტყობინება (Message)*, *შეცდომა (Error)*, *გაუქმება (Cancel)*, *კომპენსაცია (Compensation)*, *სიგნალი (Signal)*, *დამთავრება (Terminate)*, და *მრავლობითი (Multiple)*. ეს ტიპები განსაზღვრავენ დამთავრების ხდომილების მიღწევის შედეგს. ეს იქნება განხილული როგორც დამთავრების ხდომილების შედეგი.

ცხრილ 3.6-ში ნაჩვენებია შედეგების ტიპები და ყოველი მათგანისათვის გამოყენებული გრაფიკული მარკერები.

ცხრილი 3.6. დამთავრების ხდომილების ტიპები.

შედეგი	აღწერა	მარკერი (აღნიშვნა)
ცარიელი (None)	მოდელიორი არ აჩვენებს ხდომილების ტიპს. ის ასევე გამოიყენება იმ ქვე-პროცესის დამთავრების საჩვენებლად რომელიც მთავრდება და რომელიც იწვევს მშობელი პროცესისაკენ უკან მიმავალ ნაკადს.	
შეტყობინება (Message)	დამთავრების ეს ტიპი აჩვენებს, რომ შეტყობინება არის გაგზავნილი მონაწილისათვის პროცესის დასკვნით ნაწილში.	
შეცდომა (Error)	დამთავრების ეს ტიპი აჩვენებს, რომ დასახელებული შეცდომა უნდა იყოს გენერირებული. შეცდომა იქნება დაჭერილი (აღმოჩენილი) შეცდომის შუალედური ხდომილების საშუალებით იმავე შეცდომაკოდით ან არა შეცდომაკოდით, რომელიც არის უახლოესი (იერარქიულად) მშობელი მოქმედების საზღვარზე. პროცესის მოქმედება არ არის ზუსტად განსაზღვრული თუ უმოქმედობას პროცესში აქვს ასეთივე შუალედური ხდომილების შეცდომა. პროცესის შემსრულებელ სისტემას შეუძლია განსაზღვროს ამ შემთხვევის შესახებ დამატებითი შეცდომა, რომელიც არის პროცესის საერთო დამამთავრებელი.	
გაუქმება (Cancel)	დამთავრების ეს ტიპი გამოიყენება შესრულებად ქვე-პროცესებში. ის მიუთითებს, რომ შესრულება უნდა იქნას	

	<p>გაუქმებული და გადაირთვება (ჩაირთვება) გაუქმების შუალედური ხდომილება, რომელიც მიმაგრებულია ქვე-პროცესის საზღვარზე. გარდა ამისა, ის მიუთითებს, რომ შესრულების პროტოკოლის გაუქმების შეტყობინება უნდა იქნას გაგზავნილი რომელიმე ორგანიზაციისათვის, რომელიც ჩართულია (მონაწილეობს) შესრულებაში.</p>	
<p>კომპენსაცია (Compensation)</p>	<p>დამთავრების ეს ტიპი მიუთითებს, რომ აუცილებელია კომპენსაცია. თუ მოქმედება არის განსაზღვრული, მაშინ ეს არის მოქმედება, რომელიც იქნება კომპენსირებული. სხვანაირად, ყველა მოქმედება, რომელიც იყო დასრულებული პროცესის შიგნით, დაწყებული პროცესის საწყისს დონეზე და შემდგომი ყველა ქვე-პროცესების, არიან კომპენსაციის საგნები, განხორციელებული უკუ მიმდევრობით. კომპენსირებისათვის მოქმედებას უნდა ჰქონდეს კომპენსაციის შუალედური ხდომილება მიმაგრებული მის საზღვარზე.</p>	
<p>სიგნალი (Signal)</p>	<p>დამთავრების ეს ტიპი მიუთითებს, რომ სიგნალი იქნება გავრცელებული (გადაცემული), როდესაც დამთავრება იქნება მიღწეული. შეგნიშნოთ, რომ სიგნალი, რომელიც არის გადაცემული რომელიმე პროცესისათვის, რომელსაც შეუძლია მიიღოს სიგნალი, შეიძლება იქნას გაგზავნილი პროცესის დონეების ან გუბეების გავლით (across), მაგრამ ის არ არის შეტყობინება (რომელსაც აქვს სპეციფიური წყარო ან მიზანი).</p>	
<p>დამთავრება (Terminate)</p>	<p>დამთავრების ეს ტიპი მიუ-</p>	

	თითებს, რომ ყველა მოქმედება პროცესში დაუყოვნებლად უნდა იყოს დამთავრებული. ეს შეიცავს მრავლობითი შემთხვევების ყველა შემთხვევას. პროცესი მთავრდება კომპენსაციის გარეშე ან ხდომილების დამუშავების გარეშე.	
მრავლობითი (Multiple)	ეს ნიშნავს, რომ არსებობს პროცესის დამთავრების მრავალი შედეგი. ყველა მათგანი მოხდებიან (მაგალითად, შეიძლება გაგზავნილი იყოს მრავალი შეტყობინება). დამთავრების ხდომილების ატრიბუტები განსაზღვრავენ შედეგების სხვადასხვა ტიპებიდან რომელი გამოიყენება.	

3.3.3.2. ატრიბუტები

შემდეგ ცხრილში მოცემულია დამთავრება ხდომილების ატრიბუტების სიმრავლე, რომლებიც აფართოვებენ ხდომილების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.3).

ცხრილი 3.7. დამთავრების ხდომილების ატრიბუტები

ატრიბუტები	აღწერა
<p>შედეგი (0-n) : ხდომილებადეტალი Result (0-n) : EventDetail</p>	<p>შედეგი (ხდომილებადეტალი) არის ატრიბუტი, რომელიც განსაზღვრავს შედეგის ტიპს მოსალოდნელს დამთავრების ხდომილებისათვის. ხდომილებადეტალიტიპების სიმრავლიდან (იხ. პარაგრაფი 3.3.5. “ხდომილების დეტალები,” გვ. 67) მხოლოდ ექვსი (6) შეიძლება იყოს გამოყენებული დამთავრების ხდომილებისათვის: <i>შეტყობინება, შეცდომა, გაუქმება, კომპენსაცია, სიგნალი და დამთავრება</i> (იხ. ცხრილი 3.6).</p> <ul style="list-style-type: none"> • თუ ხდომილებადეტალი არ არის განსაზღვრული, მაშინ ეს განხილული როგორც ცარიელი დამთავრების ხდომილება და ხდომილებას არ ექნება შუალედური მარკერი (იხ. ცხრილი 3.6). • თუ ერთზე მეტი ხდომილებადეტალი

	არის განსაზღვრული, მაშინ ეს განხილულია როგორც მრავლობითი დამთავრების ხდომილება და ხდომილებას ექნება ხუთკუთხა შიდა მარკერი (იხ. ცხრილი 3.6).
--	---

3.3.3.3. შესრულების ნაკადის შეერთებები

პარაგრაფ 2.4.1-ში “შესრულების ნაკადის წესები”, გვ. 34, მოცემულია ობიექტების მთელი სიმრავლე და აღწერილია როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო (source) ან მიზნები (targets).

- დამთავრების ხდომილება უნდა იყოს შესრულების ნაკადის მიზანი.
- დამთავრების ხდომილებას შეიძლება ჰქონდეს მრავლობითი შემავალი შესრულების ნაკადი.

ნაკადი შეიძლება მოვიდეს როგორც ალტერნატიული, ასევე პარალელური ბილიკებიდან. მოდელირების მოხერხებულობისათვის ყოველი ბილიკი შეიძლება უერთდებოდეს ცალკეულ დამთავრების ხდომილების ობიექტს. დამთავრების ხდომილება გამოიყენება როგორც ჩამრეცხი (გამქრობი) ყველა სიმბოლოსათვის, რომლებიც მოდიან ხდომილებასთან. ყველა სიმბოლო (Token), რომლებიც ამ პროცესისათვის არიან გენერირებული დაწყების ხდომილებასთან, საბოლოოდ უნდა მივიდნენ დამთავრების ხდომილებასთან. პროცესი იქნება გაშვებულ (მუშა) (running) მდგომარეობაში, სანამ ყველა სიმბოლო არ არიან შთანთქმული (დახარჯული).

- დამთავრების ხდომილება არ უნდა იყოს შესრულების ნაკადის წყარო; ეს ნიშნავს, რომ მას არ უნდა ჰქონდეს გამოშვებული შესრულების ნაკადი.
 - ამისგან გამონაკლისია როდესაც დამთავრების ხდომილება გამოიყენება გაფართოებულ ქვე-პროცესში და მიდგმულია ამ ქვე-პროცესის საზღვარზე. ამ შემთხვევაში უმაღლესი დონის პროცესის შესრულების ნაკადი შეიძლება შეუერთდეს დამთავრების ხდომილებას ნაცვლად ქვე-პროცესის არსებულ საზღვარზე მიერთებისა (იხ. ნახ. 4.15).

3.3.3.4. შეტყობინების ნაკადის შეერთებები

პარაგრაფ 2.4.2-ში “შეტყობინების ნაკადის წესები”, გვ. 35, მოცემულია ობიექტების მთელი სიმრავლე და აღწერილია როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო (source) ან მიზნები (targets).

შენიშვნა. ყველა შეტყობინების ნაკადმა უნდა შეაერთოს ორი განცალკევებული გუბე. მათ შეუძლიათ შეუერთდნენ გუბის საზღვარს ან ნაკად ობიექტებს გუბის საზღვრებში. მათ არ შეუძლიათ შეაერთონ ორი ობიექტი ერთი და იგივე გუბეში.

- დამთავრების ხდომილება არ უნდა იყოს შეტყობინების ნაკადის მიზანი; მას შეიძლება ჰქონდეს არა შემავალი შეტყობინების ნაკადი. თუ შეაღებულ

ხლომილებას აქვს შემავალი შეტყობინების ნაკადი, მაშინ მას არ უნდა ჰქონდეს გამომავალი შეტყობინების ნაკადი.

- შეტყობინების ტიპის შუალედური ხლომილება, თუ ის გამოიყენება ნორმალური ნაკადის შიგნით, შეიძლება იყოს შეტყობინების ნაკადის წყარო; მას შეიძლება ჰქონდეს ერთი გამომავალი შეტყობინების ნაკადი. *თუ შუალედურ ხლომილებას აქვს გამომავალი შეტყობინების ნაკადი, მაშინ მას არ უნდა ჰქონდეს შემავალი შეტყობინების ნაკადი.*

3.3.4. შუალედური

როგორც სათაურიდან ჩანს, შუალედური ხლომილება მიუთითებს პროცესის დაწყებასა და დამთავრებას შორის სად ხდება რაღაც (ხლომილება). ის იმოქმედებს პროცესის ნაკადზე, მაგრამ არ დაიწყებს ან (უშუალოდ) არ დაამთავრებს პროცესს.

შუალედური ხლომილებები შეიძლება გამოყენებული იყვნენ:

- აჩვენოს პროცესის შიგნით სად არიან შეტყობინებები მოსალოდნელი ან გაგზავნილი;
- აჩვენოს პროცესში სად არიან მოსალოდნელი დაყოვნებები;
- შეწყვიტოს ნორმალური ნაკადი *გამონაკლისის* დამუშავების საშუალებით, ან
- აჩვენოს დამატებითი სამუშაო მოთხოვნილი *კომპენსაციისათვის*.

შუალედურ ხლომილებას აქვს იგივე ძირითადი ფორმა რაც დაწყებისა და დამთავრების ხლომილებებს. ეს არის წრე გახსნილი ცენტრით ისე, რომ მარკერები შეიძლება იყვნენ განთავსებული წრის შიგნით, რომ მიუთითონ ხლომილების ვარიაცია (ცვალებადობა).

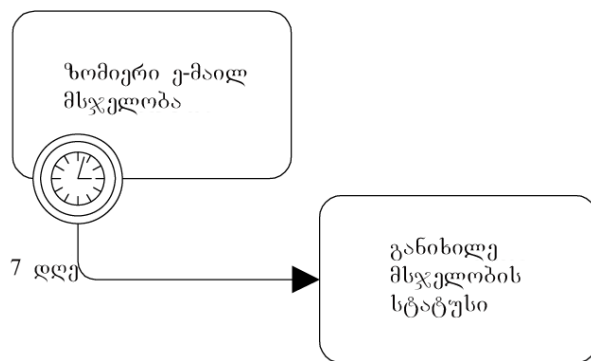
- შუალედური ხლომილება არის წრე, რომელიც დახაზული უნდა იყოს ორი წვრილი შავი ხაზით (იხ. ნახ. 3.3).
 - შუალედური ხლომილებისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს იმ წესების თანახმად, რომლებიც მოცემულია პარაგრაფში 2.3, “ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში”, გვ. 33 იმის გამოკლებით, რომ:
 - ხაზის სისქე უნდა დარჩეს ორმაგი ისე, რომ შუალედური ხლომილების განსხვავება შეიძლებოდეს დაწყებისა და დამთავრების ხლომილებებისაგან.



ნახ. 3.3. შუალედური ხლომილება.

შუალედური ხლომილების ერთი გამოყენებაა გამონაკლისის ან კომპენსაციის დამუშავების წარმოდგენა. ეს ნაჩვენებია იქნება შუალედური ხლომილების განთავსებით ამოცანის ან ქვე-პროცესის (ან დანგრეულის ან გაფართოებულის) საზღვარზე. ნახ. 3.4-ზე ნაჩვენებია ამოცანაზე მიდგმული შუალედური ხლომილების მაგალითი. შუალედური ხლომილება შეიძლება მიდგმული იყოს მოქმედების

საზღვრის ნებისმიერ ადგილას და გამომავალი შესრულების ნაკადი შეიძლება მიედინებოდეს ნებისმიერი მიმართულებით. თუმცა, დიაგრამის სიცხადისათვის, რეკომენდირებულია, რომ მოდელიორმა შეარჩიოს საზღვარზე მიმდევრობითი განთავსება. მაგალითად, თუ დიაგრამის ორიენტაცია არის ჰორიზონტალური, მაშინ შუალედური ხდომილება შეიძლება იყოს მიმაგრებული მოქმედების ძირზე და შესრულების ნაკადი მიმართული იყოს ქვემოთ და არა მარჯვნივ. თუ დიაგრამის ორიენტაცია არის ვერტიკალური, მაშინ შუალედური ხდომილება შეიძლება იყოს მიმაგრებული მოქმედების მარცხენა ან მარჯვენა მხარეს და შესრულების ნაკადი მიმართული იყოს მარცხნივ ან მარჯვნივ და არა ქვემოთ.



ნახ. 9.4. ამოცანა მის საზღვარზე მიდგმული შუალედური ხდომილებით.

3.3.4.1. შუალედური ხდომილების გადამრთველები (Triggers)

BPMN-ში არის 10 ტიპის შუალედური ხდომილება: ცარიელი (None), შეტყობინება (Message), ტაიმერი (Timer), შეცდომა (Error), გაუქმება (Cancel), კომპენსაცია (Compensation), პირობითი (Conditional), რგოლი (Link), სიგნალი (Signal) და მრავლობითი (Multiple). შუალედური ხდომილების ყოველ ტიპს აქვს სხვადასხვა პიქტოგრამა (icon), რომელიც განთავსებულია შუალედური ხდომილების ფორმის ცენტრში და განარჩევს ერთს მეორესაგან.



შუალედური ხდომილება, რომელიც განთავსებულია პროცესის ნორმალური ნაკადის შიგნით, შეიძლება გამოყენებული იქნას ორიდან ერთი მიზნისათვის. ხდომილება შეიძლება პასუხობდეს (“იჭერდეს” (“catch”)) ხდომილების გადამრთველს ან ხდომილება შეიძლება გამოყენებული იქნას ხდომილების გადამრთველის (Trigger) გასაშვებად (“გადასაგდებად” (“throw”)). შუალედური ხდომილება, რომელიც მიდგმულია (attached) მოქმედების საზღვარზე, შეიძლება გამოყენებული იქნას მხოლოდ ხდომილების გადამრთველის (Trigger) “დასაჭერად” (“catch”).




როდესაც სიმბოლო (Token) მოდის შუალედურ ხდომილებასთან, რომელიც განთავსებულია პროცესის ნორმალურ ნაკადში, ორიდან ერთი რამე მოხდება. თუ ხდომილება გამოიყენება ხდომილების გადამრთველის (Trigger) “გადასაგდებად”, მაშინ ხდომილების გადამრთველი დაუყოვნებლივ შეგვხვდება (ანუ შეტყობინება

იქნება გაგზავნილი) და სიმბოლო გამომავალ შესრულების ნაკადს გადაადგილებს ქვემოთ. თუ ხდომილება გამოიყენება ხდომილების გადამრთველის (Trigger) “დასაჭერად”, მაშინ სიმბოლო დარჩება ხდომილებასთან გადამრთველის გამოჩენამდე (ანუ სანამ შეტყობინება იქნება მიღებული). შემდეგ სიმბოლო გამომავალ შესრულების ნაკადს გადაადგილებს ქვემოთ.




ნახ. 3.8-ზე ნაჩვენებია გადამრთველების (Triggers) ტიპები და გრაფიკული მარკერები, რომლებიც გამოიყენებიან ყოველი მათგანისათვის.


ცხრილი 3.8. შუალედური ხდომილების ტიპები.



გადამრთველი (Trigger)	აღწერა	მარკერი (Marker)
ცარიელი (None)	ეს არის მნიშვნელოვანი მხოლოდ შუალედური ხდომილებებისათვის, რომლებიც არიან პროცესის მთავარ ნაკადში. მოდელიორი არ აჩვენებს ხდომილების ტიპს. ის გამოიყენება მეთოდოლოგიების მოდელირებისათვის, რომლებიც ხდომილებებს გამოიყენებენ პროცესში მდგომარეობის რაიმე ცვლილების საჩვენებლად.	
შეტყობინება (Message)	შეტყობინება მოდის მონაწილეთაგან და ხდომილების გადამრთველებისაგან. ეს იწვევს პროცესის გაგრძელებას, თუ ის ელოდებოდა შეტყობინებას, ან ცვლის ნაკადს გამონაკლისის დამუშავებისათვის. როცა გამოიყენება შეტყობინების “დაჭერა” (“catch”) (ანუ მიღება), მაშინ ხდომილების მარკერი იქნება შეუფერადებელი (იხ. მარჯვენა მხარეს ზედა ფიგურა). ნორმალურ ნაკადში, შუალედური ხდომილებების შეტყობინება შეიძლება იყოს გამოყენებული მონაწილისათვის შეტყობინებების გასაგზავნად. როცა გამოიყენება შეტყობინების “გადაგდება” (“throw”) (ანუ გაგზავნა), ხდომილების მარკერი იქნება შეუფერადებელი (იხ. მარჯვნივ ქვედა ფიგურა). თუ გამოიყენება გამო-	

	ნაკლისის დამუშავება, ის შეცვლის ნორმალურ ნაკადს გამონაკლისის ნაკადის შიგნით.	
ტაიმერი (Timer)	სპეციფიურ დრო-თარიღზე ან სპეციფიურ ციკლზე (მაგალითად, ყოველ ორშაბათს დილის 9 საათზე) შეიძლება იყოს დადგენილი, რომ ხდომილება იქნება გადართული. თუ გამოიყენება მთავარი ნაკადის შიგნით, ის მოქმედებს როგორც დამყოვნებელი მექანიზმი. თუ გამოიყენება გამონაკლისის დამუშავებისათვის, ის შეცვლის ნორმალურ ნაკადს გამონაკლისის ნაკადის შიგნით.	
შეცდომა (Error)	ამ ტიპის ხდომილება შეიძლება მიდგმული იყოს მხოლოდ მოქმედების საზღვარზე. ამრიგად, ის რეაგირებს (იჭერს) დასახელებულ შეცდომას, ან ნებისმიერ შეცდომას თუ სახელი არ არის განსაზღვრული.	
გაუქმება (Cancel)	ამ ტიპის შუალედური ხდომილება გამოიყენება ქვეპროცესის შესრულებისათვის (Transaction). ამ ტიპის ხდომილება უნდა იყოს მიმაგრებული ქვეპროცესის საზღვარზე. ის იქნება გადართული (triggered) თუ დამთავრების ხდომილების გაუქმება არის მიღწეული ქვეპროცესის შესრულებისას. ის აგრეთვე იქნება გადართული თუ შესრულების პროტოკოლი “გაუქმება” შეტყობინება იყო მიღებული სანამ შესრულება დამთავრდებოდა.	
კომპენსაცია (Compensation)	ეს გამოიყენება კომპენსაციის დამუშავებისათვის – ორივეს გააქტიურების და შესრულების კომპენსაციებისათვის. როდესაც გამოიყენება	

	<p>ნორმალურ ნაკადში, ეს შუალედური ხდომილება მიუთითებს, რომ კომპენსაცია არის საჭირო. ამრიგად, ის გამოიყენება კომპენსირების ხდომილების “გადაგდებისათვის” და ხდომილების მარკერი უნდა იყოს შეფერადებული (იხ. მარჯვნივ ქვედა ფიგურა). თუ ხდომილება იდენტიფიცირებას უკეთებს მოქმედებას, მაშინ ეს არის მოქმედება (და არა სხვა რამე), რაც იქნება კომპენსირებული. სხვანაირად, კომპენსაცია გვატყობინებს ყველა მოქმედებას, რომლებიც დამთავრდნენ პროცესის შიგნით, პროცესის ზედა დონის და ყველა ქვე-პროცესების ჩათვლით. ყოველი დამთავრებული მოქმედება, რომელიც არის კომპენსაციის საგანი, კომპენსირებული იქნება მოქმედების შესრულების უკუ მიმართულებით. იმისათვის, რომ მოქმედება იყოს კომპენსირებული, მას უნდა ჰქონდეს კომპენსაციის შუალედური ხდომილება მიდგმული მის საზღვარზე. როდესაც მიდგმულია მოქმედების საზღვარზე, ხდომილება იქნება გადართული გადაგდებული კომპენსირებით, რაც აიდენტიფიცირებს ამ მოქმედებას ან შეტყობინებულ კომპენსირებას. როდესაც გამოიყენება კომპენსაციის ხდომილების “დეჭერა”, ხდომილების მარკერი არ უნდა იყოს შეფერადებული (იხ. მარჯვნივ ზედა ფიგურა). როდესაც ხდომილება არის გადართული, კომპენსირების მოქმედება, რომელიც არის მიერთებული ხდომილებასთან, იქნება შესრულებული</p>	
--	--	---

	(იხ. ნახ. 3.13).	
პირობითი (Conditional)	ამ ტიპის ხდომილება არის გადართული, როდესაც პირობა ხდება ჭეშმარიტი.	
რგოლი (Link)	რგოლი არის მექანიზმი პროცესის ორი სექციის შესაერთებლად. რგოლი ხდომილება შეიძლება იყოს გამოყენებული მარყუჟული სიტუაციის შესაქმნელად ან გრძელი შესრულების ნაკადის ხაზის თავიდან ასაცილებლად. შეერთების ხდომილების გამოყენება არის შეზღუდული მარტივი პროცესის დონეზე (ე.ი. მათ არ შეუძლიათ შეაერთონ მშობელი პროცესი ქვეპროცესთან). შეწყვილებული შუალედური ხდომილებები კიდევ შეიძლება იყვნენ გამოყენებული როგორც “გვერდის გარეთ შემაერთებლები” პროცესის დაბეჭდვისათვის მრავალ გვერდზე. ისინი შეიძლება კიდევ გამოყენებული იყვნენ როგორც საერთო “გადასვლა” (“Go To”) ობიექტები პროცესის დონის შიგნით. შეიძლება არსებობდნენ მრავლობითი წყაროს შეერთების ხდომილებები, მაგრამ აქ შეიძლება იყოს მხოლოდ ერთი მიზნის შეერთების ხდომილება. როდესაც გამოიყენება “დაჭერა” წყაროს შეერთებიდან, ხდომილების მარკერი იქნება შეუფერადებელი (იხ. მარჯვნივ ზედა ფიგურა). როდესაც გამოიყენება “გადაგდება” (to “throw”) მიზნის შეერთებასთან, ხდომილების მარკერი იქნება შეფერადებული (იხ. მარჯვნივ ქვედა ფიგურა).	 
სიგნალი (Signal)	ამ ტიპის ხდომილება გამო-	

	<p>იყენება სიგნალების გასაგზავნად ან მისაღებად. სიგნალი გამოიყენება პროცესის დონეებს, გუბეებს და ბიზნეს პროცესის დიაგრამებს შორის საერთო კავშირისათვის (შეტყობინებისათვის, კომუნიკაციისათვის). BPMN-ის სიგნალი არის ნათების მაგვარი სიგნალი, რომელიც გაისროლეს ცაში ვილაციისათვის, ვისაც შეუძლია მისი შემჩნევა და შემდეგ მასზე რეაგირება. ამრიგად, არსებობს სიგნალის წყარო, მაგრამ არა სპეციფიური, განზრახული მიზანი. ეს არის განსხვავდება BPMN-ის შეტყობინებისაგან, რომელსაც აქვს სპეციფიური წყარო და სპეციფიური მიზანი (რომელიც შეიძლება იყოს რაიმე ორგანიზაცია ან რაიმე აბსტრაქტული როლი). ამ ტიპის შუალედურ ხდომილებას შეუძლია გაგზავნოს ან მიღოს სიგნალი თუ ხდომილება არის ნორმალური ნაკადის ნაწილი. ხდომილებას შეუძლია მხოლოდ მიიღოს სიგნალი როდესაც მიმაგრებულია მოქმედების საზღვარზე. სიგნალის ხდომილება განსხვავდება შეცდომის ხდომილებისაგან, რომელშიც სიგნალი განსაზღვრავს უფრო ზოგად, არა მცდარ პირობას წყვეტადი მოქმედებებისათვის (ისეთის როგორიც არის სხვა მოქმედების წარმატებული დამთავრება), ასევე აქვს უფრო ფართო კომპეტენცია ვიდრე შეცდომის ხდომილებებს. როდესაც გამოიყენება სიგნალის “დაჭერა”, ხდომილების მარკერი იქნება შეუფერადებელი (იხ. მარჯვნივ ზედა ფიგურა). როდესაც გამოიყენება სიგ-</p>	
--	---	---

	ნალის “გადაგდება”, ხლომილების მარკერი იქნება შეფერადებული (იხ. მარჯვნივ ქვედა ფიგურა).	
მრავლობითი (Multiple)	ეს ნიშნავს, რომ ხლომილებისათვის გამოყენებული გვაქვს მრავლობითი გადამრთველები. თუ გამოიყენება ნორმალური ნაკადის შიგნით, ხლომილებას შეუძლია “დაიჭიროს” (“catch”) ან “გადააგდოს” (“throw”) გადამრთველი. როდესაც მიმაგრებულია მოქმედების საზღვარზე, ხლომილებას შეუძლია მხოლოდ “დაიჭიროს” გადამრთველი. როდესაც გამოიყენება გადამრთველის “დაჭერა”, მხოლოდ ერთი გადამრთველია მოთხოვნილი და ხლომილების მარკერი იქნება შეფერადებული (იხ. მარჯვნივ ზედა ფიგურა). როდესაც გამოიყენება გადამრთველის “გადაგდება” (იგივე რაც მრავლობითი დამთავრების ხლომილება), ყველა დანიშნული გადამრთველები იქნებიან გადაგებული და ხლომილების მარკერი იქნება შეფერადებული (იხ. მარჯვნივ ქვედა ფიგურა).	 

3.3.4.2. ატრიბუტები

შემდეგ ცხრილში მოცემულია შუალედური ხლომილების ატრიბუტების სიმრავლე, რომლებიც აფართოებენ ხლომილებების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.9).

ცხრილი 3.9 შუალედური ხლომილების ატრიბუტები

ატრიბუტები	აღწერა
გადამრთველი (0-n) : ხლომილებადეტალი Trigger (0-n) : EventDetail	გადამრთველი (ხლომილებადეტალი) არის ატრიბუტი, რომელიც განსაზღვრავს გადამრთველის ტიპს გამოყენებულს შუალედური

	<p>ხდომილებისათვის. ხდომილება დეტალიტიპების სიმრავლიდან (იხ. პარაგრაფი 3.3.5, “ხდომილების დეტალები”, გვ. 49) მხოლოდ რვა შეიძლება იყოს გამოყენებული შუალედური ხდომილებისათვის: <i>შეტყობინება (Message)</i>, <i>ტაიმერი (Timer)</i>, <i>შეცდომა (Error)</i>, <i>გაუქმება (Cancel)</i>, <i>კომპენსაცია (Compensation)</i>, <i>პირობითი (Conditional)</i>, <i>რგოლი (Link)</i> და <i>სიგნალი (Signal)</i> (იხ. ცხრილი 3.8).</p> <ul style="list-style-type: none"> • თუ ხდომილება დეტალი არ არის განსაზღვრული, მაშინ ეს არის განხილული როგორც ცარიელი (None) შუალედური ხდომილება და ხდომილებას არ ექნება შიდა მარკერი (იხ. ცხრილი 3.8). • თუ ერთზე მეტი ხდომილება დეტალი არის განსაზღვრული, მაშინ ეს არის განხილული როგორც მრავლობითი შუალედური ხდომილება და ხდომილებას ექნება ვარსკვლავური შიდა მარკერი (იხ. ცხრილი 3.8).
<p>მიზანი (0-1) : მოქმედება</p>	<p>მიზანი შეიძლება იყოს ჩართული (included) შუალედური ხდომილებისათვის. ეს მიზანი უნდა იყოს მოქმედება (ქვე-პროცესი ან ამოცანა). ეს ნიშნავს, რომ შუალედური ხდომილება მიმაგრებულია მოქმედების საზღვარზე და გამოიყენება ამ მოქმედებისათვის გამონაკლისის ან კომპენსირების გამოსახატავად.</p>

3.3.4.3. მოქმედების საზღვრის შეერთებები

შუალედური ხდომილება შეიძლება იყოს მიმაგრებული მოქმედების საზღვარზე შემდეგი პირობების დაცვით:

- (ერთი ან მეტი) შუალედური ხდომილებები შეიძლება იყვნენ მიმაგრებული (მიდგმული) უშუალოდ მოქმედების საზღვარზე.
 - იმისათვის, რომ მოქმედების საზღვარზე იყოს მიდგმული, შუალედური ხდომილება უნდა იყოს ერთ-ერთი შემდეგი გადამრთველებიდან (Triggers): *შეტყობინება (Message)*, *ტაიმერი (Timer)*, *შეცდომა (Error)*, *გაუქმება (Cancel)*, *კომპენსაცია (Compensation)*, *პირობითი (Conditional)*, *სიგნალი (Signal)* და *მრავლობითი (Multiple)*.
 - შუალედური ხდომილება *გაუქმება (Cancel)* გადამრთველით შეიძლება იყოს მიდგმული (attached) ქვე-პროცესის საზღვარზე მხოლოდ თუ ქვე-პროცესის არის შესრულება (IsATransaction) ატრიბუტი არის დაყენებული ჭეშმარიტზე (True).

3.3.4.4. შესრულების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების სრული სიმრავლისათვის და იმისათვის, თუ როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

- შემდეგი შუალედური ხდომილებები შეიძლება მიმაგრდნენ მოქმედების საზღვარზე: *შეტყობინება (Message)*, *ტაიმერი (Timer)*, *შეცდომა (Error)*, *გაუქმება (Cancel)* (მხოლოდ ქვე-პროცესი, რომელიც არის შესრულება (Transaction)), *კომპენსაცია (Compensation)*, *პირობითი (Conditional)*, *სიგნალი (Signal)* და *მრავლობითი (Multiple)*. ამრიგად, შემდეგი შუალედური ხდომილებები არ შეიძლება: *ცარიელი (None)* და *რგოლი (Link)*.
- თუ შუალედური ხდომილება არის მიმაგრებული მოქმედების საზღვარზე:
 - შუალედური ხდომილება არ უნდა იყოს შესრულების ნაკადის მიზანი; მას არ შეუძლია ჰქონდეს შემავალი ნაკადი.
 - შუალედური ხდომილება უნდა იყოს შესრულების ნაკადის წყარო; მას შეიძლება ჰქონდეს ერთი (და მხოლოდ ერთი) გამომავალი შესრულების ნაკადი.
 - გამონაკლისს შეადგენს: შუალედური ხდომილება კომპენსაციის გადამრთველით რომელსაც არ უნდა ჰქონდეს გამომავალი შესრულების ნაკადი (მას შეიძლება ჰქონდეს ერთი გამომავალი გაერთიანება).
- შემდეგი შუალედური ხდომილებები შეიძლება გამოყენებული იყვნენ ნორმალურ ნაკადში: *ცარიელი (None)*, *შეტყობინება (Message)*, *ტაიმერი (Timer)*, *კომპენსაცია (Compensation)*, *პირობითი (Conditional)*, *რგოლი (Link)*, და *სიგნალი (Signal)*. ამრიგად, არ გამოიყენება შემდეგი მოქმედებები: *გაუქმება (Cancel)*, *შეცდომა (Error)* და *მრავლობითი (Multiple)*.
 - თუ შუალედური ხდომილება გამოიყენება ნორმალური ნაკადის შიგნით:
 - შემდეგი ტიპის შუალედური ხდომილებები უნდა იყვნენ შესრულების ნაკადის მიზანი: *ცარიელი (None)* და *კომპენსაცია (Compensation)*. მათ უნდა ჰქონდეთ ერთი (და მხოლოდ ერთი) შემავალი ნაკადი.
 - შემდეგი ტიპის შუალედური ხდომილებები შეიძლება იყვნენ შესრულების ნაკადის მიზანი: *შეტყობინება (Message)*, *ტაიმერი (Timer)*, *პირობითი (Conditional)*, *რგოლი (Link)*, და *სიგნალი (Signal)*. მათ შეიძლება ჰქონდეთ ერთი (და მხოლოდ ერთი) შემავალი ნაკადი.

შენიშვნა. ამ ტიპის შუალედური ხდომილებები ყოველთვის იქნებიან მზად მიიღონ ხდომილების გადამრთველები (Event Triggers) (ერთხელ) სანამ პროცესი, რომელშიც

ისინი შედიან, არის აქტიური. ისინი არ არიან არასავალდებულო (ფაკულტატიური) და არიან გადართვის მოლოდინში პროცესის შესრულების დროს.

- შუალედური ხდომილება უნდა იყოს შესრულების ნაკადის წყარო; მას უნდა ჰქონდეს ერთი (და მხოლოდ ერთი) გამომავალი შესრულების ნაკადი.
 - გამონაკლისს შეადგენს: წყარო *რგოლი (Link)* შუალედური ხდომილება (როგორც განსაზღვრულია ქვემოთ). არ მოითხოვება ჰქონდეს გამომავალი შესრულების ნაკადი.
- შუალედური ხდომილება *რგოლი (Link)* გადამრთველით არ უნდა იყოს შესრულების ნაკადის არც მიზანი და არც წყარო.

განვსაზღვროთ *რგოლი (Link)* შუალედური ხდომილების გამოყენება როგორც “გვერდის გარეთ შემართებელი” (Off-Page Connector) ან “გადასვლა” (Go To) ობიექტი:

- *რგოლი (Link)* შუალედური ხდომილება შეიძლება იყოს მიზანი (მიზანი რგოლი ხდომილება) ან წყარო (წყარო რგოლი ხდომილება) შესრულების ნაკადის.
 - თუ გვაქვს წყარო *რგოლი (Link)* ხდომილება, აქვე უნდა იყოს შესაბამისი მიზანი *რგოლი* ხდომილება (მათ აქვთ ერთნაირი სახელი).
 - შეიძლება არსებობდნენ მრავლობითი წყარო *რგოლი* ხდომილებები მარტივი მიზანი *რგოლი* ხდომილებისათვის.
 - აქ არ უნდა იყვნენ მრავლობითი მიზანი *რგოლი* ხდომილებები მარტივი წყარო *რგოლი* ხდომილებისათვის.

3.3.4.5. შეტყობინების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.2, “შეტყობინების ნაკადის წესები”, გვ. 35, ობიექტების მთელი სიმრავლისათვის და იმისათვის, თუ როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

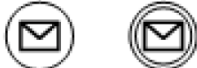

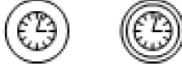






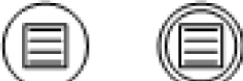


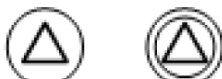


შენიშვნა. ყველა შეტყობინების ნაკადმა უნდა შეაერთოს ორი განცალკევებული გუბე. მათ შეუძლიათ მიუერთდნენ გუბის საზღვარს ან ნაკადის ობიექტებს გუბის საზღვრებში. მათ არ შეუძლიათ შეაერთონ ორი ობიექტი ერთი გუბის ფარგლებში.

- შეტყობინების (Message) ტიპის შუალედური ხდომილება შეიძლება იყოს მიზანი შეტყობინების ნაკადისათვის; მას შეიძლება ჰქონდეს ერთი შემავალი შეტყობინების ნაკადი.
- შეტყობინების (Message) ტიპის შუალედური ხდომილება შეიძლება იყოს წყარო შეტყობინების ნაკადისათვის; მას შეიძლება ჰქონდეს ერთი გამომავალი შეტყობინების ნაკადი.

- შეტყობინების (Message) ტიპის შუალედურ ხდომილებას შეიძლება ჰქონდეს ერთი შემავალი შეტყობინების ნაკადი ან ერთი გამომავალი შეტყობინების ნაკადი, მაგრამ არა ორივე.

3.3.5. ხდომილების დეტალები

ხდომილების დეტალებს გადავყავართ დაწყების გადამრთველებთან და შუალედურ ხდომილებებთან და დამთავრების ხდომილებების შედგებთან. ხდომილების დეტალების ტიპები არიან: შეტყობინება (Message), ტაიმერი (Timer), შეცდომა (Error), გაუქმება (Cancel), კომპენსაცია (Compensation), პირობითი (Conditional), რგოლი (Link), სიგნალი (Signal) და დამთავრება (Terminate). ცარიელი (None) ხდომილება არის განსაზღვრული ხდომილებით, რომელიც არ განსაზღვრავს (არ აზუსტებს) ხდომილების დეტალს. მრავლობითი (Multiple) ხდომილება არის განსაზღვრული ხდომილებით, რომელიც განსაზღვრავს (აზუსტებს) ხდომილების ერთზე მეტ დეტალს. ხდომილებების სხვადასხვა ტიპები (დაწყება (Start), შუალედური (Intermediate) და დამთავრება (End)) იყენებენ ხდომილების დეტალების შესაძლო ტიპების ქვე-სიმრავლეს (იხ. ნახ. 3.5).

	“დაჭერა” (“Catching”)	“გადაგდება” (“Throwing”)
შეტყობინება (Message)		
ტაიმერი (Timer)		
შეცდომა (Error)		
გაუქმება (Cancel)		
კომპენსაცია (Compensation)		
პირობითი (Conditional)		
რგოლი (Link)		
სიგნალი (Signal)		
დამთავრება (Terminate)		

მრავლობითი (Multiple)	
-----------------------	--

ნახ. 3.5. ხდომილების დეტალები გამოყენებული დაწყების, შუალედურ და დამთავრების ხდომილებებთან.

შემდეგ პარაგრაფებში მოცემულია ატრიბუტები, რომლებიც საერთოა ყველა ხდომილების დეტალებისათვის და სპეციფიური ატრიბუტები იმ ხდომილების დეტალებისათვის, რომლებსაც აქვთ დამატებითი ატრიბუტები. შევნიშნოთ, რომ გაუქმების (Cancel) და დამთავრების (Terminate) ხდომილების დეტალებს არა აქვთ დამატებითი ატრიბუტები.

3.3.5.1. ხდომილების დეტალის საერთო ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია ატრიბუტების სიმრავლე, რომელიც საერთოა ხდომილებადეტალის (EventDetail) ტიპებისათვის და რომელიც აფართოებს BPMN-ის ელემენტების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.1).

ცხრილი 3.10. საერთო ხდომილებადეტალის ატრიბუტები.

ატრიბუტები	აღწერა
<p>ხდომილებადეტალიტიპი (შეტყობინება ტაიმერი შეცდომა პირობითი რგოლი სიგნალი კომპენსაცია გაუქმება დამთავრება) შეტყობინება : სტრიქონი EventDetailType (Message Timer Error Conditional Link Signal Compensate Cancel Terminate) Message : String</p>	<p>ხდომილებადეტალიტიპის ატრიბუტი განსაზღვრავს გადამრთველის ტიპს, რომელიც მოსალოდნელია ხდომილებისათვის. ტიპების სიმრავლე შეიცავს: <i>შეტყობინებას (Message)</i>, <i>ტაიმერს (Timer)</i>, <i>შეცდომას (Error)</i>, <i>პირობითობის (Conditional)</i>, <i>რგოლს (Link)</i>, <i>სიგნალს (Signal)</i>, <i>კომპენსაციას (Compensation)</i>, <i>გაუქმებას (Cancel)</i> და <i>დამთავრებას (Terminate)</i>. ყოველ ხდომილებატიპს (დაწყება (Start), შუალედური (Intermediate) და დამთავრება (End)) ექნება ხდომილებადეტალიტიპების ქვე-სიმრავლე, რომელიც შეიძლება იყოს გამოყენებული. ხდომილებადეტალიტიპის ჩამონათვალი შეიძლება იყოს გაფართოებული ახალი ტიპების ჩართვით. ამ ახალ ტიპებს შეიძლება ჰქონდეთ მოდელიორის ან ინსტრუმენტის განმსაზღვრელის მიერ ჩასმული ახალი მარკერი ხდომილების საზღვრებში.</p>

3.3.5.2. პირობითი ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია პირობითი ხდომილებადეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილების დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.11. პირობითი ხდომილებადეტალის ატრიბუტები

ატრიბუტები	აღწერა
პირობაRef : პირობა ConditionRef : Condition	თუ გადამრთველი (Trigger) არის პირობითი, მაშინ პირობა უნდა იყოს შემოტანილი. არსებობენ პირობის ატრიბუტები, რომლებსაც ჩვენ არ განვიხილავთ.

3.3.5.3. კომპენსაციის ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია კომპენსაციის ხდომილებადეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილების დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.12. კომპენსაციის ხდომილებადეტალის ატრიბუტები

ატრიბუტები	აღწერა
მოქმედებაRef (0-1) : მოქმედება ActivityRef (0-1) : Activity	<p>დამთავრების დხომილებისათვის:</p> <ul style="list-style-type: none"> • თუ შედეგი არის კომპენსაცია, მაშინ მოქმედება რომელიც უნდა იყოს კომპენსირებული შეიძლება იყოს შემოტანილი. თუ მოქმედება არ არის შემოტანილი, მაშინ ხდომილება არის გამოყენებული ყოველი დამთავრებული მოქმედებისათვის პროცესში. <p>შუალედური ხდომილებისათვის ნორმალურ ნაკადში:</p> <ul style="list-style-type: none"> • თუ გადამრთველი (Trigger) არის კომპენსაცია, მაშინ კომპენსირებული მოქმედება შეიძლება იყოს შემოტანილი. თუ მოქმედება არ არის შემოტანილი, მაშინ ხდომილება არის გამოყენებული ყოველი დამთავრებული მოქმედებისათვის პროცესში. ეს “გადააგდებს” (“throws”) (ანუ შეასრულებს) კომპენსაციას. <p>შუალედური ხდომილებისათვის, რომელიც</p>

	<p>მიდგმულია მოქმედების საზღვარზე:</p> <ul style="list-style-type: none"> ეს ხდომილება “იჭერს” (“catches”) კომპენსაციას. შემდეგი ინფორმაცია არ მოითხოვება. მოქმედება, რომელზედაც ხდომილება არის მიდგმული, უზრუნველყოფს Id-ს, რომელიც საჭიროა კომპენსაციის ხდომილების მოსანიშნად ხდომილებით, რომელიც “გადააგდებს” (“threw”) კომპენსაციას.
--	---

3.3.5.4. შეცდომის ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია შეცდომის ხდომილება დეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილების დეტალების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.13. შეცდომის ხდომილება დეტალის ატრიბუტები

ატრიბუტები	აღწერა
<p>შეცდომა კოდი : სტრიქონი ErrorCode : String</p>	<p>დამთავრების ხდომილებისათვის:</p> <ul style="list-style-type: none"> თუ შედეგი არის შეცდომა, მაშინ შეცდომა კოდი უნდა იყოს შემოტანილი. ეს “გადააგდებს” (“throws”) შეცდომას. <p>შუალედური ხდომილებისათვის, რომელიც მიდგმულია მოქმედების საზღვარზე:</p> <ul style="list-style-type: none"> თუ გადამრთველი (Trigger) არის შეცდომა, მაშინ შეცდომა კოდი შეიძლება იყოს შემოტანილი. ეს ხდომილება “იჭერს” (“catches”) შეცდომას. თუ არა გვაქვს შეცდომა კოდი, მაშინ ნებისმიერი შეცდომა გადართავს (trigger) ხდომილებას. თუ გვაქვს შეცდომა კოდი, მაშინ მხოლოდ შეცდომა, რომელიც შეესაბამება შეცდომა კოდს, გადართავს ხდომილებას.

3.3.5.5. რგოლი ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია რგოლი ხდომილებადეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილების დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.14. რგოლის ხდომილებადეტალის ატრიბუტები

ატრიბუტები	აღწერა
სახელი : სტრიქონი Name: String	თუ გადამრთველი არის რგოლი, მაშინ სახელი უნდა იყოს შემოტანილი.

3.3.5.6. შეტყობინების ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია შეტყობინების ხდომილებადეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილების დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.15. შეტყობინების ხდომილებადეტალის ატრიბუტები

ატრიბუტები	აღწერა
შეტყობინებაRef : შეტყობინება MessageRef : Message	თუ ხდომილებადეტალიტიპი არის შეტყობინებაRef-ი, მაშინ შეტყობინება უნდა იყოს შემოტანილი. შეტყობინების ატრიბუტებს აქ არ განვიხილავთ.
შესრულება (ვებ სერვისი სხვა არა სპეციფიცირებული) ვებ სერვისი : სტრიქონი Implementation (Web Service Other Unspecified) Web Service : String	ეს ატრიბუტი განსაზღვრავს ტექნოლოგიას, რომელიც იქნება გამოყენებული შეტყობინების გასაგზავნად ან მისაღებად. ვებ სერვისი არის გამოუცხადებელი (default) ტექნოლოგია.

3.3.5.7. სიგნალის ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია სიგნალის ხდომილებადეტალის ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილება დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.16. სიგნალის ხდომილებადეტალის ატრიბუტები

ატრიბუტები	აღწერა
სიგნალიRef : სიგნალი SignalRef : Signal	თუ გადამრთველი არის სიგნალი, მაშინ სიგნალი იქნება შემოტანილი. სიგნალის ატრიბუტებს აქ არ განვიხილავთ.

3.3.5.8. ტაიმერის ხდომილების დეტალები

შემდეგ ცხრილში ნაჩვენებია ტაიმერის ხდომილება-დეტალების ატრიბუტების სიმრავლე, რომელიც აფართოებს ხდომილება დეტალების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.10).

ცხრილი 3.17. ტაიმერის ხდომილება-დეტალების ატრიბუტები

ატრიბუტები	აღწერა
დროთარიღი (0-1) : დროთარიღი-გამოსახულება TimeDate (0-1) : TimeDateExpression	<ul style="list-style-type: none"> • თუ გადამრთველი არის ტაიმერი, მაშინ დროთარიღი შეიძლება იყოს შემოტანილი. • თუ დროთარიღი არ არის შემოტანილი, მაშინ დროციკლი უნდა იყოს შემოტანილი (იხ. ატრიბუტები ქვემოთ). დროთარიღი-გამოსახულების ატრიბუტებს აქ არ განვიხილავთ.
დროციკლი (0-1) : დროთარიღი-გამოსახულება TimeCycle (0-1) : TimeDateExpression	<ul style="list-style-type: none"> • თუ გადამრთველი არის ტაიმერი, მაშინ ტაიმერიციკლი შეიძლება იყოს შემოტანილი. • თუ ტაიმერიციკლი არ არის შემოტანილი, მაშინ ტაიმერთარიღი უნდა იყოს შემოტანილი (იხ. ატრიბუტები ზემოთ).

3.4. მოქმედებები

მოქმედება არის მუშაობა, რომელიც შესრულებულია ბიზნეს პროცესის შიგნით. მოქმედება შეიძლება იყოს ატომური (ელემენტარული) ან არა ატომური (რთული). მოქმედებების ტიპები, რომლებიც არიან ბიზნეს პროცესს დიაგრამის ნაწილი, არიან: *პროცესი (Process)*, *ქვე-პროცესი (Sub-Process)* და *ამოცანა (Task)*. თუმცა, პროცესი არ არის განსაზღვრული (კონკრეტული) გრაფიკული ობიექტი. პირიქით, ის არის გრაფიკული ობიექტების სიმრავლე. შემდეგ პარაგრაფებში მოცემული იქნება ქვე-პროცესის და ამოცანის გრაფიკული ობიექტები. პროცესების შესახებ მეტი ინფორმაცია შეიძლება ნახოთ პარაგრაფში 2.6, “პროცესები”, გვ. 37.

3.4.1. მოქმედების საერთო ატრიბუტები

შემდეგ ცხრილში მოცემულია ატრიბუტების სიმრავლე, რომლებიც საერთოა ქვე-პროცესისთვისაც და ამოცანისათვისაც და რომელიც აუარყოფს ნაკადი ობიექტის ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.3). შევნიშნოთ, რომ ცხრილი 3.19 და ცხრილი 3.20 შეიცავს დამატებით ატრიბუტებს, რომლებიც უნდა იყვნენ ჩართული ამ სიმრავლეში თუ მოხდა გაფართოება რაიმე სხვა ატრიბუტების ცხრილით:

ცხრილი 3.18. მოქმედების საერთო ატრიბუტები

ატრიბუტები	აღწერა
მოქმედებატიპი (ამოცანა ქვე-პროცესი) ამოცანა : სტრიქონი ActivityType (Task Sub-Process) Task : String	მოქმედებატიპი უნდა იყოს ამოცანის ან ქვე-პროცესის ტიპის.
სტატუსი (ცარიელი მზა აქტიური გაუქმებული შეჩერება შეჩერებული დამთავრება დამთავრებული) ცარიელი : სტრიქონი Status (None Ready Active Cancelled Aborting Aborted Completing Completed) None : String	მოქმედების სტატუსი არის განსაზღვრული როდესაც მოქმედება შესრულდა პროცესის მამოძრავებლის (ძრავის) საშუალებით. მოქმედების სტატუსი შეიძლება იყოს გამოყენებული დანიშნულების გამოსახულების შიგნით (ფარგლებში).
შემსრულებლები (0-n) : სტრიქონი	ერთი ან მეტი შემსრულებელი შეიძლება იყოს შემოტანილი. შემსრულებლის ატრიბუტი განსაზღვრავს რესურსს, რომელიც შესრულდება ან იქნება პასუხისმგებელი მოქმედებისათვის. შემსრულებლის შესასვლელი შეიძლება იყოს კონკრეტული პიროვნების, ჯგუფის, ორგანიზაციის როლის ან პოზიციის, ან ორგანიზაციის ფორმაში.
თვისებები (0-n) : თვისება Properties (0-n) : Property	მოდელირ-განმსაზღვრელის თვისებები შეიძლება დამატებული იყოს მოქმედებებზე. ეს თვისებები არის “ლოკალური” მოქმედებებთან მიმართებაში. ეს თვისებები გამოიყენებიან მხოლოდ მოქმედების დამუშავებაში. ამ თვისებების სრულად განსაზღვრული სახელი არის “<პროცესის სახელი>.<მოქმედების სახელი>.<თვისების სახელი>” (მაგალითად, “მყიდველის დამატება.კრედიტის მიმოხილვა.სტატუსი”).
შეტანასიმრავლეები (0-n) : შეტანასიმრავლე InputSets (0-n) : InputSet	შეტანასიმრავლეების ატრიბუტი განსაზღვრავს მოქმედებაში შესატან მონაცემებზე მოთხოვნებს. შეიძლება იყოს განსაზღვრული ნოლი ან მეტი შეტანასიმრავლეები. ყოველი შეტანასიმრავლე არის საკმარისი მოქმედების შესასრულებლად.
გამოტანასიმრავლეები (0-n) : OutputSets (0-n) : OutputSet	გამოტანასიმრავლეების ატრიბუტი განსაზღვრავს მოქმედებიდან გამოტანისათვის მონაცემებზე მოთხოვნებს. შეიძლება იყოს განსაზღვრული ნოლი ან მეტი გამოტანასიმრავლეები. მოქმედების დასრულებისას გა-

	მოტანასიმრავლეებიდან მხოლოდ ერთი შეიძლება იყოს გამომუშავებული. ის არის მოქმედების შესრულების ზემოთ და განსაზღვრავს რომელი სიმრავლე იქნება გამომუშავებული. თუმცა, IOწესების ატრიბუტი შეიძლება მიუთითებდეს გამოტანასიმრავლესა და შეტანასიმრავლეს (რომელმაც დაიწყო მოქმედება) შორის ურთიერთდამოკიდებულებას.
IOწესები (0-n) : გამოსახულება IORules (0-n) : Expression	IOწესების ატრიბუტი არის გამოსახულებების კრებული, რომელთაგან ყოველი აზუსტებს მოთხოვნილ ურთიერთკავშირს ერთ შეტანა და ერთ გამოტანას შორის. ეს ნიშნავს, თუ მოქმედება არის დაწყებული სპეციფიური შეტანით, ეს მოქმედება დასრულდება სპეციფიური გამოტანით.
დაწყებარაოდენობა 1 : მთელი StartQuantity 1 : Integer	გამოუცხადებლად მნიშვნელობა არის 1. მნიშვნელობა არ უნდა იყოს ერთზე ნაკლები. ეს ატრიბუტი განსაზღვრავს სიმბოლოების (Tokens) რაოდენობას, რომლებიც უნდა მოვიდნენ სანამ მოქმედება შეძლებს დაწყებას.
დამთავრებარაოდენობა 1 : მთელი CompletionQuantity 1 : Integer	გამოუცხადებლად მნიშვნელობა არის 1. მნიშვნელობა არ უნდა იყოს ერთზე ნაკლები. ეს ატრიბუტი განსაზღვრავს სიმბოლოების (Tokens) რაოდენობას, რომელიც უნდა იყოს გენერირებული მოქმედებიდან. გაგზავნილი სიმბოლოების ეს რაოდენობა შექმნის რაიმე გამომავალ შესრულების ნაკადს (იმ დაშვებით, რომ რაიმე შესრულების ნაკადის პირობები არის დაკმაყოფილებული).
მარყუჟიტიპი (ცარიელი სტანდარტი მრავალშემთხვევიანი) ცარიელი : სტრიქონი LoopType (None Standard MultiInstance) None : String	მარყუჟიტიპი არის ატრიბუტი და გამოუცხადებლად არის ცარიელი, მაგრამ შეიძლება იყოს დაყენებული სტანდარტზე ან მრავალშემთხვევიანზე. თუ ეს ასეა, მარყუჟის მარკერი იქნება გადაადგილებული მოქმედების ფორმის ქვემო ცენტრისაკენ (იხ. ნახ. 3.9 და ნახ. 3.15). ტიპის მიღების ამოცანას, რომელსაც აქვს მისი მყისიერი ატრიბუტი დაყენებული ჭეშმარიტ მნიშვნელობაზე, არ უნდა ჰქონდეს მარყუჟის ტიპი სტანდარტული ან მრავალშემთხვევიანი.

3.4.1.1. მარყუჟის სტანდარტული ატრიბუტები

მარყუჟის სტანდარტულ მოქმედებას ექნება ბულის გამოსახულება, რომელიც შეფასებულია მარყუჟის ყოველი ციკლის შემდეგ. თუ გამოსახულება არის

ჭეშმარიტი, მაშინ მარყუჟი გაგრძელდება. მარყუჟის ორი ვარიანტი არსებობს, რომლებიც ასახავენ პროგრამულ კონსტრუქციებს while-ს და until-ს. ეს ნიშნავს, რომ while მარყუჟი შეაფასებს გამოსახულებას მოქმედების შესრულების წინ, რაც ნიშნავს, რომ მოქმედება სინამდვილეში შეიძლება არ შესრულდეს. until მარყუჟი შეაფასებს გამოსახულებას მოქმედების შესრულების შემდეგ, რაც ნიშნავს, რომ მოქმედება შესრულდება სულ მცირე ერთხელ (ანუ ერთხელ მაინც).

შემდეგი არის სტანდარტული მარყუჟის მოქმედების დამატებითი ატრიბუტები (სადაც მარყუჟი ატრიბუტი არის დაყენებული “სტანდარტზე”), რაც აფართოებს მოქმედებების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.18):

ცხრილი 3.19. მარყუჟის მოქმედების სტანდარტული ატრიბუტები

ატრიბუტები	აღწერა
მარყუჟიპირობა : გამოსახულება LoopCondition : Expression	სტანდარტულ მარყუჟებს უნდა ჰქონდეთ შეფასებული ბულის გამოსახულება, პლიუს არჩეული დრო, როდესაც გამოსახულება იქნება შეფასებული. გამოსახულების ატრიბუტებს აქ არ განვიხილავთ.
მარყუჟიმთვლეელი : მთელი LoopCounter : Integer	მარყუჟიმთვლეელი ატრიბუტი გამოიყენება გაშვებისას (დაწყებისას) მარყუჟების რიცხვის დასათვლელად და ავტომატურად არის განახლებული პროცესის ძრავით. მარყუჟიმთვლეელი ატრიბუტი უნდა იყოს გაზრდილი მარყუჟის დაწყებისას. მოდელიორს შეუძლია გამოიყენოს ატრიბუტი მარყუჟიპირობის გამოსახულებაში.
მარყუჟიმაქსიმუმი (0-1) : მთელი LoopMaximum (0-1) : Integer	მაქსიმუმი არის არა სავალდებულო ატრიბუტი. ის შეიძლება დამატებული იყოს მარყუჟიპირობაში განსაზღვრულ გამოსახულებაზე.
ტესტიდრო (წინ შემდეგ) შემდეგ : სტრიქონი TestTime (Before After) After : String	<p>გამოსახულებები, რომლებიც არიან შეფასებული მოქმედების დაწყების წინ, არიან პროგრამირების while ფუნქციის ექვივალენტი.</p> <p>გამოსახულებები, რომლებიც არიან შეფასებული მოქმედების დამთავრების შემდეგ, არიან პროგრამირების until ფუნქციის ექვივალენტი.</p>

3.4.1.2. მარყუჟის მრავალ-შემთხვევიანი ატრიბუტები

მრავალ-შემთხვევიანი მარყუჟები ასახავენ პროგრამულ კონსტრუქციას ყოველი შემთხვევისათვის. მრავალ-შემთხვევიანი მარყუჟის მარყუჟი გამოსახულება არის რიცხვითი გამოსახულება შეფასებული მხოლოდ ერთხელ, მოქმედების შესრულების წინ. გამოსახულების შეფასების შედეგი იქნება მთელი რიცხვი, რაც განსაზღვრავს დროის მნიშვნელობას, როდესაც მოქმედება იქნება გამეორებული.

მრავალ-შემთხვევიანი მარყუჟის კოდექსი ორი ვარიანტი არსებობს, სადაც შემთხვევები არიან შესრულებული ან მიმდევრობით, ან პარალელურად.

ქვემოთ მოყვანილი არის მრავალ-შემთხვევიანი მარყუჟი მოქმედების დამატებითი ატრიბუტები (სადაც მარყუჟიტიპი ატრიბუტი არის დაყენებული “მრავალი-შემთხვევა“-ზე), რომლებიც აფართოებენ მოქმედების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.18):

ცხრილი 3.20. მრავალ-შემთხვევიანი (MI) მარყუჟის მოქმედების ატრიბუტები

ატრიბუტები	აღწერა
MI-პირობა : გამოსახულება MI_Condition : Expression	მრავალ-შემთხვევიან მარყუჟებს უნდა ჰქონდეთ რიცხვითი გამოსახულება შესაფასებლად; გამოსახულება უნდა დებულობდეს მთელ მნიშვნელობას. გამოსახულების ატრიბუტები აქ არ მოგვეყვას.
მარყუჟიმთვლელი : მთელი LoopCounter : Integer	მარყუჟიმთვლელის ატრიბუტი არის გამოყენებული მხოლოდ მიმდევრობითი მრავალ-შემთხვევიანი მარყუჟებისათვის და პროცესებისათვის, რომლებსაც დაწესებული აქვთ შესრულება პროცესის მამოძრავებელიდან (მოტორიდან). ატრიბუტი განახლებულია პროცესის ძრავით გაშვების (დაწყების) მომენტში, მარყუჟის რიცხვის დასათვლელად იმ მიმდევრობით როგორცაც ისინი წარმოიქმნებიან. მარყუჟიმთვლელი ატრიბუტი უნდა იყოს გაზრდილი მარყუჟის დაწყებისას. სტანდარტული მარყუჟისაგან განსხვავებით, მოდელიორი არ იყენებს ამ ატრიბუტს MI-პირობის გამოსახულებაში, მაგრამ ის შეიძლება იყოს გამოყენებული მარყუჟის სტატუსისათვის თვალის მისაღწენებლად.
MI-მოწესრიგება (მიმდევრობითი პარალელური) მიმდევრობითი : სტრიქონი MI_Ordering (Sequential Parallel) Sequential : String	ეს გამოიყენება მხოლოდ მრავალ-შემთხვევიან მარყუჟებში. MI-მოწესრიგების ატრიბუტი განსაზღვრავს მარყუჟის შემთხვევები იქნებიან შესრულებული მიმდევრობით თუ პარალელურად. მიმდევრობითი MI-მოწესრიგება არის უფრო ტრადიციული მარყუჟი. პარალელური MI-მოწესრიგება არის მრავალი-შემთხვევიანი სპეციფიკაციების ექვივალენტი. თუ ის დაყენებულია პარალელურზე, პარალელური მარკერი გადაადგილებს მარყუჟის მარკერს მოქმედების ფორმის ქვედა ცენტრში (იხ. ნახ. 3.9 და ნახ. 3.18).
[პარალელური MI-მოწესრიგება მხოლოდ] MI-ნაკადიპირობა (ცარიელი ერთი ყველა რთული) ყველა : სტრიქონი	ეს ატრიბუტი გასასვლელების (Gateway) გამოყენების ექვივალენტურია პარალელური ბილიკების სიმრავლეში გასული ნაკადის

<p>[Parallel MI_Ordering only] MI_FlowCondition (None One All Complex) All : String</p>	<p>მართვისათვის.</p> <p>MI–ნაკადიპირობა დაყენებული “ცარიელი”-ზე (None) იგივეა რაც არა კონტროლირებადი ნაკადი (არა გასასვლელი) და ნიშნავს, რომ მოქმედების ყველა შემთხვევა გენერირებას გაუკეთებენ სიმბოლოს (token), რომელიც გააგრძელება მაშინ, როდესაც ეს შემთხვევა დამთავრდება.</p> <p>MI–ნაკადიპირობა დაყენებული “ერთზე” იგივეა რაც ერთადერთი გასასვლელი და ნიშნავს, რომ სიმბოლო გააგრძელებს წარსულ მოქმედებას იმის შემდეგ რაც მხოლოდ მოქმედების ერთი შემთხვევა დამთავრდება. მოქმედება გააგრძელებს მის სხვა შემთხვევებს, მაგრამ დამატებითი სიმბოლოები (Tokens) არ უნდა იყვნენ გამოსული მოქმედებიდან.</p> <p>MI–ნაკადიპირობა დაყენებული “ყველაზე” იგივეა რაც პარალელური გასასვლელი და ნიშნავს, რომ სიმბოლო (Token) გააგრძელებს წარსულ მოქმედებას იმის შემდეგ, რაც მოქმედების ყველა შემთხვევა დამთავრდებიან.</p> <p>MI–ნაკადიპირობა დაყენებული “რთულზე” არის რთული გასასვლელის ანალოგიური. რთული MI–ნაკადიპირობა ატრიბუტი განსაზღვრავს სიმბოლოების ნაკადს.</p>
<p>[რთული MI–ნაკადიპირობა მხოლოდ] რთული MI–ნაკადიპირობა (0-1) : გამოსახულება</p> <p>[Complex MI_FlowCondition only] ComplexMI_FlowCondition (0-1) : Expression</p>	<p>თუ MI–ნაკადიპირობა ატრიბუტი არის განსაზღვრული როგორც “რთული”, მაშინ გამოსახულება უნდა იყოს შემოტანილი. ამ გამოსახულებას შეუძლია მიუთითოს პროცესის მონაცემები. გამოსახულება შეფასებული იქნება მოქმედების ყოველი იტერაციის შემდეგ და მიიღებს ბულის მნიშვნელობას.</p> <p>თუ გამოსახულების შეფასების შედეგი არის ჭეშმარიტი, მაშინ სიმბოლო გაიგზავნება ქვემოთ, მოქმედების გამოსასვლელი შესრულების ნაკადისაკენ. სხვანაირად, ამ იტერაციისათვის სიმბოლო არ გაიგზავნება. გამოსახულების ატრიბუტი აქ არ მოგვყავს.</p>

MI – MultiInstance (მრავალი შემთხვევა)

3.4.2. ქვე-პროცესი

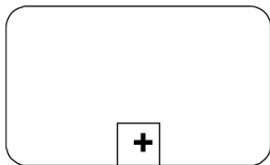
ქვე-პროცესი არის შედგენილი (რთული) მოქმედება რომელშიც მას აქვს დეტალი, რომელიც არის განსაზღვრული როგორც სხვა მოქმედებების ნაკადი. ქვე-

პროცესი არის გრაფიკული ობიექტი პროცესის ნაკადის ფარგლებში, მაგრამ ის აგრეთვე შეიძლება იყოს “გახსნილი” სხვა პროცესის საჩვენებლად (ან ჩადგმულის (Вложенный) ან მრავალჯერადი გამოყენების). ქვე-პროცესის ობიექტს აქვს იგივე ფორმა რაც ამოანის ობიექტს აქვს. ეს არის კუთხეებ მომრგვალებული სწორკუთხედი.

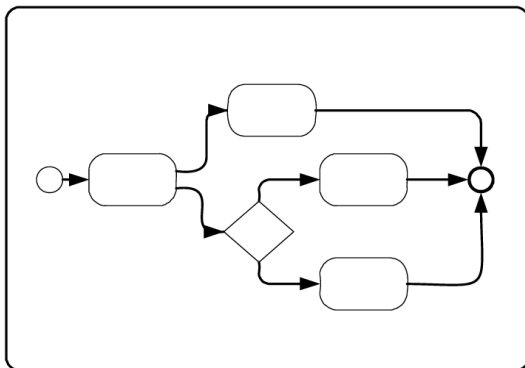
- ქვე-პროცესი არის მომრგვალებულ კუთხეებიანი სწორკუთხედი, რომელიც უნდა იყოს დახაზული ერთი წვრილი შავი ხაზით.
 - ქვე-პროცესებისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში”, გვ. 33, იმის გამოკლებით, რომ:
 - საზღვარი დახაზული ორი ხაზით იქნება დარეზერვირებული ქვე-პროცესისათვის, რომელსაც ატრიბუტი არის შესრულება (IsATransaction) დაყენებული აქვს მნშვნელობაზე ჭეშმარიტი (True).

ქვე-პროცესი შეიძლება იყოს დანგრეული სახით (a collapsed view), რაც მაღავს მის დეტალებს (იხ. ნახ. 3.6) ან ქვე-პროცესი შეიძლება იყოს გაფართოებული სახით, რაც აჩვენებს მის დეტალებს პროცესის ფორმაში რომელშიც ის შედის (იხ. ნახ. 3.7). დანგრეულ ფორმაში ქვე-პროცესის ობიექტი იყენებს მარკერს, რომელიც მას გამოარჩევს ამოცანისაგან როგორც ქვე-პროცესს.

- ქვე-პროცესის მარკერი უნდა იყოს პატარა კვადრეტი შუაში ჩაწერილი პლიუს (+) ნიშნით. კვადრეტი უნდა იყოს განთავსებული ფორმის ქვედა ცენტრში.



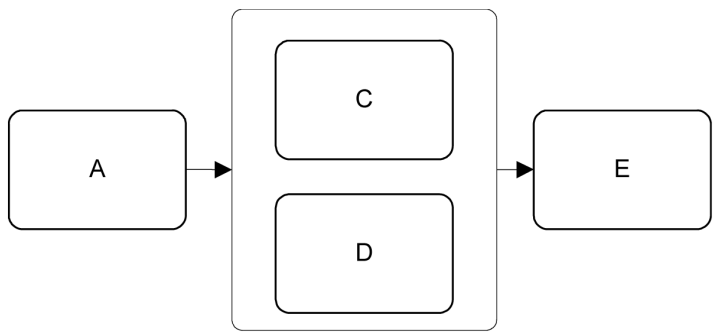
ნახ. 3.6. დანგრეული ქვე-პროცესი.



ნახ. 3.7. გაფართოებული ქვე-პროცესი.

გაფართოებული ქვე-პროცესი შეიძლება იყოს გამოყენებული მრავალი მიზნისათვის. ისინი შეიძლება იყვნენ გამოყენებული იერარქიული პროცესების გასასწორებლად (“გასაუთოებლად”) (“to flatten”) ისე, რომ ყველა დეტალი ნაჩვენები იყოს ერთდროულად. ისინი გამოიყენება კონტექსტის შესაქმნელად გამონაკლისის განმარტებისათვის, რომელიც გამოიყენება მოქმედებების ჯგუფთან (მეტი დეტალებისათვის იხ. პარაგრაფი 4.2.2, “გამონაკლისი ნაკადი”, გვ. 174). კომპენსაციები შეიძლება იყვნენ გამოყენებული ანალოგიურად (მეტი დეტალებისათვის იხ. პარაგრაფი 4.3, “კომპენსაციის გაერთიანება”, გვ. 177).

გაფართოებული ქვე-პროცესი შეიძლება იყოს გამოყენებული როგორც პარალელური მოქმედებების ჯგუფის საჩვენებელი მექანიზმი, როგორც უფრო კომპაქტური გზა. ნახ. 3.8-ზე მოქმედებები “C” და “D” არიან შემოფარგლული (enclosed) დაუჭდევებელ (unlabeled) გაფართოებულ ქვე-პროცესში. ეს ორი მოქმედება შესრულდება პარალელურად. შევნიშნოთ, რომ გაფართოებული ქვე-პროცესი არ შეიცავს დაწყების ხდომილებას (Start Event) ან დამთავრების ხდომილებას (End Event) და შესრულების ნაკადი მიედინება ამ ხდომილებებისაკენ ან გამოედინება ამ ხდომილებებიდან. “პარალელური ყუთებისათვის” გაფართოებული ქვე-პროცესების ეს გამოყენება არის დაწყების და დამთავრების ხდომილებების ქონის მოტივაცია, რომლებიც არიან არა სავალდებულო ობიექტები.

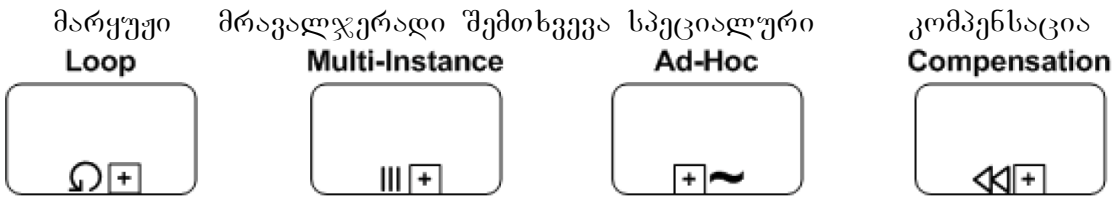


ნახ. 3.8. გაფართოებული ქვე-პროცესი გამოყენებული როგორც “პარალელური ყუთი”.

ქვე-პროცესებისათვის BPMN განსაზღვრავს სტანდარტული მარკერების ხუთ ტიპს. (დანგრეული) ქვე-პროცესის მარკერი, ნაჩვენები ნახ. 3.6-ზე, შეიძლება გაერთიანებული იყოს სხვა ოთხ მარკერთან: მარყუჟი მარკერი ან პარალელური მარკერი, კომპენსირება მარკერი და სპეციალური (Ad Hoc) მარკერი. დანგრეულ ქვე-პროცესს შეიძლება ჰქონდეს ერთიდან სამამდე ეს მარკერები ნებისმიერ კომბინაციაში, იმის გამოკლებით, რომ მარყუჟი და მრავლობითი მაგალითი არ შეიძლება იყოს ნაჩვენები ერთდროულად (იხ. ნახ. 3.9).

- ქვე-პროცესის მარკერი, რომელიც აკეთებს მარყუჟს, უნდა იყოს პატარა ისრის თავიანი ხაზი, რომელიც თვითონვე არის უკან ჩახვეული.
 - მარყუჟის მარკერი შეიძლება გამოყენებული იყოს სხვა რომელიმე მარკერთან ერთად, გარდა მრავალჯერადი შემთხვევების (Multiple Instance) მარკერისა.

- მრავალჯერადი შემთხვევების ქვე-პროცესის მარკერი არის სამი ვერდიკალური პარალელური ხაზი.
 - მრავალჯერადი შემთხვევების მარკერი შეიძლება იყოს გამოყენებული ნებისმიერ სხვა მარკერებთან კომბინაციაში გარდა მარყუჟი მარკერისა.
- სპეციალური (Ad Hoc) ქვე-პროცესის მარკერი არის “ტილდა” სიმბოლო.
 - სპეციალური (Ad Hoc) ქვე-პროცესის მარკერი შეიძლება გამოყენებული იყოს ნებისმიერ სხვა მარკერებთან კომბინაციაში.
- კომპენსაციისათვის ქვე-პროცესის მარკერი არის მარცხნივ მიმართული წყვილი სამკუთხედი.
 - კომპენსაციის მარკერი შეიძლება იყოს გამოყენებული ნებისმიერ სხვა მარკერებთან კომბინაციაში.
- ყველა წარმოდგენილი მარკერები შეიძლება იყვნენ დაჯგუფებული და მთლიანი ჯგუფი იყოს ცენტრირებული ფორმის ქვემოთ.



ნახ. 3.9. დანერგული ქვე-პროცესის მარკერები.

3.4.2.1. ატრიბუტები

ცხრილში მოცემულია ქვე-პროცესის ატრიბუტების სიმრავლე, რომელიც აფართოებს მოქმედების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.21).

ცხრილი 3.21. ქვე-პროცესის ატრიბუტები.

ატრიბუტები	აღწერა
<p>ქვეპროცესიტიპი (ჩადგმული მრავალჯერადი გამოყენების მითითება) ჩადგმული: სტრიქონი SubProcessType (Embedded Reusable Reference) Embedded : String</p>	<p>ქვეპროცესიტიპი არის ატრიბუტი, რომელიც განსაზღვრავს ქვე-პროცესის დეტალები არის ჩადგმული უმაღლესი დონის პროცესში თუ მიეკუთვნება სხვა, ხელმეორედ გამოყენებულ პროცესს. გამოუცხადებლად (default) არის ჩადგმული.</p>
<p>არისAშესრულება მცდარი : ბულის IsATransaction False : Boolean</p>	<p>არისAშესრულება განსაზღვრავს ქვე-პროცესის ქცევა მიჰყვება თუ არა შესრულების ქცევას (იხ. 3.4.2.5, “ქვე-პროცესის ქცევა როგორც შესრულება, გვ. 86”).</p>
<p>შესრულებაRef (0-1) : შესრულება TransactionRef (0-1) : Transaction</p>	<p>თუ არისAშესრულება ატრიბუტი არის მცდარი (False), მაშინ შესრულება არ უნდა იყოს იდენტიფიცირებული. თუ არისAშეს-</p>

	რულების ატრიბუტი არის ჭეშმარიტი (True), მაშინ შესრულება (პროტოკოლი) უნდა იყოს იდენტიფიცირებული. შევნიშნოთ, რომ შესრულებებს (პროტოკოლებს), რომლებიც არიან სხვადასხვა გუბეში (Pool) და დაკავშირებული არიან შეტყობინების ნაკადის საშუალებით, უნდა ჰქონდეთ ერთი და იგივე შესრულებაId (პროტოკოლიId).
--	---

3.4.2.2. ჩადგმული ქვე-პროცესი

ჩადგმული (ან დაბუდებული) ქვე-პროცესის ობიექტი არის მოქმედება, რომელიც შეიცავს სხვა მოქმედებებს (პროცესს). პროცესი პროცესში დამოკიდებულია მშობელი პროცესის მითითებაზე და ხედავს მშობელი პროცესის გლობალურ მონაცემებს. მოითხოვება მონაცემების არ გამოხაზვა (განფენა) (no mapping).

ჩადგმული ქვე-პროცესის შიგნით ობიექტებს, რომლებიც დამოკიდებული არიან მათი მშობელი პროცესებისაგან, არა აქვთ მთლიანი ბიზნეს პროცესის დიაგრამის მახასიათებლები, ისეთები როგორცაა გუბეები და ბილიკები. ამრიგად, ჩადგმული ქვე-პროცესის გაფართოებული სახე შეიცავს მხოლოდ ნაკადის ობიექტებს (Flow Objects), შეერთების ობიექტებს (Connecting Objects) და არტიფაქტებს (Artifacts) (იხ. ნახ. 3.8).

- ჩადგმული ქვე-პროცესებისათვის ყველა დაწყების ხდომილებები უნდა იყვნენ ცარიელი (None) ტიპის.

ქვემოთ მოცემული არიან ჩადგმული ქვე-პროცესის დამატებითი ატრიბუტები (სადაც ქვე-პროცესიტიპი ატრიბუტი არის განსაზღვრული როგორც “ჩადგმული”), რომლებიც აფართოებენ ქვე-პროცესის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.22).

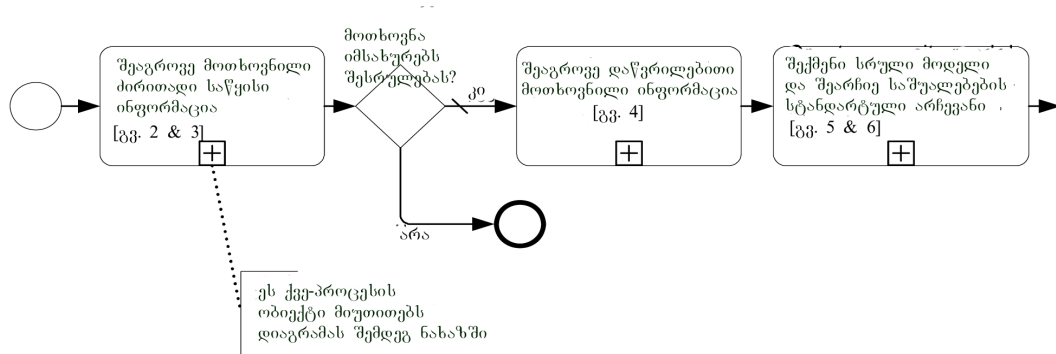
ცხრილი 3.22. ჩადგმული ქვე-პროცესის ატრიბუტები.

ატრიბუტები	აღწერა
გრაფიკულიელემენტები (0-n) : ობიექტი GraphicalElements (0-n) : Object	ატრიბუტი გრაფიკულიელემენტები იდენტიფიცირებას უკეთებს ყველა ობიექტს (მაგალითად, ხდომილებები, მოქმედებები, გასასვლელელები და არტიფაქტები) (Events, Activities, Gateways, and Artifacts), რომლებსაც შეიცავს ჩადგმული ქვე-პროცესი.
AdHoc არა ჭეშმარიტი : ბულის AdHoc False : Boolean	სპეციალური (AdHoc) არის ბულის ატრიბუტი, რომელსაც გამოუცხადებლად აქვს არა ჭეშმარიტი (False) მნიშვნელობა. ეს განსაზღვრავს არის თუ არა ჩადგმული ქვე-პროცესი სპეციალური . სპეციალური დაფარული ქვე-პროცესის შიგნით მოქმედებები არ არიან კონტროლირებადი ან მოწესრიგე-

	ბული სპეციფიური მიმდევრობით, მათი შესრულება განსაზღვრულია მოქმედებების შესრულებებით.
[AdHoc = მხოლოდ ჭეშმარიტი] სპეციალური მოწვევრიგება (0-1) (მიმდევრობითი პარალელური) პარალელური : სტრიქონი [AdHoc = True only] AdHocOrdering (0-1) (Sequential Parallel) Parallel : String	თუ დაფარულ ქვე-პროცესი არის სპეციალური (AdHoc ატრიბუტი არის ჭეშმარიტი), მაშინ სპეციალური მოწვევრიგება ატრიბუტი უნდა იყოს ჩართული. ეს ატრიბუტი განსაზღვრავს მოქმედებები პროცესის შიგნით შეიძლება შესრულდნენ პარალელურად თუ უნდა შესრულდნენ მიმდევრობით. გამოუცხადებელის დაყენება არის პარალელური და მიმდევრობითის დაყენება არის შეზღუდვა შესრულებაზე, რომელიც შეიძლება იყოს მოთხოვნილი განაწილებული (დაყოფილი) რესურსების გამო.
[AdHoc = მხოლოდ ჭეშმარიტი] AdHocდამთავრებაპირობა (0-1) : გამოსახულება [AdHoc = True only] AdHocCompletionCondition (0-1) : Expression	თუ ჩადგმული ქვე-პროცესი არის სპეციალური (AdHoc ატრიბუტი არის ჭეშმარიტი), მაშინ დამთავრების პირობა უნდა იყოს ჩართული, რომელიც განსაზღვრავს პროცესის დამთავრების პირობებს. AdHoc მარკერი განთავსდება პროცესის ქვედა ცენტრში ან ქვე-პროცესის ფორმის ქვედა ცენტრში სპეციალური (AdHoc) პროცესებისათვის.

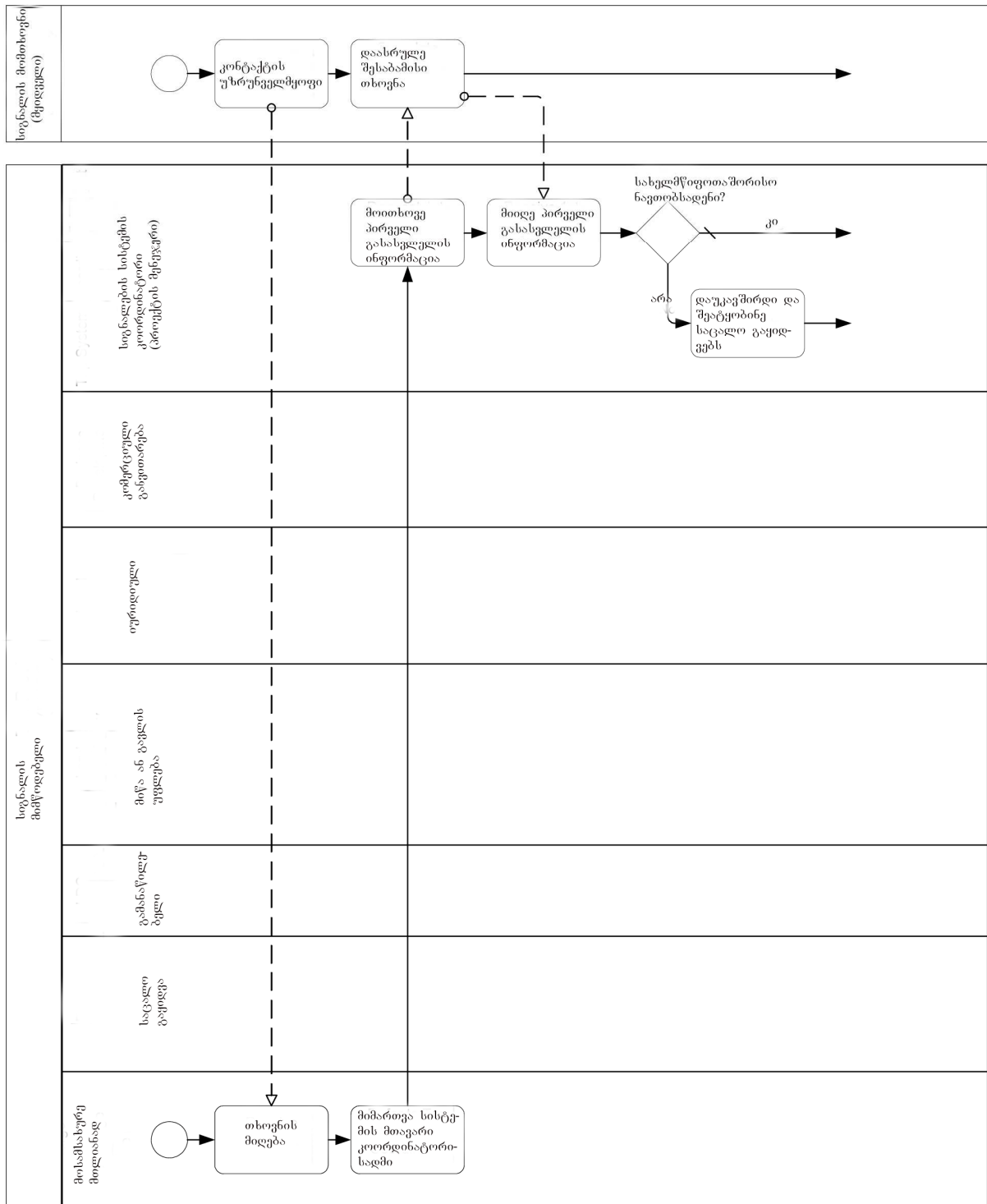
3.4.2.3. მრავალჯერადი გამოყენების ქვე-პროცესი

მრავალჯერადი გამოყენების ქვე-პროცესის ობიექტი არის მოქმედება პროცესის შიგნით, რომელიც “იძახებს” სხვა პროცესს, რომელიც არსებობს BDP-ში (ბიზნესის განვითარების პროცესში) (იხ. ნახ. 3.10). გამოძახებული პროცესი არ არის დამოკიდებული მრავალჯერადი გამოყენების ქვე-პროცესის მშობელ პროცესზე გლობალური მონაცემებისათვის. მრავალჯერადი გამოყენების ქვე-პროცესის ობიექტმა შეიძლება გაატაროს მონაცემები გამოძახებული პროცესისაკენ ან პროცესიდან.



ნახ. 3.10. ქვე-პროცესის ობიექტი თავისი დეტალებით, ნაჩვენები შემდეგი ნახაზის დიაგრამაში.

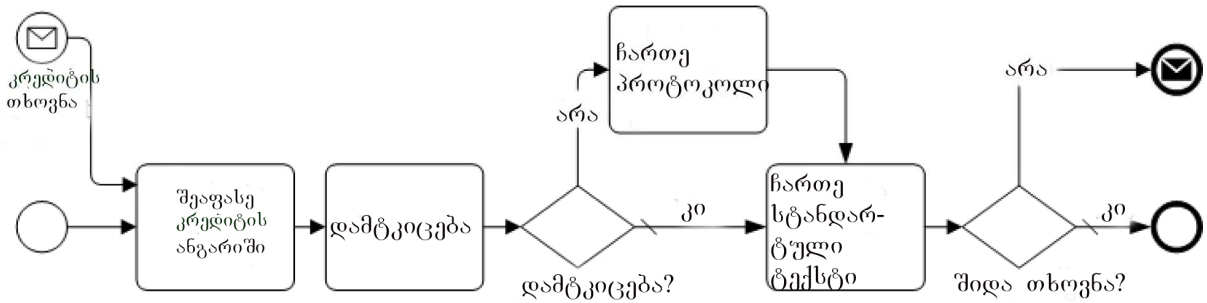
გამოძახებული პროცესი იარსებებს ცალკე დიაგრამაში, რომელსაც შეიძლება ჰქონდეს მრავალი გუბეები. გამოძახებული პროცესის ნებისმიერი ფორმა (view) (გამოძახებულ პროცესში გაფართოებული ფორმის (view) ჩათვლით) აჩვენებს მთელ დიაგრამას, რომელსაც გამოძახებული პროცესი მიეკუთვნება (იხ. ნახ. 3.11), მაგრამ მონაცემების ნებისმიერი განთავსება (განფენა) (mapping) იქნება მხოლოდ გამოძახებულ პროცესთან და არა ნებისმიერთან სხვა პროცესებიდან, რომლებიც შეიძლება იყვნენ გამოძახებულ დიაგრამაში.



ნახ. 3.11. წინა ნახაზის ქვე-პროცესი ობიექტის პროცესის და დიაგრამის დეტალები.

გამოახებული პროცესი იქნება (უნდა იყოს) მყისიერად შესრულებული როგორც ქვე-პროცესი ცარიელზე (None) დაყენებული დაწყების ხდომილების

საშუალებით. არის რა განმეორებით გამოყენებული, პროცესი შეიძლება აგრეთვე იყოს მყისიერად შესრულებული, როგორც ქვე-პროცესი სხვა დამოუკიდებელი ქვე-პროცესის ობიექტების საშუალებით (იმავე ან სხვა დიაგრამებში). გარდა ამისა, ის შეიძლება იყოს მყისიერად შესრულებული, როგორც უმაღლესი დონის პროცესი გამოყოფილი დაწყების ხდომილების საშუალებით, რომელსაც აქვს გადამროველი (Trigger) (დაყენებული არა ცარიელზე (other than None) – იხ. ნახ. 3.12).



ნახ. 9.12. პროცესი, რომელიც გამოყენებულია როგორც ქვე-პროცესი ან უმაღლესი დონის პროცესი.

ქვემოთ მოყვანილი არის ახლიდან მრავალჯერ გამოყენებული (Reusable) პროცესის დამატებითი ატრიბუტები (სადაც ქვეპროცესიტიპი ატრიბუტი დაყენებულია “მრავალჯერ გამოყენება”-ზე), რაც აფართოებს ქვე-პროცესის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.23).

ცხრილი 3.23. მრავალჯერ გამოყენებული პროცესის ატრიბუტები

ატრიბუტები	აღწერა
დიაგრამაRef: ბიზნეს პროცესი დიაგრამა DiagramRef: Business Process Diagram	ბიზნეს პროცესის დიაგრამა (BPD) უნდა იყოს იდენტიფიცირებული. ბიზნეს პროცესის დიაგრამის ატრიბუტები შეიძლება ნახოთ პარაგრაფში 2.5, “ბიზნეს პროცესის დიაგრამების ატრიბუტები”, გვ. 36.
პროცესიRef: პროცესი ProcessRef: Process	პროცესი უნდა იყოს იდენტიფიცირებული. პროცესის ატრიბუტები შეიძლება ნახოთ პარაგრაფში 2.6, “პროცესები”, გვ. 37.
შეტანაგანფენები (0-n) : გამოსახულება InputMaps (0-n) : Expression	მრავალჯერადი შეტანა განფენები (Multiple input mappings) შეიძლება გაკეთდეს მრავალჯერ გამოყენებულ ქვე-პროცესს და ამ ობიექტით მითითებულ პროცესს შორის. ეს განფენები (mappings) მოცემული არიან გამოსახულების ფორმით. სპეციალური განფენის გამოსახულება უნდა განსაზღვრავდეს ორ პროცესს შორის თვისებების განფენას ან ორ პროცესს შორის არტიფაქტების განფენას.

გამოტანაგანფენები (0-n) : გამოსახულება OutputMaps (0-n) : Expression	მრავალჯერადი გამოსვლის განფენები (mapping) შეიძლება გაკეთდეს მრავალჯერ გამოყენებულ ქვე-პროცესსა და ამ ობიექტით მითითებული პროცესს შორის. ამ განფენებს აქვთ გამოსახულების ფორმა. სპეციალური განფენის გამოსახულება უნდა განსაზღვრავდეს ორ პროცესს შორის თვისებების განფენას ან ორ პროცესს შორის არტიფაქტების განფენას.
---	--

3.4.2.4. მითითების (Reference) ქვე-პროცესი

შეიძლება წარმოიშვას სიტუაცია, როდესაც მოდელიორს უნდა მიუთითოს ქვე-პროცესი, რომელიც უკვე იყო განსაზღვრული. თუ ორ ქვე-პროცესს აქვს ზუსტად ერთი და იგივე ქცევა და თვისებები, მაშინ ეს შესაძლებელია გაკეთდეს ერთიდან მეორეზე ატრიბუტების მითითებით, რომლებიც განსაზღვრავენ ქცევას განსაზღვრულს მხოლოდ ერთხელ და ჩაწერილს მხოლოდ ერთ ადგილას.

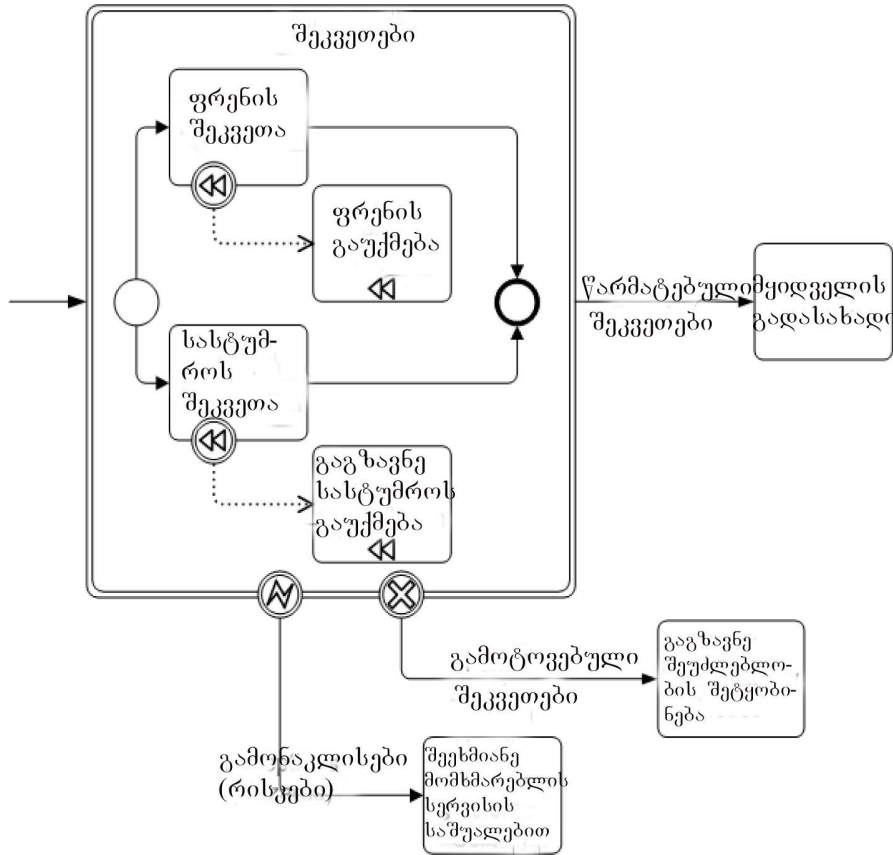
შემდეგ ცხრილში მოყვანილია მითითების ქვე-პროცესის ატრიბუტების სიმრავლე (სადაც ქვე-პროცესიტიპი ატრიბუტი არის დაყენებული “მითითება“-ზე), რომლებიც აფართოებენ ქვე-პროცესის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.24).

ცხრილი 3.24. მითითების ქვე-პროცესის ატრიბუტები

ატრიბუტები	აღწერა
ქვეპროცესიRef : ქვე-პროცესი SubProcessRef : Sub-Process	ქვე-პროცესი, რომელიც არის მითითებული, უნდა იყოს იდენტიფიცირებული. ქვე-პროცესის ელემენტის ატრიბუტები შეიძლება ნახოთ ცხრილში 3.21.

3.4.2.5. ქვე-პროცესის ქცევა როგორც შესრულება

ქვე-პროცესი, ორივე დანგრეულიც და გაფართოებულიც, შეიძლება იყოს დაყენებული შესრულების მდგომარეობაში, რომელსაც ექნება სპეციალური (განსაკუთრებული) ქცევა, რომელიც იმართებს შესრულების პროტოკოლის საშუალებით. მოქმედების საზღვარი იქნება ორმაგი ხაზი, რომელიც მიუთითებს, რომ ის არის შესრულება (იხ. ნახ. 3.13).



ნახ. 3.13. გაფართოებული ქვე-პროცესის შესრულების მაგალითი.

არსებობს შესრულების (of a Transaction) სამი ძირითადი შედეგი:

1. წარმატებული დამთავრება: ეს იქნება ნაჩვენები როგორც ნორმალური შესრულების ნაკადი, რომელიც ტოვებს ქვე-პროცესს.
2. განუხორციელებელი დამთავრება (გაუქმება): როდესაც შესრულება არის გაუქმებული, მაშინ შესრულების შიგნით მოქმედებები იქნებიან შეზღუდული მოქმედებებით, რომლებიც შეიძლება შეიცავდნენ პროცესის უკან შებრუნებას და სპეციფიკური მოქმედებების კომპენსაციას. შევნიშნოდ, რომ ქვე-პროცესის შეწყვეტის სხვა მექანიზმები არ გამოიწვევენ კომპენსაციას (ანუ შეცდომა, ტაიმერი და რაიმე, არ-შესრულების (non-Transaction) მოქმედებისათვის). გაუქმების შუალედური ხდომილება, მიდგმული მოქმედების საზღვარზე, მიმართავს ნაკადს იმის შემდეგ რაც შესრულება შებრუნდება უკან და ყველა კომპენსაცია დასრულდება. გაუქმების შუალედური ხდომილება შეიძლება მხოლოდ გამოყენებული იყოს, როდესაც შესრულება მიდგმულია მოქმედების საზღვარზე. ის არ შეიძლება იყოს გამოყენებული რაიმე ნორმალურ ნაკადში და არ შეიძლება იყოს მიდგმული არ-შესრულების მოქმედებაზე. არსებობს ორი მექანიზმი, რომელსაც შეუძლია მოგვცეს შესრულების გაუქმების სიგნალი:

- დამთავრების ხდომილების გაუქმება არის მიღწეული შესრულების ქვეპროცესის შიგნით. დამთავრების ხდომილების გაუქმება შეიძლება იქნეს გამოყენებული მხოლოდ ქვეპროცესის შიგნით, რომელიც არის დაყენებული შესრულებაზე.
 - გაუქმების შეტყობინება შეიძლება მიღებული იყოს შესრულების პროტოკოლის საშუალებით, რაც უზრუნველყოფს ქვეპროცესის შესრულებას.
3. რისკი (Hazard): ეს ნიშნავს, რომ რაღაც წავიდა ძალიან ცუდად და ნორმალური წარმატება ან გაუქმება არ არის შესაძლებელი. რისკის საჩვენებლად გამოიყენება შეცდომა (Error). როდესაც რისკი ხდება, მოქმედება ჩერდება (კომპენსაციის გარეშე) და ნაკადი გაგრძელდება შეცდომის შუალედური ხდომილებიდან.

წარმატებული შესრულების ქვეპროცესის დამთავრებისას ქცევა არის ცოტათი სხვანაირი ვიდრე ეს ნორმალურ ქვეპროცესშია. როდესაც ყოველი შესრულების ქვეპროცესის ბილიკი აღწევს დამთავრების ხდომილების(ბების) არა-გაუქმებას, ნაკადი დაუყოვნებლივ არ გადაადგილდება უკან, უმაღლესი დონის მშობელი პროცესისაკენ, როგორც ამას აკეთებს ნორმალური ქვეპროცესი. პირველი, შესრულების პროტოკოლმა უნდა შეამოწმოს, რომ ყველა მონაწილეებმა წარმატებულად მიაღწიეს თავიანთი შესრულების დამთავრებას (ბოლოს). უმეტეს შემთხვევაში ეს იქნება ჭეშმარიტი (true) და შემდეგ ნაკადი გადაადგილდება უმაღლესი დონის პროცესისაკენ. მაგრამ შესაძლებელია, რომ ერთ-ერთი მონაწილე დასრულდეს პრობლემით, რომელიც იწვევს ანულირებას (გაუქმებას) ან რისკს. ამ შემთხვევაში, ნაკადი შემდეგ გადაადგილდება შესაბამისი შუალედური ხდომილებისაკენ, თუნდაც ის ცხადად დასრულდეს წარმატებით.

3.4.2.6. შესრულების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, შემდეგი საკითხების გასახსენებლად: ობიექტების სიმრავლე და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყაროები ან მიზნები.

- ქვეპროცესი შეიძლება იყოს შესრულების ნაკადის მიზანი; მას შეიძლება ჰქონდეს მრავალი შემავალი წყარო. შემავალი წყარო შეიძლება იყოს ალტერნატიული ბილიკიდან და/ან პარალელური ბილიკიდან.
 - შემავალი შესრულების ნაკადის ატრიბუტი მიზანიRef შეიძლება იყოს გაფართოებული, რომელიც შეიცავს ორივეს: ქვეპროცესის ობიექტს (მშობელის დონესთან) და დაწყების ხდომილებას, რომელიც მიეკუთვნება ქვეპროცესის დეტალებს. ეს უზრუნველყოფს პირდაპირ შეერთებას მშობელის-დონის შესრულების ნაკადიდან დაბალი დონის დაწყების ხდომილებისაკენ სიტუაციებისათვის, რომლებშიც არსებობენ ერთზე მეტი დაწყების ხდომილებები ქვეპროცესში. გაფართოების ფორმა იქნება “ქვეპროცესი.დაწყება” (“Sub- Process.Start”).
 - თუ ქვეპროცესის დეტალები (ე.ი. მისი დაწყების ხდომილებები) არ არიან ხილული ან ხელმისაწვდომი მოდელირებისათვის,

მაშინ განსაზღვრა, როგორც არის რომელი დაწყების ხდომილება (თუ მრავალი არსებობს) იქნება გადართული არის განუსაზღვრელი. მაგრამ, დაწყების ხდომილებებიდან მხოლოდ ერთი იქნება გადართული (will be triggered).

შენიშვნა. თუ ქვე-პროცესს აქვს მრავალი შემავალი შესრულების ნაკადი, მაშინ ეს არის განხილული როგორც არა კონტროლირებადი ნაკადი. ეს ნიშნავს, რომ როდესაც სიმბოლო (Token) მოდის ერთ-ერთი ბილიკიდან, ქვე-პროცესი შესრულდება მყისიერად. ის არ დაელოდება სიმბოლოს მოსვლას სხვა ბილიკებიდან. თუ სხვა სიმბოლო მოდის იგივე ან სხვა ბილიკიდან, მაშინ ქვე-პროცესის განსხვავებული შემთხვევა შეიქმნება. თუ ნაკადს სჭირდება, რომ იყოს კონტროლირებული, მაშინ ნაკადი უნდა მიისწრაფოდეს გასასვლელისაკენ (Gateway), რომელიც წინ უძღვის ქვე-პროცესს (იხ. პარაგრაფი 3.5, “გასასვლელები” (“Gateways”), გვ. 99, გასასვლელების შესახებ უფრო მეტი ინფორმაციისათვის).

- თუ ქვე-პროცესს არა აქვს შემავალი შესრულების ნაკადი და პროცესისათვის არ არსებობს დაწყების ხდომილება, მაშინ ქვე-პროცესი უნდა იყოს შესრულებული მყისიერად, როდესაც პროცესი არის მყისიერი.
 - აქედან გამონაკლისია ქვე-პროცესები, რომლებიც განსაზღვრული არიან როგორც კომპენსაციის მოქმედებები (აქვთ კომპენსაციის მარკერი). კომპენსაციის ქვე-პროცესები არ განიხილებიან როგორც ნორმალური ნაკადის ნაწილები და არ უნდა იყვნენ შესრულებული მყისიერად, როდესაც პროცესი არის მყისიერი.
- ქვე-პროცესი შეიძლება იყოს შესრულების ნაკადის წყარო; მას შეიძლება ჰქონდეს მრავალი გამომავალი ნაკადი. თუ არსებობს მრავალი გამომავალი შესრულების ნაკადი, მაშინ ეს ნიშნავს, რომ ყოველი ნაკადისათვის შექმნილი არის განცალკევებული პარალელური ბილიკი.

სიმბოლოები (Tokens) იქნებიან გენერირებული ქვე-პროცესიდან გამომავალი ყოველი შესრულების ნაკადისათვის. სიმბოლოId-ები (TokenIds) ყოველი სიმბოლოსათვის იქნება დაყენებული ისე, რომ მათ შეეძლოთ იდენტიფიცირება შემდეგი ფაქტების: 1) რომ ყველა სიმბოლო არის ერთი და იგივე პარალელური “ჩანგლიდან” (Fork) (ანუ პარალელური განშტოებიდან) და 2) პარალელურად არსებული სიმბოლოების რაოდენობა.

- თუ ქვე-პროცესს არა აქვს გამომავალი შესრულების ნაკადი და პროცესისათვის არ არსებობს დამთავრების ხდომილება, მაშინ ქვე-პროცესი აღნიშნავს (marks) ერთი ან მეტი ბილიკის დამთავრებას პროცესში. როდესაც ქვე-პროცესი მთვარდება და არ არსებობენ სხვა პარალელური აქტიური (მოქმედი) ბილიკები, მაშინ პროცესი უნდა დასრულდეს.
 - აქედან გამონაკლისია ქვე-პროცესები, რომლებიც განსაზღვრული არიან როგორც კომპენსაციის მოქმედებები (აქვთ კომპენსაციის მარკერი). კომპენსაციის ქვე-პროცესები არ განიხილებიან როგორც ნორმალური ნაკადის ნაწილი და მათ არ უნდა აღნიშნონ (MUST NOT mark) პროცესის დამთავრება.

3.4.2.7. შეტყობინების ნაკადის შეერთებები

პარაგრაფ 2.4.2–ში, “შეტყობინების ნაკადის წესები”, გვ. 35, მოცემულია ობიექტების მთელი სიმრავლე და თუ როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

შენიშვნა. ყველა შეტყობინების ნაკადმა უნდა შეაერთოს ორი განცალკევებული გუბე. მათ შეუძლიათ მიუერთდნენ გუბის საზღვარს ან ნაკადის ობიექტებს გუბის საზღვრებს შიგნით. მათ არ შეუძლიათ შეაერთონ ორი ობიექტი ერთი და იგივე გუბეში.

- ქვე-პროცესი შეიძლება იყოს შეტყობინების ნაკადის მიზანი; მას შეიძლება ჰქონდეს ნოლი ან მეტი შემავალი შეტყობინების ნაკადი.
- ქვე-პროცესი შეიძლება იყოს შეტყობინების ნაკადის წყარო; მას შეიძლება ჰქონდეს ნოლი ან მეტი გამომავალი შეტყობინების ნაკადი.

3.4.3. ამოცანა

ამოცანა არის ატომური (ელემენტარული) მოქმედება, რომელიც შედის პროცესში. ამოცანა გამოიყენება, როდესაც მუშაობა პროცესში არ არის შეჩერებული (გაწყვეტილი) (broken down) პროცესის მოდელის დეტალის (ელემენტის) კარგ დონესთან.

ამოცანის ობიექტი იყენებს ქვე-პროცესის იგივე ფორმებს, როგორცაა სწორკუთხედი, რომელსაც აქვს მომრგვალებული კუთხეები (იხ. ნახ. 3.14).

- ამოცანა არის მომრგვალებულ კუთხეებიანი სწორკუთხედი, რომელიც უნდა იყოს დახაზული ერთი წვრილი შავი ხაზით.
 - ამოცანისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში”, გვ. 33.



ნახ. 3.14. ამოცანის ობიექტი

BPMN ამოცანისათვის განსაზღვრავს მარკერების სამ ტიპს: *მარყუვის მარკერი ან მრავალ შემთხვევიანი მარკერი და კომპენსაციის მარკერი.* ამოცანას შეიძლება ჰქონდეს ამ მარკერებიდან ერთი ან ორი (იხ. ნახ. 3.15).

- ამოცანის მარკერი, რომელიც არის სტანდარტული მარყუვი, უნდა იყოს პატარა ხაზი ისრის თავით, რომელიც შემოტრიალებულია უკან.

- მარყუჟის მარკერი შეიძლება გამოყენებული იყოს კომპენსაციის მარკერთან კომბინაციაში.
- ამოცანის მარკერი, რომელიც არის მრავალ-შემთხვევიანი მარყუჟი, უნდა იყოს სამი ვერდიკალური პარალელური ხაზის სიმრავლე.
 - მრავალ-შემთხვევიანი მარყუჟი შეიძლება გამოყენებული იყოს კომპენსაციის მარკერთან კომბინაციაში.
- ამოცანის მარკერი, რომელიც არის გამოყენებული კომპენსაციისათვის, უნდა იყოს მარცხნივ მიმართული სამკუთხედების წყვილი.
 - კომპენსაციის მარკერი შეიძლება იყოს გამოყენებული მარყუჟის მარკერთან ან მრავალ შემთხვევიან მარკერთან კომბინაციაში.
- ყველა წარმოდგენილი მარკერი უნდა იყოს დაჯგუფებული და მთელი ჯგუფი უნდა იყოს ცენტრირებული ფორმის ბოლოში.



ნახ. 3.15. ამოცანის მარკერები.

ზემოთ ნაჩვენები ამოცანის კატეგორიების დამატებით BPMN-ში განსაზღვრული არიან სხვადასხვა ტიპის ამოცანები, რომლებიც განაცალკევებენ დამახასიათებელი ქცევის ტიპებს, როლებიც ამოცანებს შეუძლიათ წარმოადგინონ (იხ. ცხრილი 3.2). თუმცა BPMN არ განსაზღვრავს რაიმე გრაფიკულ ინდიკატორს ამოცანების სხვადასხვა ტიპებისათვის. მოდელირებს ან მოდელირების ინსტრუმენტებს შეუძლიათ აირჩიონ: შექმნან მათი საკუთარი ინდიკატორები ან მარკერები, რათა დიაგრამის წამკითველს აჩვენონ ამოცანის ტიპი. ეს დასაშვებია BPMN-თვის, რადგან ამოცანის ძირითადი ფორმა (დამრგვალებული სწორკუთხედი) არ არის მოდიფიცირებული. ამოცანის ტიპების ჩამონათვალი შეიძლება იყოს გაფართოებული რაიმე შესაბამისი ინდიკატორების საშუალებით.

3.4.3.1. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია ამოცანის ატრიბუტების სიმრავლე, რომელიც აფართოებს მოქმედებების ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.18).

ცხრილი 3.25. ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
ამოცანატიპი (მომსახურება მიღება ...)	ამოცანატიპი არის ატრიბუტი, რომელიც გა-

<p>გაგზავნა მომხმარებელი ხელნაწერი სახელმძღვანელო მითითება ცარიელი) ცარიელი : სტრიქონი TaskType (Service Receive Send User Script Manual Reference None) None : String</p>	<p>მოუცხადებლად არის ცარიელი (None), მაგრამ შეიძლება დაყენებული იყოს შემდეგ მნიშვნელობებზე: გაგზავნა, მიღება, მომხმარებელი, ხელნაწერი, სახელმძღვანელო, მითითება ან მომსახურება (Send, Receive, User, Script, Manual, Reference, or Service). ამოცანატიპზე იმოქმედებს შეტყობინების ნაკადი მიმართული ამოცანისაკენ და/ან ამოცანიდან, თუ შეტყობინების ნაკადი არის გამოყენებული. მიღების ამოცანატიპს არ უნდა ჰქონდეს გამომავალი შეტყობინების ნაკადი. გაგზავნის ამოცანატიპს არ უნდა ჰქონდეს შემავალი შეტყობინების ნაკადი. ხელნაწერის ან სახელმძღვანელოს ამოცანატიპს არ უნდა ჰქონდეს შემავალი ან გამომავალი შეტყობინების ნაკადი. ამოცანატიპის ჩამონათვალი შეიძლება იყოს გაფართოებული ახალი ტიპების ჩართვით. ამოცანატიპის სპეციფიური მნიშვნელობების ატრიბუტები შეიძლება ნახოთ ცხრილ 3.26–ში და ცხრილ 3.31–ში.</p>
---	---

3.4.3.2. მომსახურების ამოცანა

მომსახურების ამოცანა არის ამოცანა, რომელიც უზრუნველყოფს მომსახურების რაიმე სახეს, რომელიც შეიძლება იყოს ვებ სერვისი ან ავტომატური გამოყენება. შემდეგი ცხრილი გვიჩვენებს მომსახურების ამოცანის ატრიბუტების სიმრავლეს (სადაც ამოცანატიპი ატრიბუტი არის “მომსახურების” სიმრავლე), რაც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.25).

ცხრილი 3.26. მომსახურების ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
<p>შეტყობინებაშიRef : შეტყობინება InMessageRef : Message</p>	<p>შეტყობინებაშიRef ატრიბუტისათვის უნდა იყოს შემოტანილი შეტყობინება. ეს მიუთითებს, რომ შეტყობინება იქნება მიღებული ამოცანის დასაწყისში, იმის შემდეგ რაც ხელმისაწვდომი იქნებიან რაიმე განსაზღვრული შეყვანისიმრავლეები (InputSets). დიაგრამაზე შეიძლება იყოს ნაჩვენები ერთი</p>

	<p>ან მეტი შესაბამისი შემომავალი შეტყობინების ნაკადი. მიუხედავად ამისა შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყოველი შემავალი შეტყობინების ნაკადისათვის, მაგრამ ამოცანის ერთი მაგალითისათვის შეიძლება მოვიდეს მხოლოდ ერთი შემომავალი შეტყობინების ნაკადი.</p>
<p>შეტყობინებისგარეთშიRef : შეტყობინება OutMessageRef : Message</p>	<p>შეტყობინებისგარეთშიRef ატრიბუტისათვის უნდა იყოს შემოტანილი შეტყობინება. ამ შეტყობინების გაგზავნა აღნიშნავს ამოცანის დამთავრებას, რასაც შეუძლება გამოიწვიოს გამომავლისიმრავლეს წარმოშობა. ერთი ან მეტი შესაბამისი შემომავალი შეტყობინების ნაკადი შეიძლება იყოს ნაჩვენები დიაგრამაზე. მიუხედავად ამისა, შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყველა გამომავალ შეტყობინების ნაკადთან და შეტყობინება გამორთავს ყველა გამომავალ შეტყობინების ნაკადს ამოცანის ერთი მაგალითის დამთავრებისას.</p>
<p>შესრულება (ვებ სერვისი სხვა არა სპეციფიცირებული) ვებ სერვისი : სტრიქონი Implementation (Web Service Other Unspecified) Web Service : String</p>	<p>ეს ატრიბუტი განსაზღვრავს ტექნოლოგიას, რომელიც იქნება გამოყენებული შეტყობინების გასაგზავნად და მისაღებად. ვებ სერვისი არის გამოუცხადებელი (default) ტექნოლოგია.</p>

3.4.3.3. მიღების ამოცანა

მიღების ამოცანა არის მარტივი ამოცანა, რომელიც არის შექმნილი შეტყობინების მოსაცდელად, რომელიც მოდის გარე მონაწილისაგან (ბიზნეს პროცესთან დაკავშირებული). როგორც კი შეტყობინება მიიღება, ამოცანა მთავრდება.

მიღების ამოცანა ხშირად გამოიყენება პროცესის დასაწყებად. გარკვეული აზრით, პროცესი გაუმჯობესებულია შეტყობინების მიღებით. პროცესის მყისიერი შესრულების ამოცანისათვის მან უნდა დააკმაყოფილოს შემდეგი პირობებიდან ერთ-ერთი:

- პროცესს არა აქვს დაწყების ხდომილება და მიღების ამოცანას აქვს არა შემავალი შესრულების ნაკადი.

- მიღების ამოცანის შემავალ შესრულების ნაკადს აქვს დაწყების ხდომილების წყარო.
 - შევნიშნოთ, რომ არც ერთი სხვა შემავალი შესრულების ნაკადი არ არის დასაშვები მიღების ამოცანისათვის (კერძოდ, მარყუჟი შეერთება დინების მიმართულებით ქვემო ობიექტიდან).

შემდეგი ცხრილი გვიჩვენებს მიღების ამოცანის ატრიბუტების სიმრავლეს (სადაც ამოცანატიპი ატრიბუტი არის დაყენებული “მიღებაზე”), რაც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.25).

ცხრილი 3.27. მიღების ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
<p>შეტყობინებაRef : შეტყობინება MessageRef : Message</p>	<p>შეტყობინებაRef ატრიბუტისათვის შეტყობინება უნდა იყოს შემოტანილი. ეს მიუთითებს, რომ შეტყობინება იქნება მიღებული ამოცანის მიერ. ამ კონტექსტში შეტყობინება არის მხოლოდ შეტყობინების ნიმუშის ექვივალენტი (ვებ სერვისი). ერთი ან მეტი შესაბამისი შემავალი შეტყობინების ნაკადი შეიძლება იყოს ნაჩვენები დიაგრამაზე. მიუხედავად ამისა შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყველა შემავალი შეტყობინების ნაკადებისათვის, მაგრამ შეიძლება მოვიდეს მხოლოდ ერთი შემავალი შეტყობინების ნაკადისათვის ამოცანის ერთი შემთხვევისათვის.</p>
<p>გამყისიერება არა ჭეშმარიტი : ბულის Instantiate False : Boolean</p>	<p>მიღების ამოცანები შეიძლება იყვნენ განსაზღვრული, როგორც გამყისიერების მექანიზმი გამყისიერების ატრიბუტებიანი პროცესისათვის. ეს ატრიბუტი შეიძლება დაყენებული იყოს ჭეშმარიტზე თუ ამოცანა არის პირველი მოქმედება დაწყების ხდომილების შემდეგ ან დამწყები ამოცანა, თუ არ არსებობს დაწყების ხდომილება (ე.ი. არ არის შემავალი შესრულების ნაკადი). მრავალჯერად ამოცანებს ამ ატრიბუტების სიმრავლე შეიძლება დაყენებული ჰქონდეთ მდგომარეობაზე ჭეშმარიტი.</p>
<p>შესრულება (ვებ სერვისი სხვა არა სპეციფიცირებული) ვებ სერვისი : სტრიქონი Implementation (Web Service Other Unspecified) Web Service : String</p>	<p>ეს ატრიბუტი განსაზღვრავს ტექნოლოგიას, რომელიც იქნება გამოყენებული შეტყობინების მისაღებად. ვებ სერვისი არის გამოუცხადებელი (default) ტექნოლოგია.</p>

3.4.3.4. გაგზავნის ამოცანა

გაგზავნის ამოცანა არის მარტივი ამოცანა, რომელიც დამუშავებულია შეტყობინების გასაგზავნად გარე მონაწილისათვის (ბიზნეს პროცესთან დაკავშირებული). როგორც კი შეტყობინება გაიგზავნება, ამოცანა დასრულდება.

შემდეგი ცხრილი გვიჩვენებს გაგზავნის ამოცანის ატრიბუტების სიმრავლეს (სადაც ამოცანატიპი ატრიბუტი არის “გაგზავის” სიმრავლე), რაც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.25).

ცხრილი 3.28. გაგზავნის ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
შეტყობინებაRef : შეტყობინება MessageRef : Message	შეტყობინებაRef ატრიბუტისათვის შეტყობინება უნდა იყოს შემოტანილი. ეს მიუთითებს, რომ შეტყობინება იქნება გაგზავნილი ამოცანით. ამ კონტექსტში შეტყობინება ექვივალენტურია შეტყობინების ერთადერთი ნიმუშის (ვებ სერვისი). ერთი ან მეტი შესაბამისი გამომავალი შეტყობინების ნაკადი შეიძლება ნაჩვენები იყოს დიაგრამაზე. თუმცა, შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყველა გამომავალი შეტყობინების ნაკადისათვის და შეტყობინება იქნება გაგზავნილი ქვემოთ ყველა გამომავალი შეტყობინების ნაკადისაკენ ამოცანის ერთი მაგალითის დამთავრებისას.
შესრულება (ვებ სერვისი სხვა არა სპეციფიცირებული) ვებ სერვისი : სტრიქონი Implementation (Web Service Other Unspecified) Web Service : String	ეს ატრიბუტი განსაზღვრავს ტექნოლოგიას, რომელიც გამოიყენება შეტყობინების გასაგზავნად. ვებ სერვისი არის გამოუცხადებელი ტექნოლოგია.

3.4.3.5. მომხმარებლის ამოცანა

მომხმარებლის ამოცანა არის ტიპური “ტექნოლოგიური” (“workflow”) ამოცანა, სადაც ადამიანი შემსრულებელი ასრულებს ამოცანას პროგრამული პროდუქციის გამოყენებით, რომელიც განსაზღვრულია გარკვეული ტიპის მართვის ამოცანების ჩამონათვლით.

შემდეგი ცხრილი გვიჩვენებს მომხმარებლის ამოცანის ატრიბუტების სიმრავლეს (სადაც ამოცანატიპი ატრიბუტი დაყენებულია “მომხმარებელზე”), რომელიც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.25).

ცხრილი 3.29. მომხმარებლის ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
<p>შეტყობინებაშიRef : შეტყობინება InMessageRef : Message</p>	<p>შეტყობინებაშიRef ატრიბუტისათვის შეტყობინება უნდა იყოს შემოტანილი. ეს უჩვენებს, რომ შეტყობინება იქნება მიღებული ამოცანის დაწყებისას, რაიმე განსაზღვრული შეტანასიმრავლების (InputSets) ხელმისაწვდომობის შემდეგ. ერთი ან მეტი შესაბამისი შემაჯავალი შეტყობინების ნაკადი შეიძლება იყოს ნაჩვენები დიაგრამაზე. თუმცა, შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყველა შემაჯავალი შეტყობინების ნაკადისათვის, მაგრამ ამოცანის ერთი შემთხვევისათვის შემაჯავალი შეტყობინების ნაკადისათვის შეიძლება მოვიდეს მხოლოდ ერთი.</p>
<p>შეტყობინებისგარეთშიRef : შეტყობინება OutMessageRef : Message</p>	<p>შეტყობინებისგარეთRef ატრიბუტისათვის შეტყობინება უნდა იყოს შემოტანილი. ამ შეტყობინების გამოგზავნა აღნიშნავს ამოცანის დამთავრებას, რასაც შეუძლია გამოიწვიოს გამოსვლასიმრავლეს (OutputSet) შექმნა. მიუხედავად ამისა, შეტყობინების ნაკადის ჩვენება არ მოითხოვება. შეტყობინება გამოიყენება ყველა გამომავალი შეტყობინების ნაკადისათვის და ამოცანის ერთი შემთხვევის დამთავრებისას შეტყობინება იქნება გაგზავნილი ქვემოთ ყველა გამომავალი შეტყობინების ნაკადისაკენ.</p>
<p>შესრულება (ვებ სერვისი სხვა არა სპეციფიცირებული) სხვა : სტრიქონი Implementation (Web Service Other Unspecified) Other : String</p>	<p>ეს ატრიბუტი განსაზღვრავს ტექნოლოგიას, რომელიც იქნება გამოყენებული შემსრულებლის მიერ ამოცანის შესრულებისას.</p>

3.4.3.6. ხელნაწერის ამოცანა

ხელნაწერის ამოცანა შესრულებულია ბიზნეს პროცესის ძრავით (მამოძრავებელით). მოდელიორი ან შემსრულებელი ხელნაწერს განსაზღვრავს ენაში, რომლის ინტერპრეტირებაც (გაგებაც) შეუძლია ძრავას. როდესაც ამოცანა მზად არის გასაშვებად, ძრავა შეასრულებს ხელნაწერს. როდესაც ხელნაწერი დამთავრებულია, ამოცანაც აგრეთვე იქნება დამთავრებული.

შემდეგ ცხრილში ნახვენებია ხელნაწერის ამოცანის ატრიბუტების სიმრავლე (სადაც ამოცანატიპი ატრიბუტი დაყენებული არის “ხელნაწერზე”), რაც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ, ცხრილი 3.25).

ცხრილი 3.30. ხელნაწერის ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
ხელნაწერი (0-1) : სტრიქონი Script (0-1) : String	მოდელიორს შეუძლია ჩართოს (include) ხელნაწერი, რომელიც შეიძლება იყოს გაშვებული (can be run), როდესაც ამოცანა არის შესრულებული. თუ ხელნაწერი არ არის ჩართული, მაშინ ამოცანა იმოქმედებს ცარიელი (None) ამოცანატიპის ექვივალენტურად.

3.4.3.7. სახელმძღვანელოს ამოცანა

სახელმძღვანელოს ამოცანა არის ამოცანა, რომელიც მოსალოდნელია იყოს შესრულებული რაიმე ბიზნეს პროცესის შესრულების ძრავის ან რაიმე გამოყენების დახმარების გარეშე. ამის მაგალითი შეიძლება იყოს ტელეფონის ტექნიკოსის მიერ ტელეფონის დაყენება კლიენტის ყოფნის ადგილზე.

3.4.3.8. მითითების ამოცანა

შეიძლება მოხდეს, რომ მოდელიორს მოუნდეს მიუთითოს სხვა ამოცანა, რომელიც ადრე იყო განსაზღვრული. თუ ორ (ან მეტ) ამოცანას აქვს ზუსტად ერთი და იგივე ქცევა, მაშინ ერთის მითითებით მეორეზე, ატრიბუტები, რომლებიც განსაზღვრავენ ქცევას, მხოლოდ ერთხელ უნდა იყვნენ განსაზღვრული და შენახული მხოლოდ ერთ ადგილზე.

შემდეგი ცხრილი გვიჩვენებს მითითების ამოცანის ატრიბუტების სიმრავლეს (სადაც ამოცანატიპი ატრიბუტი დაყენებული არის “მითითებაზე”), რაც აფართოებს ამოცანის ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.25).

ცხრილი 3.31. მითითების ამოცანის ატრიბუტები

ატრიბუტები	აღწერა
ამოცანაRef : ამოცანა TaskRef : Task	მითითებული ამოცანა უნდა იყოს განსაზღვრული. ამოცანის ელემენტების ატრიბუტები შეიძლება ნახოთ ცხრილში 3.25.

3.4.3.9. შესრულების ნაკადის კავშირები

პარაგრაფში 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, მოცემულია ობიექტების მთელი სიმრავლე და თუ როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

- ამოცანა შეიძლება იყოს შესრულების ნაკადის მიზანი; მას შეიძლება ჰქონდეს მრავალი შემავალი ნაკადი. შემავალი ნაკადი შეიძლება იყოს ალტერნატიული ბილიკიდან და/ან პარალელური ბილიკებიდან.

შენიშვნა. თუ ამოცანას აქვს მრავალი შემავალი შესრულების ნაკადი, მაშინ ეს განხილულია როგორც არა კონტროლირებადი ნაკადი. ეს ნიშნავს, რომ როდესაც სიმბოლო მოდის ერთ-ერთი ბილიკიდან, ამოცანა შესრულდება მყისიერად. ის არ დაელოდება სიმბოლოების მოსვლას სხვა ბილიკებიდან. თუ სხვა სიმბოლო მოდის იგივე ან სხვა ბილიკიდან, მაშინ ამოცანის ცალკეული (სხვადასხვა) შემთხვევები იქნება შექმნილი. თუ საჭიროა ნაკადის კონტროლი (მართვა), მაშინ ნაკადი უნდა მიისწრაფოდეს გასასვლელისაკენ (Gateway), რომელიც წინ უსწრებს ამოცანას (იხ. პარაგრაფი 3.5, “გასასვლელები”, გვ. 99, გასასვლელების შესახებ მეტი ინფორმაციისათვის).

- თუ ამოცანას არა აქვს შემავალი შესრულების ნაკადი და პროცესს არა აქვს დაწყების ხდომილება, მაშინ ამოცანა უნდა შესრულდეს მყისიერად, როდესაც პროცესი არის შესრულდებული მყისიერად.
 - ამის გამონაკლისია ამოცანები, რომლებიც განსაზღვრულია როგორც კომპენსაციის მოქმედებები (აქვს კომპენსაციის მარკერი). კომპენსაციის ამოცანები არ განიხილებიან ნორმალური ნაკადის ნაწილად და არ უნდა იყვნენ შესრულდებული მყისიერად, როდესაც პროცესი არის შესრულდებული მყისიერად.
- ამოცანა შეიძლება იყოს წყარო შესრულების ნაკადისათვის; მას შეიძლება ჰქონდეს მრავალი გამომავალი ნაკადი. თუ არსებობს მრავალი გამომავალი შესრულების ნაკადი, მაშინ ეს ნიშნავს, რომ განცალკევებული პარალელური ბილიკი იწყება ყოველი ნაკადისათვის.

სიმბოლოები (Tokens) იქნებიან გენერირებული ამოცანიდან გამომავალი ყოველი შესრულების ნაკადისათვის. ყოველი სიმბოლოსათვის სიმბოლოId-ები იქნება დაყენებული ისეთნაირად, რომ მათ შეეძლოთ შემდეგი ფაქტის იდენტიფიცირება: ყველა

სიმბოლო არის იგივე პარალელური ჩანგალიდან (განშტოებიდან) (parallel Fork), ისევე როგორც სიმბოლოების რაოდენობა, რომლებიც არსებობენ პარალელურად.

- თუ ამოცანას არა აქვს გამომავალი შესრულების ნაკადი და პროცესს არა აქვს დამთავრების ხდომილება, მაშინ ამოცანა აღნიშნავს (marks) ერთი ან მეტი ბილიკის დამთავრებას პროცესში. როდესაც ამოცანა მთავრდება და არ არსებობენ სხვა პარალელური აქტიური (მოქმედი) ბილიკები, მაშინ პროცესი უნდა დამთავრდეს.
 - აქედან გამონაკლისია ამოცანები, რომლებიც არიან განსაზღვრული როგორც კომპენსაციის მოქმედებები (აქვს კომპენსაციის მარკერი). კომპენსაციის ამოცანები არ განიხილებიან როგორც ნორმალური ნაკადის ნაწილი და არ უნდა დაამთავრონ პროცესი.

3.4.3.10. შეტყობინების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.2, “შეტყობინების ნაკადის წესები”, გვ. 35, ობიექტების მთელი სიმრავლისათვის და თუ როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

შენიშვნა. ყველა შეტყობინების ნაკადმა უნდა შეაერთოს (დააკავშიროს) ორი პარალელური გუბე. მათ შეუძლიათ მიუერთდნენ გუბის საზღვარს ან ნაკად ობიექტებს გუბის საზღვრებში. მათ არ შეუძლიათ დააკავშირონ ორი ობიექტი ერთი და იმავე გუბეში.

- ამოცანა შეიძლება იყოს შეტყობინების ნაკადის მიზანი; მას შეიძლება ჰქონდეს ნოლი ან მეტი შემავალი შეტყობინების ნაკადი. თუ არსებობს მრავლობითი შემავალი შეტყობინების ნაკადი, მაშინ ერთი შეტყობინება გამოიყენება ყველა შეტყობინების ნაკადისათვის. თუმცა, მხოლოდ ერთი შეტყობინება შეიძლება იყოს მიღებული ერთი შეტყობინების ნაკადიდან ამოცანის მოცემული შემთხვევისათვის.
- ამოცანა შეიძლება იყოს შეტყობინების ნაკადის წყარო; მას შეიძლება ჰქონდეს ნოლი ან მეტი გამომავალი შეტყობინების ნაკადი. თუ გვაქვს მრავალი გამომავალი შეტყობინების ნაკადი, მაშინ ერთი შეტყობინება იქნება გამოყენებული ყველა შეტყობინების ნაკადისათვის. ეს შეტყობინება იქნება გაგზავნილი ქვემოთ ყველა გამომავალი შეტყობინების ნაკადისაკენ.

3.5. გასასვლელები

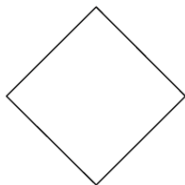
გასასვლელები (Gareways) არიან მოდელირების ელემენტები, რომლებიც გამოიყენებიან იმის საკონტროლებლად თუ როგორ ურთიერთმოქმედებენ შესრულების ნაკადები როდესაც ისინი იკრიფებიან და იშლებიან პროცესის შიგნით. თუ არ არის საჭირო ნაკადის კონტროლი (მართვა), მაშინ გასასვლელები არ არიან

საჭირო. ტერმინი “გასასვლელი” (“Gateway”) გულისხმობს, რომ არსებობს გასვლის მექანიზმი, რომელიც იძლევა ან არ იძლევა გასასვლელში გასვლის საშუალებას; ანუ, როდესაც სიმბოლოები მოვლენ გასასვლელში, ისინი შეიძლება იყვნენ შერწყმული ერთად შესასვლელზე და/ან გახლეჩილი გამოსასვლელებს შორის, როგორც გასასვლელის მექანიზმები მოითხოვენ. უფრო დაწვრილებით, გასასვლელი არის ნამდვილად “კარებების” ნაკრები. თუმცა კარებები (Gates) არ არიან ნაჩვენები გრაფიკულად, ისინი არიან გამოყენებული შესრულების ნაკადის მიერ დასაკავშირებლად (მისაერთებლად) რამეს შესასვლელთან ან გასასვლელის (Gateway) გამოსასვლელთან.

არსებობენ გასასვლელების (Gateways) სხვადასხვა ტიპები (როგორც ქვემოთ არის აღწერილი) და ყოველი ტიპის გასასვლელის ქცევა განსაზღვრავს რამდენი კარებები (Gates) იქნება შესაძლებელი ნაკადის გასაგრძელებლად. არსებობს ერთი კარები (Gate) გასასვლელის (Gateway) ყოველი გამომავალი შესრულების ნაკადისათვის.

გასასვლელი (Gateway) არის რომბი (იხ. ნახ. 3.16), რომელიც გამოყენებული იყო მრავალ გრაფიკულ აღნიშვნებში სპეციალური განშტოებისათვის და არის ნაცნობი უმრავლესი მოდელიორისათვის.

- გასასვლელი არის რომბი, რომელიც უნდა დაიხაზოს ერთი წვრილი შავი ხაზით.
 - გასასვლელისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში, “ტექსტის, ფერის, ზომის და ხაზების გამოყენება დიაგრამაში”, გვ. 33, განსაზღვრული წესების თანახმად.



ნახ. 3.16. გასასვლელი.

შენიშვნა. თუმცა გასასვლელის ფორმა არის რომბი, არ არის მოთხოვნილი, რომ შემავალი და გამომავალი შესრულების ნაკადი უნდა უერთდებოდნენ რომბის კუთხეებს. შესრულების ნაკადი შეიძლება შეუერთდეს გასასვლელის ფორმის საზღვარს ნებისმიერ ადგილას.

გასასვლელებს შეუძლიათ განსაზღვრონ ყველა ტიპის ბიზნეს პროცესის შესრულების ნაკადის საქციელი: გადაწყვეტილებები/განშტოება (გამონაკლისი, შემცველი (თავის თავში) და რთული), შერწყმა, განშტოება და მიერთება (Decisions/branching (exclusive, inclusive, and complex), merging, forking, and joining). ამრიგად, მაშინ როდესაც რომბი ტრადიციულად გამოიყენებოდა გამონაკლისი გადაწყვეტილებებისათვის, BPMN აფართოებს რომბის ქცევას, ასახავს რა ნებისმიერი ტიპის შესრულების ნაკადის მართვას. გასასვლელის (Gateway) ნებისმიერ ტიპს ექნება შიდა ინდიკატორი ან მარკერი გამოყენებული გასასვლელის ტიპის საჩვენებლად (იხ. ნახ. 3.17).



ნახ. 9.17. გასასვლელების სხვადასხვა ტიპები.

- გასასვლელთან დაკავშირებული შიდა მარკერი უნდა იყოს განთავსებული ფორმის შიგნით, ნებისმიერი ზომით და განთავსებით, მოდელირების შეხედულების ან მოდელირების ინსტრუმენტის გამყიდველის მიხედვით, იმის გამოკლებით, რომ მონაცემებზე დაფუძნებული გასასვლელისათვის მარკერი არ არის მოთხოვნილი.

გასასვლელები მართავენ ორივეს კრებად და/ან განშლად შესრულების ნაკადს. ეს ნიშნავს, რომ კონკრეტულ გასასვლელს შეიძლება ჰქონდეს მრავალი შესასვლელი კარები და მრავალი გასასვლელი კარები ერთი და იგივე დროს (ყოველი კარებისათვის არსებობს ერთი შესრულების ნაკადი). გასასვლელის ტიპი განსაზღვრავს ერთი და იგივე ტიპის ქცევას ორივე, განშლადი და კრებადი შესრულების ნაკადისათვის. მოდელირებს და მოდელირების ინსტრუმენტებს შეიძლება სურდეთ გასასვლელის საუკეთესო თვისების გაძლიერება ამ ფუნქციებიდან მხოლოდ ერთის შესრულებით. ამრიგად, შეიძლება ავიღოთ ორი მიმდევრობითი გასასვლელი, რომელთაგან პირველი შეკრებს და შემდეგ გაშლის შესრულების ნაკადს.

3.5.1. გასასვლელის საერთო თავისებურებები

3.5.1.1. გასასვლელის საერთო ატრიბუტები

შემდეგ ცხრილში მოცემულია ატრიბუტები საერთო გასასვლელებისათვის, რომლებიც აფართოებენ ნაკადი ობიექტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.2).

ცხრილი 3.32. გასასვლელის საერთო ატრიბუტები

ატრიბუტები	აღწერა
გასასვლელიტიპი (განსაკუთრებული	გასასვლელიტიპი გამოუცხადებლად არის

<p>შემცველი რთული პარალელური) გამონაკლისი : სტრიქონი GatewayType (Exclusive Inclusive Complex Parallel) Exclusive : String</p>	<p>გამონაკლისი. გასასვლელიტიპი შეიძლება იყოს დაყენებული შემდეგ მდგომარეობაში: შემცველი, რთული ან პარალელური (Inclusive, Complex, or Parallel). გასასვლელიტიპი განსაზღვრავს გასასვლელის ქცევას ორივე, შემაჯავალი და გამომავალი შესრულების ნაკადებისათვის და განსაზღვრავს შიდა ინდიკატორს (როგორც ნაჩვენებია ნახ. 3.17-ზე).</p>
<p>კარებები (0-n) : კარები</p>	<p>შეიძლება იყოს ნოლი ან მეტი კარები (Gate) (გარდა ქვემოთ გაკეთებული შენიშვნის). ნოლი კარები დასაშვებია თუ გასასვლელი არის ბოლო ობიექტი პროცესის ნაკადში და პროცესისათვის არ არის დაწყების ან დამთავრების ხდომილებები. თუ არსებობს ნოლი ან მხოლოდ ერთი შემაჯავალი შესრულების ნაკადი, მაშინ უნდა იყოს სულ ცოტა ორი კარები.</p> <p>მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელისათვის: როცა ორი კარები არის მოთხოვნილი, ერთ-ერთი მათგანი შეიძლება იყოს გამოუცხადებელიკარები.</p> <p>ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელები: უნდა იყოს ორი ან მეტი კარები. (შეგნიშნოთ, რომ გასასვლელის ეს ტიპი არ მოქმედებს მხოლოდ როგორც შერწყმა; ის ყოველთვის არის, სულ ცოტა, გადაწყვეტილება).</p> <p>შემცველი გასასვლელისათვის: როცა ორი კარები არის მოთხოვნილი, ერთი მათგანი შეიძლება იყოს გამოუცხადებელიკარები.</p>

3.5.1.2. გასასვლელის შესრულების ნაკადის საერთო შეერთებები

ეს პარაგრაფი ეხება ყველა გასასვლელს. დამატებით შესრულების ნაკადის შეერთების წესები გასასვლელის ყოველი ტიპისათვის იქნება აღწერილი შემდეგ პარაგრაფებში. იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო და მიზნები.

- გასასვლელი (Gateway) შეიძლება იყოს შესრულების ნაკადის მიზანი; მას შეიძლება ჰქონდეს ნოლი ან მეტი შემაჯავალი შესრულების ნაკადი. შემაჯავალი ნაკადი შეიძლება იყოს ალტერნატიული ბილიკიდან ან პარალელური ბილიკიდან.
 - თუ გასასვლელს არა აქვს შემაჯავალი შესრულების ნაკადი და პროცესს არა აქვს დაწყების ხდომილება, მაშინ გასასვლელის

განშლადობის ქცევა იქნება შესრულებული გასასვლელიტიპი ატრიბუტისაგან დამოკიდებულებით (იხ. ქვემოთ), როდესაც პროცესი არის შესრულებული მყისიერად.

- გასასვლელი შეიძლება იყოს შესრულების ნაკადის წყარო; მას შეიძლება ჰქონდეს ნოლი ან მეტი გამომავალი ნაკადი.
- გასასვლელს შეიძლება ჰქონდეს ორივე მრავალბოთი (multiple) შემავალი და გამომავალი შესრულების ნაკადი.

შენიშვნა. არ მოითხოვება, რომ შემავალი და გამომავალი შესრულების ნაკადები იყვნენ მიდგომული გასასვლელის რომბის ფორმის კუთხეებზე. შესრულების ნაკადი შეიძლება მიედგას გასასვლელის საზღვრის ნებისმიერ ადგილას.

3.5.1.3. შეტყობინების ნაკადის შეერთებები

ეს პარაგრაფი ეხება ყველა გასასვლელს. იხ. პარაგრაფი 2.4.2, “შეტყობინების ნაკადის წესები”, გვ. 35, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

- გასასვლელი არ უნდა იყოს შეტყობინების ნაკადის მიზანი.
- გასასვლელი არ უნდა იყოს შეტყობინების ნაკადის წყარო.

3.5.1.4. კარებები (Gates)

შემდეგ ცხრილში მოცემულია კარებების ატრიბუტები, რომელიც აფართოვებს BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.1).

ცხრილი 3.33. კარების ატრიბუტები

ატრიბუტები	აღწერა
<p>გამომავალი შესრულების ნაკადი შესრულების ნაკადი OutgoingSequenceFlow : SequenceFlow</p>	<p>ყოველ კარებს უნდა ჰქონდეს მიერთებული (გამომავალი) შესრულების ნაკადი. შესრულების ნაკადის ატრიბუტები შეიძლება ნახოთ პარაგრაფში 4.1.2-ში, “შესრულების ნაკადი”, გვ. 139.</p> <p>გამონაკლისი ხდომილებაზე დაფუძნებული, რთული და პარალელური გასასვლელებისათვის: შესრულების ნაკადს უნდა ჰქონდეს პირობითი ატრიბუტი დაყენებული ცარიელზე (None) (აქ არ არის პირობითი გამოსახულების შეფასება).</p> <p>გამონაკლისი მონაცემებზე დაფუძნებული და შემცველი გასასვლელებისათვის: შესრულების ნაკადს უნდა ჰქონდეს პირობითი ატრიბუტის სიმრავლე გამოსახულებაში და</p>

	<p>უნდა ჰქონდეს მოქმედი (მნიშვნელობის მქონე) პირობაგამოსახულება. გასასვლელის შიგნით ყველა კარებისათვის პირობაგამოსახულება უნდა იყოს უნიკალური (ერთადერთი). თუ მარტო ერთი კარები არსებობს (ე.ი. გასასვლელი მოქმედებს მხოლოდ როგორც შერწყმა), მაშინ შესრულების ნაკადის პირობა უნდა იყოს დაყენებული ცარიელზე.</p> <p>გამოუცხადებელი კარებისათვის (For DefaultGates): შესრულების ნაკადს უნდა ჰქონდეს პირობითი ატრიბუტი დაყენებული სხვანაირადზე (to Otherwise).</p>
დავალებები (0-n) : დავალება	<p>ყოველი კარებისათვის შეიძლება იყოს შექმნილი ერთი ან მეტი დანიშნულების გამოსახულება. დანიშნულება უნდა შესრულდეს, როდესაც კარები არის შერჩეული.</p>

3.5.2. განსაკუთრებული (Exclusive) გასასვლელები

განსაკუთრებული გასასვლელები (გადაწყვეტილებები) არიან ადგილმდებარეობები ბიზნეს პროცესის შიგნით, სადაც შესრულების ნაკადს შეუძლია ჰქონდეს ორი ან მეტი ალტერნატიული ბილიკი. ეს ძირითადად არის “ჩანგალი გზაში” (“fork in the road”) პროცესისათვის. პროცესის მოცემული შესრულებისათვის (ან მაგალითისათვის) მხოლოდ ერთი ბილიკი შეიძლება იყოს აღებული (შერჩეული) (ეს არ უნდა იყოს არეული ბილიკების განშტოებასთან, იხ. 4.2.1.1. “განშტოება ნაკადი”, გვ. 151). გადაწყვეტილება არ არის მოქმედება ბიზნეს პროცესის თვალსაზრისით, მაგრამ არის გასასვლელის ტიპი, რაც მართავს შესრულების ნაკადს მოქმედებებს შორის. კითხვის სახით უნდა იყოს მოფიქრებული ის, რაც არის კითხვის სახით დასმული პროცესში ამ წერტილთან. კითხვას ქვს ალტერნატიული პასუხების (კარების) განსაზღვრული სიმრავლე. ყოველი გადაწყვეტილების კარები არის დაკავშირებული პირობის გამოსახულებასთან, რომელიც იმყოფება გამომავალ შესრულების ნაკადში. როდესაც პროცესის შესრულებისას კარები არის არჩეული, მაშინ შესაბამისი შესრულების ნაკადი არის არჩეული. გადაწყვეტილებასთან მოსული სიმბოლო მიმართული იქნება ქვემოთ შესაბამისი ბილიკისაკენ, რომელიც დაფუძნებულია არჩეულ კარებზე.

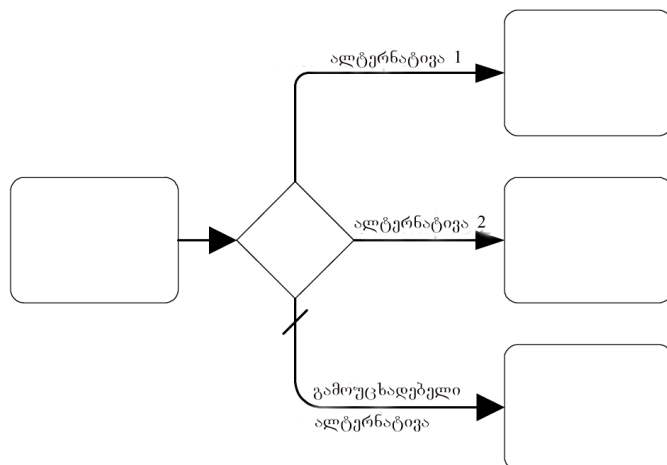
განსაკუთრებულ გადაწყვეტილებას აქვს ორი ან მეტი გამომავალი შესრულების ნაკადი, მაგრამ მათგან მხოლოდ ერთი შეიძლება იყოს შერჩეული პროცესის შესრულების პროცესში. ამრიგად, განსაკუთრებული გადაწყვეტილება სიმბოლოსათვის განსაზღვრავს ალტერნატიული ბილიკების სიმრავლეს, შერჩეულს თუ როგორ უნდა გადაკვეთოს სიმბოლომ ნაკადი. არსებობს განსაკუთრებული გადაწყვეტილების ორი ტიპი: მონაცემებზე დაფუძნებული და ხდომილებაზე დაფუძნებული.

3.5.2.1. მონაცემებზე დაფუძნებული

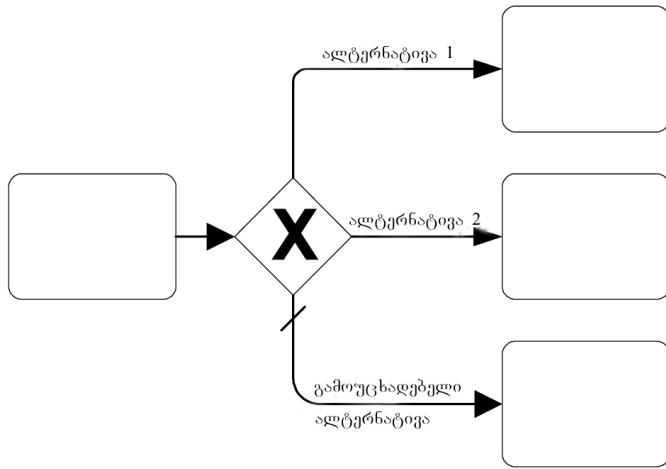
მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელები არიან გასასვლელების უფრო მეტად გამოყენებული ტიპები. მონაცემებზე დაფუძნებული განსაკუთრებული გადაწყვეტილებებისათვის კარგების სიმრავლე დაფუძნებულია ბულის გამოსახულებაზე, რომელიც შედის გასასვლელის გამომავალი შესრულების ნაკადის პირობაგამოსახულება ატრიბუტში. ეს გამოსახულებები იყენებენ პროცესის მონაცემების მნიშვნელობებს, რომ განსაზღვრონ რომელი ბილიკი უნდა იყოს ადებული (არჩეული) (აქედან გამომდინარეობს სახელი მონაცემებზე დაფუძნებული).

შენიშვნა. BPMN არ განსაზღვრავს გამოსახულებების ფორმატს გასასვლელებში ან რაიმე სხვა BPMN-ის ელემენტს, რომელიც იყენებს გამოსახულებებს.

- მონაცემებზე დაფუძნებულმა განსაკუთრებულმა გასასვლელმა შეიძლება გამოიყენოს მარკერი, რომელსაც აქვს “X”-ის ფორმა და განთავსებულია გასასვლელის რომში (იხ. ნახ. 3.19) მის განსასხვავებლად სხვა გასასვლელისაგან. ზოგადად ეს მარკერი არ მოითხოვება (იხ. ნახ. 3.18).
 - დიაგრამა უნდა იყოს მიმდევრობითი “X” შიდა ინდიკატორების გამოყენებაში. ეს ნიშნავს, რომ დიაგრამას არ უნდა ჰქონდეს ზოგი გასასვლელები ინდიკატორებით (მაჩვენებლებით) და ზოგი გასასვლელები ინდიკატორების გარეშე.



ნახ. 3.18. განსაკუთრებული მონაცემებზე დაფუძნებული გადაწყვეტილების (გასასვლელის) მაგალითი შიდა ინდიკატორის გარეშე.



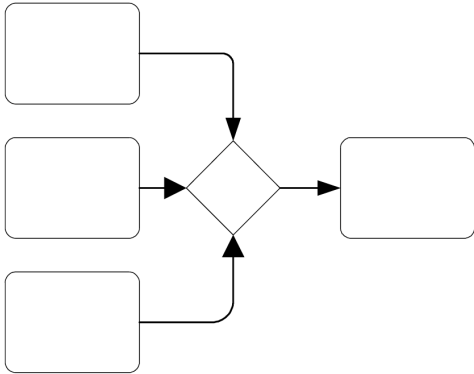
ნახ. 3.19. განსაკუთრებული მონაცემებზე დაფუძნებული გადაწყვეტილების (გასასვლელის) მაგალითი შიდა ინდიკატორით.

შენიშვნა. “X” შიდა ინდიკატორი განსაკუთრებული მონაცემებზე დაფუძნებული გასასვლელისათვის იყო შეტანილი BPMN-ში ინდიკატორების სიმრავლის დასაკომპლექტებლად გასასვლელების სხვადასხვა ტიპებისათვის (იხ. ნახ. 3.17). თუმცა, გასაგებია, რომ მოდელირების უმრავლესობა უნდა იცნობდნენ გადაწყვეტილების ცარიელი რომბს, რაც წარმოადგენს პროცესის განსაკუთრებულ განშტოებას და გადაწყვეტილებების უმრავლესობამ შესაძლებელია მიიღოს ეს ფორმა. ამრიგად, მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელის შიდა ინდიკატორი იყო შექმნილი როგორც არა აუცილებელი (ფაკულტატიური) ისე, რომ მოდელირებს და მოდელირების ინსტრუმენტებს შეეძლოთ შექმნან დიაგრამები, რომლებიც მიესადაგებიან მოდელირებისაგან მოსალოდნელ ძირითად ნაკადს.

ალტერნატიული კარებებისათვის პირობები უნდა იყვნენ შეფასებული სპეციალური წესით. პირველი მათგანი შეფასებული როგორც ჭეშმარიტი, განსაზღვრავს შესრულების ნაკადს, რომელიც იქნება არჩეული. იმის გამო, რომ ამ გასასვლელის ქცევა არის განსაკუთრებული, ნებისმიერი სხვა პირობა, რომელიც ნამდვილად შეიძლება იყოს ჭეშმარიტი, იქნება იგნორირებული; მხოლოდ ერთი კარები შეიძლება იყოს არჩეული. კარებიდან ერთი შეიძლება იყოს “გამოუცხადებელი” (“default”) (ან სხვანაირად), და არის უკანასკნელად განხილული კარები. ეს ნიშნავს, რომ თუ სხვა კარებიდან არც ერთი არ არის არჩეული, მაშინ გამოუცხადებელი კარები იქნება არჩეული მასთან დაკავშირებულ შესრულების ნაკადთან ერთად.

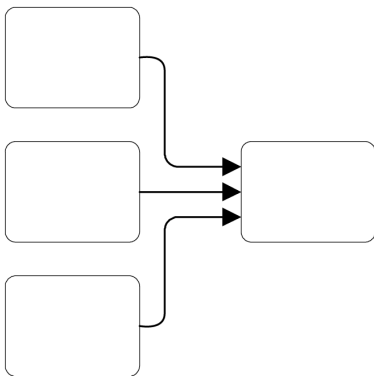
გამოუცხადებელი კარები არ არის აუცილებელი გასასვლელისათვის. ეს ნიშნავს, რომ თუ ის არ არის გამოყენებული, მაშინ მოდელიორი უნდა იყოს დარწმუნებული, რომ სულ მცირე ერთი კარები იქნება მოქმედი შესრულების მომენტში. BPMN არ განსაზღვრავს რა მოხდება თუ არ იქნებიან მოქმედი კარები. და მაიც, BPMN განსაზღვრავს, რომ არ უნდა იყოს ნაგულისხმევი ნაკადი და რომ პროცესის ყველა ნორმალური ნაკადი უნდა იყოს გამოსახული შესრულების ნაკადის საშუალებით. ეს ნიშნავს, რომ პროცესის მოდელი, რომელსაც აქვს გასასვლელი,

რომელსაც პოტენციულად არა აქვს მოქმედი კარები შესრულებისას, არის არასწორი (მახინჯი) (invalid) მოდელი.



ნახ. 3.20. განსაკუთრებული შერწყმა (გასასვლელი) (შიდა ინდიკატორის გარეშე)

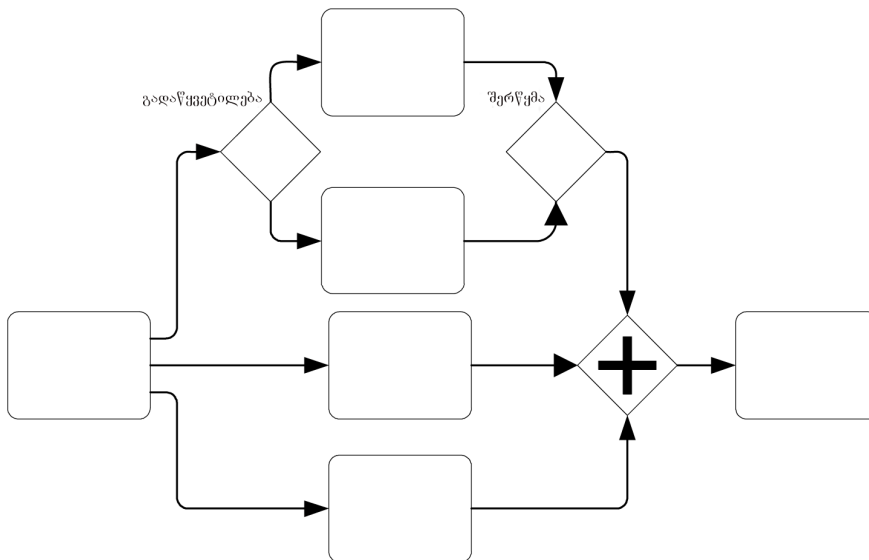
განსაკუთრებული გასასვლელი შეიძლება აგრეთვე იყვნენ გამოყენებული როგორც შერწყმა (იხ. ნახ. 3.20) ალტერნატიული შესრულების ნაკადისათვის, თუმცა იშვიათად არის მოთხოვნილი, რომ მოდელირებმა გამოიყენონ ისინი ასეთნაირად. გასასვლელის ქცევა შერწყმისას შეიძლება აგრეთვე იყოს მოდელირებული როგორც ჩანს ნახ. 3.21-დან. ნახ. 3.20 და ნახ. 3.21-ის ქცევა ერთნაირია, თუ ყველა შემავალი ნაკადები არიან ალტერნატიული.



ნახ. 3.21. შესრულების ნაკადის არა კონტროლირებადი (არა მართვადი) შერწყმა.

ცნობილია სიტუაციები, სადაც განსაკუთრებული გასასვლელი არის მოთხოვნილი იმისათვის, რომ მოიქცეს როგორც შერწყმის ობიექტი. ნახ. 3.22-ზე განსაკუთრებული გასასვლელი (დაჭდევებული “შერწყმა”) (labeled “Merge”) შერწყავს (აერთებს) ორ ალტერნატიულ შესრულების ნაკადს, რომელიც გენერირებული იყო ზემოთ მიმართული გადაწყვეტილებით (Decision). ალტერნატიული შესრულების ნაკადი არის შერწყმული პარალელური გასასვლელების მომზადებაში, რაც სინქრონიზაციას უკეთებს პარალელური შესრულების ნაკადების სიმრავლეს, რაც გენერირებული იყო კიდევ უფრო ზემოთ დინების მიმართულებით. თუ შერწყმის გასასვლელი არ იქნებოდა გამოყენებული, მაშინ პარალელურ გასასვლელში იქნე-

ბოდა ოთხი შემავალი შესრულების ნაკადი. თუმცა, ოთხიდან მხოლოდ სამი შესრულების ნაკადი ოდესმე გადასცემდა სიმბოლოს ერთი და იგივე მომენტში (დროს). ამრიგად, გასასვლელი დაელოდებოდა მეოთხე სიმბოლოს, რომელიც არასოდეს მოვიდოდა. ამრიგად, პროცესი გაიჭედებოდა პარალელური გასასვლელის წერტილში (პუნქტში).



ნახ. 3.22. განსაკუთრებული გასასვლელი, რომელიც შერწყავს (აერთებს) შესრულების ნაკადს პარალელურ გასასვლელამდე.

მარტივ სიტუაციებში, განსაკუთრებული გასასვლელები შეიძლება არ იყვნენ გამოყენებული შესრულების ნაკადის შერწყმისათვის, მაგრამ უფრო რთულ სიტუაციები ისინი არიან მოთხოვნილი. ამრიგად, მოდელიორმა ყოველთვის უნდა იცოდეს იმ სიტუაციებში ქცევა, რომლებშიც შესრულების ნაკადი არის არაკონტროლირებადი. ზოგიერთ მოდელიორს ან მოდელირების ინსტრუმენტებს შეუძლიათ ფაქტიურად მოითხოვონ, რომ განსაკუთრებული გასასვლელები იყვნენ გამოყენებული ყველა სიტუაციაში როგორც საუკეთესო მეთოდები.

3.5.2.2. ატრიბუტები

შემდეგ ცხრილში მოცემულია მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელის ატრიბუტები. ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში თუ გასასვლელი ატრიბუტი არის განსაკუთრებული სიმრავლე. შემდეგი ატრიბუტები აფართოებენ გასასვლელის საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.32).

ცხრილი 3.34. მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელის ატრიბუტები.

ატრიბუტები	აღწერა
განსაკუთრებულიტიპი (მონაცემები ხლომილება) მონაცემები : სტრიქონი ExclusiveType (Data Event) Data : String	განსაკუთრებულიტიპი გამოუცხადებლად არის მონაცემები. განსაკუთრებულიტიპი შეიძლება დაყენებული იყოს ხლომილებაზე. რადგან მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელები არიან ამ პარაგრაფში განხილული საკითხები (თემები), ატრიბუტი უნდა იყოს დაყენებული მონაცემებზე ატრიბუტებისათვის და ამ პარაგრაფში განსაზღვრული ქცევა იყოს გამოყენებული გასასვლელისათვის.
მარკერიცხადი არაჩქეშმარიტი : ბულის MarkerVisible False : Boolean	ეს ატრიბუტი განსაზღვრავს არის თუ არა განსაკუთრებული მარკერი ნაჩვენები გასასვლელის რომის ცენტრში ("X"). მარკერი ნაჩვენებია თუ ატრიბუტი არის ჭეშმარიტი და არ არის ნაჩვენები თუ ატრიბუტი არის არაჩქეშმარიტი. გამოუცხადებლად მარკერი არ არის ნაჩვენები.
გამოუცხადებელიკარი (0-1) : კარი DefaultGate (0-1) : Gate	გამოუცხადებელი კარი შეიძლება იყოს განსაზღვრული (იხ. პარაგრაფი 3.5.1.4, "კარები", გვ. 103).

3.5.2.3. შესრულების ნაკადის შეერთებები

პარაგრაფი აფართოვებს გასასვლელის შესრულების ნაკადის ძირითად წესებს როგორც განსაზღვრულია 3.5.1.2.-ში "გასასვლელის შესრულების ნაკადის საერთო შეერთებები", გვ. 102. იხ. პარაგრაფი 2.4.1, "შესრულების ნაკადის წესები", გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

ამ გასასვლელის ქცევის განსაკუთრებული ბუნების განსაზღვრა კრებადი შესრულების ნაკადისათვის:

- თუ გვაქვს მრავალი შემავალი შესრულების ნაკადი, ყველა მათგანი იქნება გამოყენებული პროცესის ნაკადის გასაგრძელებლად (ისე თითქოს აქ არ იყო გასასვლელი). ეს ნიშნავს,
 - პროცესის ნაკადი გაგრძელდება როდესაც სიგნალი (სიმბოლო) მოდის შესრულების ნაკადის რომელიმე სიმრავლიდან.
 - სიგნალები სხვა შესრულების ნაკადიდან ამ სიმრავლის შიგნით (ფარგლებში) შეიძლება მოვიდნენ სხვა დროს და ნაკადი გაგრძელდება მათი წარმატებით მოსვლის შემთხვევაში, იმ სიგნალების განხილვის და სინქრონიზაციის გარეშე, რომლებიც მოვიდნენ სხვა შესრულების ნაკადიდან.

ამ გასასვლელის ქცევის განსაკუთრებული ბუნების განსაზღვრა განშლადი შესრულების ნაკადისათვის:

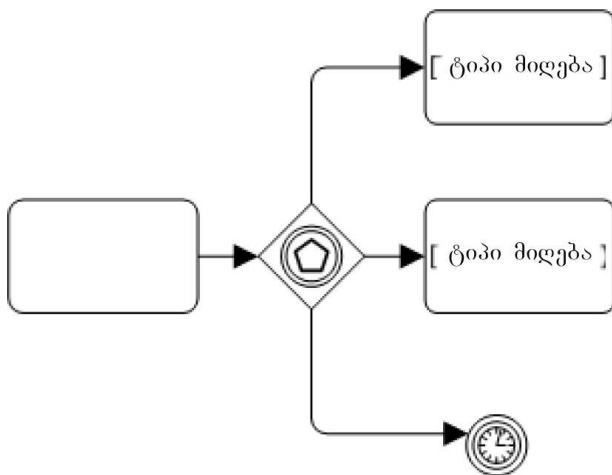
- თუ გვაქვს მრავალი გამომავალი შესრულების ნაკადი, მაშინ მხოლოდ ერთი კარები (ან გამოუცხადებელიკარები) იქნება შერჩეული პროცესის შესრულების დროს.
 - კარები იქნება არჩეული პირობაგამოსახულების შეფასების შედეგზე დაყრდნობით, რომელიც არის განსაზღვრული კარებთან დაკავშირებული შესრულების ნაკადისათვის.
 - კარებებთან დაკავშირებული პირობები იქნება შეფასებული იმ მიმდევრობით, რომლითაც კარებები გამოჩნდებიან გასასვლელის სიაში.
 - თუ პირობაგამოსახულება არის შეფასებული როგორც “ჭეშმარიტი”, მაშინ ეს კარები იქნება ამორჩეული და სიაში დარჩენილი დანარჩენი კარებები არ იქნებიან შეფასებული.
 - თუ არც ერთი პირობაგამოსახულება კარებებისათვის არ იქნებიან შეფასებული როგორც “ჭეშმარიტი”, მაშინ გამოუცხადებელიკარები იქნება არჩეული.

შენიშვნა. თუ გასასვლელს არა აქვს გამოუცხადებელიკარები და არც ართი კარები პირობაგამოსახულება არ არის შეფასებული როგორც “ჭეშმარიტი”, მაშინ პროცესი არის მახინჯი (არასწორე) მოდელის მქონე.

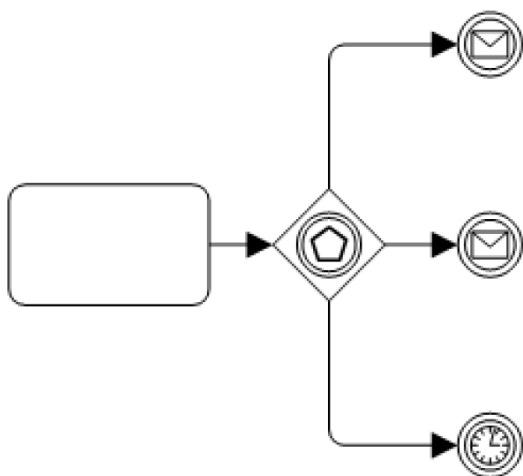
3.5.2.4. ხდომილებაზე დაფუძნებული

ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელების ჩართვა განაწილებულ სისტემებში მოხდა უახლოეს წარსულში. მონაცემების შეტანის მხრივ მისი ქცევა მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელის ანალოგიურია (იხ. 3.5.2.1. მონაცემებზე დაფუძნებული, გვ. 105). მონაცემების გამოტანის მხრივ, ძირითადი იდეა მდგომარეობს იმაში, რომ ეს გადაწყვეტილება წარმოადგენს განშტოების წერტილს პროცესში, სადაც ალტერნატივები არიან ხდომილებაზე დაფუძნებული, რომლებიც გვხვდება პროცესის ამ წერტილში, და არა გამოსახულებების შეფასებებზე, რომლებიც იყენებენ პროცესის მონაცემებს. განსაზღვრული ხდომილება, ჩვეულებრივ შეტყობინების მიღება, განსაზღვრავს რომელი ბილიკი იქნება არჩეული. მაგალითად, თუ კომპანია ელოდება მყიდველისაგან პასუხს, ისინი შეასრულებენ მოქმედებების ერთ სიმრავლეს თუ მყიდველი უპასუხებს “დიახ” და სხვა მოქმედებების სიმრავლეს თუ მყიდველი უპასუხებს “არა”. მყიდველის პასუხი განსაზღვრავს თუ რომელი ბილიკი იქნება არჩეული. შეტყობინების იდენტურობა განსაზღვრავს ბილიკის არჩევას. ეს ნიშნავს, რომ “დიახ” შეტყობინება და “არა” შეტყობინება არიან სხვადასხვა შეტყობინებები; ისინი არ არიან ერთი და იგივე შეტყობინება სხვადასხვა მნიშვნელობებით შეტყობინების თვისების შიგნით. შეტყობინების მიღება შეიძლება იყოს მოდელირებული ამოცანატიპის მიღების ამოცანასთან ერთად ან შუალედურ ხდომილებასთან ერთად შეტყობინების გადამრთველით. შეტყობინებების დამატებით შუალედური ხდომილებების სხვა გადამრთველები (Triggers) შეიძლება იყვნენ გამოყენებული, ისეთები როგორც ტაიმერებია.

- ხლომილებაზე დაფუძნებულმა განსაკუთრებულმა გასასვლელმა უნდა გამოიყენოს მარკერი (რაც იგივეა რაც მრავალჯერადი (Multiple) შუალედური ხლომილება), რომელიც განთავსებულია გასასვლელის რომბის შიგნით (იხ. ნახ. 3.23 და ნახ. 3.24) იმისათვის, რომ ის განსხვავდებოდეს სხვა გასასვლელებისაგან.
- ხლომილებაზე დაფუძნებული განსაკუთრებული გადაწყვეტილებები არიან კონფიგურირებული გამოძავალი შესრულების ნაკადის მიზნის მქონე ამოცანატიპის მიღების ამოცანით ან შუალედური ხლომილებით (იხ. ნახ. 3.23 და ნახ. 3.24).
 - ყველა გამოძავალ შესრულების ნაკადს უნდა ჰქონდეს ამ ტიპის მიზანი; მოცემული გადაწყვეტილებისათვის არ უნდა იყოს არეული პირობის გამოსახულებები და შუალედური ხლომილებები.



ნახ. 3.23. ხლომილებაზე დაფუძნებული გადაწყვეტილების (გასასვლელის) მაგალითი, რომელიც იყენებს მიღების ამოცანებს.



ნახ. 3.24. ხლომილებაზე დაფუძნებული გადაწყვეტილების (გასასვლელის) მაგალითი, რომელიც იყენებს შეტყობინების ხლომილებებს.

რადგან ეს გასასვლელი არის განსაკუთრებული გასასვლელი, შერწყმის ფუნქციონალურობა ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელისათვის იგივეა რაც მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელი, რომელიც აღწერილია წინა პარაგრაფში.

გასასვლელი შეიძლება იყოს გამოყენებული პროცესის დასაწყებად. ამ გაგებით, პროცესი გაუმჯობესებულია შეტყობინების მიღებით. რაიმე შეტყობინების მიღება, განსაზღვრული გასასვლელის კონფიგურაციით, მოახდენს პროცესის მყისიერ (დაუყოვნებლივ) შესრულებას. ამრიგად, გასასვლელი უზრუნველყოფს ალტერნატიული გზების სიმრავლეს დაწყებული პროცესისათვის.

გასასვლელისათვის, რომელიც ახდენს პროცესის მყისიერ შესრულებას, უნდა შეგვხვდეს შემდეგი პირობებიდან ერთ-ერთი:

- პროცესს არა აქვს დაწყების ხდომილება და გასასვლელს არა აქვს შემავალი შესრულების ნაკადი.
- გასასვლელისათვის შემავალ შესრულების ნაკადს აქვს დაწყების ხდომილების წყარო.
 - შევნიშნოთ, რომ არც ერთი სხვა შემავალი შესრულების ნაკადი არ არის ნებადართული გასასვლელისათვის (კერძოდ, მარყუქი შეერთება დინების მიმართულებით ქვემოთ განთავსებული ობიექტიდან).
- გასასვლელის გამომავალი შესრულების ნაკადისათვის მიზანი არ უნდა იყოს ტაიმერი, შუალედური ხდომილება.

3.5.2.5. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელის ატრიბუტები. ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში თუ გასასვლელიტიპი ატრიბუტი არის დაყენებული განსაკუთრებულზე. შემდეგი ატრიბუტები აფართოებენ გასასვლელის ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.38).

ცხრილი 3.35. ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელის ატრიბუტები.

ატრიბუტები	აღწერა
განსაკუთრებულიტიპი (მონაცემები ხდომილება) ხდომილება : სტრიქონი ExclusiveType (Data Event) Event : String	განსაკუთრებულიტიპი გამოუცხადებლად არის მონაცემები (Data). განსაკუთრებულიტიპი შეიძლება იყოს დაყენებული ხდომილებაზე (Event). რადგან ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელი არიან ამ პარაგრაფის მიზანი, ატრიბუტი უნდა იყოს დაყენებული ხდომილებაზე ატრიბუტებისათვის და ამ პარაგრაფში განსაზღვრული ქცევა უნდა იყოს გამოყენებული გასასვლელისათვის.

<p>გამყისიერება არა ჭეშმარიტი : ბულის Instantiate False : Boolean</p>	<p>ხდომილებაზე დაფუძნებული გასასვლელი შეიძლება იყვნენ განსაზღვრული როგორც გამყისიერების მაქანიზმი პროცესისათვის მყისიერად შესრულების ატრიბუტით. ეს ატრიბუტი შეიძლება იყოს დაყენებული ჭეშმარიტზე თუ გასასვლელი არის პირველი ელემენტი დაწყების ხდომილების შემდეგ ან არის დაწყების გასასვლელი, თუ არა გვაქვს დაწყების ხდომილება (ე.ი. არ არის შემომავალი შესრულების ნაკადი).</p>
--	---

3.5.2.6. შესრულების ნაკადის შეერთებები

ეს პარაგრაფი აფართოებს გასასვლელის შესრულების ნაკადის შეერთების ძირითად წესებს, როგორც ეს არის განსაზღვრული 3.5.1.2.-ში “გასასვლელის შესრულების ნაკადის საერთო შეერთებები”, გვ. 102. იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადი წყარო და მიზნები.

კრებადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის განსაკუთრებული ბუნების განსაზღვრა:

- თუ არსებობს მრავლობითი შემავალი შესრულების ნაკადი, ყველა მათგანი იქნება გამოყენებული პროცესის ნაკადის გასაგრძელებლად (ისე თითქოს არ არის გასასვლელი). ეს არის
 - პროცესის ნაკადი გაგრძელდება როდესაც სიგნალი (სიმბოლო) მოდის შესრულების ნაკადის რაიმე სიმრავლიდან.
 - ამ სიმრავლის სხვა შესრულების ნაკადიდან სიგნალები შეიძლება მოვიდნენ სხვა დროს და ნაკადი ასევე გაგრძელდება როდესაც ისინი მოვლენ, იმ სიგნალების განხილვის ან სინქრონიზაციის გარეშე, რომლებიც მოვიდნენ სხვა შესრულების ნაკადიდან.

განშლადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის განსაკუთრებული ბუნების განსაზღვრა:

- მხოლოდ ერთი კარები იქნება შერჩეული პროცესის შესრულების განმავლობაში.
 - კარები იქნება არჩეული კარების შესრულების ნაკადის მიზანზე დაყრდნობით.
 - თუ მიზანი შესრულებულია მყისიერად (ანუ შეტყობინება მიღებულია ან დრო არის გადაჭარბებული (ამოწურული)), მაშინ კარები იქნება არჩეული და დარჩენილი კარებები არ უნდა იყვნენ შეფასებული (ე.ი. მათი მიზნები იქნებიან განუსორციელებელი (disabled)).
- გამომავალი შესრულების ნაკადის პირობის ატრიბუტი უნდა იყოს დაყენებული ცარიელზე (None).

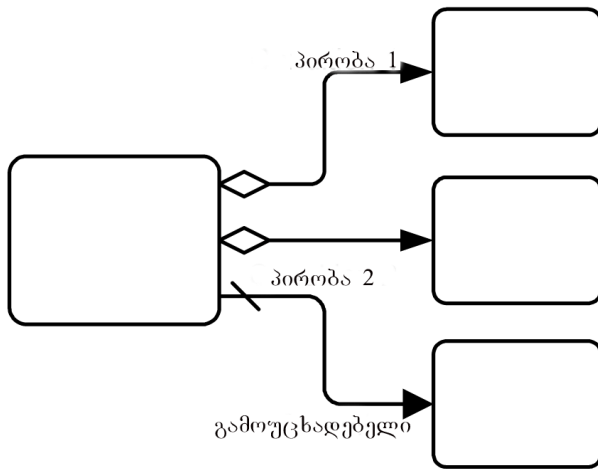
- გასასვლელის გამომავალი შესრულების ნაკადის მიზანი უნდა იყოს შემდეგი ობიექტებიდან ერთ-ერთი:
 - ამოცანა ამოცანატიპი ატრიბუტით დგება მიღებაზე.
 - შუალედური ხდომილება მიზნის ატრიბუტით დგება შეტყობინებაზე, ტაიმერზე, სიგნალიზე.
 - თუ ერთი კარების მიზანი არის ამოცანა, მაშინ შუალედური ხდომილება გადამრთველით შეტყობინება არ უნდა იყოს გამოყენებული როგორც სხვა კარების მიზანი. *ეს ნიშნავს, რომ შეტყობინება უნდა იყოს მიღებული მხოლოდ მიღების ამოცანებით ან მხოლოდ შეტყობინების ხდომილებებით, მაგრამ არა ორივეს ნარევით მოცემული გასასვლელისათვის.*

3.5.3. შემცველი გასასვლელები

ეს გადაწყვეტილება წარმოადგენს განშტოების წერტილს, სადაც ალტერნატივები არიან დაფუძნებული გამომავალ შესრულების ნაკადში მოცემულ პირობით გამოსახულებებზე. და მაინც, ამ შემთხვევაში, ერთი პირობითი გამოსახულების ჭეშმარიტი შეფასება არ გამოორიცხავს სხვა პირობითი გამოსახულებების შეფასებას. ყველა შესრულების ნაკადი ჭეშმარიტი შეფასებით იქნება გადაკვეთილი სიმბოლოთი. რაღაც გაგებით ეს გავს დაკავშირებული დამოუკიდებელი ორობითი (კი/არა) გადაწყვეტილებების დაჯგუფებას და შეიძლება იყოს მოდელირებული ამ გზით. რადგან ყოველი ბილიკი არის დამოუკიდებელი, ბილიკების ყველა კომბინაცია შეიძლება იყოს შერჩეული, ნოლიდან ყველამდე. და მაინც, ის უნდა იყოს შედგენილი (დაპროექტებული) ისე, რომ სულ მცირე ერთი ბილიკი იყოს არჩეული.

შენიშვნა. თუ არც ერთი შემცველი გადაწყვეტილების კარების პირობაგამოსახულება არ არიან შეფასებული როგორც “ჭეშმარიტი”, მაშინ პროცესი განხილულია მახინჯი (ინვალიდი) მოდელის მქონედ.

ამ ტიპის გადაწყვეტილების მოდელირებისათვის არსებობს ორი მექანიზმი. შემცველი გადაწყვეტილების ადგილმდებარეობების მოდელირების პირველი მეთოდი არსებითად არ იყენებს შემცველ გასასვლელს, მაგრამ სამაგიეროდ იყენებს პირობითი შესრულების ნაკადის კრებულს, რომელიც აღნიშნულია მინი რომბებით; კარებები გასასვლელების გარეშე (იხ. ნახ. 3.25). პირობით შესრულების ნაკადს მისი პირობითი ატრიბუტი დაყენებული აქვს გამოსახულებაზე და პირობაგამოსახულება დაყენებული აქვს ბულის მათემატიკურ გამოსახულებაზე, პროცესში არსებულ ინფორმაციაზე დაყრდნობით. ეს შესრულების ნაკადები ნაჩვენებია არიან “მინი-რომბების” მარკერებით შესრულების ნაკადის ხაზის დასაწყისში.



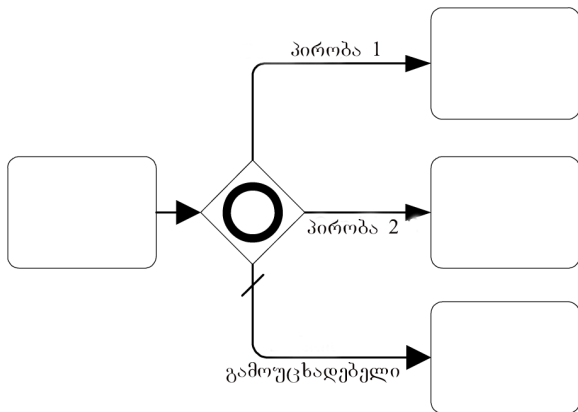
ნახ. 3.25. შემცველი გადაწყვეტილება რომელიც იყენებს პირობით შესრულების ნაკადს.

პირობითი შესრულების ნაკადის (მინი-რომბებით) გამოყენებაში არსებობს ზოგიერთი შეზღუდვა:

- წყარო ობიექტი არ უნდა იყოს ხდომილება. წყარო ობიექტი შეიძლება იყოს გასასვლელი, მაგრამ მინი-რომბი არ უნდა იყოს ნაჩვენები ამ შემთხვევაში. წყარო ობიექტი შეიძლება იყოს მოქმედება (ამოცანა ან ქვე-პროცესი) და მინი-რომბი იქნება ნაჩვენები ამ შემთხვევაში.
 - წყარო გასასვლელი არ უნდა იყოს პარალელური ან რთული ტიპის.
- თუ პირობითი შესრულების ნაკადი არის გამოყენებული წყარო მოქმედებიდან, მაშინ უნდა იყოს სულ ცოტა ერთი სხვა გამომავალი შესრულების ნაკადი ამ მოქმედებიდან.
 - დამატებითი შესრულების ნაკადი(ები) შეიძლება აგრეთვე იყოს პირობითი, მაგრამ არ მოითხოვება, რომ ისინი იყვნენ პირობითი.

შემცველი გადაწყვეტილების ადგილმდებარეობის მოდელირების მეორე მეთოდი გამოიყენებს შემცველ გასასვლელს (იხ. ნახ. 3.26), ზოგჯერ სხვა გასასვლელებთან კომბინაციაში. მარკერი განთავსებული იქნება გასასვლელის ცენტრში იმის საჩვენებლად, რომ გასასვლელის ქცევა არის შემცველი (inclusive).

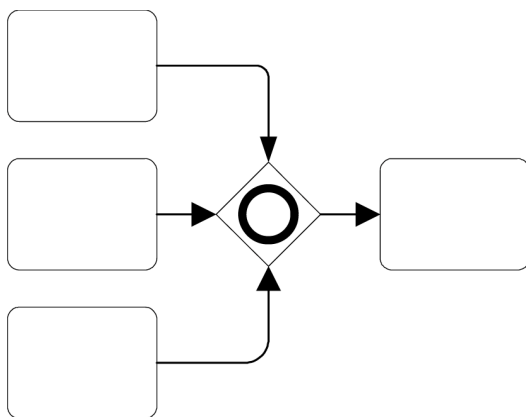
- შემცველმა გასასვლელმა უნდა გამოიყენოს მარკერი, რომელიც არის წრის ფორმაში ან “O” და განთავსებულია გასასვლელის რომბის შიგნით (იხ. ნახ. 3.26) სხვა გასასვლელებისაგან მის განსასხვავებლად.



ნახ. 3.26. შემცველი გადაწყვეტილება, რომელიც იყენებს შემცველ გასასვლელს.

ნახ. 3.25-ზე ნაჩვენები მოდელის ქცევა ექვივალენტურია ნახ. 3.26-ზე ნაჩვენები მოდელის ქცევისა. ვიმეორებთ, რომ ის არის მოდელიორამდე იმისათვის, რომ მოგვცეს გარანტია, რომ სულ მცირე ერთი პირობა იქნება ჭეშმარიტი როდესაც პროცესი შესრულდება.

როდესაც შემცველი გასასვლელი არის გამოყენებული როგორც შერწყმა, ის გაუკეთებს სინქრონიზაციას ყველა სიმბოლოს, რომლებიც იყენენ წარმოებული მანამდე (დინების მიმართულებით ზევით), მაგრამ მაქსიმუმ ერთს ყოველი შემავალი შესრულების ნაკადისათვის. **შენიშვნა:** მოითხოვება, რომ სიმბოლოები ყოველი შესრულების ნაკადისათვის, რომლებიც ნამდვილად იყენენ გამოიმუშავებული ზემოთ (დინების მიმართულებით ზევით) (შემცველი ადგილმდებარეობით, მაგალითად) იყენენ სინქრონიზირებული. თუ დინების მიმართულებით ზევითა შემცველი აწარმოებს (გამოიმუშავებს) სამი შესაძლო სიმბოლოდან ორს, მაშინ დინების მიმართულებით ქვევითა შემცველი სინქრონიზაციას გაუკეთებს ამ ორ სიმბოლოს და არ დაელოდება სხვა სიმბოლოს, მაშინაც კი როცა არსებობს სამი შესრულების ნაკადი (იხ. ნახ. 3.27).



ნახ. 3.27. შემცველი გასასვლელი რომელიც შერწყავს (აერთებს) შესრულების ნაკადს.

3.5.3.1. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია ატრიბუტები შემცველი გასასვლელისათვის. ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში, თუ გასასვლელიტიპი ატრიბუტი არის დაყენებული შემცველზე. შემდეგი ატრიბუტები აფართოებენ გასასვლელის ატრიბუტების საერთო სიმრავლეს (იხ. ცხრილი 3.32).

ცხრილი 3.36. შემცველი გასასვლელის ატრიბუტები

ატრიბუტები	აღწერა
გამოუცხადებელიკარი (0-1) : კარი	გამოუცხადებელი კარი შეიძლება იყოს განსაზღვრული (იხ. პარაგრაფი 3.5.1.4, “კარები”, გვ. 103)

3.5.3.2. შესრულების ნაკადის შეერთებები

ეს პარაგრაფი აფართოებს გასასვლელის შესრულების ნაკადის შეერთების ძირითად წესებს, როგორც განსაზღვრულია 3.5.1.2.-ში “გასასვლელის შესრულების ნაკადის საერთო შეერთებები”, გვ. 102. იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

კრებადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის შემცველი ბუნების განსაზღვრა:

- თუ მოცემულია მრავლობითი შემავალი შესრულების ნაკადი, ერთი ან მეტი მათგანი იქნება გამოყენებული პროცესის ნაკადის გასაგრძელებლად. ეს არის,
 - პროცესის ნაკადი გაგრძელდება, როდესაც სიგნალები (სომბოლოები) მოვლენ ყველა შემავალი შესრულების ნაკადებიდან, რომლებიც ელოდებიან პროცესის ზევით აღმავალ (დინების მიმართულებით) სტრუქტურაზე დაფუძნებულ სიგნალს (ანუ ზემოთ მიმართული დინების მიმართულებით) შემცველი გადაწყვეტილება).
 - შემავალი შესრულების ნაკადიდან ნაწილს არ ექნება სიგნალები, ხოლო ნიმუში, რომლის შესრულების ნაკადს (Sequence Flow) ექნება სიგნალები, შეიძლება შეიცვალოს პროცესის სხვადასხვა შესრულებისათვის.

შენიშვნა. შემავალი შესრულების ნაკადი, რომელსაც აქვს წყარო, რომელიც არის (დინების მიმართულებით) ქვემოთ მიმართული მოქმედება (რაც ნიშნავს, რომ არის მარყუჟის ნაწილი), განხილული იქნება სხვანაირად, ვიდრე ისინი, რომლებსაც აქვთ ზემოთ მიმართული (დინების მიმართულებით) წყარო. ისინი იქნებიან განხილული, როგორც შესრულების ნაკადის სხვადასხვა სიმრავლეების ნაწილი, იმ

შესრულების ნაკადებიდან, რომლებსაც აქვთ წყარო, რომელიც არის დინების მმართველობით მოქმედება.

განშლადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის შემცველი ბუნების (inclusive nature) განსაზღვრა:

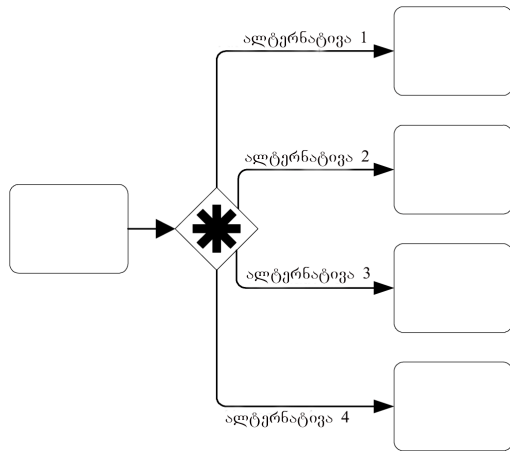
- ერთი ან მეტი კარებები უნდა იყვნენ შერჩეული პროცესის შესრულების დროს.
- კარებები უნდა იყვნენ შერჩეული პირობითი გამოსახულების საფუძველზე, ანუ უნდა იყვნენ განსაზღვრული კარებებთან დაკავშირებული შესრულების ნაკადისათვის.
 - ყველა კარებთან დაკავშირებული პირობა უნდა იყოს შეფასებული.
 - თუ პირობა შეფასებულია როგორც “ჭეშმარიტი”, მაშინ ეს კარები უნდა იყოს არჩეული, იმისგან დამოუკიდებლად რომელიმე სხვა კარებები იყვნენ თუ არ იყვნენ შერჩეული.
 - თუ პირობა გამოსახულებებიდან არც ერთი არ არის შეფასებული როგორც “ჭეშმარიტი”, მაშინ გამოუცხადებელი კარები უნდა იყოს არჩეული.

3.5.4. რთული გასასვლელები

BPMN შეიცავს რთულ გასასვლელებს იმ სიტუაციებთან გასამკლავებლად, რომლებიც არ არიან ადვილი გასამკლავებელი სხვა ტიპის გასასვლელებისათვის. რთულ გასასვლელები შეიძლება აგრეთვე გამოყენებული იყოს დაკავშირებული მარტივი გასასვლელების სიმრავლის ერთადერთ, უფრო კომპაქტურ სიტუაციაში, გასაერთიანებლად. მოდელირებს შეუძლიათ გამოიყენონ რთული გამოსახულებები, რომლებიც განსაზღვრავენ გასასვლელის შერწყმის და/ან გახლეჩის ქცევას.

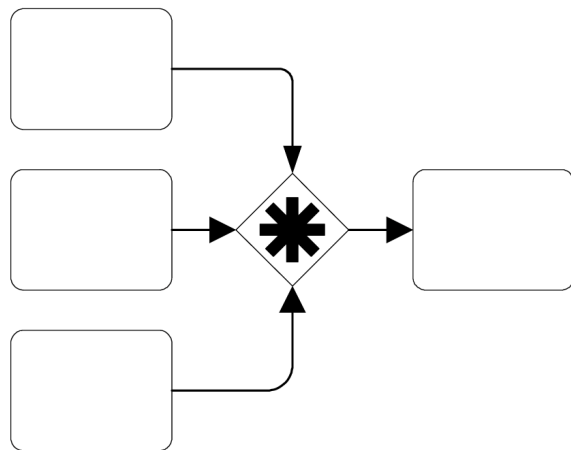
- რთულმა გასასვლელმა უნდა გამოიყენოს მარკერი, რომელიც არის ვარსკვლავის ფორმის და განთავსებულია გასასვლელის რომში (იხ. ნახ. 3.28) იმისათვის, რომ ის განსხვავდებოდეს სხვა გასასვლელებისაგან.

როდესაც გასასვლელი არის გამოყენებული როგორც გადაწყვეტილება (იხ. ნახ. 3.28), მაშინ გამოსახულება განსაზღვრავს გამომავალი შესრულების ნაკადებიდან რომელი იქნება არჩეული პროცესის გასაგრძელებლად. გამოსახულება შეიძლება იყენებდეს პროცესის მონაცემებს და შემაჯავლი შესრულების ნაკადის სტატუსს. მაგალითად, გამოსახულებამ შეიძლება შეაფასოს პროცესის მონაცემები და შემდეგ შეარჩიოს გამომავალი შესრულების ნაკადის სხვადასხვა სიმრავლეები, შეფასების შედეგებზე დაყრდნობით. თუმცა, გამოსახულება უნდა იყოს შექმნილი ისე, რომ სულ მცირე ერთი გამომავალი შესრულების ნაკადი იყოს შერჩეული.



ნახ. 3.28. რთული გადაწყვეტილება (გასასვლელი)

როდესაც გასასვლელი არის გამოყენებული როგორც შერწყმა (იხ. ნახ. 3.29), მაშინ იქნება გამოყენებული გამოსახულება, რომელიც განსაზღვრავს რომელი შემომაველი შესრულების ნაკადი იქნება მოთხოვნილი პროცესის გასაგრძელებლად. გამოსახულება შეიძლება იყენებდეს პროცესის მონაცემებს და შემავალი შესრულების ნაკადის სტატუსს. მაგალითად, გამოსახულება შეიძლება განსაზღვრავდეს, რომ ხუთი შემავალი სიმბოლოდან რომელიმე სამში გააგრძელებს პროცესს. სხვა მაგალითი შეიძლება იყოს გამოსახულება, რომელიც განსაზღვრავს, რომ სიმბოლო არის მოთხოვნილი “ა” შესრულების ნაკადიდან და რომ სიმბოლო სხვა ან “ბ” ან “გ” შესრულების ნაკადიდან არის მისაღები (is acceptable). თუმცა, გამოსახულება უნდა იყოს დამუშავებული ისე, რომ პროცესი არ გაიჭედოს (შეფერხდეს) ამ ადგილას.



ნახ. 3.29. რთული შერწყმა (გასასვლელი)

3.5.4.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია რთული გასასვლელის ატრიბუტები. ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში თუ გასასვლელიტიპი ატრიბუტი არის დაყენებული რთულზე. შემდეგი ატრიბუტები აფართოებენ გასასვლელის საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.32).

ცხრილი 3.37. რთული გასასვლელის ატრიბუტები

ატრიბუტები	აღწერა
შემავალიპირობა (0-1) : გამოსახულება IncomingCondition (0-1) : Expression	თუ გვაქვს მრავალი შემავალი შესრულების ნაკადი, შემავალიპირობა გამოსახულება უნდა იყოს დაყენებული მოდელიორის მიერ. მის შემადგენლობაში იქნება გამოსახულება, რომელსაც შეუძლია მიანიშნოს შესრულების ნაკადის სახელები და/ან პროცესის თვისებები (მონაცემები).
გამომავალიპირობა (0-1) : გამოსახულება OutgoingCondition (0-1) : Expression	თუ გვაქვს მრავალი გამომავალი შესრულების ნაკადი, გამომავალიპირობა გამოსახულება უნდა იყოს დაყენებული მოდელიორის მიერ. მის შემადგენლობაში იქნება გამოსახულება, რომელსაც შეუძლია მიანიშნოს (გამომავალი) შესრულების ნაკადის Ids (იდენტიფიკატორები) და/ან პროცესის თვისებები (მონაცემები).

3.5.4.2. შესრულების ნაკადის შეერთებები

ეს პარაგრაფი აფართოებს გასასვლელის შესრულების ნაკადის შეერთების ძირითად წესებს როგორც განსაზღვრულია 3.5.1.2.-ში “გასასვლელის შესრულების ნაკადის საერთო შეერთებები”, გვ. 102. იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

კრებადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის რთული ბუნების განსაზღვრა:

- თუ გვაქვს მრავალი (მრავლობითი) შემავალი შესრულების ნაკადი, ერთი ან მეტი მათგანი იქნება გამოყენებული პროცესის ნაკადის გასაგრძელებლად.
 - პროცესის ნაკადი გაგრძელდება, როდესაც სიგნალების (სიმბოლოების) შესაბამისი

რაოდენობა მოვა შესაბამისი შემავალი შესრულების ნაკადიდან.

- ამ სიმრავლის სხვა შესრულების ნაკადიდან სიგნალები შეიძლება მოვიდნენ, მაგრამ ისინი არ უნდა იყვნენ გამოყენებული პროცესის ნაკადის გასაგრძელებლად.

შენიშვნა. შემომავალი შესრულების ნაკადი, რომელსაც აქვს წყარო, რომელიც არის დინების მიმართულებით ქვემოთ მდებარე მოქმედება (ანუ არის მარყუჟის ნაწილი), იქნება სხვანაირად დამუშავებული ვიდრე ისინი, რომლებსაც აქვთ დინების მიმართულებით ზემოთ მდებარე წყარო. ისინი იქნებიან განხილული როგორც შესრულების ნაკადის სხვადასხვა სიმრავლის ნაწილი, იმ შესრულების ნაკადიდან, რომელსაც აქვს წყარო, რომელიც არის დინების მიმართულებით ზემოთ მდებარე მოქმედება.

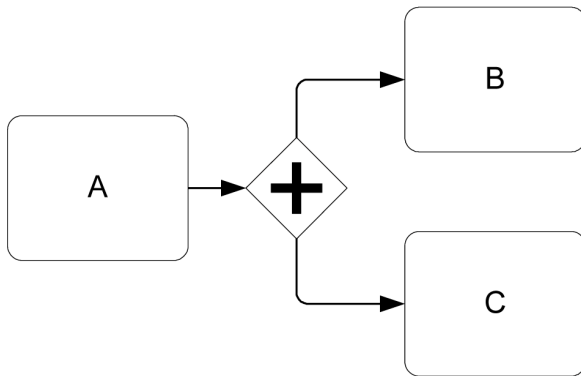
განშლადი შესრულების ნაკადისათვის ამ გასასვლელის შემცველი ბუნების განსაზღვრა:

- ერთი ან მეტი კარებები იქნება არჩეული პროცესის შესრულების დროს.
 - კარებები იქნება არჩეული გასასვლელის გამომავალიპირობა გამოსახულებაზე დაყრდნობით.

3.5.5. პარალელური გასასვლელები

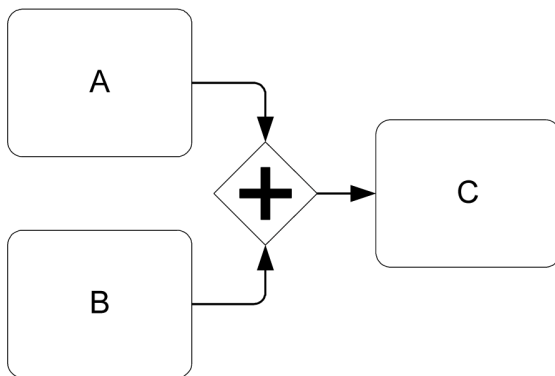
პარალელური გასასვლელები უზრუნველყოფენ პარალელური ნაკადის სინქრონიზაციის მექანიზმს და ქმნიან პარალელურ ნაკადს. ეს გასასვლელები არ მოითხოვება პარალელური ნაკადის შესაქმნელად, მაგრამ ისინი შეიძლება იყვნენ გამოყენებული რთული სიტუაციების ქცევის განმარტებისათვის, სადაც გასასვლელების მიმდევრობა არის გამოყენებული და პარალელური ნაკადი არის მოთხოვნილი. დამატებით, ზოგიერთ მოდელიორს შეიძლება სურდეს შექმნას “თვალსაჩინო გამოყენება”, სადაც ყოველთვის გამოყენებულია პარალელური გასასვლელები პარალელური ბილიკების შესაქმნელად. ეს გამოყენება შექმნის მოდელირების დამატებით ელემენტს, სადაც ელემენტების განშტოება და მიერთება შეიძლება იყვნენ შეწყვილებული.

- პარალელურმა გასასვლელმა უნდა გამოიყენოს მარკერი, რომელიც არის პლიუს ნიშნის ფორმის და განთავსებულია გასასვლელის რომბის შიგნით (იხ. ნახ. 3.30) მისი სხვა გასასვლელებისაგან განსასხვავებლად.



ნახ. 3.30. პარალელური გასასვლელი.

პარალელური გასასვლელები გამოიყენება პარალელური ნაკადის სინქრონიზაციისათვის.



ნახ. 3.31. შეერთება – პარალელური ბილიკების შეერთება.

3.5.5.1. ატრიბუტები

პარალელურ გასასვლელებს არა აქვთ რაიმე დამატებითი ატრიბუტები გასასვლელის საერთო ატრიბუტების გარდა (იხ. ცხრილი 3.32).

3.5.5.2. შესრულების ნაკადის შეერთებები

ეს პარაგრაფი აფართოებს გასასვლელის შესრულების ნაკადის შეერთების ძირითად წესებს როგორც ეს გამსაზღვრულია 3.5.1.2.-ში “გასასვლელის შესრულების ნაკადის საერთო შეერთებებ”, გვ. 102. იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყენენ შესრულების ნაკადის წყარო ან მიზნები.

კრებადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის პარალელური ბუნების განსაზღვრა:

- თუ გვაქვს მრავალი (მრავლობითი) შემავალი შესრულების ნაკადი, ყველა მათგანი იქნება გამოყენებული პროცესის ნაკადის გასაგრძელებლად; ნაკადი იქნება სინქრონიზირებული. ეს ნიშნავს,
 - პროცესის ნაკადი უნდა გაგრძელდეს, როდესაც სიგნალი (სიმბოლო) მოსულია შესრულების ნაკადის ყველა სიმრავლიდან (ე.ი. პროცესი დაელოდება ყველა სიგნალის მოსვლას მის გაგრძელებამდე).

შენიშვნა. შემომავალი შესრულების ნაკადი, რომელსაც აქვს წყარო, რომელიც არის დინების მიმართულებით ქვემოთ განთავსებული მოქმედება (ანუ არის მარჯულის ნაწილი) დამუშავებული იქნება სხვანაირად, ვიდრე ისინი რომლებსაც აქვთ დინების მიმართულებით ზემოთ განთავსებული წყარო. ისინი იქნებიან განხილული როგორც შესრულების ნაკადის სხვა სიმრავლის ნაწილი, იმ შესრულების ნაკადებიდან, რომლებსაც აქვთ წყარო, რომელიც არის დინების მიმართულებით ზემოთ განთავსებული მოქმედება.

განშლადი შესრულების ნაკადისათვის ამ გასასვლელის ქცევის პარალელური ბუნების განსაზღვრა:

- ყველა კარებები იქნებიან შერჩეული პროცესის შესრულების დროს (განმაგლობაში).

3.6. პროცესის მონაწილენი (Swimlanes) (გუბეები და ბილიკები) (Pools and Lanes)

BPMN იყენებს კონცეფციას რომელიც ცნობილია როგორც “პროცესის მონაწილენი” (გუბეები და ბილიკები), რომ განახორციელოს დაყოფა და ორგანიზება გაუკეთოს მოქმედებებს. შესაძლებელია, რომ BPMN-ის დიაგრამა წარმოადგენდეს ერთზე მეტ კერძო პროცესს, ანალოგიურად პროცესებისა, რომლებიც აჩვენებენ თანამშრომლობას კერძო პროცესებს და მონაწილეებს შორის. თუ ეს ასეა, მაშინ ყოველი კერძო ბიზნეს პროცესი იქნება განხილული როგორც შესრულებული სხვადასხვა მონაწილის მიერ. გრაფიკულად, ყოველი მონაწილე იქნება გამოყოფილი; ეს ნიშნავს, რომ იქნება შეცვლილი სწორკუთხა ყუთში, რომელსაც ეწოდება “გუბე” (“Pool”). გუბეებს შეიძლება ჰქონდეთ ქვე-პროცესის მონაწილენი (sub-Swimlanes), რომლებსაც მარტივად ეწოდებათ “ბილიკები” (“Lanes”).

პარაგრაფი 1.1.1, “BPMN-ის გამოყენება”, გვ. 10, აღწერს BPMN-ის გამოყენებას კერძო პროცესების მოდელირებისათვის და პროცესების ურთიერთმოქმედებას. გუბეები და ბილიკები არიან დამუშავებული BPMN-ის ამ გამოყენებების უზრუნველსაყოფად.

3.6.1. პროცესის მონაწილის საერთო ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია პროცესის მონაწილეებისათვის (გუბეების და ბილიკებისათვის) საერთო ატრიბუტების სიმრავლე და აფართოებს BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.1):

ცხრილი 3.38. პროცესის მონაწილეების საერთო ატრიბუტები

ატრიბუტები	აღწერა
სახელი : სტრიქონი Name : String	სახელი არის ატრიბუტი, რომელიც არის პროცესის მონაწილის ტექსტური აღწერა.

3.6.2. გუბე

გუბე წარმოადგენს პროცესის მონაწილეს. მონაწილე შეიძლება იყოს კოკრეტული ბიზნეს ორგანიზაცია (მაგალითად, კომპანია) ან შეიძლება იყოს უფრო ზოგადი საქმიანი როლი (მაგალითად, მყიდველი, გამყიდველი ან მწარმოებელი). გრაფიკულად, გუბე არის კონტეინერი პროცესის გამოყოფისათვის სხვა გუბეებიდან, კორპორაციული (შიდა საქმიანი) (business-to-business) სიტუაციების მოდელირებისათვის, თუმცა გუბეს არ სჭირდება რაიმე შიდა დეტალების ქონა (ე.ი. ის შეიძლება იყოს “შავი ყუთი”).

- გუბე არის სწორკუთხედი, რომელიც უნდა იყოს დახაზული ერთი მასიური შავი ხაზით (როგორც ჩანს ნახაზი 3.32-დან).
 - ერთი და მხოლოდ ერთი გუბე შეიძლება იყოს წარმოდგენილი საზღვრების გარეშე დიაგრამაში. თუ დიაგრამაში ერთზე მეტი გუბეა, მაშინ დანარჩენ გუბეებს უნდა ჰქონდეთ საზღვრები.
 - გუბეებისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, განსაზღვრული წესების თანახმად, გვ. 33.



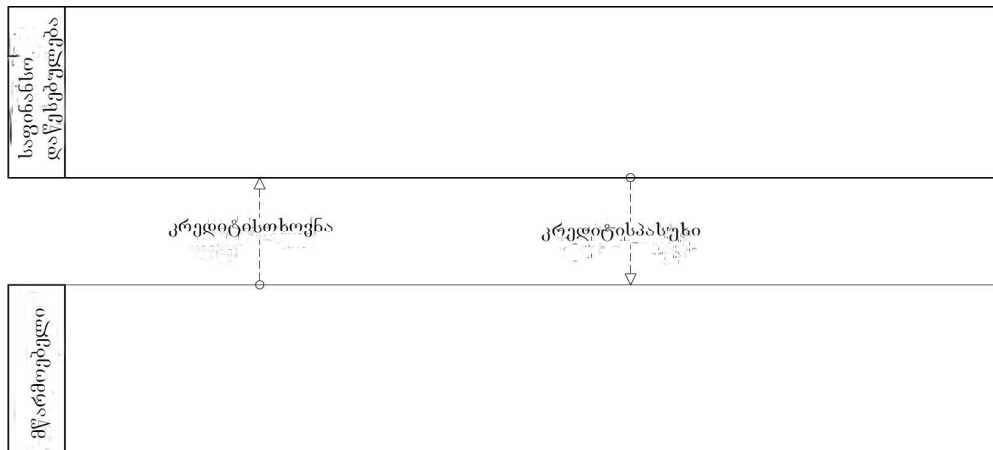
ნახ. 3.32. გუბე.

დიაგრამაში სიცხადის შეტანისათვის, გუბე გააფართოებს დიაგრამის საერთო სიგრძეს, ან ჰორიზონტალურს, ან ვერტიკალურს. თუმცა, აქ არ არსებობს

სპეციფიური შეზღუდვები გუბის ზომებზე და/ან განთავსებაზე. მოდელირებს და მოდელირების ინსტრუმენტებს შეუძლიათ გამოიყენონ გუბები (და ბილიკები) მოქნილად დიაგრამის “რეალური ქონების” შენახვის მიზნებისათვის ეკრანზე ან საბეჭდო გვარდზე.

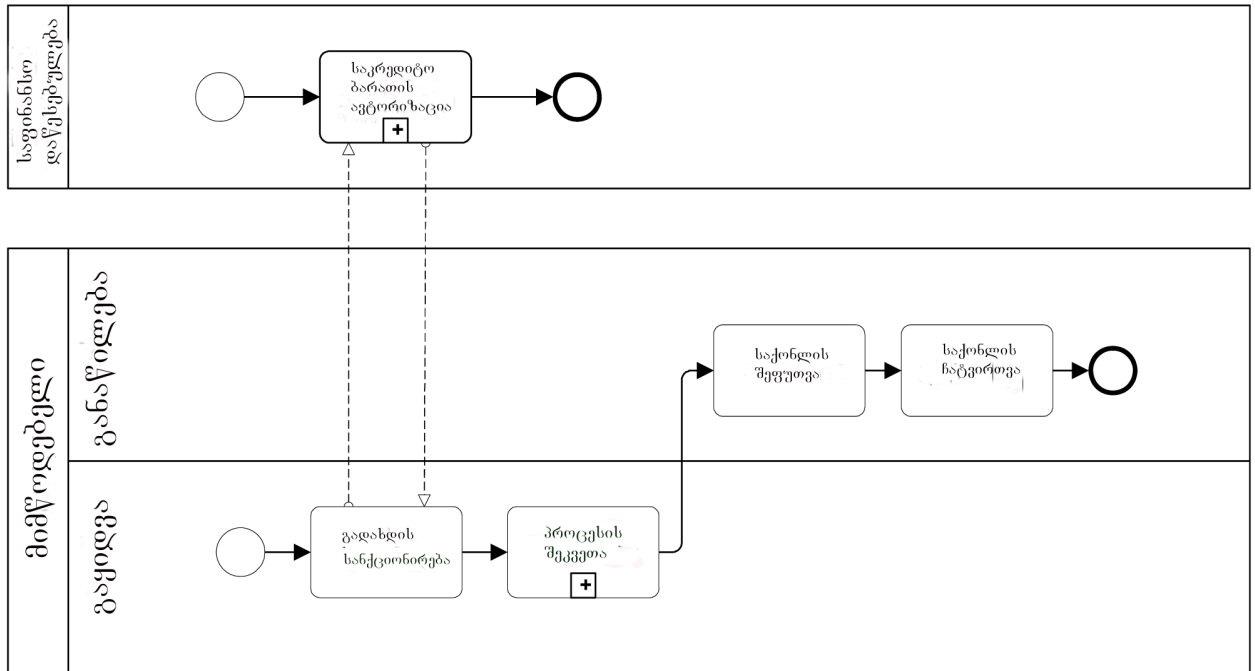
გუბე მოქმედებს როგორც კონტეინერი (შემცველი) შესრულების ნაკადისათვის მოქმედებს შორის. შესრულების ნაკადს შეუძლია გადაკვეთოს საზღვრები გუბის ბილიკებს შორის, მაგრამ არ შეუძლიათ გადაკვეთონ გუბის საზღვრები. გუბების ურთიერთზემოქმედება ნაჩვენებია შეტყობინების ნაკადის საშუალებით.

გუბების სხვა მახასიათებელია შემდეგი: არის თუ არა რაიმე მოქმედება დეტალურად ნაჩვენები გუბის შიგნით. ამრიგად, მოცემული გუბე შეიძლება იყოს ნაჩვენები როგორც “თეთრი ყუთი” ყველა ნაჩვენები დეტალით ან როგორც “შავი ყუთი” ყველა დამალული დეტალით. არანაირი შესრულების ნაკადი არ არსებობს “შავი ყუთის” გუბეში, მგრამ შეტყობინების ნაკადი შეიძლება მიედგას მის საზღვრებს (იხ. ნახ. 3.33).



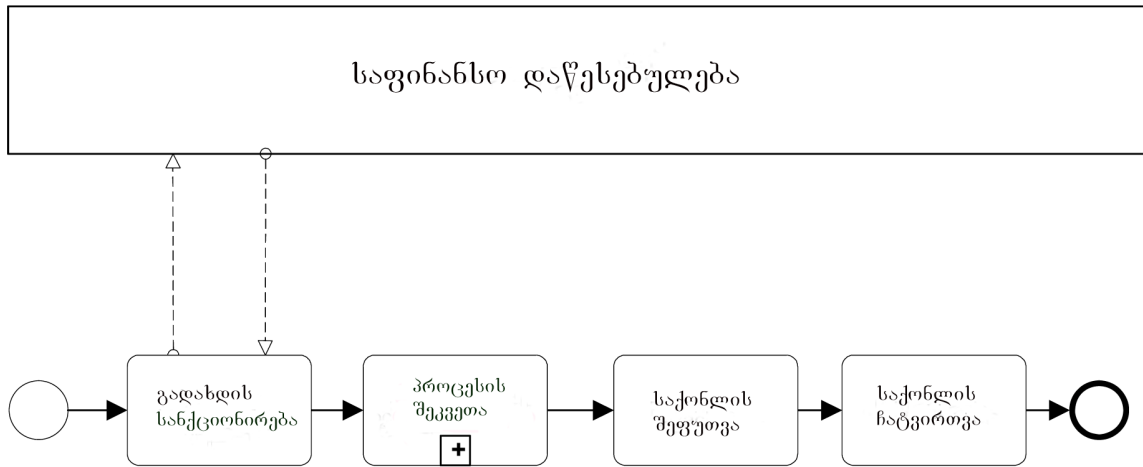
ნახ. 9.33. შეტყობინების ნაკადი დაკავშირებული ორი გუბის საზღვრებთან.

“თეთრი ყუთის” გუბისათვის, მოქმედებები საზღვრებში ორგანიზებული არიან შესრულების ნაკადით. შეტყობინების ნაკადმა შეიძლება გადაკვეთოს გუბის საზღვარი, რომ მიედგას (მიმაგრდეს) შესაბამის მოქმედებას (იხ. ნახ. 3.34).



ნახ. 3.34. შეტყობინების ნაკადი დაკავშირებული ნაკადის ობიექტებთან ორი გუბის შიგნით.

ბიზნეს პროცესების ყველა დიაგრამა (BPD) შეიცავს სულ ცოტა ერთ გუბეს. უმრავლეს შემთხვევაში, ბიზნეს პროცესების დიაგრამა, რომელიც შეიცავს ერთადერთ გუბეს, აჩვენებს მხოლოდ პროცესის მოქმედებებს და არ აჩვენებს გუბის საზღვრებს. გარდა ამისა, ბიზნეს პროცესების დიაგრამამ შეიძლება აჩვენოს “მთავარი” გუბე საზღვრების გარეშე. ასეთ შემთხვევებში დიაგრამაში შეიძლება იყოს, მხოლოდ ერთი უხილავად შემოსადგურული გუბე და ამ გუბის სახელი უნდა იყოს იგივე რაც დიაგრამისა. შესაბამისად, მოქმედებები, რომლებიც წარმოადგენენ სამუშაოს, შესრულებულს მოდელიორის ან მოდელიორის ორგანიზაციის თვალსაზრისით, არის განხილული “შიდა” მოქმედებებად და არ უნა იყოს შემოფარგლული გუბის საზღვრებით, მაშინ როდესაც სხვა გუბეებს დიაგრამაში უნდა ჰქონდეთ თავიანთი საზღვრები (იხ. ნახ. 3.35).



ნახ. 3.35. მთავარი (შიდა) გუბე საზღვრების გარეშე.

3.6.2.1. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია გუბის იდენტიფიცირებული ატრიბუტები; ის აფართოებს პროცესის მონაწილის (Swimlane) საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.38).

ცხრილი 3.39. გუბის ატრიბუტები

ატრიბუტები	აღწერა
პროცესიRef (0-1) : პროცესი ProcessRef (0-1) : Process	პროცესიRef ატრიბუტი განსაზღვრავს პროცესს, რომელიც განთავსებულია გუბეში. ყოველ გუბეს შეიძლება ჰქონდეს პროცესი. პროცესის ატრიბუტები შეიძლება ნახოთ პარაგრაფ 2.6-ში, “პროცესები”, გვ. 37.
მონაწილეRef : მონაწილე ParticipantRef : Participant	მოდელირება გუბისათვის უნდა განსაზღვროს მონაწილე. მონაწილე შეიძლება იყოს ან როლი ან ორგანიზაცია. მონაწილისათვის არსებობენ სპეციალური ატრიბუტები, რომლებსაც აქ არ განვიხილავთ.
ბილიკები (1-n) : ბილიკი Lanes (1-n) : Lane	გუბის შიგნით უნდა იყოს ერთი ან მეტი ბილიკი. ბილიკისათვის ატრიბუტები შეიძლება ნახოთ პარაგრაფში 3.6.3-ში, “ბილიკი”, გვ. 128.
საზღვარიხილული ჭეშმარიტი : ბულის BoundaryVisible True : Boolean	ეს ატრიბუტი განსაზღვრავს არის თუ არა გუბის სწორკუთხედი საზღვარი ხილული. დიაგრამაში მხოლოდ ერთ გუბეს შეიძლება ჰქონდეს ატრიბუტი დაყენებული არა ჭეშმარიტზე (set to False).
მთავარიგუბე არა ჭეშმარიტი : ბულის	ეს ატრიბუტი განსაზღვრავს არის თუ არა

MainPool False : Boolean	გუბე “მთავარი” გუბე. დიაგრამაში მხოლოდ ერთ გუბეს შეიძლება ჰქონდეს ატრიბუტი დაყენებული ჰეშმარიტზე.
---------------------------------	---

3.6.3. ბილიკი (Lane)

ბილიკი არის გუბის შიგნით ქვე-დაყოფა და აფართოებს გუბის მთლიან სიგრძეს ან ვერდიკალურად (იხ. ნახ. 3.36) ან ჰორიზონტალურად (იხ. ნახ. 3.37). თუ გუბე არის უხილავად შემოსაზღვრული, გუბესთან დაკავშირებულმა ბილიკმა უნდა გააფართოვოს გუბის მთლიანი სიგრძე. ბილიკთან დაკავშირებული ტექსტი (ანუ მისი სახელი და/ან რაიმე ატრიბუტი) შეიძლება იყოს განთავსებული ფორმის შიგნით ნებისმიერი მიმართულებით ან მდებარეობით, მოდელიორის ან მოდელირების ინსტრუმენტის გამყიდველის შეხედულების მიხედვით. მოყვანილ მაგალითებში სახელი განლაგებულია დროშასავით მარცხენა მხარეს (ჰორიზონტალური გუბეებისათვის) ან ზემოთ (წვეროში) (ვერდიკალური გუბეებისათვის) იმ ხაზის მეორე მხარეს, რომელიც გამოყოფს გუბის სახელს, თუმცა, ეს არ არის მკაცრი მოთხოვნა.

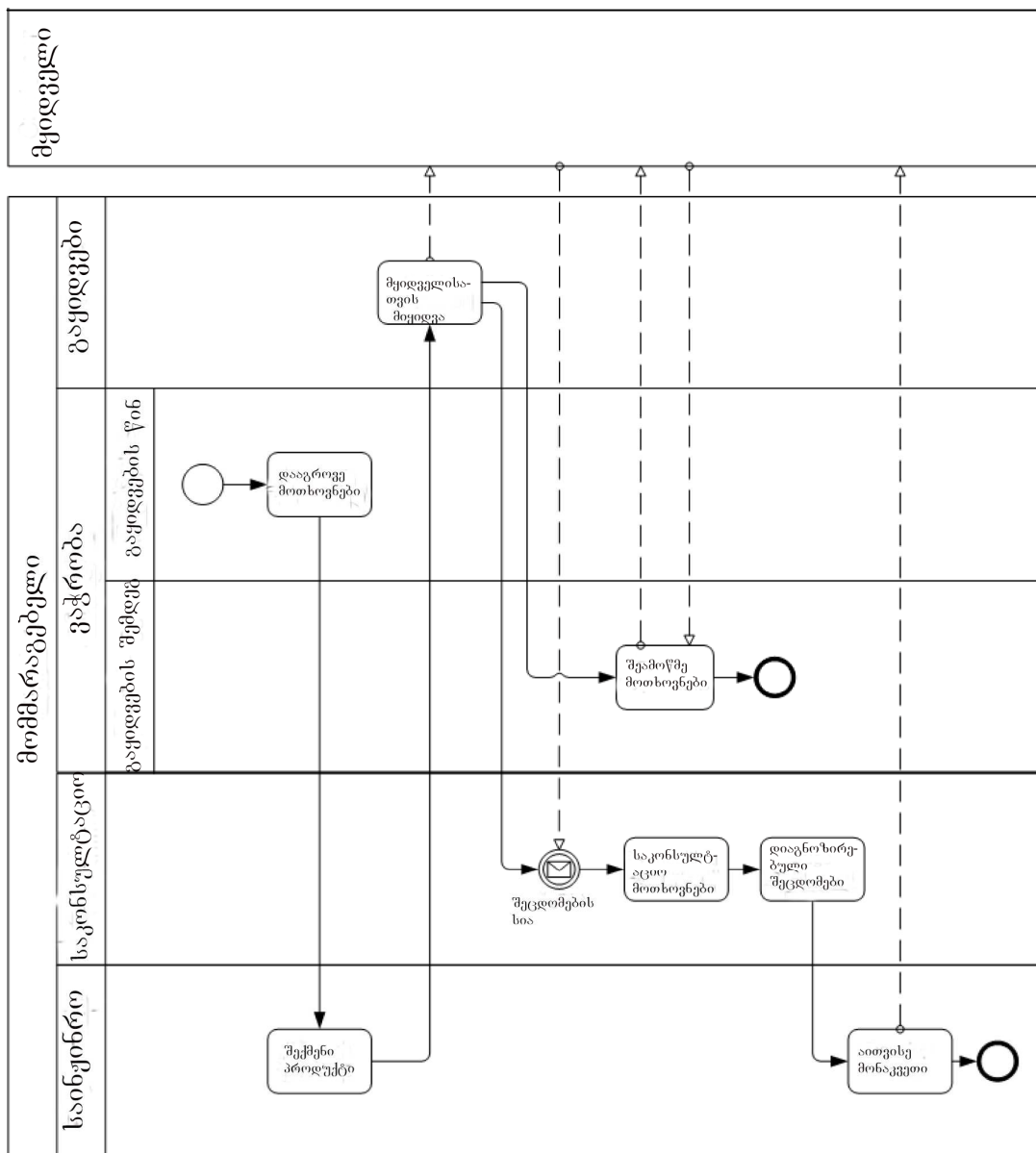
გუბე	
ბილიკი	ბილიკი

ნახ. 3.36. ორი ბილიკი ვერდიკალურ გუბეში.

გუბე ბილიკი	

ნახ. 3.37. ორი ბილიკი ჰორიზონტალურ გუბეში.

ბილიკები გამოიყენება გუბის შიგნით მოქმედებების ორგანიზებისათვის და კატგორიებად განაწილებისათვის. ბილიკების მნიშვნელობები დამოკიდებულია მოდელიორზე. *BPMN* არ აზუსტებს (აკონკრეტებს) ბილიკების გამოყენებას. ბილიკები ხშირად გამოიყენება ისეთი რამეებისათვის, როგორცაა შიდა როლები (მაგალითად, მენეჯერი, კომპანიონი (პარტნიორი)), სისტემები (მაგალითად, საწარმოს გამოყენება), შიდა დეპარტამენტი (მაგალითად, გადაზიდვა, ფინანსები), და სხვა. დამატებით ბილიკები შეიძლება იყვნენ ჩალაგებული (nested) (вложенны) (იხ. ნახ. 3.38) ან მატრიცაში განსაზღვრული. მაგალითად, შეიძლება იყოს ბილიკების გარე სიმრავლე კომპანიის დეპარტამენტებისათვის და შემდეგ ბილიკების შიდა სიმრავლე როლებისათვის ყოველი დეპარტამენტის შიგნით.



ნახ. 3.38. ჩალაგებული ბილიკების მაგალითი.

3.6.3.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია ბილიკების მაიდენტიფიცირებელი ატრიბუტები, რომლებიც აფართოებენ პროცესის მონაწილეთა საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.38).

ცხრილი 3.40. ბილიკის ატრიბუტები

ატრიბუტები	აღწერა
ბილიკები (0-*) : ბილიკი Lanes (0-*) : Lane	ეს ატრიბუტი იდენტიფიცირებას უკეთებს ნებისმიერ ბილიკებს, რომლებიც არიან ჩალაგებული მიმდინარე ბილიკის შიგნით.

3.7. არტიფაქტები (ნიმუშები) (Artifacts)

BPMN-ი მოდელიორს აძლევს შესაძლებლობას დიაგრამაზე აჩვენოს დამატებითი ინფორმაცია პროცესის შესახებ, რომელიც არ არის პირდაპირ დაკავშირებული პროცესის შესრულების ნაკადთან ან შეტყობინების ნაკადთან.

ამ ეტაპზე, BPMN-ი უზრუნველყოფს სამ სტანდარტულ არტიფაქტებს (ნიმუშებს) (Artifacts): მონაცემები, ჯგუფი და ტექსტური ანოტაცია (A Data Object, a Group, and an Annotation). დამატებითი სტანდარტული არტიფაქტები (ნიმუშები) შეიძლება დაემატოს BPMN-ის სპეციფიკაციებს შემდგომ ვერსიებში. მოდელიორმა ან მოდელირების ინსტრუმენტების შემქმნელებმა შეიძლება გააფართოონ ბიზნეს პროცესების დიაგრამა (BPD) და მას დაუმატონ არტიფაქტების (ნიმუშების) ახალი ტიპები. ნებისმიერი ახალი არტიფაქტი (ნიმუში) უნდა დაემორჩილოს შესრულების ნაკადის და შეტყობინების ნაკადის შეერთების წესებს (ჩამოთვლილია ქვემოთ). გაერთიანებები (Associations) შეიძლება იყვნენ გამოყენებული არტიფაქტების (ნიმუშების) დასაკავშირებლად ნაკადის ობიექტებთან (იხ. პარაგრაფი 4.1.4, “გაერთიანება”, გვ. 144).

3.7.1. არტიფაქტების (ნიმუშების) საერთო განსაზღვრებები

მომდევნო პარაგრაფები უზრუნველყოფენ განსაზღვრებებს, რომლებიც საერთოა ყველა არტიფაქტებისათვის (ნიმუშებისათვის).

3.7.1.1. არტიფაქტების (ნიმუშების) საერთო ატრიბუტები

შემდეგ ცხრილში მოცემულია არტიფაქტებისათვის (ნიმუშებისათვის) საერთო საიდენტიფიკაციო ატრიბუტები, რომლებიც აფართოებენ BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 3.1):

ცხრილი 3.41. არტიფაქტების (ნიმუშების) საერთო ატრიბუტები

ატრიბუტები	აღწერა
არტიფაქტიტიპი (მონაცემებისობიექტი ჯგუფი ანოტაცია) მონაცემებისობიექტი : სტრიქონი	არტიფაქტიტიპი შეიძლება იყოს დაყენებული მონაცემებისობიექტზე, ჯგუფზე ან ანოტაციაზე (DataObject, Group, or Annotation). ნიმუშის ჩამონათვალი შეიძლება იყოს გაფართოებული ახალი ტიპების ჩართვით.

3.7.1.2. არტიფაქტის (ნიმუშის) და შესრულების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.1, “შესრულების ნაკადის წესები”, გვ. 34, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შესრულების ნაკადის წყარო ან მიზნები.

- არტიფაქტი (ნიმუში) არ უნდა იყოს შესრულების ნაკადის მიზანი.
- არტიფაქტი (ნიმუში) არ უნდა იყოს შესრულების ნაკადის წყარო.

3.7.1.3. არტიფაქტის (ნიმუშის) და შეტყობინების ნაკადის შეერთებები

იხ. პარაგრაფი 2.4.2, “შეტყობინების ნაკადის წესები”, გვ. 35, ობიექტების მთელი სიმრავლისათვის და როგორ შეიძლება ისინი იყვნენ შეტყობინების ნაკადის წყარო ან მიზნები.

- არტიფაქტი (ნიმუში) არ უნდა იყოს შეტყობინების ნაკადის მიზანი.
- არტიფაქტი (ნიმუში) არ უნდა იყოს შეტყობინების ნაკადის წყარო.

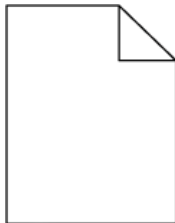
3.7.2. მონაცემების ობიექტი

BPMN-ში მონაცემების ობიექტი განხილულია როგორც არტიფაქტი (ნიმუში) და არა როგორც ნაკადის ობიექტი. ისინი განხილულია როგორც არტიფაქტი (ნიმუში) იმიტომ, რომ მათ არა აქვთ რაიმე პირდაპირი ზემოქმედება პროცესის შესრულების ნაკადზე ან შეტყობინების ნაკადზე, მაგრამ ისინი უზრუნველყოფენ ინფორმაციას იმის შესახებ თუ რას აკეთებს პროცესი. ეს არის, თუ როგორ არიან გამოყენებული და მოწესრიგებული დოკუმენტები, მონაცემები და სხვა ობიექტები პროცესის შესრულებისას. მიუხედავად ამისა, სახელი “მონაცემების ობიექტი” შეიძლება გულისხმობდეს ელექტრონულ დოკუმენტს; ისინი შეიძლება გამოყენებული იყვნენ ობიექტების ბევრი სხვადასხვა ტიპის წარმოსადგენად, როგორც ელექტრონულის, ასევე ფიზიკურის.

ზოგადად, BPMN-ი არ ახდენს ბევრი მოდელირების არტიფაქტების (ნიმუშების) სტანდარტიზაციას. ეს ძირითადად დამოკიდებულია მოდელირებისაგან და

მოდელირების ინსტრუმენტების გამყიდველებისაგან, რომლებიც ქმნიან მათ პირადი მიზნებისათვის. თუმცა, BPMN-ის მონაცემების ობიექტების ექვივალენტები გამოიყენება მართვის ტექნოლოგიური პროცესის ორიენტირებული სისტემების და ბევრი სხვა პროცესის მოდელირების მეთოდოლოგიების მიერ. ამრიგად, ეს ობიექტი საკმაოდ გამოიყენება იმისათვის, რომ სტანდარტიზირებული იყოს მისი ფორმა და საქციელი.

- მონაცემების ობიექტი არის პორტრეტ-ორიენტირებული (გამოსახულება-ორიენტირებული) სწორკუთხედი, რომელსაც აქვს ზედა მარჯვენა კუთხე გადაკეცილი და დახაზულია ერთი უწყვეტი შავი ხაზით (როგორც ნაჩვენებია ნახ. 3.39-ზე).
 - მონაცემების ობიექტისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება ხდება პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.

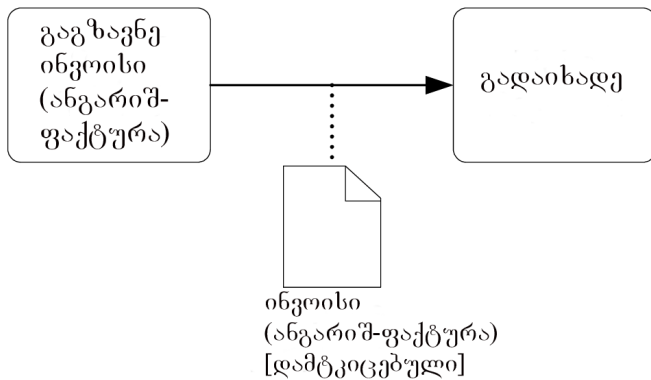


სახელი
[მდგომარეობა]

ნახ. 3.39. მონაცემების ობიექტი.

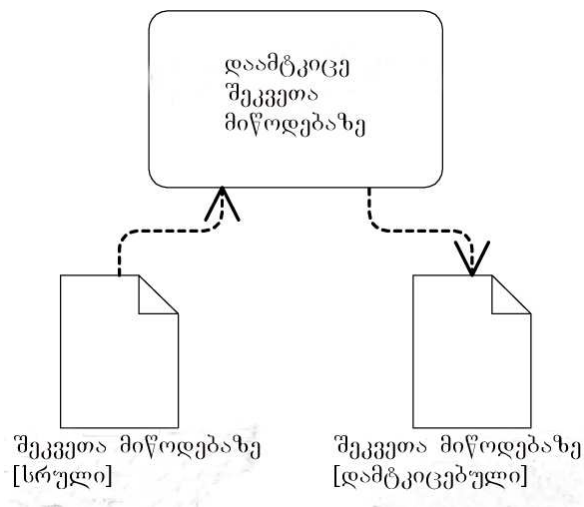
როგორც არტიფაქტი (ნიმუში), მონაცემების ობიექტები ზოგადად იქნება დაკავშირებული ნაკადის ობიექტებთან. დაკავშირება იქნება გამოყენებული მონაცემების ობიექტსა და ნაკადის ობიექტს შორის შეერთების შესაქმნელად. ეს ნიშნავს, რომ პროცესის ქცევა შეიძლება იყოს მოდელირებული მონაცემების ობიექტების გარეშე მოდელირებისათვის, რომლებსაც სურთ უწესრიგობის (ქაოსის) შემცირება. იგივე პროცესი შეიძლება იყოს მოდელირებული მონაცემების ობიექტებით მოდელირებისათვის, რომლებსაც სურთ ჩართონ მეტი ინფორმაცია პროცესის ძირითადი ქცევის შეცვლის გარეშე.

ზოგ შემთხვევებში, მონაცემების ობიექტი იქნება ნაჩვენები ერთი მოქმედებიდან მეორესაკენ გაგზავნილ შესრულების ნაკადის საშუალებით (იხ. ნახ. 3.40). მონაცემების ობიექტები იქნებიან აგრეთვე დაკავშირებული შეტყობინების ნაკადთან. ისინი არ უნდა იყვნენ არეული შეტყობინებასთან თავისთავად, მაგრამ შეიძლება იყვნენ გაგებული როგორც “სასრუბლო ტვირთი” ან რაიმე შეტყობინებების შინაარსი.



ნახ. 340. შესრულების ნაკადთან დაკავშირებული მონაცემების ობიექტი.

სხვა შენთხვევებში, იგივე მონაცემების ობიექტი იქნება ნაჩვენები როგორც პროცესის შესასვლელი (input), შემდეგ გამოსასვლელი (output) (იხ. ნახ. 341). მიმართულება დამატებული შეერთებასთან აჩვენებს მონაცემების ობიექტი არის შესასვლელი თუ გამოსასვლელი. გარდა ამისა, მონაცემების ობიექტის მდგომარეობის ატრიბუტს შეუძლია აჩვენოს პროცესის გავლენა მონაცემების ობიექტზე.



ნახ. 341. მონაცემების ობიექტები ნაჩვენები როგორც შესასვლელი და გამოსასვლელი.

3.7.2.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია ატრიბუტები მონაცემების ობიექტებისათვის, რომლებიც აფართოებენ არტიფაქტების (ნიმუშების) საერთო ატრიბუტების

სიმრავლეს (იხ. ცხრილი 3.41 და პარაგრაფი 3.7.1.2). ეს ატრიბუტები გამოიყენება მხოლოდ თუ არტიფაქტივი არის დაყენებული მონაცემების ობიექტზე:

ცხრილი 3.42. მონაცემების ობიექტის ატრიბუტები

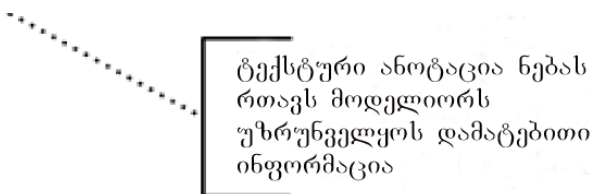
ატრიბუტები	აღწერა
სახელი : სტრიქონი Name : String	სახელი არის ატრიბუტი, რომელიც არის ობიექტის ტექსტური აღწერა.
მდგომარეობა (0-1) : სტრიქონი State (0-1) : String	მდგომარეობა (State) არის არა სავალდებულო ატრიბუტი, რომელიც მიუთითებს პროცესის გავლენზე, რომელიც მან მოახდინა მონაცემების ობიექტზე.
თვისებები (0-n) : თვისება Properties (0-n) : Property	მოდელირების მიერ განსაზღვრული თვისებები შეიძლება იყოს დამატებული მონაცემების ობიექტზე. ამ თვისებების სრული ჩანაწერი არის “<პროცესის სახელი>.<ამოცანის სახელი>.<თვისების სახელი>” (მაგალითად, “მყიდველის დამატება.საკრედიტო ოპერაციების ანგარიში.ანგარიში”).

3.7.3. ტექსტური ანოტაცია

ტექსტური ანოტაციები მოდელირებისათვის არის მექანიზმი, რომ უზრუნველყოს BPMN-ის დიაგრამის მკითხველისათვის დამატებითი ინფორმაცია.

- ტექსტური ანოტაცია არის გახსნილი სწორკუთხედი, რომელიც უნდა იყოს დახაზული ერთი უწყვეტი შავი ხაზით (როგორც ჩანს ნახ. 3.42-დან).
 - ტექსტის, ფერის, ზომის და ხაზების გამოყენება ტექსტის ანოტაციისათვის უნდა განხორციელდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.

ტექსტის ანოტაციის ობიექტი შეიძლება დაკავშირებული იყოს კონკრეტულ ობიექტთან დიაგრამაზე გაერთიანებით (with an Association) (იხ. ნახ. 3.42), მაგრამ გავლენას არ ახდენს პროცესის ნაკადზე. ანოტაციასთან დაკავშირებული ტექსტი შეიძლება განთავსებული იყოს ღია სწორკუთხედის საზღვრებს შიგნით.



ნახ. 3.42. ტექსტური ანოტაცია.

3.7.3.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია ატრიბუტები ანოტაციისათვის, რაც აფართოებს არტიფაქტის (ნიმუშის) საერთო ატრიბუტების სიმრავლეს (იხ. 3.7.1.2 და ცხრილი 3.41). ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში თუ არტიფაქტი ატრიბუტი არის დაყენებული ანოტაციაზე:

ცხრილი 3.43. ტექსტური ანოტაციის ატრიბუტები

ატრიბუტები	აღწერა
ტექსტი : სტრიქონი Text : String	ტექსტი არის ატრიბუტი, რომელიც არის ტექსტი, რომელიც მოდელიორს სურს შეატყობინოს დიაგრამის მკითხველს.

3.7.4. ჯგუფი

ჯგუფი ობიექტი არის არტიფაქტი (ნიმუში), რომელიც დიაგრამის ჯგუფის ელემენტებს არაფორმალურად უზრუნველყოფს ვიზუალური მექანიზმით. დაჯგუფება არის მიბმული (დაკავშირებული) კატეგორიის მხარდამჭერ ელემენტთან (რომელიც არის BPMN-ის ყველა ელემენტების ატრიბუტი). ამრიგად, ჯგუფი არის ერთი კატეგორიის ვიზუალური გამოსახულება. ჯგუფში გრაფიკული ელემენტები განსაზღვრავენ ჯგუფის კატეგორიას. **(შენიშვნა:** კატეგორიები შეიძლება გამოისახონ სხვა მექანიზმების საშუალებითაც, როგორც არის ფერი, იმის და მიხედვით თუ როგორ განსაზღვრავს მოდელიორი ან მოდელირების ინსტრუმენტი).

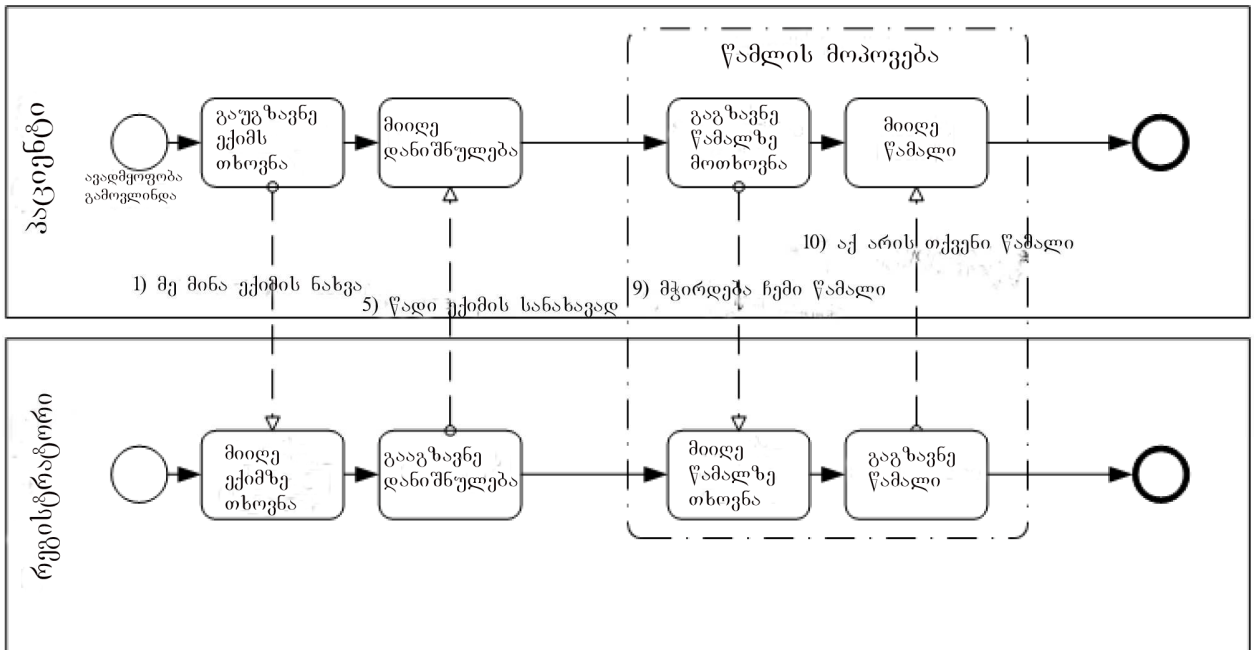
- ჯგუფი არის მომრგვალებულ კუთხეებიანი სწორკუთხედი, რომელიც უნდა იყოს დახაზული წერტილ წყვეტილი შავი ხაზით (როგორც ნაჩვენებია ნახ. 3.43-ზე).
 - ჯგუფისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.



ნახ. 3.43. ჯგუფის არტიფაქტი (ნიმუში).

როგორც არტიფაქტი (ნიმუში), ჯგუფი არ არის მოქმედება ან რაიმე ნაკადის ობიექტი და ამიტომ არ შეუძლია შეუერთდეს (მიუერთდეს) შესრულების ნაკადს ან

შეტყობინების ნაკადს. გარდა ამისა, ჯგუფებზე არ ვრცელდება გუბეების და ბილიკების შეზღუდვები. ეს ნიშნავს, რომ ჯგუფი შეიძლება გაიწელოს გუბის საზღვრებს გარეთ, რომ შემოფარგლოს დიაგრამის ელემენტები (იხ. ნახ. 3.44), ხშირად მოქმედებების იდენტიფიცირებისათვის, რომლებიც არსებობენ კორპორაციული კლიენტებისათვის განაწილებული გარიგების ფარგლებში.



ნახ. 3.44. ჯგუფი სხვადასხვა გუბეებში მოქმედებების გარშემო.

ჯგუფები ხშირად გამოიყენება დიაგრამის გარკვეული სექციების პირველ პლანზე წამოსაწევად შესრულებაზე დამატებითი შეზღუდვების დადების გარეშე, როგორც იქნებოდა ქვე-პროცესი. პირველ პლანზე წამოწეული (დაჯგუფებული) დიაგრამის სექცია შეიძლება იყოს გამოყოფილი შეტყობინებისა და ანალიზის მიზნებისათვის. ჯგუფები არ ზემოქმედებენ პროცესის ნაკადზე.

3.7.4.1. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია ატრიბუტები ჯგუფებისათვის, რომლებიც აფართოებენ არტიფაქტის (ნიმუშის) საერთო ატრიბუტების სიმრავლეს (იხ. 3.7.1.2 და ცხრილი 3.41). ეს ატრიბუტები გამოიყენება მხოლოდ იმ შემთხვევაში, თუ არტიფაქტიტიპი (ArtifactType) ატრიბუტი არის დაყენებული ჯგუფზე:

ცხრილი 3.44. ჯგუფის ატრიბუტები

ატრიბუტები	აღწერა
კატეგორიაRef : კატეგორია	კატეგორიაRef განსაზღვრავს კატეგორიას,

CategoryRef : Category	რომელსაც ჯგუფი წარმოადგენს. კატეგორიის სახელი უზრუნველყოფს ჯგუფისათვის ჭდეს. ჯგუფის საზღვრებში გრაფიკულ ელემენტებს მიენიჭებათ კატეგორია.
გრაფიკულიელემენტები (0-n) : გრაფიკული ელემენტი GraphicalElements (0-n) : Graphical Element	ატრიბუტი გრაფიკულიელემენტები იდენტიფიცირებას უკეთებს ყველა გრაფიკულ ელემენტს (მაგალითად, ხდომილებები, მოქმედებები, გასასვლელი და არტიფაქტები (ნიმუშები)), რომლებიც არიან ჯგუფის საზღვრებში.

თავი 4. ბიზნეს პროცესის დიაგრამის შემაერთებელი (დამაკავშირებელი) ობიექტები

ამ თავში განსაზღვრული არიან გრაფიკული ობიექტები, რომლებიც გამოიყენებიან ორი ობიექტის შესაერთებლად (დასაკავშირებლად) (ე.ი. დიაგრამის შემაერთებელი ხაზები) და ნაკადის მოძრაობა პროცესში (პირდაპირი მიმდევრობის სახით ან პარალელური ან ალტერნატიული ბილიკების შექმნის საშუალებით).

4.1. გრაფიკული შემაერთებელი (დამაკავშირებელი) ობიექტები

BPMN-ში რსებობს ობიექტების შეერთების (დაკავშირების) ორი გზა: *ნაკადი (ან შესრულების ან შეტყობინების)* და *გაერთიანება (Association)*. შესრულების ნაკადი და შეტყობინების ნაკადი, გარკვეული თვალსაზრისით, წარმოადგენენ მოდელში აღწერილ ბიზნეს პროცესების ორთოგონალურ ასპექტებს, თუმცა ორივე ისინი მოქმედებენ პროცესში მოქმედებების შესრულებაზე. ამის შესაბამისად, შესრულების ნაკადი ძირითადად მიედინება ერთი მიმართულებით (ან მარცხნიდან მარჯვნივ ან ზემოდან ქვემოთ) და შეტყობინების ნაკადის დინება შეადგენს 90° შესრულების ნაკადთან. ეს დაგვეხმარება ურთიერთ კავშირების გარკვევაში დიაგრამისათვის, რომელიც შეიცავს ორივეს, შესრულების ნაკადს და შეტყობინების ნაკადს. თუმცა, BPMN არ ზღუდავს ამ ურთიერთდამოკიდებულებას ნაკადის ორ ტიპს შორის. მოდელიორს შეუძლია შეაერთოს ნაკადის ყოველი ტიპი ნებისმიერი მიმართულებით დიაგრამის ნებისმიერ ადგილას.

შემდეგ სამ პარაგრაფში აღწერილი იქნება ამ ტიპის შეერთებების ფუნქციონირება BPMN-ში.

4.1.1. შემაერთებელი (დამაკავშირებელი) ობიექტების საერთო ატრიბუტები (ნიშნები) (Attributes)

შემდეგ ცხრილში მოცემულია ატრიბუტების სიმრავლე საერთო შემაერთებელი (დამაკავშირებელი) ობიექტებისათვის (შესრულების ნაკადი, შეტყობინების ნაკადი და გაერთიანება (Association)), რომელიც აფართოებს BPMN-ის ელემენტების საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 4.1).

ცხრილი 4.1. შემაერთებელი ობიექტის საერთო ატრიბუტები

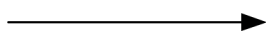
ატრიბუტები	აღწერა
სახელი (0-1) : სტრიქონი Name (0-1) : String	სახელი არის არა სავალდებულო ატრიბუტი, რომელიც არის შემაერთებელი ობიექტის ტექსტური აღწერა.
წყაროRef : გრაფიკული ელემენტი SourceRef : Graphical Element	წყაროRef არის ატრიბუტი, რომელიც იდენტიფიცირებას უკეთებს რომელი გრაფიკული

	ელემენტიდან არის გამოძვალა შამაერთე-ბელი ობიექტი. შენიშვნა: არსებობენ შეზ-ღუდვები თუ რომელ ობიექტებს შეიძლება მიუერთდნენ შესრულების ნაკადი და შეტ-ყობინების ნაკადი. ნახეთ შესრულების ნაკა-დის შეერთებების პარაგრაფი და შეტყო-ბინების ნაკადის შეერთებების პარაგრაფი ყოველი ნაკადის ობიექტისათვის, პროცესის მონაწილისათვის (Swimlane) და არტიფაქ-ტისათვის (Artifact).
მიზანიRef : გრაფიკული ელემენტი TargetRef : Graphical Element	მიზანიRef არის ატრიბუტი, რომელიც იდენ-ტიფიცირებას უკეთებს რომელი გარფიკული ელემენტისაკენ არის მიმართული შამაერთე-ბელი ობიექტი. შენიშვნა: არსებობენ შეზ-ღუდვები თუ რომელ ობიექტებს შეიძლება მიუერთდნენ შესრულების ნაკადი და შეტ-ყობინების ნაკადი. ნახეთ შესრულების ნაკადის შეერთებების პარაგრაფი და შეტ-ყობინების ნაკადის შეერთებების პარაგრაფი ყოველი ნაკადის ობიექტისათვის, პროცესის მონაწილისათვის (Swimlane) და ნიმუშისა-თვის (Artifact).

4.1.2. შესრულების ნაკადი

შესრულების ნაკადი გამოიყენება იმ მოქმედებების მიმდევრობის საჩვენებლად, რომლებიც შესრულდებიან პროცესში. ყოველ ნაკადს აქვს მხოლოდ ერთი წყარო და მხოლოდ ერთი მიზანი. *წყარო და მიზანი უნდა იყვნენ შემდეგი ნაკადის ობიექტების სიმრავლიდან: ხდომილებები (დაწყება, შუალედური და დამთავრება), მოქმედებები (ამოცანა და ქვე-პროცესი) და გასასვლელები (Gateways).* პროცესის შესრულებისას (ან მოდელირებისას), სიმბოლო დატოვებს წყარო ნაკადის ობიექტს, გადაკვეთს ქვემოთ შესრულების ნაკადს და შევა მიზან ნაკადის ობიექტში.

- შესრულების ნაკადი არის ხაზი მასიური ისრიანი თავით, რომელიც უნდა იყოს დახაზული მასიური ერთი ხაზით (როგორც ნაჩვენებია ნახ. 4.1-ზე).
- შესრულების ნაკადისათვის ტექსტის, ფერის და ზომის გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.



ნახ. 4.1. შესრულების ნაკადი.

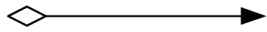
BPMN არ გამოიყენებს ტერმინს “მართვის ნაკადი” (“Control Flow”), როდესაც იყენებს ხაზებს წარმოდგენილს შესრულების ნაკადით ან შეტყობინების ნაკადით. მოქმედების დაწყება არის “მართვადი” (“controlled”) არა მარტო შესრულების ნაკადით (მოქმედებების მიმდევრობით), არამედ ასევე შეტყობინების ნაკადით (მოსული შეტყობინება), ისევე როგორც პროცესის სხვა ფაქტორებით, ისეთის როგორიც არის დაგეგმილი რესურსები. არტიფაქტები (ნიმუშები) შეიძლება იყენებენ გაერთიანებული მოქმედებებთან ამ სხვა ფაქტორებიდან ზოგიერთის საჩვენებლად. ამრიგად, გამოიყენება უფრო სპეციფიური (კონკრეტული) ტერმინი “შესრულების ნაკადი”, რადგან ეს ხაზები ძირითადად აჩვენებენ მიმდევრობას რომლითაც მოქმედებები იქნება შესრულებული.

- შესრულების ნაკადს შეიძლება ჰქონდეს პირობითი გამოსახულების ატრიბუტი, მისი წყარო ობიექტისაგან დამოკიდებულებით.

ეს ნიშნავს, რომ პირობითი გამოსახულება უნდა იყოს შეფასებული სიმბოლოს გენერირების წინ, რომელიც შემდეგ ტოვებს წყაროს ობიექტს და გადაკვეთს ნაკადს. პირობები ჩვეულებრივ დაკავშირებულია გადაწყვეტილების გასასვლელებთან, მაგრამ შეიძლება ასევე იყოს გამოყენებული მოქმედებებთან.

- თუ შესრულების ნაკადის წყარო არის მოქმედება, და არა გასასვლელი, მაშინ პირობითი მარკერი, აღნიშნული როგორც “მინი რომბი”, უნდა იყოს გამოყენებული შესრულების ნაკადის დასაწყისში (იხ. ნახ. 4.2).

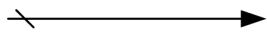
რომბის ფორმა გამოიყენება ქცევის გასასვლელთან (ასევე რომბით აღნიშნული) დასაკავშირებლად, რომელიც მართავს ნაკადს პროცესის შეგნით. უფრო მეტი ინფორმაცია იმის შესახებ, თუ როგორ გამოიყენება პირობითი შესრულების ნაკადი შეიძლება ნახოთ პარაგრაფში 4.2.1.3. “ნაკადის გახლეჩვა”, გვ. 156.



ნახ. 4.2. პირობითი შესრულების ნაკადი.

შესრულების ნაკადს, რომელსაც აქვს განსაკუთრებული მონაცემებზე დაფუძნებული გასასვლელი ან მოქმედება როგორც მისი წყარო, შეიძლება აგრეთვე იყოს განსაზღვრული გამოუცხადებელი (Default) პირობითი გამოსახულებით.

- გამოუცხადებელი მარკერი უნდა იყოს დახრილი ხაზი ხაზის დასაწყისთან ახლოს (იხ. ნახ. 4.3).



ნახ. 4.3. გამოუცხადებელი (Default) შესრულების ნაკადი.

4.1.2.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია შესრულების ნაკადის ატრიბუტების სიმრავლე, რომლებიც აფართოებენ შემაერთებული ობიექტების საერთო ატრიბუტების სიმრავლეს (იხ. ნახ. 4.43):

ცხრილი 4.2. შესრულების ნაკადის ატრიბუტები

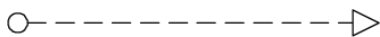
ატრიბუტები	აღწერა
<p>პირობატიპი (ცარიელი გამოსახულება გამოუცხადებელი) ცარიელი : სტრიქონი ConditionType (None Expression Default) None : String</p>	<p>შესრულების ნაკადის პირობატიპი გამოუცხადებლად არის ცარიელი. ეს ნიშნავს, რომ არ არსებობს შეფასება დაწყების დროს, რომელიც განსაზღვრავს იქნება თუ არა შესრულების ნაკადი გამოყენებული. როგორც კი სიმბოლო (Token) მზად იქნება გადაკვეთოს შესრულების ნაკადი (ე.ი. წყარო არის მოქმედება, რომელიც დამთავრდა), მაშინ სიმბოლო გააკეთებს ამას. შესრულების ნაკადის ნორმალურ, არა კონტროლირებად გამოყენებას შესრულების მოქმედებაში ექნება ცარიელი პირობატიპი (იხ. ნახ. 4.11). ცარიელი პირობატიპი არ უნდა იყოს გამოყენებული თუ შესრულების ნაკადის წყარო არის განსაკუთრებული მონაცემებზე დაფუძნებული ან შემცველი (Inclusive) გასასვლელი.</p> <p>პირობატიპი ატრიბუტი შეიძლება იყოს დაყენებული გამოსახულებაზე თუ შესრულების ნაკადის წყარო არის ამოცანა, ქვეპროცესი ან განსაკუთრებული მონაცემებზე დაფუძნებული ან შემცველი გასასვლელი. თუ პირობატიპი ატრიბუტი არის დაყენებული გამოსახულებაზე, მაშინ პირობის მარკერი შეიძლება იყოს დამატებული ხაზზე თუ შესრულების ნაკადი არის მოქმედებიდან გამომავალი (იხ. ნახ. 4.2). თუმცა, პირობის მანქანებელი არ უნდა იყოს დამატებული ხაზზე თუ შესრულების ნაკადი არის გამომავალი გასასვლელიდან.</p> <p>პირობატიპი გამოსახულება არ უნდა იყოს გამოყენებული თუ შესრულების ნაკადის წყარო არის ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელი, რთული გასასვლელი, პარალელური გასასვლელი, დაწყების ხდომილება ან შუალედური ხდომილება. გარდა ამისა, პირობატიპი გამოსა-</p>

	<p>სულეა არ უნდა იყოს გამოყენებული თუ შესრულების ნაკადი არის დაკავშირებული გასასვლელის გამოუცხადებელ კარებთან.</p> <p>პირობატიპი ატრიბუტი შეიძლება იყოს დაყენებული გამოუცხადებელზე მხოლოდ თუ შესრულების ნაკადის წყარო არის მოქმედა ან განსაკუთრებული მონაცემებზე დაფუძნებული გასასვლელი. თუ პირობატიპი არის გამოუცხადებელი, მაშინ გამოუცხადებელი მარკერი შეიძლება იყოს ნაჩვენები (იხ. ნახ. 4.3).</p>
<p>[პირობატიპი არის დაყენებული გამოსახულებაზე მხოლოდ] პირობაგამოსახულება : გამოსახულება [ConditionType is set to Expression only] ConditionExpression: Expression</p>	<p>თუ პირობატიპი ატრიბუტი არის დაყენებული გამოსახულებაზე, მაშინ პირობაგამოსახულება ატრიბუტი უნდა იყოს განსაზღვრული როგორც ნამდვილი გამოსახულება (valid expression). გამოსახულება იქნება შეფასებული შესრულების დროს. თუ შეფასების შედეგი არის ჭეშმარიტი, მაშინ სიმბოლო იქნება გენერირებული და გადაკვეთს მიმდევრობას (ყოველი შეზღუდვის არსი დადებული წყაროს მიერ, რომელიც არის გასასვლელი).</p>

4.1.3. შეტყობინების ნაკადი

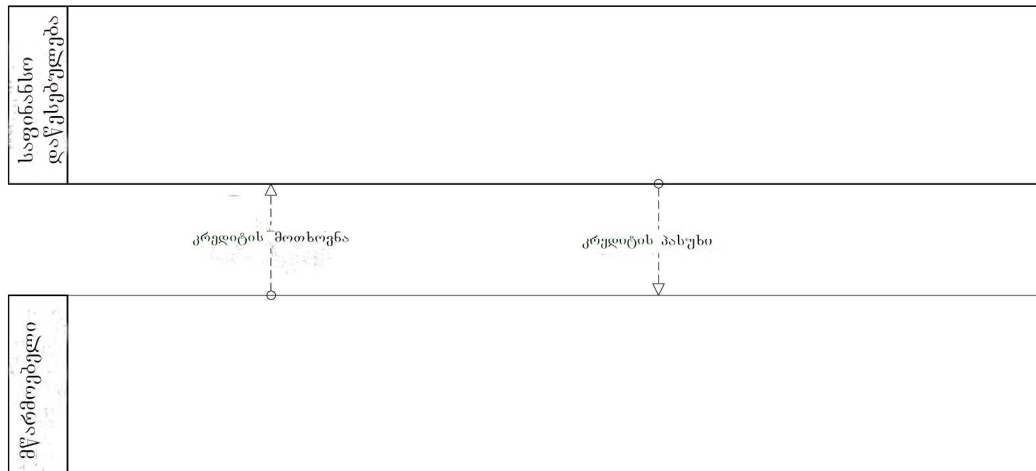
შეტყობინების ნაკადი გამოიყენება შეტყობინებების ნაკადის საჩვენებლად ორ იურიდიულ პირს შორის, რომლებიც მზად არიან გააგზავნონ ან მიიღოს ისინი. BPMN-ში ორი განცალკევებული გუბე დიაგრამაში წარმოადგენს ორ იურიდიულ პირს, ამრიგად,

- შეტყობინების ნაკადმა უნდა შეაერთოს ორი გუბე, ან დაუკავშირდნენ გუბეებს თავისთავად, ან დაუკავშირდნენ ნაკადის ობიექტებს გუბეების შიგნით. მათ არ შეუძლიათ შეაერთონ ორი ობიექტი ერთი და იგივე გუბის შიგნით.
- შეტყობინების ნაკადი არის ხაზი ცარიელი (შეუფერადებელი) ისრის თავით, რომელიც უნდა იყოს დახაზული ერთი შავი წყვეტილი ხაზით (როგორც ჩანს ნახ. 4.4-ზე).
 - შეტყობინების ნაკადისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.



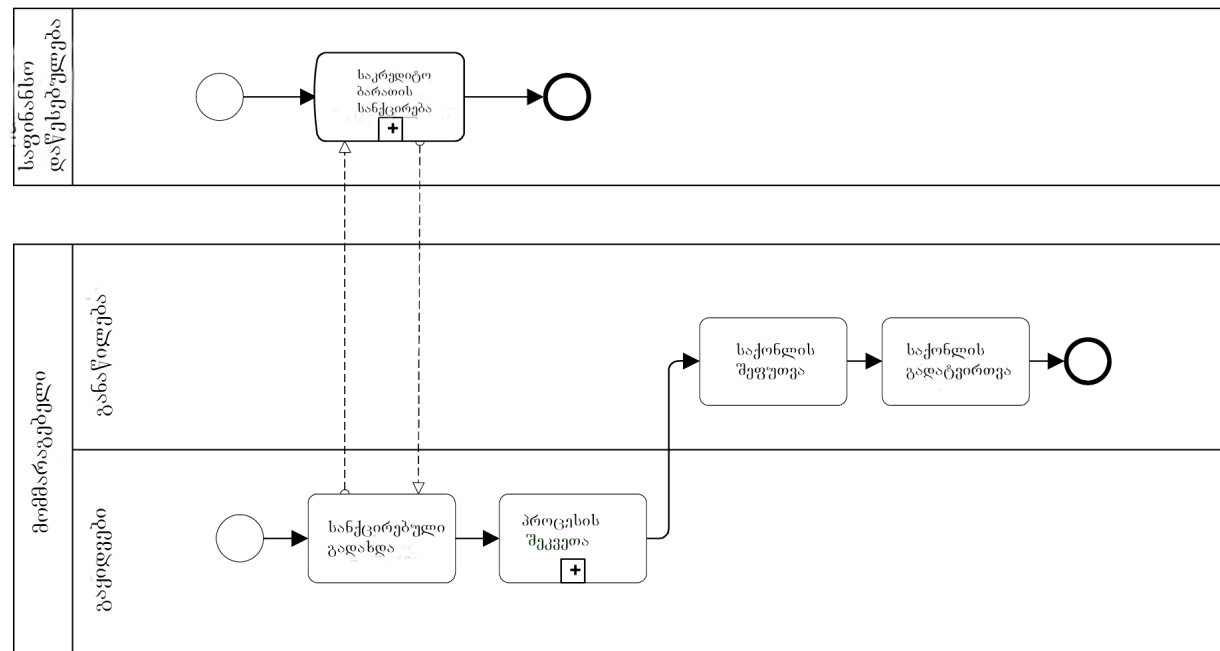
ნახ. 4.4. შეტყობინების ნაკადი.

შეტყობინების ნაკადი შეიძლება უშუალოდ მიუერთდეს გუბის საზღვარს (იხ. ნახ. 4.5), განსაკუთრებით თუ გუბეს არა აქვს პროცესის რაიმე დეტალები შიგნით (ანუ არის “შავი ყუთი”).



ნახ. 4.5. ორი გუბის საზღვრების შემაერთებელი შეტყობინების ნაკადი.

შეტყობინების ნაკადს შეუძლია ასევე გადაკვეთოს გუბის საზღვარი და შეუერთდეს ნაკადის ობიექტს გუბის შიგნით (იხ. ნახ. 4.6).



ნახ. 4.6. ორი გუბის შიგნით მდებარე ნაკადის ობიექტების დამაკავშირებელი შეტყობინების ნაკადი.

თუ გვაქვს გაფართოებული ქვე-პროცესი ერთ-ერთ გუბეში, მაშინ შეტყობინების ნაკადი შეიძლება იყოს შეერთებული (მიერთებული) ქვე-პროცესის საზღვართან ან ობიექტებთან ქვე-პროცესის შიგნით.

4.1.3.1. ატრიბუტები

შემდეგ ცხრილში ნაჩვენებია შეტყობინების ნაკადის იდენტიფიცირებული (არსებული) ატრიბუტები, რომლებიც აფართოებენ შემაერთებელი ობიექტის საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 4.1):

ცხრილი 4.3. შეტყობინების ნაკადის ატრიბუტები

ატრიბუტები	აღწერა
შეტყობინებაRef (0-1) : შეტყობინება MessageRef (0-1) : Message	შეტყობინებაRef არის არა სავალდებულო ატრიბუტი, რომელიც იდენტიფიცირებას უკეთებს გაგზავნილ შეტყობინებას.

4.1.4. გაერთიანება (Association)

გაერთიანება გამოიყენება ინფორმაციის ან არტიფაქტების (ნიმუშების) გასაერთიანებლად ნაკადის ობიექტებთან. ტექსტური და გრაფიკული არა ნაკადის ობიექტები შეიძლება იყვნენ გაერთიანებული ნაკადის ობიექტებთან და ნაკადთან. გაერთიანება აგრეთვე გამოიყენება მოქმედებების საჩვენებლად, რომლებიც გამოიყენება მოქმედების კომპენსირებისათვის. კომპენსაციის შესახებ მეტი ინფორმაცია შეიძლება ნახოთ პარაგრაფ 4.3-ში, “კომპენსაციის გაერთიანება”, გვ. 177.

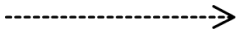
- გაერთიანების ნაკადი არის ხაზი, რომელიც უნდა იყოს დახაზული ერთი შავი პუნქტორის ხაზით (როგორც ნაჩვენებია ნახ. 4.7-ზე).
 - გაერთიანებისათვის ტექსტის, ფერის, ზომის და ხაზების გამოყენება უნდა მოხდეს პარაგრაფ 2.3-ში განსაზღვრული წესების თანახმად, “ტექსტების, ფერების, ზომების და ხაზების გამოყენება დიაგრამებში”, გვ. 33.

ნახ. 4.7. გაერთიანება.

თუ არსებობს მიზეზი, რომ გაერთიანებას მიეცეს მიმართულება, მაშინ:

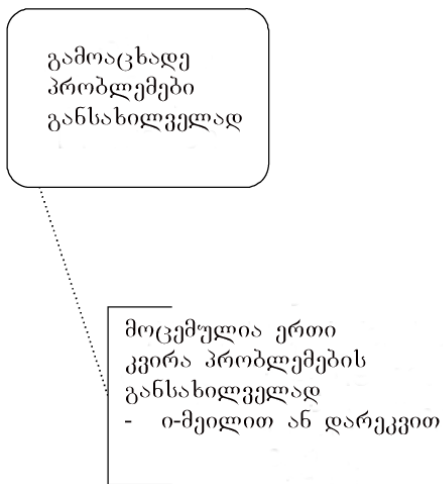
- ისარი შეიძლება იყოს დამატებული გაერთიანების ხაზს (იხ. ნახ. 4.8).

გაერთიანების მიმართულება ხშირად გამოიყენება მონაცემების ობიექტებთან იმის საჩვენებლად, რომ მონაცემების ობიექტი არის ან მოქმედებაში შემავალი (input to) ან მოქმედებიდან გამომავალი (output from).



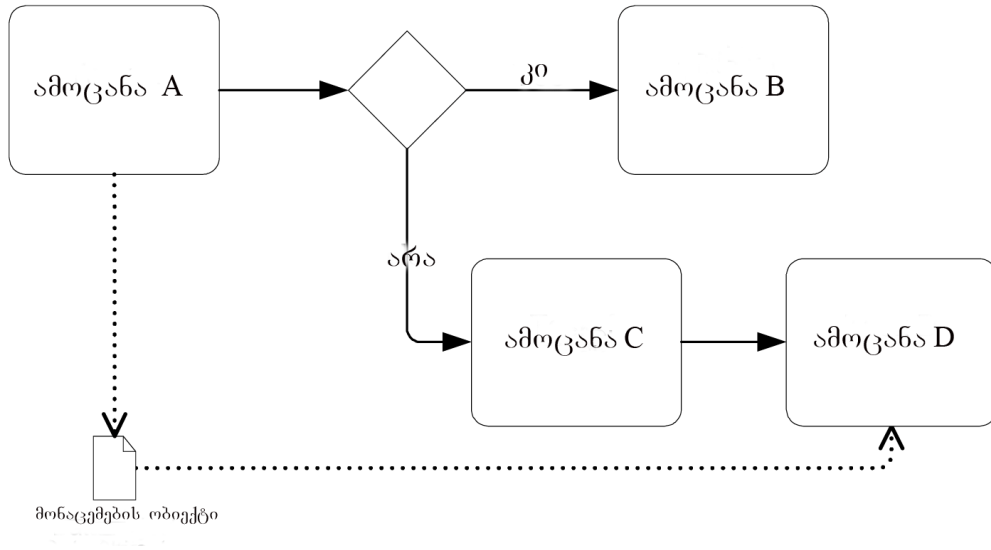
ნახ. 10.8. გაერთიანების მიმართულება.

გაერთიანება გამოიყენება მომხმარებლის მიერ განსაზღვრული ტექსტის (შენიშვნის (Annotation)) ნაკად ობიექტთან დასაკავშირებლად (შესაერთებლად, მისაერთებლად) (იხ. ნახ. 4.9).



ნახ. 4.9. ტექსტური შენიშვნის გაერთიანება.

გაერთიანება აგრეთვე გამოიყენება მონაცემები ობიექტების სხვა ობიექტებთან გასაერთიანებლად (იხ. ნახ. 4.10). მონაცემების ობიექტი გამოიყენება იმის საჩვენებლად თუ როგორ არიან დოკუმენტები გამოყენებული მთელ პროცესში. მონაცემების ობიექტზე მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.7.2, “მონაცემების ობიექტი (Data Object)”, გვ. 131.



ნახ. 4.10. მონაცემების ობიექტის ნაკადთან დამაკავშირებელი გაერთიანება.

4.1.4.1. ატრიბუტები

შემდეგ ცხრილში მოცემულია გაერთიანებების იდენტიფიცირებული (განსაზღვრული) ატრიბუტები, რომლებიც აფართოებენ დამაკავშირებელი (შემაერთებელი) ობიექტის საერთო ატრიბუტების სიმრავლეს (იხ. ცხრილი 4.1).

ცხრილი 4.4. გაერთიანების ატრიბუტები.

ატრიბუტები	აღწერა
მიმართულება (ცარიელი ერთი ორივე) ცარიელი : სტრიქონი Direction (None One Both) None : String	მიმართულება არის ატრიბუტი, რომელიც განსაზღვრავს აჩვენებს თუ არა გაერთიანება რაიმე მიმართულებას ისრით. გამოუცხადებლად არის ცარიელი (არ არის ისარი). მნიშვნელობა ერთი ნიშნავს, რომ ისარი უნდა იყოს მიზან ობიექტთან. მნიშვნელობა ორივე ნიშნავს, რომ ისარი უნდა იყოს გაერთიანების ხაზის ორივე ბოლოსთან.

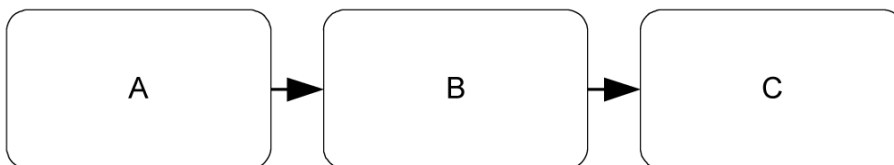
4.2. შესრულების ნაკადის მექანიზმები

შესრულების ნაკადის მექანიზმები, რომლებიც აღწერილია შემდეგ პარაგრაფებში იყოფა ოთხ ტიპად: *ნორმალური, გამონაკლისი, კავშირის ხდომილებები და სპეციალური (არა არსებობა) (Normal, Exception, Link Events, and Ad Hoc (no flow))*. ნაკადების ამ ტიპების შიგნით, BPMN შეიძლება დაკავშირებული იყოს კონკრეტულ “ტექნოლოგიური პროცესის ნიმუშებთან” (“Workflow Patterns”).

ოცდაერთი ნიმუში იყო განსაზღვრული როგორც კონკრეტული ქცევის დოკუმენტირების გზა, რომელიც შეიძლება იყოს შესრულებული ბიზნეს პროცესების მოდელირების (BPM) სისტემის მიერ. ეს ნიმუშები იცვლებიან ძალზე მარტივი ქცვიდან ძალზე რთულ ბიზნეს ქცევებამდე. ეს ნიმუშები არიან სასარგებლო იმით, რომ ისინი წარმოადგენენ ქცევის შემოწმების თვალსაჩინო ჩამონათვალს, რაც შეიძლება იყოს ჩამოთვლილი ბიზნეს პროცესების მოდელირების სისტემებისათვის. ამიტომ, ამ ნიმუშებიდან ზოგიერთი ილუსტრირებული იქნებიან შემდეგ პარაგრაფებში იმის ჩვენებით, თუ როგორ უმკლავდება BPMN-ი მარტივ და რთულ მოთხოვნებს ბიზნეს პროცესების მოდელირებისას.

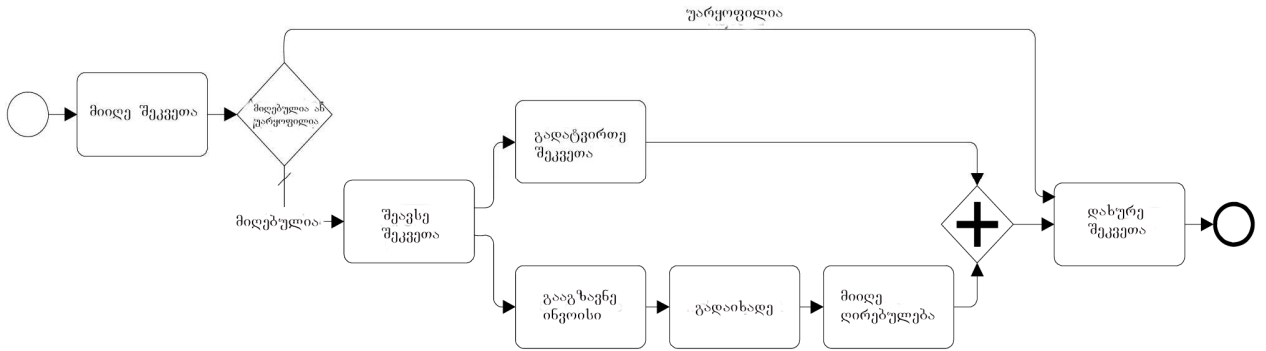
4.2.1. ნორმალური ნაკადი

ნორმალური შესრულების ნაკადი მიეკუთვნება ნაკადს, რომელიც იწყება დაწყების ხდომილებიდან და მოქმედებების საშუალებით გრძელდება ალტერნატიული და პარალელური ბილიკების გავლით მათ დამთავრებამდე დამთავრების ხდომილებასთან. პროცესის ფარგლებში ნაკადის უმარტივესი ტიპია მიმდევრობა, რომელიც განსაზღვრავს მიმდევრობის დამოკიდებულებებს მოქმედებების მწკრივისათვის, რომლებიც იქნებიან შესრულებული (მიმდევრობით). მიმდევრობა აგრეთვე არის *ტექნოლოგიური პროცესის ნიმუში №1: მიმდევრობა* (იხ. ნახ. 4.11).



ნახ. 4.11. ტექნოლოგიური პროცესის ნიმუში №1: მიმდევრობა.

როგორც ადრე იყო ნათქვამი, ნორმალური შესრულების ნაკადი უნდა იყოს მთლიანად გამოფენილი და არც ერთი ნაკადის ქცევა არ უნდა იყოს დაფარული. ეს ნიშნავს, რომ BPMN-ის დიაგრამის მიმომხილველს ექნება შესაძლებლობა მიაღწეოს თვალი ნაკადი ობიექტების და შესრულების ნაკადის მწკრივებს, დაწყებიდან პროცესის მოცემული დონის დამთავრებამდე რაიმეს გამოტოვების ან დამალული “გადახტუნების” გარეშე (იხ. ნახ. 4.12). ამ ნახაზზე, შესრულების ნაკადი დიაგრამაში აერთებს ყველა ობიექტს დაწყების ხდომილებიდან დამთავრების ხდომილებამდე. ნაჩვენები პროცესის ქცევა ასახავს შეერთებულ როგორც ნაჩვენებია და არ გადაახტება რაიმე მოქმედებებს ან არ “გადახტება” პროცესის დამთავრებასთან.



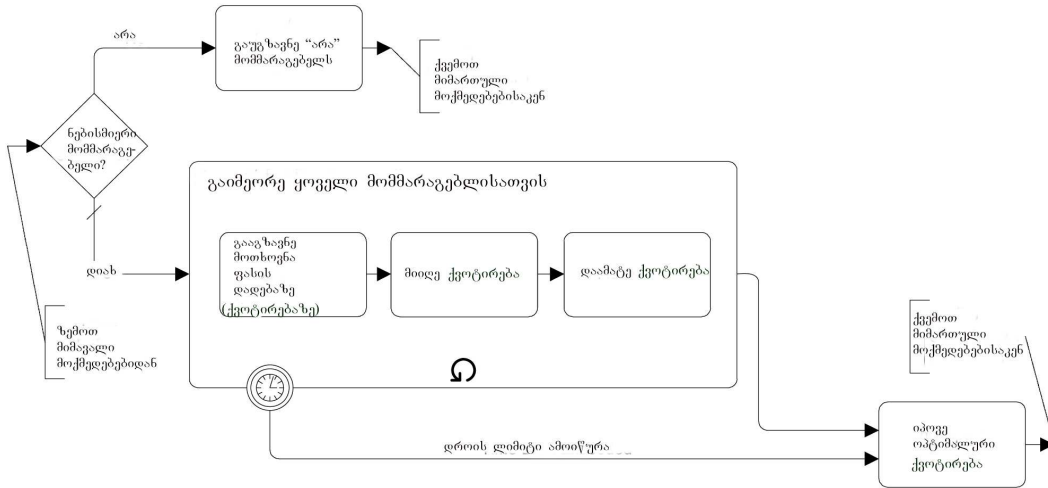
ნახ. 4.12. პროცესი ნორმალური ნაკადით.

რადგან პროცესი გრძელდება შესრულების ნაკადის მწკრივის შესაბამისად, მართვის მექანიზმებს შეუძლიათ გაყონ ან გააერთიანონ შესრულების ნაკადი, როგორც რთული ქცევის აღწერის მექანიზმი. არსებობენ მართვის მექანიზმები შესრულების ნაკადის გაყოფისათვის (გაშლა და გახლეჩვა) და გაერთიანებისათვის (გაერთიანება და შერწყმა). გასასვლელები და პირობითი შესრულების ნაკადი გამოიყენებიან ნაკადის გაყოფის და გაერთიანების დასამთავრებლად. შესაძლებელია ხარვეზების არსებობა შესრულების ნაკადში თუ გასასვლელები და/ან პირობითი შესრულების ნაკადი არ არიან ფორმირებული (კონფიგურირებული) ყველა შესაძლო შესრულების გადასაფარავად. ამ შემთხვევაში, მოდელი, რომელიც არღვევს ნაკადის ტრასირების (თვალის მიდევნების) (traceability, отслеживаемость) მოთხოვნას, იქნება განხილული როგორც ცუდი (ინვალიდი) მოდელი. საგარაუდოდ, პროცესის განვითარების პროგრამა ან ბიზნეს პროცესის მოდელირების შემოწმების გარემო მოგვცემს შესაძლებლობას შემოწმდეს პროცესის მოდელი იმაში დასარწმუნებლად, რომ მოდელი არის ვარგისი.

ამ მექანიზმების ინგლისური ტერმინების განსაზღვრებების ზედაპირული გადახედვა (მაგალითად, გაშლა (განშტოება) და გახლეჩვა (e.g., forking and splitting) გვიჩვენებს, რომ ტერმინების ყოველი წყვილი ძირითადად ერთი და იგივე მოქმედებას ნიშნავს. თუმცა, მათი შედეგი პროცესის ქცევაზე არის საკმაოდ განსხვავებული. შემდგომში, ამ სპეციფიკაციების მომდევნო რამოდენიმე პარაგრაფში, გაგრძელდება ამ ტერმინების გამოყენება, მაგრამ მოყვანილი იქნებიან სპეციალური განსაზღვრებები იმის შესახებ თუ როგორ მოქმედებენ ისინი პროცესის შესრულებაზე.

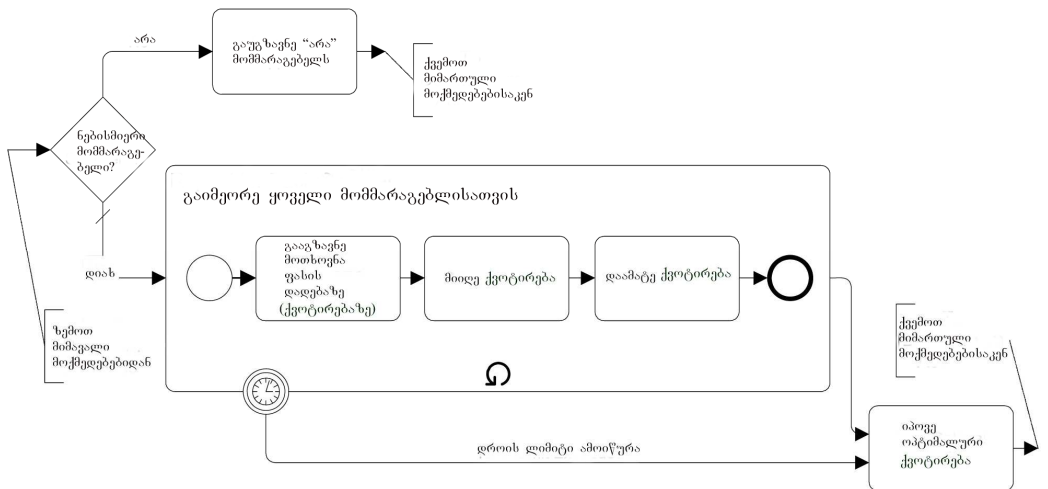
გაფართოებული ქვე-პროცესის გამოყენებას პროცესში (იხ. ნახ. 4.13), რომელიც არის პროცესის ერთი დონის ჩართვა პროცესის მეორე დონეში, შეუძლია ზოგჯერ შეუძლებელი გახადოს ნაკადისათვის თვალის მიყოლება (ტრასირება) დიაგრამაში. არ მოითხოვება, რომ ქვე-პროცესს ჰქონდეს დაწყების ხდომილება და დამთავრების ხდომილება. ეს ნიშნავს, რომ შესრულების ნაკადის მწკრივი იქნება განაწილებული ქვე-პროცესის საზღვრებიდან გაფართოებული ქვე-პროცესის შიგნით მდებარე პირველ ობიექტამდე. ნაკადი “გადახტება” გაფართოებული ქვე-პროცესის შიგნით მდებარე პირველ ობიექტამდე. როგორც ნახაზიდან ჩანს, გაფართოებული ქვე-პროცესი ხშირად იქნება გამოყენებული გამონაკლისი დამუშავების ჩასართველად (to include exception handling). მოთხოვნა, რომ მოდელირებმა ყოველთვის ჩართონ დაწყების ხდომილება და დამთავრების ხდომილება გაფართოებულ ქვე-პროცესში,

როგორც წესი, ამატებს უწესრიგობას დიაგრამაში ყოველგვარი დამატებითი სიცხადის გარეშე. ამრიგად, BPMN არ მოითხოვს დაწყების ხდომილებისა და დამთავრების ხდომილების გამოყენებას დიაგრამის (რომელიც შეიცავს მრავალ დონეს) თვალსაჩინოების (თვალის მიდევნების შესაძლებლობის) ასამაღლებლად.



ნახ. 4.13. გაფართოებული ქვე-პროცესი დაწყების ხდომილებისა და დამთავრების ხდომილების გარეშე.

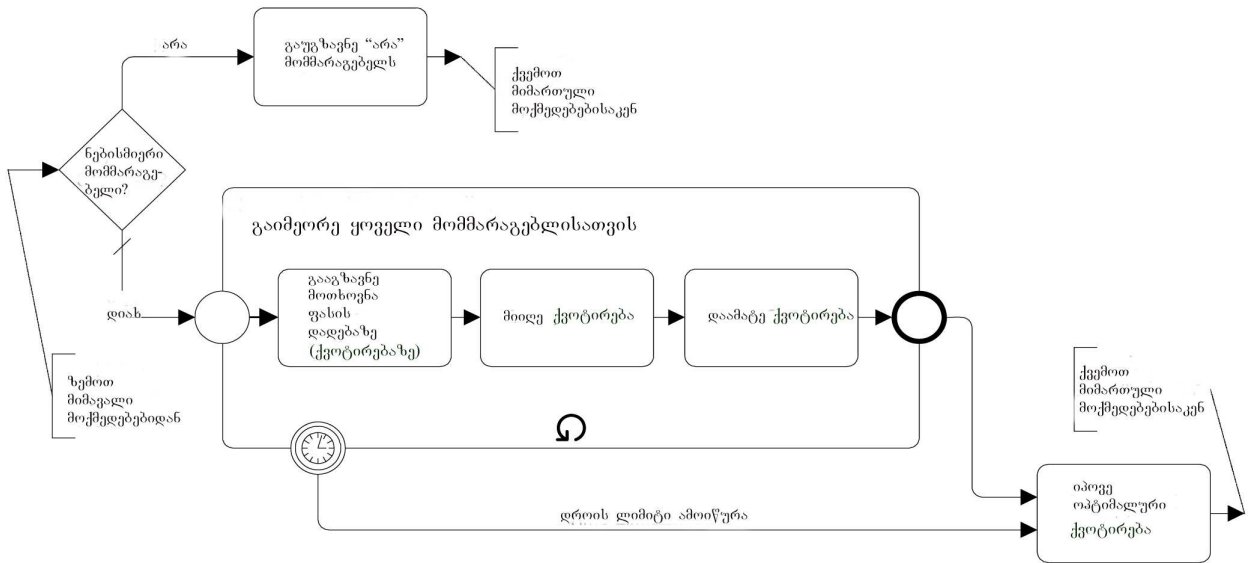
რა თქმა უნდა, დაწყების ხდომილება და დამთავრების ხდომილება გაფართოებული ქვე-პროცესისათვის შეიძლება იყოს ჩართული და განთავსებული მთლიანად მის საზღვრებში (იხ. ნახ. 4.14). ამ ტიპის მოდელი განსხვავებული ექნება აგრეთვე მთლიანად ტრასირებადი (თვალის მიდევნებადი) შესრულების ნაკადისაგან, რამდენადაც ნაკადი გრძელდება პროცესის ერთი დონიდან მეორისაკენ.



ნახ. 4.14. გაფართოებული ქვე-პროცესი შიდა დაწყების ხდომილებითა და დამთავრების ხდომილებით.

თუმცა, მოდელიორს შეიძლება უნდოდეს დიაგრამის თვალსაჩინოების გარანტირება და ამ მიზნით შეიძლება გამოიყენოს დაწყების ხდომილება და დამთავრების ხდომილება გაფართოებულ ქვე-პროცესში. ამის გაკეთების ერთი გზა იქნება ამ ხდომილებების მიღება გაფართოებული ქვე-პროცესის საზღვრებზე (იხ. ნახ. 4.15). ქვე-პროცესში შემავალი შესრულების ნაკადი შეიძლება იყოს მიდგმული უშუალოდ დაწყების ხდომილებასთან ქვე-პროცესის საზღვრის ნაცვლად. ანალოგიურად, ქვე-პროცესიდან გამომავალი შესრულების ნაკადი შეიძლება შეუერთდეს (დაუკავშირდეს) დამთავრების ხდომილებას, ქვე-პროცესის საზღვრის ნაცვლად. ამის გაკეთება საშუალებას იძლევა, რომ ნორმალური ნაკადი იყოს თვალსაჩინო მრავალ დონიანი პროცესში ყველგან.

ტექნიკურად, დაწყების და დამთავრების ხდომილებები ჯერ კიდევ იმოყოფებიან ქვე-პროცესის საზღვრებში. მოდელირების ამ ტექნიკის გამოყენება არის მხოლოდ გრაფიკული ხერხი პროცესის უფრო ზუსტი წარმოდგენისათვის (როგორც ნაჩვენებია ნახ. 4.14-ზე). ამიტომ, შესრულების ნაკადი დაკავშირებული დაწყების ხდომილებასთან და დაკავშირებული დამთავრების ხდომილებასთან არ ეწინააღმდეგება შესრულების ნაკადის შეერთების წესებს (როგორც განსაზღვრულია 3.3.2.3-ში “შესრულების ნაკადის შეერთებები”, გვ. 48 და 3.3.3.3. “შეტყობინების ნაკადის შეერთებები”-ში გვ. 49-ზე).



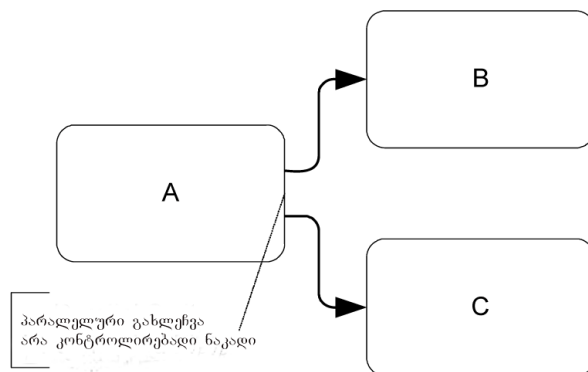
ნახ. 4.15. გაფართოებული ქვე-პროცესი საზღვარზე მიდგმული დაწყების ხდომილებითა და დამთავრების ხდომილებით.

როდესაც საქმე გვაქვს გამონაკლისებთან და კომპენსაციებთან, თვალსაჩინოების მოთხოვნა არის ასევე შერბილებული (პარაგრაფი 4.2.2, “გამონაკლისი ნაკადი”, გვ. 174 და პარაგრაფი 4.3, “კომპენსაციის გაერთიანება”, გვ. 177).

4.2.1.1. განშტოება ნაკადი (Forking Flow)

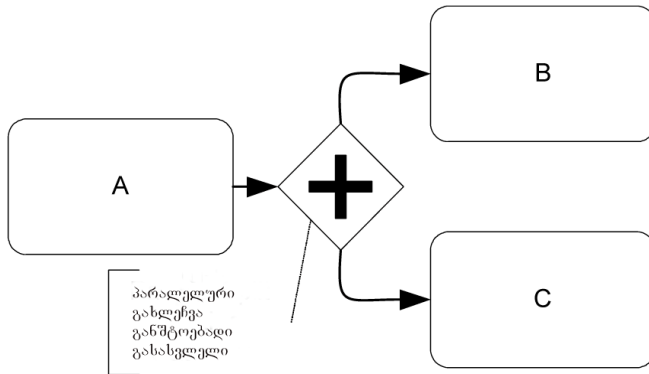
BPMN იყენებს ტერმინს განშტოება ორ ან მეტ პარალელურ ბილიკებში ბილიკის გაყოფის აღსანიშნავად (აგრეთვე ცნობილი როგორც და-გახლეჩვა (AND-Split)). ეს არის მექანიზმი, რომელიც ნებას რთავს მოქმედებებს შესრულდნენ ერთდროულად და არა მიმდევრობით. ეს აგრეთვე არის *ტექნოლოგიური პროცესის ნიმუში №2: პარალელური გახლეჩვა*. BPMN-ს აქვს *სამი კონფიგურაცია*, რომელიც უზრუნველყოფს განშტოებას.

განშტოების შექმნის *პირველი მექანიზმი არის მარტივთ: ნაკადის ობიექტს შეიძლება ჰქონდეს ორი ან მეტი გამომავალი შესრულების ნაკადი* (იხ. ნახ. 4.16). სპეციალური ნაკადის მართვის ობიექტი არ გამოიყენება ბილიკის განშტოებისათვის ამ შემთხვევაში, რადგან ის ითვლება არა კონტროლირებად ნაკადად; ეს ნიშნავს, რომ ნაკადი გააგრძელებს ქვემოთ ყოველ ბილიკს რაიმე დამოკიდებულებების ან პირობების გარეშე (არ არსებობს გასასვლელი, რომელიც აკონტროლებს ნაკადს). *განშტოებადი შესრულების ნაკადი შეიძლება იყოს წარმოშობილი ამოცანიდან, ქვე-პროცესიდან ან დაწყების ხდომილებიდან.*



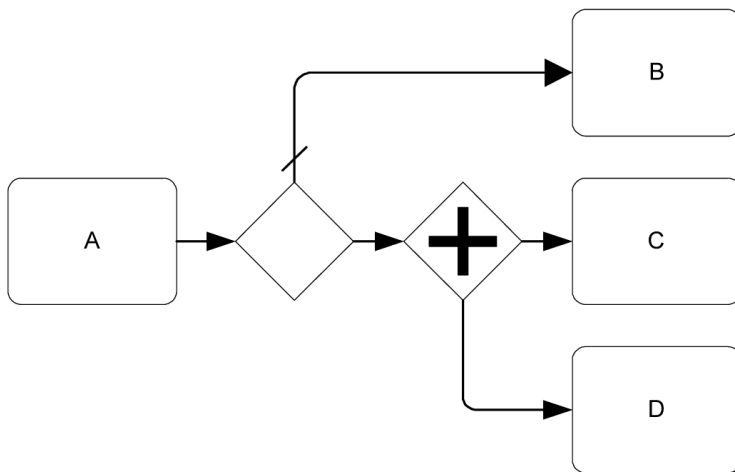
ნახ. 4.16. ტექნოლოგიური პროცესის ნიმუში №2: პარალელური გახლეჩვა – ვარიანტი 1.

მეორე მექანიზმი იყენებს პარალელურ გასასვლელს (იხ. ნახ. 4.20). ნახ. 4.17-ზე ნაჩვენები სიტუაციებისათვის გასასვლელი არ არის საჭირო, რადგან იგივე ქცევა შეიძლება იყოს შექმნილი მრავლობითი გამომავალი შესრულების ნაკადის საშუალებით, როგორც არის ნაჩვენები ნახ. 4.16-ზე. თუმცა, ზოგიერთ მოდელიორს და მოდელირების ინსტრუმენტებს შეუძლიათ გამოიყენოს განშტოებადი გასასვლელი როგორც “საუკეთესო ვარიანტი (პრაქტიკა)” (“best practice”). პარალელური გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.5, “პარალელური გასასვლელები”, გვ. 121.



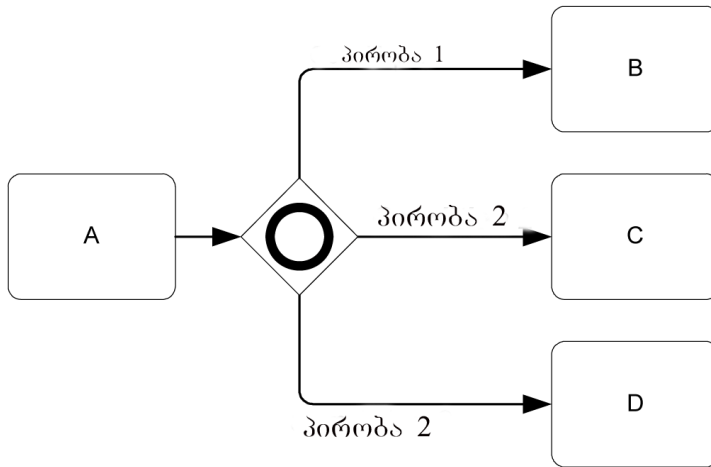
ნახ. 4.17. ტექნოლოგიური პროცესის ნიმუში №2: პარალელური გახლეჩვა – ვარიანტი 2.

მაშინაც კი, როდესაც არ არის მოთხოვნილი როგორც “საუკეთესო ვარიანტი”, არსებობენ სიტუაციები, სადაც პარალელური გასასვლელი არიან პროცესის შესრულებისათვის სასარგებლო. ნახ. 4.18-ზე ნაჩვენებია თუ როგორ გამოიყენება განშტოვადი გასასვლელი, როდესაც განსაკუთრებული გადაწყვეტილების გამოსავალი (შედეგი) მოითხოვს, რომ მრავალჯერადი მოქმედებები იქნან შესრულებული ერთ პირობაზე (კარზე) დაყრდნობით.



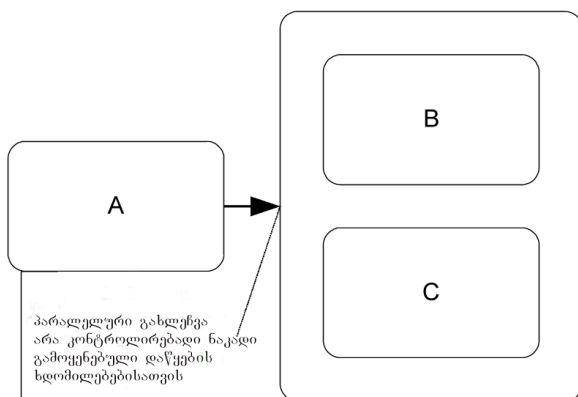
ნახ. 4.18. პარალელური ბილიკების შექმნა გასასვლელით.

მრავალჯერადი (multiple) პირობითი შესრულების ნაკადი, რომელთაგან ყოველს ზუსტად ერთი და იგივე პირობის გამოსახულება აქვს (იხ. ნახ. 10.19), შეიძლება იყოს გამოყენებული *შემცველი* გასასვლელით (with an *Inclusive Gateway*) ქცევის შესაქმნელად. ამის მაგივრად განშტოვადი გასასვლელის გამოყენება იძლევა უფრო მეტად თვალსაჩინო ქცევას (მოქმედებას).



ნახ. 4.19. პარალელური ბილიკების შექმნა ექვივალენტური პირობებით.

განშტოების მექანიზმის მესამე ვარიანტი იყენებს გაფართოებულ ქვე-პროცესს მოქმედებების სიმრავლის დასაჯგუფებლად, რომლებიც პარალელურად იქნებიან შესრულებული (იხ. ნახ. 4.20). ქვე-პროცესი არ შეიცავს დაწყებისა და დამთავრების ხდომილებებს და აჩვენებს მის ფარგლებში შესრულებულ მოქმედებებს. ამდაგვარი კონფიგურაცია შეიძლება იყოს წოდებული “პარალელურ ყუთებად” და ის შეიძლება იყოს კომპაქტური და ნაკლებად გადატვირთული საშუალება (გზა) პროცესის პარალელიზმის საჩვენებლად. ამ გზით მოდელირების შესაძლებლობა არის მიზეზი, რის გამოც დაწყებისა და დამთავრების ხდომილებები არ არიან სავალდებულო BPMN-ში.



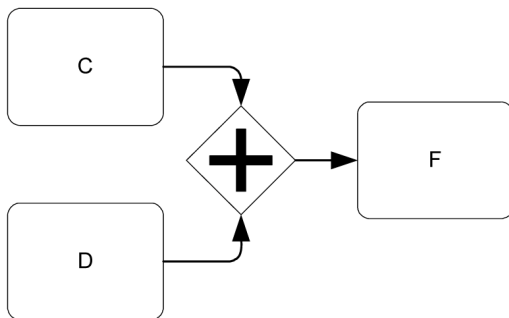
ნახ. 4.20. ტექნოლოგიური პროცესის ნიმუში №2: პარალელური გახლეჩვა – ვარიანტი 3.

უმეტესწილად, ბილიკები, რომლებიც იყენენ გაყოფილი განშტოებით, არიან შეერთებული ერთად უკანვე გაერთიანების გზით (იხ. შემდეგი პარაგრაფი) და სინქრინიზირებული ნაკადის გაგრძელებამდე. თუმცა, BPMN აქვს მეტი საშუალებები, რომ უზრუნველყოფს თანამედროვე მოდელის მოქნილობა რთულ პროცე-

სებში სიტუაციების აღწერისას. კერძოდ, ზუსტი ქცევა იქნება განსაზღვრული გამოყენებული შესრულების ნაკადის და გასასვლელების კონფიგურაციით.

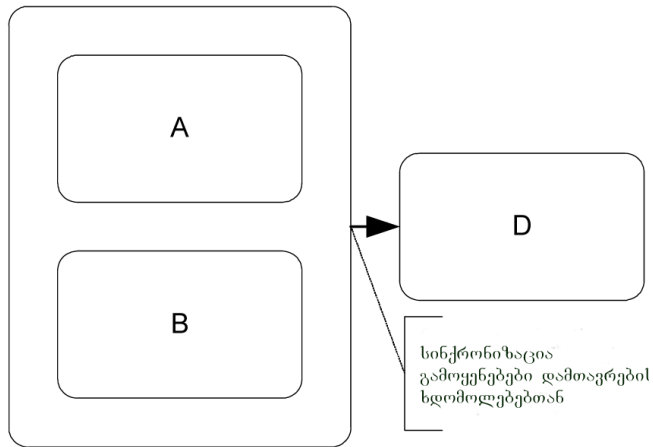
4.2.12. ნაკადის შეერთება (Joining Flow)

BPMN იყენებს ტერმინს შეერთება ორი ან მეტი პარალელური ბილიკის ერთ ბილიკში გაერთიანების აღსანიშნავად (ასევე ცნობილია როგორც და-შეერთება (AND-Join)). პარალელური გასასვლელი გამოიყენება ორი ან მეტი შემავალი შესრულების ნაკადის სინქრონიზაციისათვის (იხ. ნახ. 10.21). ზოგადად, ეს ნიშნავს, რომ განშტოებასთან წარმოშობილი სიმბოლოები წავლენ ქვემოთ პარალელური ბილიკებით და შემდეგ შეხვდებიან პარალელურ გასასვლელებთან. აქედან, მხოლოდ ერთი სიმბოლო გაგრძელდება. ეს არის ასევე ტექნოლოგიური პროცესის ნიმუში №3: სინქრონიზაცია. პარალელური გასასვლელების შესახებ უფრო მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.5, “პარალელური გასასვლელები”, გვ. 121.



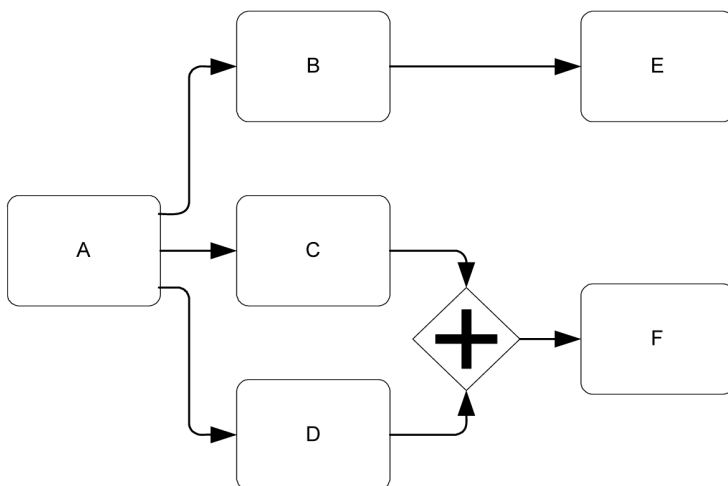
ნახ. 4.21. ტექნოლოგიური პროცესის ნიმუში №3: სინქრონიზაცია – ვარიანტი 1.

სინქრონიზაციისათვის სხვა მექანიზმია ქვე-პროცესის დამთავრება (იხ. ნახ. 4.22). თუ ქვე-პროცესში პარალელურ გასასვლელთან არსებობენ არა სინქრონიზირებული პარალელური ბილიკები, მაშინ ისინი საბოლოოდ მიაღწევენ დამთავრების ხდომილებას (მაშინაც კი თუ დამთავრების ხდომილება იქნება ნაგულისხმევი). ქვე-პროცესის გამოუცხადებელი (default) ქცევა არის ლოდინი სანამ ყველა მოქმედება მის შიგნით არ იქნება დამთავრებული, ნაკადის უკან, უმაღლესი დონის პროცესისაკენ გადაადგილდების წინ. ამრიგად, ქვე-პროცესის დამთავრება არის სინქრონიზირების ადგილი.



ნახ. 4.22. ტექნოლოგიური პროცესის ნიმუში №3: სინქრონიზაცია – ვარიანტი 2.

არ არსებობს კონკრეტული კავშირი პარალელური და განშლადი ბილიკების (რომლებიც ქმნიან პარალელურ ბილიკებს) სიმრავლეების შეერთებებს შორის. მაგალითად, მოქმედებას შეიძლება ჰქონდეს სამი გამომავალი შესრულების ნაკადი, რაც ქმნის სამი პარალელური ბილიკის გაშლას, მაგრამ ეს სამი ბილიკი არ საჭიროებენ, რომ იყვნენ შეერთებული ერთი და იგივე ობიექტთან. ნახ. 4.23-ზე ნაჩვენებია, რომ სამიდან ორი პარალელური ბილიკი შეერთებული არიან ამოცანასთან “F”. ყველა ბილიკი საბოლოო ანგარიშით იქნებიან შეერთებული, მაგრამ ეს შეიძლება მოხდეს ობიექტების რაიმე კომბინაციით, რომელიც შეიცავს განმხილველ დამთავრების ხდომილებებს. ფაქტიურად, ყოველ ბილიკს შეუძლია დამთავრდეს განცალკევებულ დამთავრების ხდომილებასთან და შემდეგ იყოს სინქრონიზირებული როგორც ზემოთ იყო ნათქვამი.



ნახ. 4.23. განშტოება-შეერთების ურთიერთ დამოკიდებულება არ არის დაფიქსირებული (დადგენილი).

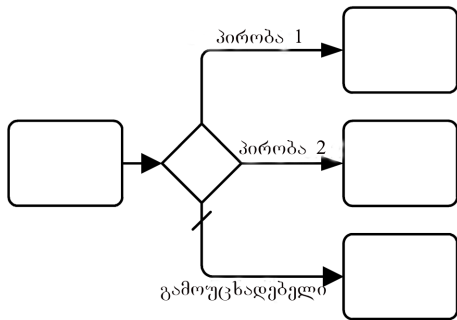
4.2.1.3. ნაკადის გახლეჩვა (Splitting Flow)

BPMN იყენებს ტერმინს გახლეჩვა ბილიკის ორ ან მეტ ალტერნატიულ ბილიკებად გაყოფის აღსანიშნავად (ასევე ცნობილია როგორც ან-გახლეჩვა (OR-Split)). ეს არის ადგილი პროცესში, სადაც კითხვა არის დასმული და პასუხი განსაზღვრავს ბილიკების რომელი სიმრავლეა აღებული. *ეს არის “გზის განშტოება”, სადაც მეზავრს, ამ შემთხვევაში სიმბოლოს, შეუძლია აირჩიოს მხოლოდ ერთი განშტოება (only one of the forks) (რომ არ აგვერიოს მოქმედება განშტოებასთან (with forking) იხ. ქვემოთ).*

ნაკადის გახლეჩვის ზოგადი კონცეფცია ძირითადად მოიხსენიება როგორც გადაწყვეტილება. ნაკადის ხაზვის ტრადიციულ მეთოდოლოგიაში, გადაწყვეტილებები წარმოდგენილი არიან რომბებით და ჩვეულებრივ არიან განსაკუთრებულები. BPMN-ი რომბს აგრეთვე იყენებს ფორმებთან ურთიერთობის გასაადვილებლად, მაგრამ აფართოებს რომბის გამოყენებას ბიზნეს პროცესის რთული ქცევის აღსაწერად (რომელიც შეუძლებელია აღიწეროს ტრადიციული ნაკადის გრაფიკებით). რომბის ფორმა გამოიყენება გასასვლელებისათვისაც და პირობითი შესრულების ნაკადის დასაწყისშიც (მოქმედების გამოსასვლელში). ამრიგად, როდესაც ბიზნეს პროცესის დიაგრამის მკითხველები ხედავენ რომბს, მათ იციან, რომ ნაკადი იქნება მართვადი რაღაც გაგებით და არა მხოლოდ გაივლის ერთი მოქმედებიდან მეორემდე. გასასვლელების შიგნით მინი-რომბის მდებარეობა და შიდა ინდიკატორები მიუთითებენ თუ როგორ იქნება ნაკადი მართული.

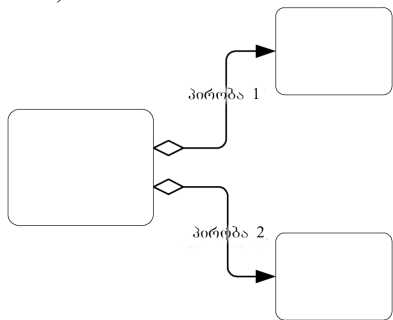
არსებობს ნაკადის გახლეჩვის მრავალი კონფიგურაცია BPMN-ში ისე, რომ შესაძლებელი იყოს რთული ქცევის სხვადასხვა ტიპების მოდელირება. ნაკადის გასახლეჩად გამოიყენება პირობითი შესრულების ნაკადი და გასასვლელების სამი ტიპი (*განსაკუთრებული, შემცველი და რთული*). პირობითი შესრულების ნაკადის დეტალებისათვის იხ. პარაგრაფი 4.1.2, “შესრულების ნაკადი”, გვ. 139. გასასვლელების დეტალებისათვის იხ. პარაგრაფი 3.5, “გასასვლელები”, გვ. 99.

პროცესის შესრულების დროს *გადაწყვეტილების მისაღებად არსებობს ორი ძირითადი მექანიზმი: პირველი არის პირობითი გამოსახულების შეფასება. არსებობს ამ მექანიზმის სამი ვარიაცია: განსაკუთრებული, შემცველი და რთული. პირველი ვარიაცია, განსაკუთრებული გადაწყვეტილება, არის იგივე რაც ტექნოლოგიური პროცესის ნიმუში №4: განსაკუთრებული არჩევანი (იხ. ნახ. 4.24). მონაცემებზე დაფუძნებული განსაკუთრებული გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. 3.5.2.1 “მონაცემებზე დაფუძნებული”, გვ. 105.*



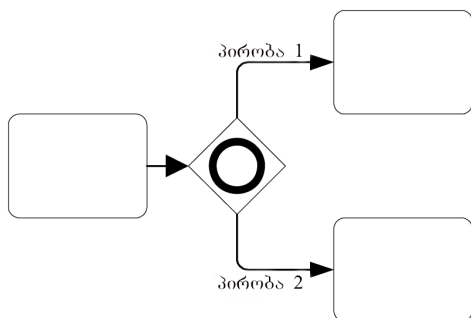
ნახ. 4.24. მონაცემებზე დაფუძნებული გადაწყვეტილების მაგალითი – ტექნოლოგიური პროცესის ნიმუში №4 -- განსაკუთრებული არჩევანი.

გამოსახულების შეფასების მეორე ტიპია შემცველი გადაწყვეტილება, რომელიც ასევე არის ტექნოლოგიური პროცესის ნიმუში №6: მრავალჯერადი არჩევანი. შემცველი გადაწყვეტილების ორი კონფიგურაცია არსებობს. შემცველი გადაწყვეტილების პირველი ტიპი იყენებს პირობით შესრულების ნაკადს მოქმედებიდან (იხ. ნახ. 4.25).



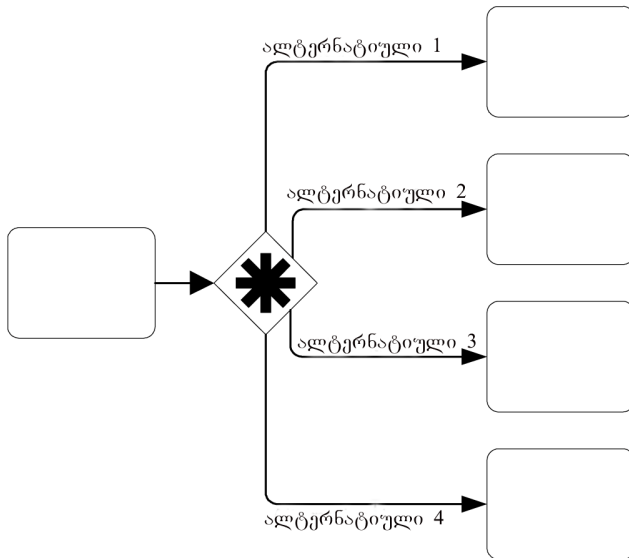
ნახ. 4.25. ტექნოლოგიური პროცესის ნიმუში №6: მრავალჯერადი არჩევანი – ვარიანტი 1.

შემცველი გადაწყვეტილებების მეორე ტიპი ნაკადის მართვისათვის იყენებს შემცველ გასასვლელს (იხ. ნახ. 4.26). შემცველი გადაწყვეტილებების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.3, “შემცველი გასასვლელები”, გვ. 114.



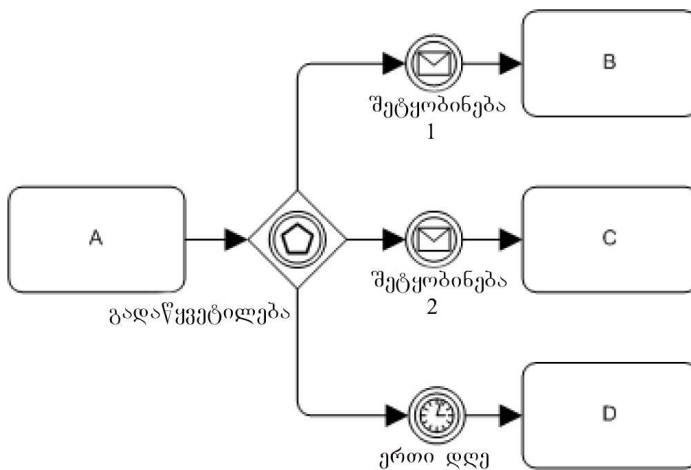
ნახ. 4.26. ტექნოლოგიური პროცესის ნიმუში №6: მრავალჯერადი არჩევანი – ვარიანტი 2.

გამოსასულების შეფასების მესამე ტიპი არის რთული გადაწყვეტილება (იხ. ნახ. 4.27). რთული გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.4, “რთული გასასვლელები”, გვ. 118.



ნახ. 4.27. რთული გადაწყვეტილება (გასასვლელი).

გადაწყვეტილების მისაღებად მეორე მექანიზმი არის სპეციფიური ხდომილების წარმოშობა, ისეთის როგორცაა შეტყობინების მიღება (იხ. ნახ. 4.28). ხდომილებაზე დაფუძნებული განსაკუთრებული გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.2.4, “ხდომილებაზე დაფუძნებული”, გვ. 110.



ნახ. 4.28. ხდომილებაზე დაფუძნებული გადაწყვეტილების მაგალითი.

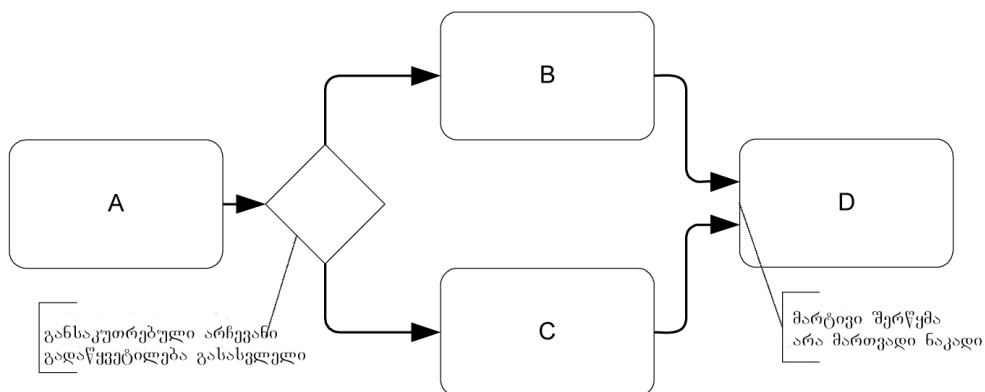
4.2.14. ნაკადის შერწყმა (Merging Flow)

BPMN იყენებს ტერმინს შერწყმა ორი ან მეტი ალტერნატიული ბილიკის ერთ ბილიკში გაერთიანების აღსანიშნავად (ასევე ცნობილია როგორც ან-შეერთება (OR-Join)). ის არის ადგილი პროცესში, სადაც ორი ან მეტი ალტერნატიული ბილიკები იწყებენ მოქმედებების გადაკვეთას, რომლებიც საერთოა ყოველი ბილიკისათვის. თეორიულად, თითოეული ალტერნატიული ბილიკი შეიძლება მოდელირებული იყოს დამთავრებისგან (დამთავრების ხდომილებისაგან) დამოუკიდებლად. თუმცა, შერწყმა საშუალებას აძლევს ბილიკებს გადაფაროს და გვერდი აუაროს იმ მოქმედებების დუბლირებას, რომლებიც საერთოა ცალკეული ბილიკისათვის. პროცესის მოცემული მაგალითისათვის, სიმბოლო შეიძლება ფაქტიურად დაინახოს მხოლოდ მოქმედებების მიმდევრობა, რომლებიც არსებობენ ერთ-ერთ ბილიკთაგანში, ისე თითქოს ის იყოს მოდელირებული დამოუკიდებლად (separately) დამთავრებისაგან.

ისევე როგორც არსებობს მრავალი გზა, რომლითაც შესრულების ნაკადი შეიძლება იყოს განშტოებული და გახლეჩილი, ასევე არსებობს შესრულების ნაკადის შერწყმის მრავალი შესაძლებლობა. *არსებობს ხუთი სხვადასხვა ტექნოლოგიური პროცესის ნიმუში, რომლებიც შეიძლება იყვნენ დემონსტრირებული (ნახვენები) შერწყმების გამოყენებით.*

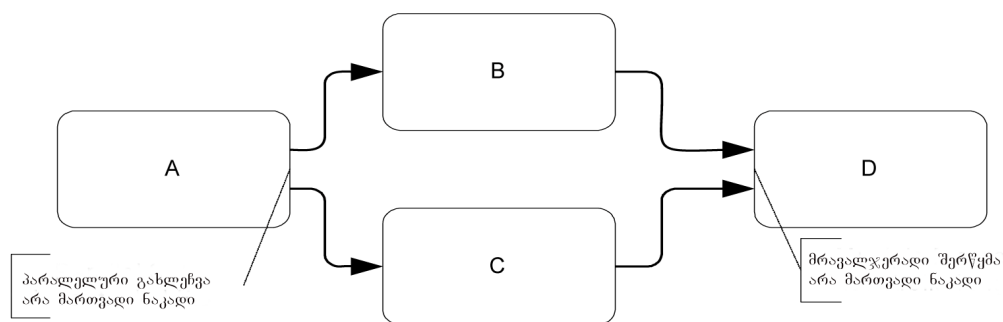
პირველი ტექნოლოგიური პროცესის ნიმუში, მარტივი შერწყმა, ალტერნატიული ბილიკების შერწყმის გრაფიკული მექანიზმი, არის მარტივი: აქ არის ორი ან მეტი შემავალი შესრულების ნაკადი ნაკადის ობიექტისაგან (იხ. ნახ. 4.29). ზოგადად, ეს ნიშნავს, რომ სიმბოლო იმოგზაურებს ქვემოთ ერთ-ერთი ალტერნატიული ბილიკით (მოცემული პროცესისათვის მაგალითად) და გაგრძელდება იქიდან. *BPMN-ს გააჩნია მარტივი შერწყმის ორი ვარიანტი.*

პირველი ვარიანტი ნახვენებია ნახ. 4.29-ზე. ორი შემავალი შესრულების ნაკადი მოქმედება “D”-თვის არიან არა მართვადი (არა კონტროლირებადი). რადგან ორი შესრულების ნაკადი არიან ორი ალტერნატიული ბილიკის ბოლოს, რომლებიც შექმნილი არიან ზემოთ მდებარე განსაკუთრებული გასასვლელით, მხოლოდ ერთი სიმბოლო მიაღწევს მოქმედება “D”-ს პროცესის ნებისმიერი მოცემული მაგალითისათვის.



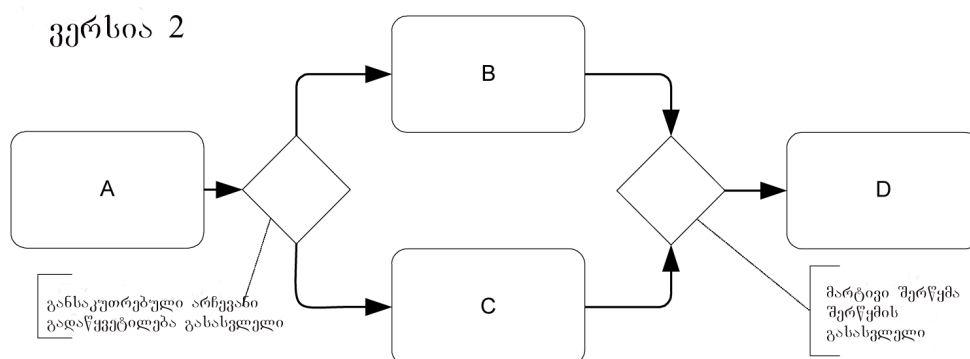
ნახ. 4.29. ტექნოლოგიური პროცესის ნიმუში №5: მარტივი შერწყმა – ვარიანტი 1.

თუ მრავალჯერადი შემავალი შესრულების ნაკადი არის ნამდვილად პარალელური და არა ალტერნატიული, მაშინ დამთავრების შედეგი იქნება განხვავებული, მაშინაც კი როდესაც შერწყმის კონფიგურაცია არის ნახ. 4.29-ზე ნაჩვენების ანალოგიური. ნახ. 4.30-ზე დინების მიმართულებით ზემოთ ქცევა არის პარალელური. ამრიგად, აქ იქნება მოქმედება “D”-თან მომავალი (სხვადასხვა დროს) ორი სიმბოლო. რადგან ნაკადი არის არა მართვადი (არა კონტროლირებადი), ყოველი სიმბოლო მომავალი “D” მოქმედებასთან გამოიწვევს ამ მოქმედების ახალ მაგალითს (მოთხოვნას). ეს არის მნიშვნელოვანი ცნება, რომელიც BPMN-ს მოდელირებას უნდა ესმოდეთ. გარდა ამისა, ამ ტიპის შერწყმა არის მრავლობითი (მრავალჯერადი) შერწყმის ტექნოლოგიური პროცესის ნიმუში.



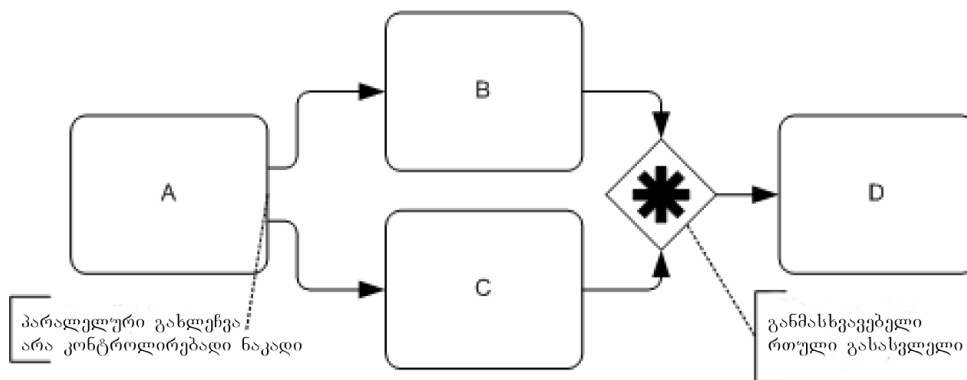
ნახ. 4.30. ტექნოლოგიური პროცესის ნიმუში №7: მრავლობითი (მრავალჯერადი) შერწყმა.

მარტივი შერწყმის მეორე ვარიანტი ნაჩვენებია ნახ. 4.31-ზე. ორი შემავალი შესრულების ნაკადის მოქმედება “D”-თვის არის მართვადი განსაკუთრებული გასასვლელით. რადგან ორი შესრულების ნაკადი არიან ორი ალტერნატიული ბილიკის ბოლოს, შექმნილი დინების მიმართულებით ზემოთ მდებარე განსაკუთრებული გასასვლელით, მხოლოდ ერთი სიმბოლო მიადწევს გასასვლელს პროცესის ნებისმიერი მოცემული მაგალითისათვის. შემდეგ სიმბოლო დაუყოვნებლივ გააგრძელებს მოქმედება “D”-ს.



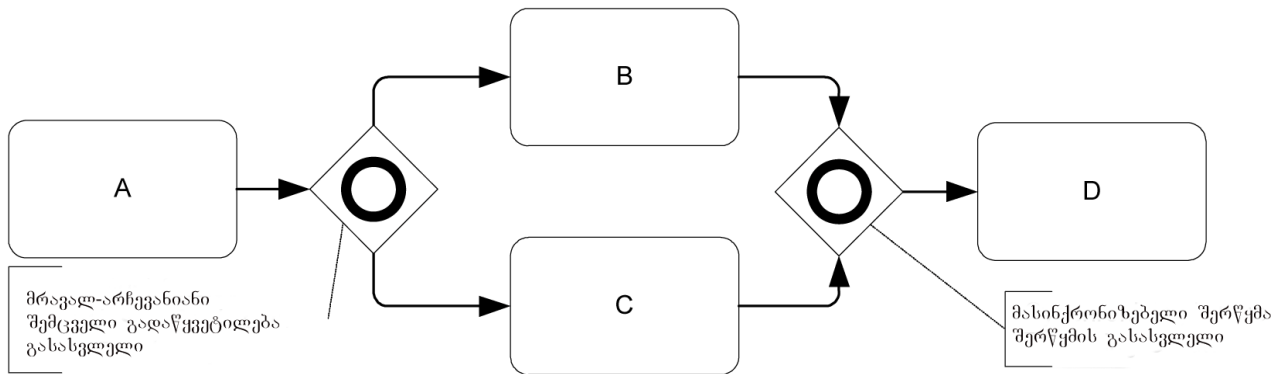
ნახ. 10.31. ტექნოლოგიური პროცესის ნიმუში №5: მარტივი შერწყმა – ვარიანტი 2.

შერწყმის სხვა ვარიანტი არის ტექნოლოგიური პროცესის ნიმუში დისკრიმინატორი (განმასხვავებელი). ამ სიტუაციაში, მრავალჯერადი შემავალი შესრულების ნაკადი არის პარალელური და არა ალტერნატიული (იხ. ნახ. 4.32). ამრიგად, აქ იქნება ორი სიმბოლო მოსული (სხვადასხვა დროს) რთულ გასასვლელთან, რომელიც წინ უსწრებს მოქმედება “D”-ს. დისკრიმინატორის (განმასხვავებელის) ნიმუშის დასაკმაყოფილებლად, რთულმა გასასვლელმა უნდა მიიღოს პირველი სიმბოლო და დაუყოვნებლად გაატაროს (გადასცეს) ის მოქმედებისაკენ. როდესაც მეორე სიმბოლო მოვა, ის იქნება გამორიცხული (исключен) ნაკადის ნარჩენისაგან. ეს ნიშნავს, რომ სიმბოლო არ იქნება გადაცემული მოქმედებაზე, ის იქნება შთანთქმული.



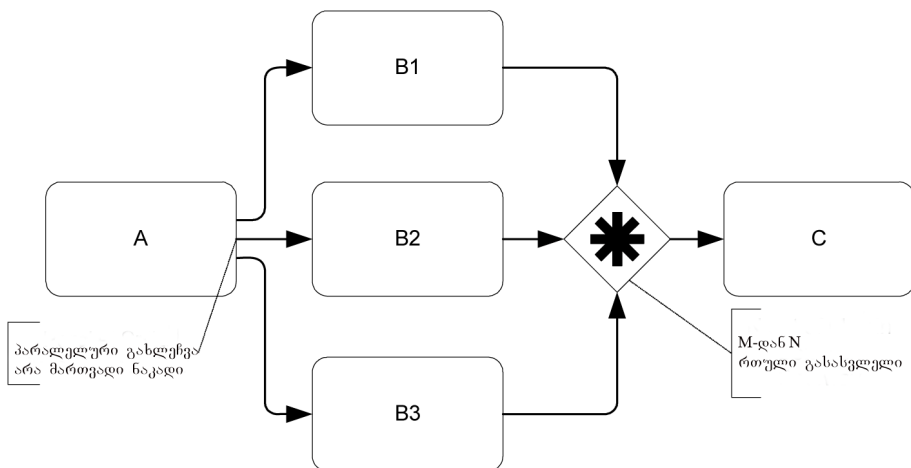
ნახ. 4.32. ტექნოლოგიური პროცესის ნიმუში №8: დისკრიმინატორი (განმასხვავებელი).

ტექნოლოგიური პროცესის ნიმუშის შერწყმის მესამე ტიპს ეწოდება მასინქრონიზირებელი შეერთება (Synchronizing Join). ეს არის სიტუაცია, როდესაც შერწყმის ადგილას წინასწარ უცნობია რამდენი სიმბოლო მოვა გასასვლელთან. ზოგიერთ პროცესებში, მაგალითად, აქ შეიძლება იყოს მხოლოდ ერთი სიმბოლო. სხვა პროცესებში, მაგალითად, აქ შეიძლება მოვიდეს ერთ სიმბოლოზე მეტი. ამ ტიპის სიტუაცია იქმნება, როდესაც შემცველი გადაწყვეტილება შექმნილია დინების მიმართულებით ზევით (იხ. ნახ. 4.33). ამ სიტუაციებთან გასამკლავებლად, შემცველი გასასვლელი შეიძლება იყოს გამოყენებული შესაბამისი რაოდენობის სიმბოლოების შესარწყმელად პროცესის ყოველი შემთხვევისათვის. გასასვლელი, რომელიც მოჰყვება მასინქრონიზირებელი შეერთების ნიმუშს, დაელოდება ყველა მოსალოდნელ სიმბოლოს სანამ ნაკადი გაგრძელდება შემდეგ მოქმედებამდე. შემცველი გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.3, “შემცველი გასასვლელები”, გვ. 114.



ნახ. 4.33. ტექნოლოგიური პროცესის ნიმუში №9: მასინქრონიზირებელი შეერთება (Synchronizing Join).

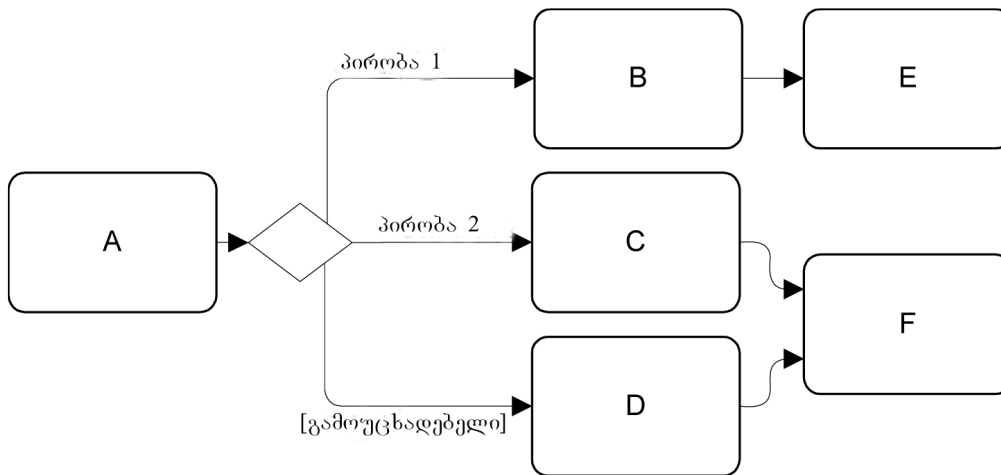
ტექნოლოგიური პროცესის ნიმუშის შერწყმის მეოთხე ტიპს ეწოდება *M* შეერთებიდან *N*-ის გამოირიცხვა (*M*-დან *N*) (*an N out of M Join*). ამ ტიპის სიტუაცია უფრო რთულია და შეიძლება განხორციელდეს რთული გასასვლელის საშუალებით (იხ. ნახ. 4.34). გასასვლელი მიიღებს სიმბოლოებს მისი შემავალი შესრულების ნაკადიდან და შეაფასებს გამოსასვლელს, რომ განსაზღვროს უნდა გაგრძელდეს თუ არა პროცესი. თუ პირობა იქნება დაკმაყოფილებული და დამატებითი სიმბოლოები მოვლენ, ისინი იქნებიან გამორიცხული (ძალზე გავს დისკრიმინატორის ნიმუშს ნახ. 4.32-დან). რთული გასასვლელების შესახებ მეტი ინფორმაციისათვის იხ. პარაგრაფი 3.5.4, “რთული გასასვლელები”, გვ. 118.



ნახ. 4.34. ტექნოლოგიური პროცესის ნიმუში №8: M-დან N.

ბილიკების სიმრავლის შერწყმას და გახლეჩვას შორის, რომლებიც წარმოიქმნებიან გასასვლელის ობიექტების საშუალებით, არ არსებობს განსაზღვრული (განსაკუთრებული) კორელაცია. მაგალითად, გადაწყვეტილებას შეუძლია გახლიჩოს ბილიკი სამ განცალკევებულ ბილიკად, მაგრამ ეს სამი ბილიკი არ საჭიროებს იყოს შერწყმული ერთი და იგივე ობიექტთან. ნახ. 4.35

გვიჩვენებს, რომ სამი ალტერნატიული ბილიკიდან ორი არის შერწყმული ამოცანა “F”-თან. ყველა ბილიკი საბოლოოდ იქნება შერწყმული, მაგრამ ეს შეიძლება მოხდეს ობიექტების ნებისმიერი კომბინაციით, მათ შორის განცალკევებული დამთავრების ხდომილებებითაც. ფაქტიურად, ყოველი ბილიკი შეიძლება დამთავრდეს განცალკევებული დამთავრების ხდომილებით.



ნახ. 4.35. გახლენვა-შერწყმის ურთიერთდამოკიდებულება არ არის დაფიქსირებული (დადგენილი).

4.2.15. გადაგრეხვა (მარყუჟის გაკეთება) (Looping) (перекручивание)

BPMN-ი უზრუნველყოფს პროცესის შიგნით გადაგრეხვის (მარყუჟის გაკეთების) ორ მექანიზმს. პირველი მდგომარეობს მოქმედების ატრიბუტის გამოყენებაში მარყუჟის განსაზღვრისათვის. მეორე მდგომარეობს შესრულების ნაკადის შეერთებაში “დინების მიმართულებით ზემოთ” მდებარე ობიექტებთან.

4.2.16. მოქმედება გადაგრეხვა (მარყუჟის გაკეთება)

ამოცანების და ქვე-პროცესების ატრიბუტები განსაზღვრავენ არიან თუ არა ისინი გამეორებული როგორც მარყუჟი. არსებობს მარყუჟების ორი ტიპი, რომლებიც შეიძლება იყვნენ განსაზღვრული: სტანდარტული და მრავალ-შემთხვევიანი (მრავალ-მაგალითიანი).

სტანდარტული მარყუჟებისათვის:

- თუ მარყუჟის პირობა არის შეფასებული მოქმედების წინ, ეს არის ზოგადად განხილული როგორც “სანამ” (“while”) მარყუჟი. ეს ნიშნავს, რომ მოქმედებები იქნებიან გამეორებული მანამ, სანამ პირობა არის ჭეშმარიტი. მოქმედებები შეიძლება არ იყვნენ შესრულებული საერთოდ (თუ პირობა არის არა ჭეშმარიტი თავიდან) ან შესრულებული მრავალჯერ.

- თუ მარყუჟის პირობა არის შეფასებული მოქმედების შემდეგ, ეს არის ზოგადად განხილული როგორც “მდე” (“until”) მარყუჟი. ეს ნიშნავს, რომ მოქმედებები იქნებიან გამეორებული სანამ პირობა არ გახდება ჭეშმარიტი. მოქმედებები შესრულდებიან სულ მცირე ერთხელ, მაგრამ შეიძლება იყვნენ შესრულებული მრავალჯერ.

მრავალ-შემთხვევიანი (Multi-Instance (MI)) მარყუჟები:

- თუ MI-მოწესრიგება (MI-მრავალ-შემთხვევიანი) არის მიმდევრობითი, მაშინ ის იქცევა “სანამ” (“while”) მარყუჟის ნაირად იტერაციების დადგენილი რიცხვით, რომელსაც მარყუჟი შეასრულებს. ეს ხშირად გამოიყენება პროცესებში, სადაც დიაგრამის ელემენტის განსაზღვრულ ტიპს ექნება ქვე-ელემენტების ან წრფე ელემენტების დადგენილი რაოდენობა. მრავალ-შემთხვევიანი მარყუჟი იქნება გამოყენებული ყოველი წრფე ელემენტის დასამუშავებლად.
- თუ MI-მოწესრიგება არის პარალელური, ეს არის ზოგადად განხილული როგორც მოქმედებების მრავლობითი მაგალითები. ამ ტიპის განსაკუთრებულობის მაგალითი იქნება გამოყენებული წიგნის დაწერის პროცესში, სადაც იქნება თავის დაწერის ქვე-პროცესი. აქ იქნება ქვე-პროცესის იმდენი კოპიები ან შემთხვევები, რამდენი თავიც არის წიგნში. ყველა შემთხვევები შეიძლება დაიწყოს ერთი და იგივე დროს.

იმ მოქმედებებს, რომლებიც არიან გამეორებული (დამარყუჟებული) ექნებათ მარყუჟის მარკერი განთავსებული მოქმედების ფორმის ქვედა ცენტრში (იხ. ნახ. 4.36). იმ მოქმედებებს, რომლებიც არიან პარალელური მრავალ-შემთხვევიანი, ექნებათ პარალელურის მარკერი განთავსებული მოქმედების ფორმის ქვედა ცენტრში (იხ. ნახ. 4.37).

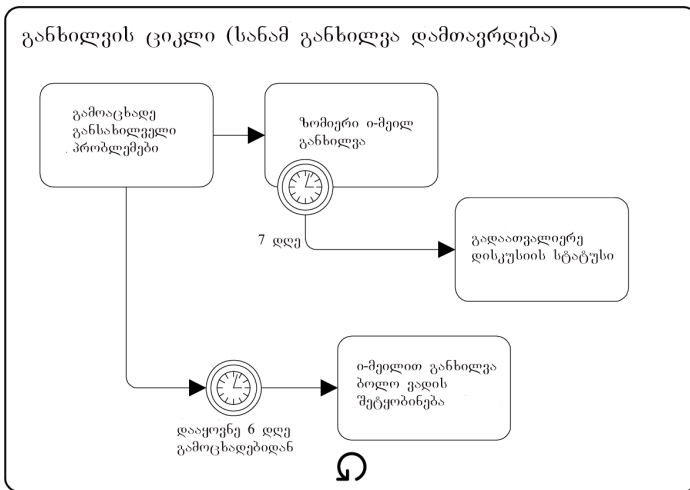


ნახ. 4.36. ამოცანა და დანგრეული (Collapsed) ქვე-პროცესი მარყუჟის მარკერით.



ნახ. 4.37. ამოცანა პარალელური მარკერით.

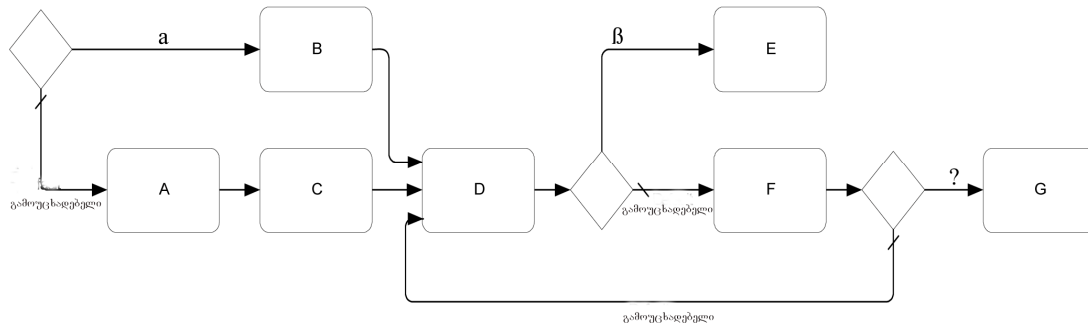
გაფართოებულ ქვე-პროცესს აგრეთვე შეიძლება ჰქონდეს მარყუჟის მარკერი განთავსებული ქვე-პროცესის სწორკუთხედის ქვედა ცენტრში (იხ. ნახ. 4.38). ქვე-პროცესის მთელი შემცველობა იქნება გამოვლენული როგორც განსაზღვრულია ატრიბუტებში.



ნახ. 4.38. გაფართოებული ქვე-პროცესი მარყუჟის მარკერით.

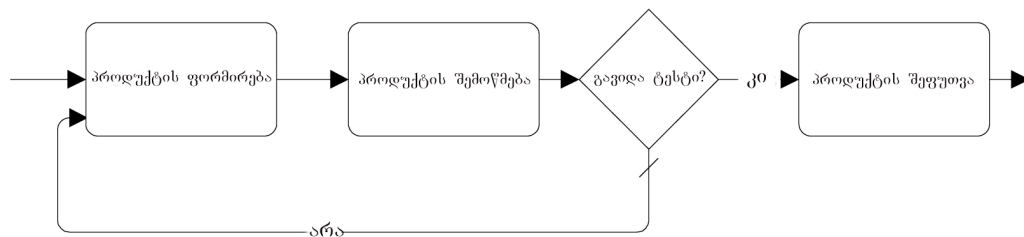
4.2.1.7. შესრულების ნაკადის გადაგრეხვა (Sequence Flow Looping)

მარყუჟი შეიძლება იყოს აგრეთვე შექმნილი შესრულების ნაკადის “დინების მიმართულებით ზემოთ მდებარე” ობიექტთან შეერთებით. ობიექტი ითვლება “დინების მიმართულებით ზემოთ მდებარედ” თუ ამ ობიექტს აქვს გამომავალი შესრულების ნაკადი, რომელსაც მიუყვართ სხვა შესრულების ნაკადის მწკრივთან (სერიასთან), რომლის ბოლოც აღმოჩნდება ორიგინალური (განსახილველი) ობიექტის შემავალი შესრულების ნაკადი. ეს ნიშნავს, რომ ობიექტი აწარმოებს სიმბოლოს და ეს სიმბოლო გადაკვეთს შესრულების ნაკადის სიმრავლეს სანამ სიმბოლო მიადწევს განსახილველ ობიექტს. შესრულების ნაკადის გადაგრეხვა არის იგივე რაც ტექნოლოგიური პროცესის ნიმუში №16: თავისუფალი ციკლი (წრე) (იხ. ნახ. 4.24).



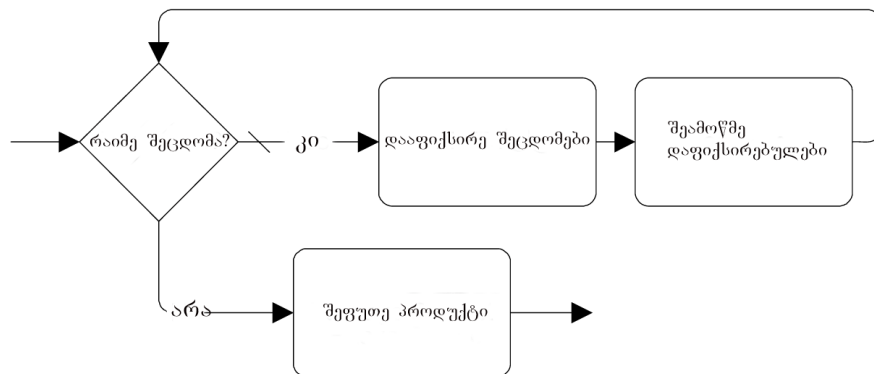
ნახ. 4.39. ტექნოლოგიური პროცესის ნიმუში №16: თავისუფალი ციკლი (წრე).

ჩვეულებრივ, ეს შეერთებები მისდევენ გადაწყვეტილებას ისე, რომ მარყუჟი არ იყოს უსასრულო (იხ. ნახ. 4.40). თუ შესრულების ნაკადი გამოდის უშუალოდ გადაწყვეტილებიდან “დინების მიმართულებით ზემოთ მდებარე” ობიექტისაკენ, ეს არის “მდე” (“until”) მარყუჟი. მომარყუჟებელი მოქმედებების (looped activities) სიმრავლე განხორციელდება (შესრულდება) მანამ, სანამ გარკვეული პირობა არის ჭეშმარიტი.



ნახ. 4.40. “მდე” მარყუჟი (An Until Loop).

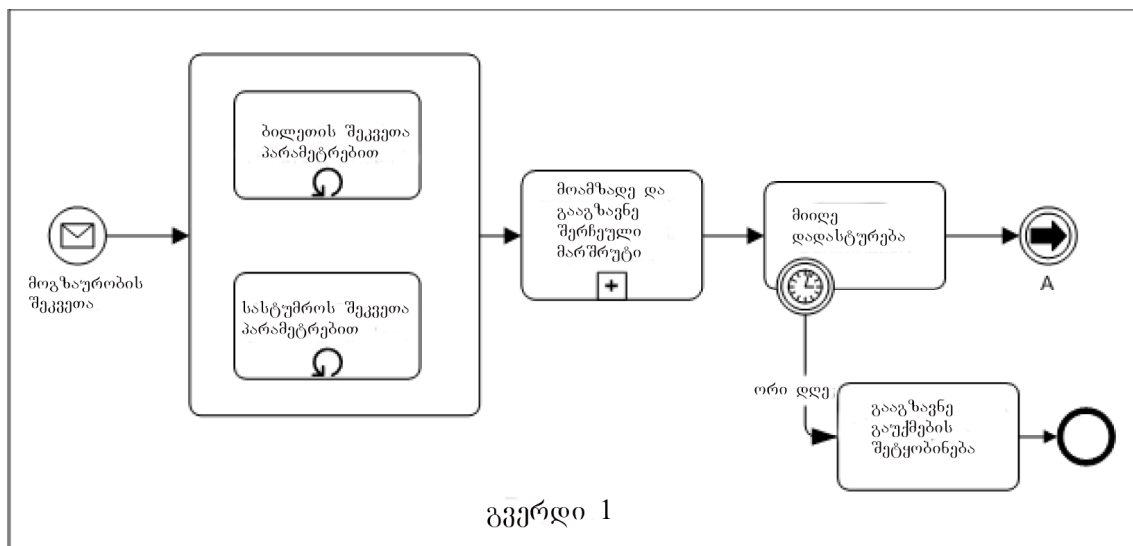
“სანამ” (“while”) მარყუჟი წარმოიქმნება გადაწყვეტილების მიღებით პირველად და შემდეგ გამეორებადი მოქმედებების განხორციელებით ან პროცესში შემდგომი გადაადგილებით (იხ. ნახ. 4.41). მომარყუჟებელი მოქმედებების (looped activities) სიმრავლე შეიძლება არ შესრულდეს ან შეიძლება შესრულდეს ბევრჯერ.

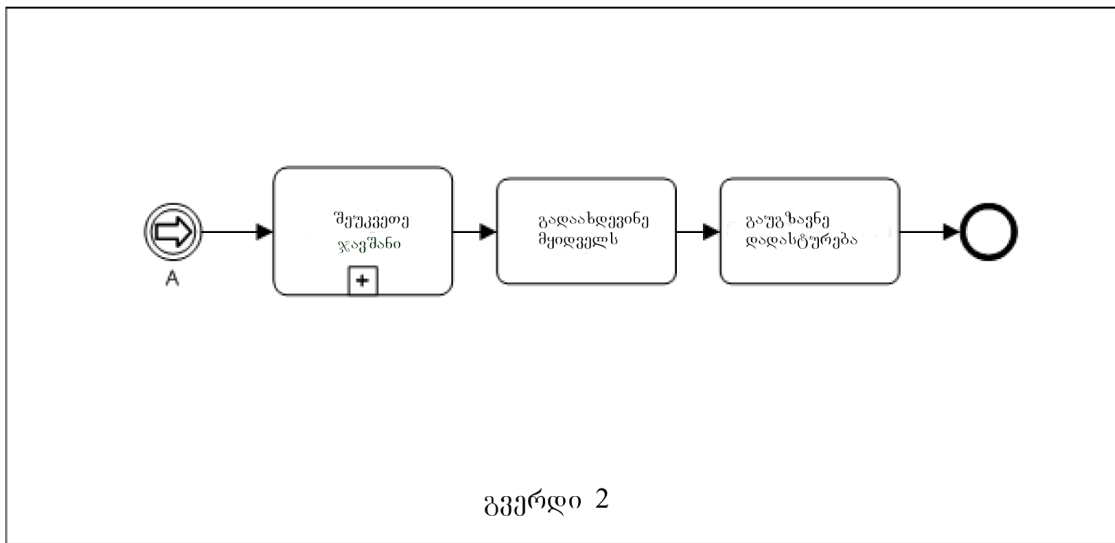


ნახ. 4.41. “სანამ” მარყუჟი

4.2.1.8. შესრულების ნაკადის გადახტომა (გვერდის გარეთ შემაერთებლები და ობიექტებზე გადასვლა) (Sequence Flow Jumping (Off-Page Connectors and Go To Objects))

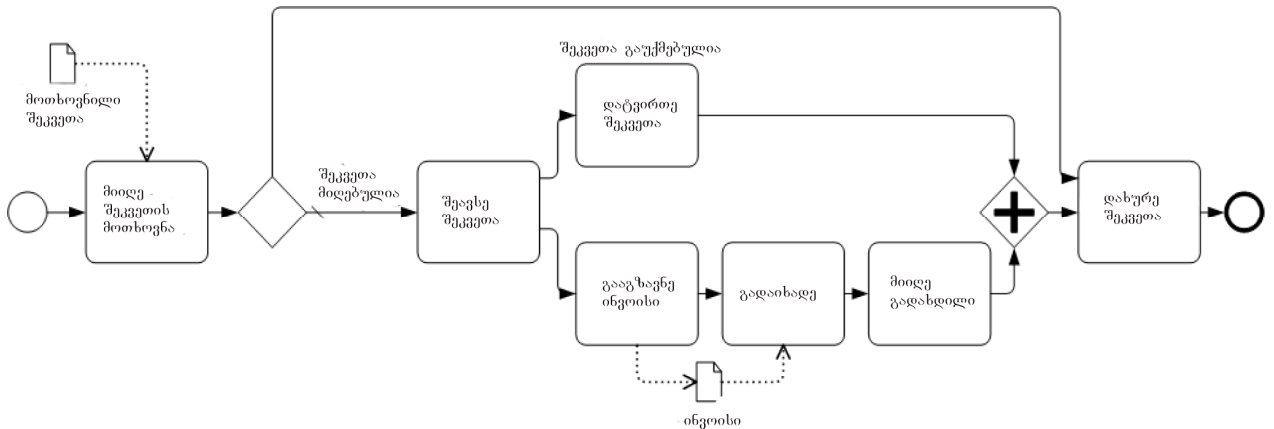
რადგან პროცესის მოდელირება ხშირად სცდება ერთი საბეჭდო გვერდის სიგრძეს, ხშირად წარმოიქმნება იმის ჩვენების აუცილებლობა თუ როგორ გავაფართოვოთ შესრულების ნაკადის შეერთებები გვერდის ფარგლებს გარეთ. ამ საკითხის ერთი გადაწყვეტა, რომელიც ხშირად გამოიყენება, არის გვერდის გარეთ შემაერთებლების გამოყენება იმის საჩვენებლად თუ სად მთავრდება ერთი გვერდი და სად იწყება მეორე. BPMN-ს აქვს შეერთების ტიპის შუალედური ხლომილებები, გვერდის ფარგლებს გარეთ შემაერთებლებად გამოსაყენებლად (იხ. ნახ. 4.42 -- შევნიშნოთ, რომ ნახაზზე ნაჩვენებია ორი სხვადასხვა საბეჭდო გვერდი და არა ორი გუბე ერთ დიაგრამაში). შეერთების შუალედური ხლომილებების წყვილი არის გამოყენებული. წყვილიდან ერთი ნაჩვენებია ერთი გვერდის ბოლოს. ამ ხლომილებას დარქმეული აქვს სახელი, მას აქვს შემავალი შესრულების ნაკადი და არა აქვს გამომავალი შესრულების ნაკადი. მეორე შეერთების ხლომილება არის შემდეგი გვერდის დასაწყისში, აქვს იგივე სახელი და აქვს გამომავალი შესრულების ნაკადი და არა აქვს შემავალი შესრულების ნაკადი.



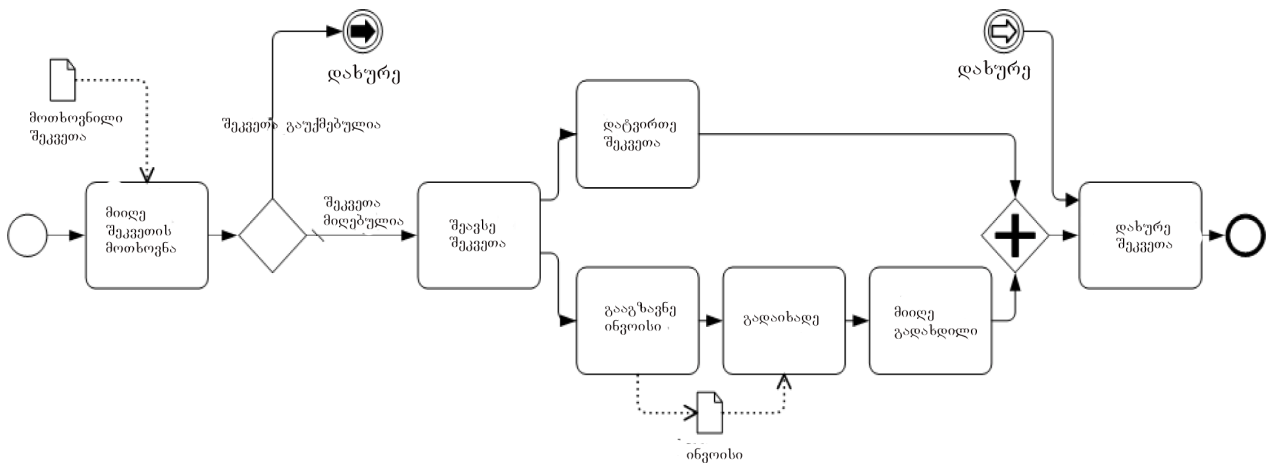


ნახ. 4.42. შეერთების შუალედური ხდომილება გამოყენებული როგორც გვერდის გარეთ შემართებული.

სხვა გზა, რომლითაც შეერთების შუალედური ხდომილება შეიძლება იყოს გამოყენებული, არის “გადადი” ობიექტებსზე (“Go To” objects) ანალოგიური. ფუნქციურად, ისინი ისევე იმუშავებენ, როგორც გვერდის გარეთ შემართებლები (აღწერილია ზემოთ), გარდა იმისა, რომ ისინი შეიძლება იყვნენ გამოყენებული დიაგრამის ნებისმიერ ადგილას: იგივე გვერდზე ან რამოდენიმე გვერდის შემდეგ. ძირითადი იდეა არის ის, რომ ისინი იძლევიან შესრულების ნაკადის ხაზების სიგრძის შემცირების მექანიზმს. ზოგიერთი მოდელიორისათვის გრძელი ხაზები შეიძლება იყოს ძნელი თვალის მისაღწევებლად და ტრასირებისათვის. “გადადი” ობიექტებზე შეიძლება იყვნენ გამოყენებული ძალზე გრძელი შესრულების ნაკადის თავიდან ასაცილებლად (იხ. ნახ. 4.43 და ნახ. 4.44). ორივე დიაგრამის ქცევა ერთნაირია. ნახ. 4.44-თვის, თუ დიაგრამიდან არჩეულია “უარყოფილი შეკვეთების” ბილიკი, მაშინ შესრულების ნაკადის გადაკვეთი სიმბოლო მიაღწევს წყარო შეერთების ხდომილებას და შემდეგ “გადახტება” მიზან შეერთების ხდომილებაზე და გააგრძელებს ქვემოთ შესრულების ნაკადს. პროცესი გაგრძელდებოდა ისე, თითქოს შესრულების ნაკადი აერთებს ორ ობიექტს.

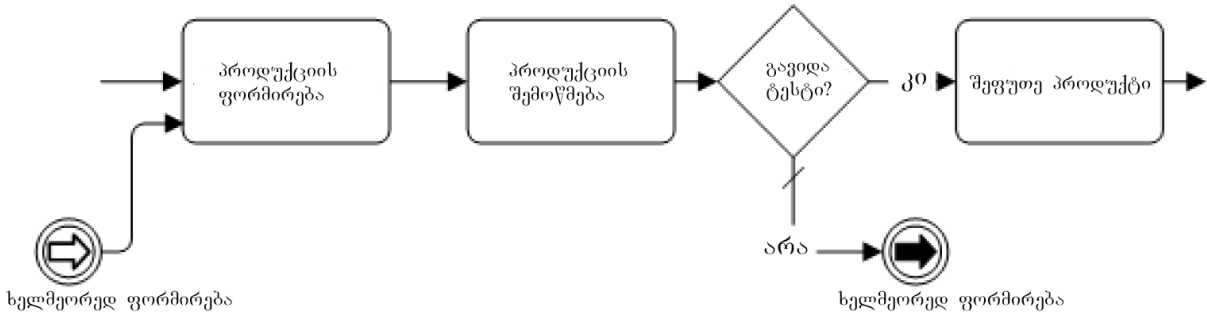


ნახ. 4.43. პროცესი გრძელი შესრულების ნაკადით.



ნახ. 4.44. პროცესი შეერთების შუალედური ხდომილებით, გამოყენებული როგორც “გადადი” ობიექტებზე.

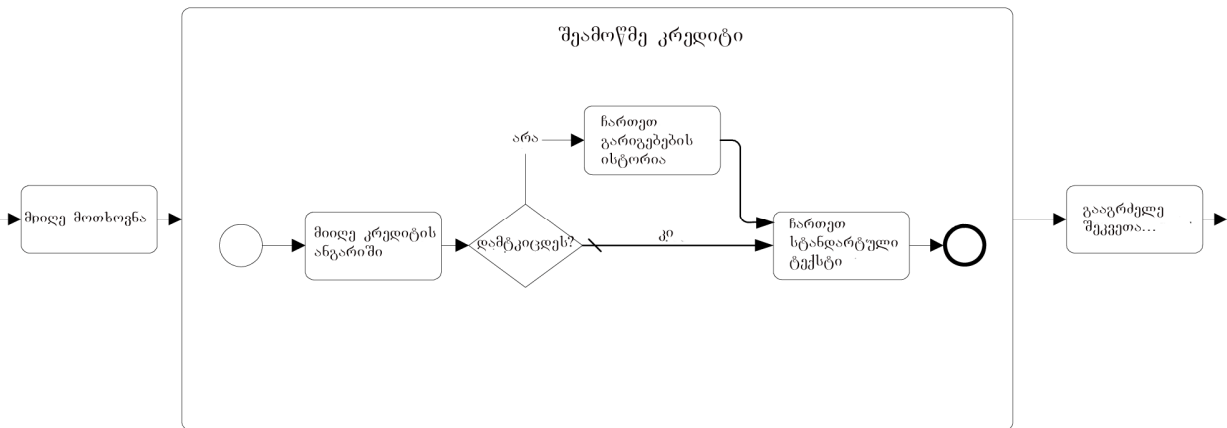
ზოგიერთი მეთოდოლოგია ამჯობინებს, რომ ყველა შესრულების ნაკადი მოძრაობდეს მხოლოდ ერთი მიმართულებით. ეს მეთოდოლოგიები არ რთავენ ნებას შესრულების ნაკადს უშუალოდ შეუერთდნენ “დინების მიმართულებით ზემოთ მდებარე” ობიექტებს (to upstream objects). მოდელირებაში გარკვეული მიმდევრობითობა შეიძლება იყოს მიღწეული ასეთი მეთოდოლოგიით, მაგრამ სიტუაციები, რომლებიც მოითხოვენ გადაგრეხვას (looping), ხდებიან პრობლემატური. კავშირის შუალედური ხდომილებები შეიძლება იყვნენ გამოყენებული “დინების მიმართულებით ზემოთ” მიმართული შეერთებების რეალიზაციისათვის და შეიძლება შექმნან მარყუებები შესრულების ნაკადის მიმართულებებზე შეზღუდვების დარღვევის გარეშე (იხ. ნახ. 4.45).



ნახ. 4.45. კავშირის შუალედური ხდომილება გამოყენებული გადაგრეხვისათვის (Used for Looping).

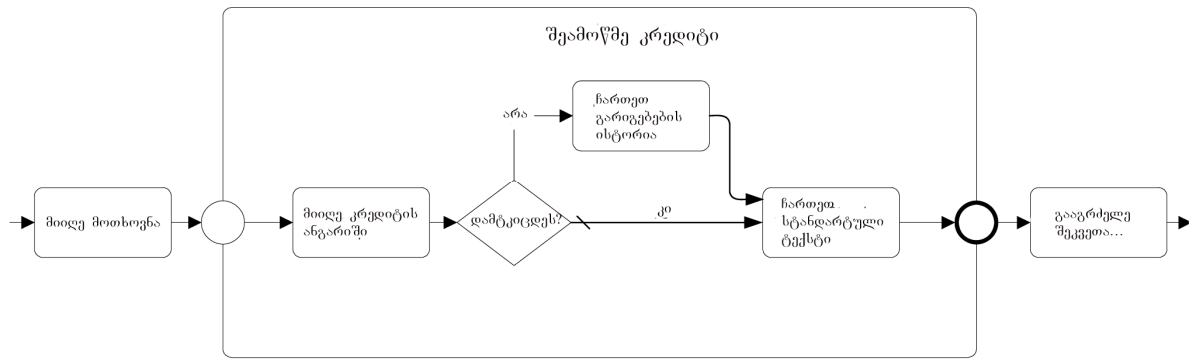
4.2.19. ნაკადის გატარება ქვე-პროცესისაკენ და ქვე-პროცესიდან

პარაგრაფში განხილულია როგორ იქნება გატარებული ნაკადი მშობელ პროცესსა და მის რომელიმე ქვე-პროცესს შორის. ნაკადი (ანუ სიმბოლო) დაიწყება მშობელ პროცესთან, შემდეგ გადაადგილდება ქვე-პროცესისაკენ და შემდეგ გადაადგილდება უკან მშობელი პროცესისაკენ (იხ. ნახ. 4.46). დროის უმეტესი ნაწილი ნაკადი იმოდრავებს ქვე-პროცესისაკენ, გადასცემს გადასაცემ ინფორმაციას ქვე-პროცესის დაწყების ხდომილებას, გადაკვეთს ქვე-პროცესის შესრულების ნაკადს, მიადწევს ქვე-პროცესის დამთავრების ხდომილებას და საბოლოოდ გადასცემს გადასაცემ ინფორმაციას უკან მშობელ პროცესს, გააგრძელებს რა ქვემოთ ქვე-პროცესის ობიექტის გამომავალ შესრულების ნაკადს. თუ ქვე-პროცესი შეიცავს პარალელურ ნაკადებს, მაშინ ყველა ნაკადი უნდა დამთავრდეს მანამ, სანამ სიმბოლო გადაეცემა უკან მშობელ პროცესს. ეს ფუნქციონალურად განიხილავს ქვე-პროცესს როგორც მოქმედებების დამოუკიდებელ “ყუთს”.



ნახ. 4.46. ქვე-პროცესის მაგალითი დაწყებისა და დამთავრების ხდომილებებით შიგნით.

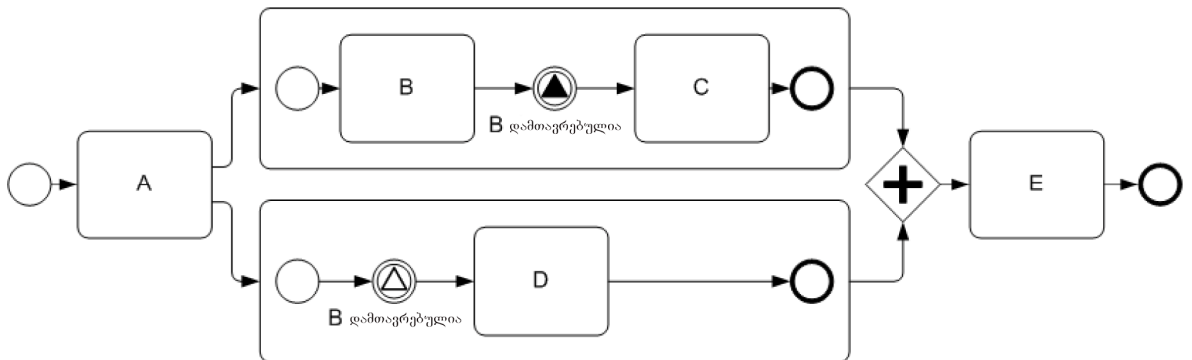
იმისათვის, რომ პროცესის დონეებს შორის ნაკადი იყოს უფრო თვალსაჩინო, მოდელიორს აქვს დაწყებისა და დამთავრების ხდომილებების ქვე-პროცესის საზღვარზე განთავსების არჩევანი და შეუერთოს შესრულების ნაკადი მშობელი პროცესის ობიექტიდან ამ ხდომილებებს (ან ამ ხდომილებებიდან მშობელი პროცესის ობიექტებს) (იხ. ნახ. 4.47).



ნახ. 4.47. ქვე-პროცესის მაგალითი დაწყებისა და დამთავრების ხდომილებებით საზღვარზე.

4.2.1.10. ნაკადის მართვა პროცესების შიგნით (Controlling Flow Across Processes)

პროცესების შიგნით შეიძლება შეგვხვდეს სიტუაციები, სადაც ნაკადზე გავლენას ახდენს მოქმედება ან ნაკადი არის დამოკიდებული მოქმედებაზე, რომელიც ხორციელდება სხვა პროცესში. ეს ხდომილებები ან პირობები შეიძლება მოხსენიებული იყვნენ როგორც ეტაპები (ნიშანსვეტები). პროცესის მოდელს უნდა შეეძლოს იდენტიფიცირება გაუკეთოს და რეაქცია მოახდინოს ამ ეტაპებზე. ეს ნიშნავს, რომ პროცესის გაგრძელება შეიძლება იყოს გადართული სიგნალის ხდომილებებით, რომლებიც ატარებენ (გადასცემენ) ნაკადს პროცესებს შორის (იხ. ნახ. 4.48). ამ ტექნოლოგიური პროცესის ნიშნის ტიპს ნიშანსვეტი (Milestone) ეწოდება.

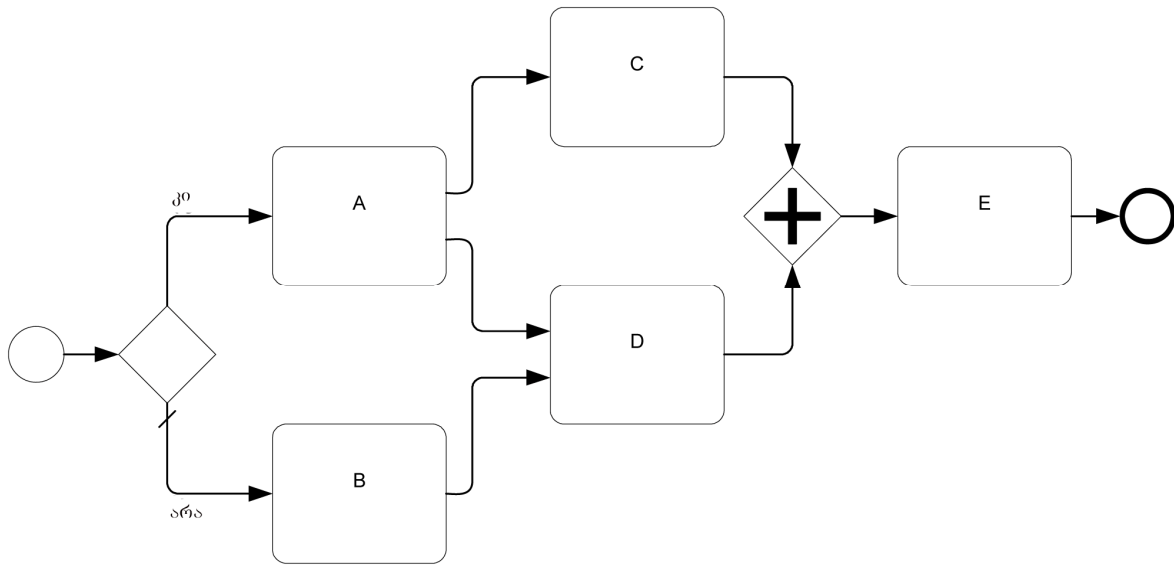


ნახ. 4.48. პროცესის შიგნით ქცევის სინქრონიზაციისათვის გამოყენებული სიგნალი ხდომილებები.

4.2.1.11. არალეგალური (უკანონო) მოდელების და მოულოდნელი ქცევის თავიდან აცილება

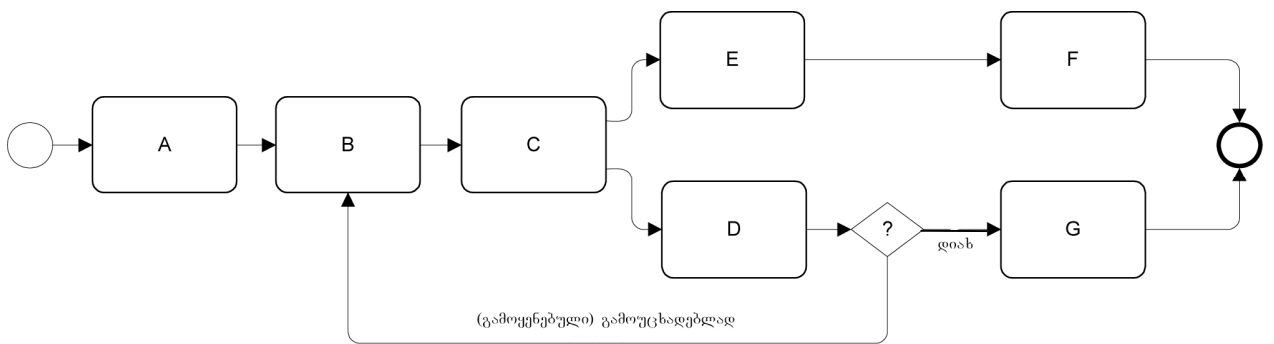
არის რა BPMN გრაფიკულ-სტრუქტურირებული დიაგრამა, ნაცვლად იმისა, რომ ჰქონდეს ბლოკური კონსტრუქცია BPEL4WS-ს ანალოგიურად, საკმაოდ მარტივი ფორმით უზრუნველყოფს დიდ მოქნილობას პროცესის რთული ქცევის წარმოსახვისათვის. თუმცა, BPMN-ის თავისუფალი ფორმის ბუნებას შეუძლია შექმნას მოდელირების სიტუაციები, რომელთა შესრულებაც შეუძლებელია ან რომლებიც მოიქცევიან მოდელირებისათვის მოულოდნელი მანერით (ფორმით). მოდელირების პრობლემების ეს ტიპები შეიძლება წარმოიშვან იმიტომ, რომ არ არსებობს მჭიდრო ურთიერთ დამოკიდებულება განშტოებას და შეერთებას შორის ან გახლეჩვას და შერწყმას შორის. ბლოკური სტრუქტურა უზრუნველყოფს ამ მჭიდრო ურთიერთ დამოკიდებულებებს, მაგრამ გრაფიკული სტრუქტურირება საშუალებას იძლევა ნაკადის მართვის ეს მაქანიზმები იყოს შერეული და ჰარმონირებული მოდელირების შეხედულებისამებრ. ამ მართვის ელემენტების ზოგიერთი კომბინაცია შექმნის პროცესებს, რომელთა შესრულებაც შეუძლებელია ან გამოიწვევს ქცევას, რომელიც მოდელირის არ უნდოდა. სიტუაციამ, როდესაც ალტერნატიული ბილიკები გადააკვეთენ პარალელური ბილიკების ჯგუფის არაცხად საზღვრებს, შეიძლება მოგვცეს ინვალიდი (მახინჯი) მოდელი.

ნახ. 4.49-ზე ნახვენებია ასეთი მოდელი. ამოცანა “D” არის მოქმედება, რომელსაც აქვს ორი შემავალი შესრულების ნაკადი; ერთი განშტოებადი ბილიკიდან და ერთი გახლეჩილი ბილიკიდან. ამან შეიძლება გამოიწვიოს პრობლემა პარალელურ გასასვლელთან, რომელიც წინ უსწრებს ამოცანა “E”-ს, რომელსაც აგრეთვე აქვს მრავალჯერადი შემავალი შესრულების ნაკადი. შესრულების ნაკადი ამოცანა “B”-დან კვეთს გაშლის (the fork) არა ცხად საზღვარს წარმოქმნილს ამოცანა “A”-ს შემდეგ. შედეგად, თუ “კი” შესრულების ნაკადი არის არჩეული გადაწყვეტილებიდან დიაგრამაში (ვარიანტ 1), მაშინ ამოცანა “E” შეიძლება დაელოდოს ორი სიმბოლოს მოსვლას: ერთი ამოცანა “C”-დან და ერთი ამოცანა “D”-დან. თუმცა, თუ “არა” შესრულების ნაკადი მოვიდა გადაწყვეტილებიდან (ვარიანტ 2), პარალელური გასასვლელი მიიღებს მხოლოდ ერთ სიმბოლოს: ერთი ამოცანა “D”-დან. რადგან გასასვლელი ელოდება ორ სიმბოლოს, პროცესი იქნება შესული ჩიხში ამ ადგილას.



ნახ. 4.49. პოტენციალურად ჩიხში შესული მოდელი.

სხვა ტიპის პრობლემა წარმოიშობა დინების მიმართულებით ზედა მოქმედებები-საკენ უკან გადაგრეხვით (უკანა მარყუქით). თუ მარყუქი გადაწყვეტილება არის შესრულებული პარალელური ბილიკების სიმრავლის არაცხად საზღვრებში, მაშინ მარყუქის ქცევა ხდება გაუგებარი (არა ცალსახა) (იხ. ნახ. 4.50), რადგან გაუგებარია ამოცანა “E” არის განმეორებადი მარყუქის საფუძველზე თუ რა შეიძლება მოხდეს თუ ამოცანა “E” იქნება ჯერ კიდევ აქტიური, როდესაც მარყუქი კვლავ მიაღწევს ამ ამოცანას.



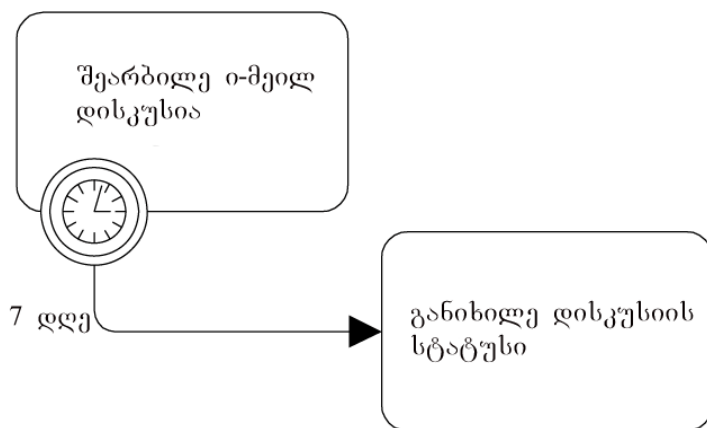
ნახ. 10.50. მცდარი გადაგრეხვა (Improper Looping)

შეერთების ხდომილებების გამოყენებას შეუძლია აგრეთვე გამოიწვიოს მოულოდნელი ქცევა. ზოგადად, შეერთების ხდომილებები, რომლებიც არ გამოიყენებიან გვერდის გარეთ შემაერთებლად, უნდა იყვნენ თანამედროვე მოდელირების ტექნიკად განხილული და მოდელიორი უნდა იყოს ყურადღებით, რომ გაიგოს შედეგად მიღებული საქციელი და სიმბოლოების ნაკადი.

ზოგადად, იმის ანალიზი თუ როგორ მიედინებიან სიმბოლოები მოდელის გავლით, დაგვეხმარება აღმოვაჩინოთ მოდელები, რომელთა სათანადო შესრულებაც შეუძლებელია.

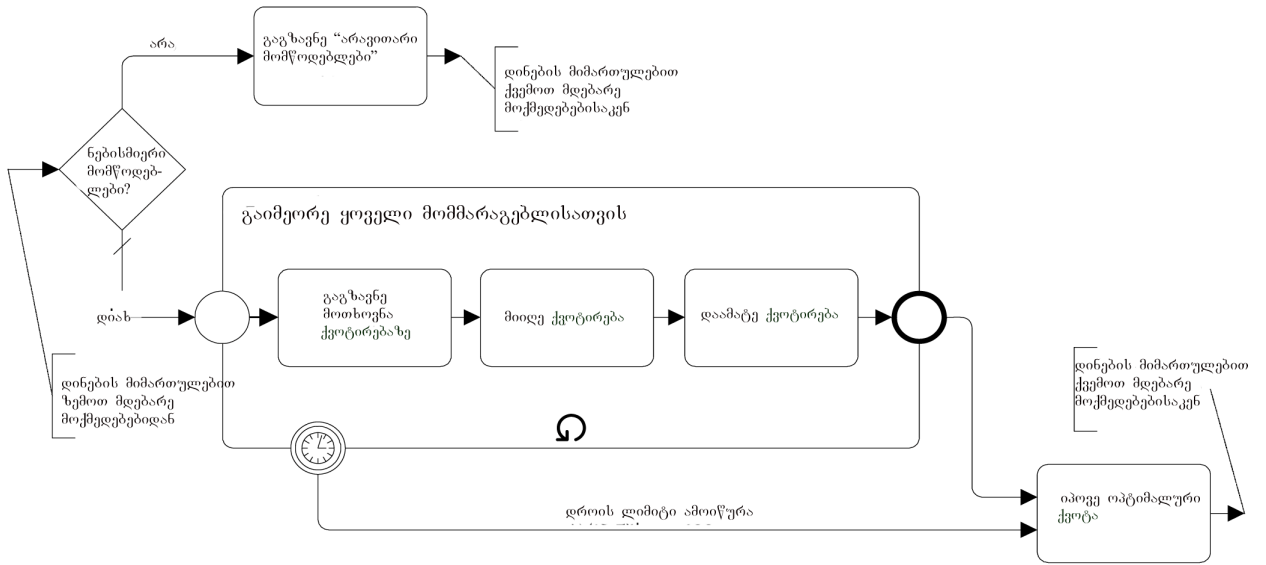
4.2.2 გამონაკლისი ნაკადი (Exception Flow)

გამონაკლისი ნაკადს ადგილი აქვს პროცესის ნორმალური ნაკადის გარეთ და დაფუძნებულია ხდომილებაზე (შუალედური ხდომილება), რომელსაც ადგილი აქვს პროცესის შესრულების დროს (განმავლობაში). მაშინ, როდესაც შუალედური ხდომილებები შეიძლება იყვნენ ჩართული ნორმალურ ნაკადში, იმისათვის, რომ აიძულოს დაყოვნებები ან გაწყვეტები დაელოდონ შეტყობინებას, როდესაც ისინი არიან მიდგმული ან ამოცანის ან ქვე-პროცესის მოქმედების საზღვარზე (იხ. ნახ. 4.51), ისინი ქმნიან გამონაკლისი ნაკადს.



ნახ. 4.51. ამოცანა გამონაკლისი ნაკადით (გაწყვეტების ხდომილების კონტექსტში).

ამის გაკეთებით, მოდელიორი ქმნის ხდომილების კონტექსტს. ხდომილების კონტექსტი უპასუხებს სპეციალურ (განსაზღვრულ) გადამრთველს, მოქმედების შესაწყვეტად ან ნაკადის გადასართველად (გადასამისამართებლად) შუალედური ხდომილების საშუალებით. ხდომილების კონტექსტი უპასუხებს მხოლოდ იმ შემთხვევაში თუ ის არის აქტიური (გაშვებული) გადართვის მომენტში. თუ მოქმედება დამთავრებულია, მაშინ გადამრთველი შეიძლება დარჩეს პასუხის გარეშე. გადამრთველის წყარო შეიძლება იყოს პროცესის შესრულებისათვის გარეშე, ისე როგორც შეტყობინების ან გამოყენების შეცდომა, ან გადამრთველი შეიძლება იყოს გამოწვეული “გადაგდების” შუალედური ხდომილებით რომელიმე სხვა მოქმედებიდან პროცესის შიგნით. ამ წესიდან გამონაკლისია შეცდომის ხდომილება, რომელიც უპასუხებს მხოლოდ მოქმედების შიგნით ან ამ მოქმედების ქვე-პროცესში გენერირებულ შეცდომის გადამრთველებს. თუ გვაქვს ამოცანების ჯგუფი, რომელიც მოდელიორს უნდა ჩართოს ხდომილების კონტექსტში, მაშინ ქვე-პროცესი შეიძლება იყოს დამატებული ამ ამოცანების მოსაცვლელად და მიმართოს ნებისმიერ ხდომილებას მათი მის საზღვარზე მიდგმით (იხ. ნახ. 4.52).



ნახ. 4.52. ქვე-პროცესი გამონაკლისი ნაკადით (გაწყვეტების ხდომილების კონტექსტში).

შეტყობინების ხდომილება წარმოიქმნება, როდესაც შეტყობინება, ზუსტი იდენტიფიკაციით, როგორც განსაზღვრულია შუალედურ ხდომილებაში, არის მიღებული პროცესის მიერ. შეცდომის ხდომილება წარმოიქმნება, როდესაც პროცესი აღმოაჩენს შეცდომას. თუ შეცდომის კოდი არის განსაზღვრული შუალედურ ხდომილებაში, მაშინ აღმოჩენილი შეცდომის კოდი უნდა შეესაბამებოდეს ხდომილების კონტექსტს (შინაარსს), რომ უპასუხოს. თუ შუალედური ხდომილება არ განსაზღვრავს შეცდომის კოდს, მაშინ ნებისმიერი შეცდომა გადართავს პასუხს ხდომილების კონტექსტიდან. თუ ეს ხდომილება არ მოხდება იმ დროის განმავლობაში, როდესაც ხდომილების კონტექსტი არის მზად, მაშინ პროცესი გაგრძელდება ნორმალური ნაკადის გავლით, როგორც არის განსაზღვრული შესრულების ნაკადის საშუალებით.

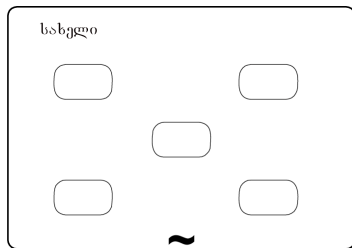
4.2.3. სპეციალური (Ad Hoc)

სპეციალური (Ad Hoc) პროცესი არის მოქმედებების ჯგუფი, რომელსაც არა აქვს წინასწარ განსაზღვრული ურთიერთკავშირების მიმდევრობა. პროცესისათვის შეიძლება იყოს განსაზღვრული მოქმედებების სიმრავლე, მაგრამ მოქმედებებისათვის შესრულების მიმდევრობა და რაოდენობა არის სრულად განსაზღვრული მოქმედებების შესრულებით და შეუძლებელია იყოს წინასწარ განსაზღვრული.

ქვე-პროცესი მონიშულია როგორც სპეციალური (Ad Hoc) ქვე-პროცესის ფორმის ქვედა ცენტრში განლაგებული "ტილდა" სიმბოლოთი (იხ. ნახ. 4.53 და ნახ. 4.54). პროცესში მოქმედებები არიან გათიშული ერთმანეთისაგან. პროცესის შესრულების დროს, ნებისმიერი ერთი ან მეტი მოქმედება შეიძლება იყოს აქტიური და ისინი შეიძლება იყვნენ შესრულებული თითქმის ნებისმიერი მიმდევრობით ან სისხიროთ.

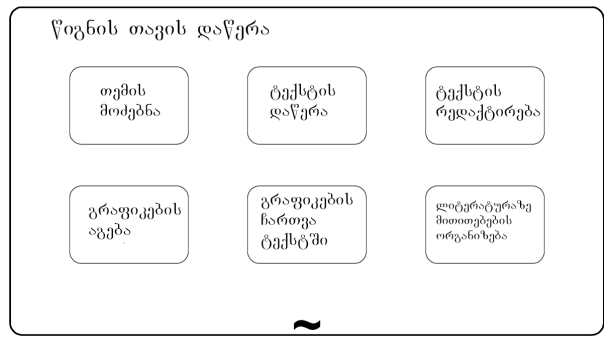


ნახ. 4.53. დანგრეული სპეციალური (Ad Hoc) ქვე-პროცესი.



ნახ. 4.54. გაფართოებული სპეციალური (Ad Hoc) ქვე-პროცესი.

შემსრულებლები განსაზღვრავენ მოქმედებების დაწყებას, მათ დამთავრებას, რომელი იქნება შემდეგი მოქმედება და ა.შ. სპეციალური (Ad Hoc) პროცესების ტიპების მაგალითებად შეიძლება განვიხილოთ: კომპიუტერის კოდის დამუშავება (დაბალ დონეზე), გაყიდვების მხარდაჭერა და წიგნის თავის დაწერა. თუ შევხვდებით წიგნის თავის დაწერის დეტალებს, დავინახავთ, რომ ამ პროცესის შიგნით მოქმედებები შეიცავენ: თემის მოძებნას, ტექსტის დაწერას, ტექსტის რედაქტირებას, გრაფიკების დახაზვას, გრაფიკების ჩართვას ტექსტში, ლიტერატურის მოწესრიგებას და ა.შ. (იხ. ნახ. 4.55). შეიძლება არსებობდნენ რაღაც დამოკიდებულებები ამოცანებს შორის პროცესში, ისეთი როგორცაა ტექსტის დაწერა ტექსტის რედაქტირების წინ, მაგრამ არ არსებობს კორელირების რაიმე აუცილებლობა, მაგალითად, ტექსტის დაწერასა და, მაგალითად, ტექსტის რედაქტირებას შორის. რედაქტირება შეიძლება მოხდეს იშვიათად და ტექსტის დაწერის ამოცანის მრავალი მაგალითისათვის ტექსტზე დაყრდნობით.



ნახ. 4.55. სპეციალური (Ad Hoc) პროცესი წიგნის თავის დასაწერად.

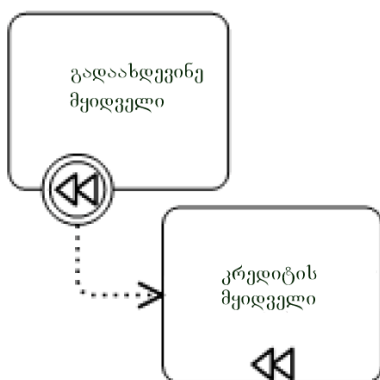
4.3. კომპენსაციის გაერთიანება

ზოგიერთი მოქმედებები წარმოქმნიან რთულ ეფექტებს ან განსაკუთრებულ შედეგებს (outputs). თუ შედეგი განსაზღვრულია როგორც არასასურველი, რაიმე განსაზღვრული კრიტერიუმებით (ისეთი როგორცაა შეკვეთის გაუქმება), მაშინ საჭირო იქნება “მოსპო” (“undo”) მოქმედებები. *არსებობს ამის გაკეთების სამი გზა:*

- მონაცემების საწყისი მნიშვნელობების ასლის აღდგენა, რითაც გადაიწერება ნებისმიერი ცვლილებები.
- არაფრის გაკეთება (თუ არაფერი შეცვლილა, რადგან ცვლილებები არ იყვნენ შესრულებული დადასტურებამდე).
- მოქმედებების გამოძახება, რომლებიც ანადგურებენ ეფექტებს – აგრეთვე ცნობილი როგორც კომპენსაცია.

მოქმედება, რომელსაც შეუძლია მოითხოვოს კომპენსაცია, შეიძლება იყოს, მაგალითად, ის, რომელიც აჯარმობს მყიდველს რაღაც მომსახურების გამო და ამას აკეთებს მისი საკრედიტო ბარათის მეშვეობით. ამ ტიპის მოქმედებებს ჩვეულებრივ სჭირდებათ განსაკუთრებული (ცალკე) მოქმედება იმისათვის, რომ დაძლიონ საწყისი მოქმედების ეფექტები. ხშირად, ორივე მოქმედების ჩანაწერი მოითხოვება, ასე რომ, ეს არის კიდევ ერთი მიზეზი, რის გამოც მოქმედება არ არის “განადგურებული”. კომპენსაციის ტიპის შუალედური ხდომილება არის მიდგმული მოქმედების საზღვარზე იმის მისათითებლად, რომ ამ მოქმედებისათვის შეიძლება კომპენსაცია იყოს საჭირო.

“განადგურების” (“undo”) მოქმედებების სამი მექანიზმიდან ერთი - კომპენსაცია, მოითხოვს სპეციალურ აღნიშვნას და ის არის სპეციალური პირობა (გარემოება), რომელიც წარმოიქმნება პროცესის ნორმალური ნაკადის გარეთ. ამის გამო, კომპენსაციის შუალედურ ხდომილებას არა აქვს გამომავალი შესრულების ნაკადი, მაგრამ სამაგიეროდ აქვს გამომავალი მიმართული გაერთიანება (Association) (იხ. ნახ. 4.56).



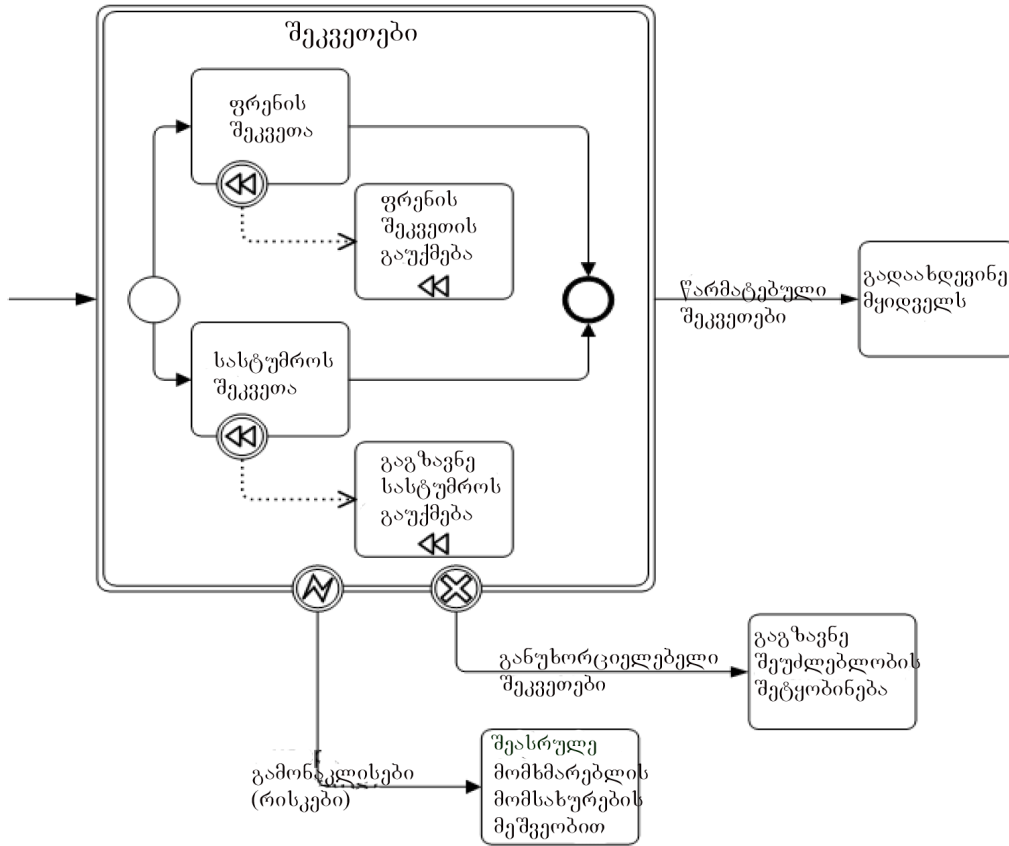
ნახ. 4.56. ამოცანა გაერთიანებული კომპენსაციის მოქმედებით.

ამ გაერთიანების მიზანი არის მოქმედება, რომელიც მოახდენს წყარო მოქმედებაში გაკეთებული მუშაობის კომპენსაციას და იქნება მითითებული

როგორც კომპენსაციის მოქმედება. კომპენსაციის მოქმედება არის განსაკუთრებული, რადგან ის არ ექვემდებარება ნორმალური ნაკადის შესრულების წესებს -- როგორც იყო აღნიშნული, ის არის პროცესი ნორმალური ნაკადის გარეთ. ამ მოქმედებას არ შეუძლია ჰქონდეს რაიმე შემავალი ან გამომავალი შესრულების ნაკადი. კომპენსაციის მარკერი (როგორც არის კომპენსაციის შუალედურ ხდომილებაში) განთავსებული იქნება მოქმედების ქვედა ცენტრში მოქმედების ამ სტატუსის საჩვენებლად (იხ. ამოცანა “კრედიტის მყიდველი” ნახ. 4.56-ზე). შევნიშნოთ, რომ კომპენსაციისათვის შეიძლება არსებობდეს მხოლოდ ერთი მიზანი მოქმედება. შეუძლებელია აქ იყოს ნაწევრები მოქმედებების მიმდევრობა. თუ კომპენსაცია მოითხოვს ერთზე მეტ მოქმედებას, მაშინ ეს მოქმედებები უნდა განთავსდეს მარტივი ქვე-პროცესის შიგნით, რომელიც არის გაერთიანების მიზანი. ქვე-პროცესი შეიძლება იყოს დაზრეული (collapsed) ან გაფართოებული (expanded). თუ ქვე-პროცესი არის გაფართოებული, მაშინ მხოლოდ ქვე-პროცესი თავის თავად მოითხოვს კომპენსაციის მარკერს – მოქმედებები ქვე-პროცესის შიგნით არ მოითხოვენ ამ მარკერს.

მხოლოდ დამთავრებული მოქმედებები შეიძლება იყვნენ კომპენსირებული. *მოქმედების კომპენსაცია შეიძლება იყოს გამოწვეული ორი გზით:*

- მოქმედება არის შესრულებადი ქვე-პროცესის შიგნით, რომელიც არის გაუქმებული (იხ. ნახ. 4.57). ამ სიტუაციაში, მთლიანად ქვე-პროცესი იქნება “გადახვეული” (“перемотан”) ან დადაბლებული წინა დონემდე – პროცესის ნაკადი წავა უკან და ნებისმიერი მოქმედება, რომელიც მოითხოვს კომპენსაციას იქნება კომპენსირებული. ამის გამო ხდომილებისათვის კომპენსაციის მარკერი გამოიყურება როგორც “გადახვევის” (“перемотки”) სიმბოლო. კომპენსაციის დამთავრების შემდეგ პროცესი გააგრძელებს თავის უკუ გადახვევას (свою обратную перемотку).
- დინების მიმართულებით ქვემოთ განლაგებული კომპენსაციის ტიპის შუალედური ან დამთავრების ხდომილება “გადააგდებს” (“throws”) კომპენსაციის იდენტიფიკატორს, რომელიც არის “დაჭერილი” (“caught”) შუალედური ხდომილების მიერ, რომელიც მიდგმულია მოქმედების საზღვართან. *კომპენსაციის გადაგდება ხდება ორნაირად:*
 - ხდომილებას შეუძლია კომპენსაციის მომთხოვნი მოქმედების კონკრეტულად იდენტიფიცირება და მხოლოდ ეს მოქმედება იქნება კომპენსირებული.
 - ხდომილებას შეუძლია გამოაცხადოს კომპენსაციის საჭიროება და შემდეგ ყველა მოქმედება, რომელსაც აქვს კომპენსაციის შუალედური ხდომილება მიდგმული მათ საზღვრებზე, იქნებიან კომპენსირებული. კომპენსაცია ეხება ყველა მოქმედებებს, რომლებიც მთლიანად დასრულდნენ პროცესის ფარგლებში (რომელიც მოიცავს პროცესის ყველა დონეს). კომპენსაცია განხორციელდება ორიგინალური მოქმედებების შესრულების უკუ მიმდევრობით.



ნახ. 4.57. კომპენსაცია ნაჩვენები შესრულების კონტექსტში

პროცესის მოდელის აგება (Building a Process Model)

სახელმძღვანელოში აღწერილია ძირითადი მოქმედებები მარტივი პროცესის მოდელის შესაქმნელად BPMN მოდელირების გამოყენებით. სახელმძღვანელოს მიზანია მოდელირების ზოგიერთი საკვანძო ცნებების (კონცეფციების) ილუსტრირება და აგრეთვე იმის ჩვენება თუ როგორ შეიძლება პროცესის მოდელის სწორად აგება პროგრამული პროდუქტის Interfacing BPMN modeler for Visio-ს გამოყენებით.

ლაბორატორიული სამუშაო №1

პროგრამული პროდუქტის Interfacing BPMN modeler for Visio-ს შესაძლებლობების და თავისებურების მიმოხილვა: ძირითადი ფუნქციების და ამოცანების გაცნობა, ინსტრუმენტების პანელის და რეალიზებული ფორმების მიმოხილვა, ამოცანის გადაწყვეტის ძირითადი ეტაპების და თავისებურების განხილვა.

მოდელის მიმოხილვა (Model overview)

ქვემოთ მოყვანილ გამარტივებულ მოდელს ვუწოდებთ “ახალი თანამშრომლების დაქირავებას”. ის აღწერს ახალი თანამშრომლის გადაბირების და დაქირავების პროცესს.

გადაბირების პროცესის დაწყების წინ ადამიანური რესურსების დეპარტამენტმა უნდა შეამოწმოს, რომ სამუშაოს აღწერა არის სწორად განახლებული. როგორც კი ეს მოხდება, არსებული სამუშაოს შესახებ გაკეთდება განცხადება თანამშრომლებისათვის კომპანიის შიგნით და ასევე საერთოდ საზოგადოებისათვის. კანდიდატურები შეირჩევა პრეტენდენტების სიიდან და გასაუბრების შემდეგ ადამიანური რესურსების მმართველი (მენეჯერი) განსაზღვრავს რომელი კანდიდატი დაიქირაოს სამუშაოზე.

ახალი ნახაზის გახსნა და სახელის დარქმევა (Open and name a new drawing)

1. “Start menu”-დან აირჩიე ოფცია **All programs > Interfacing BPMN Modeler for Visio > New BPMN Model**. სახატავი (დასახაზი) გვერდი სახელით *New Scenario* გაიხსნება. მას ეწოდება *სცენარის* გვერდი (*Scenario page*) და ყოველთვის არის მოდელის პირველი გვერდი.
2. თავუნას მარჯვენა ღილაკი დააწკაპუნეთ ცარიელ გვერდზე და აირჩიეთ ოფცია **Properties**. ეს გახსნის *შემკვეთის თვისების რედაქტორს* (the *Custom Property Editor*) სცენარის გვერდისათვის.
3. სცენარის სახელის ველი გამოიყენება საწარმოს ან უმაღლესი დონის დეპარტამენტის, რომლის მოდელირებაც თქვენ გინდათ, სახელის შესაყვანად. შეიყვანეთ სახელი “Human Resources” (“ადამიანური რესურსები”).

4. აღწერის ველი გამოიყენება თქვენი მოდელის მიზნის მოკლე ახსნისათვის. შეიყვანე ტექსტი “ახალი თანამშრომლის დაქირავების დამუშავების მოდელები” (“Models the processing of hiring a new employee”).
5. დააწკაპუნე “OK”.

ნახაზის შენახვა (Save the drawing)

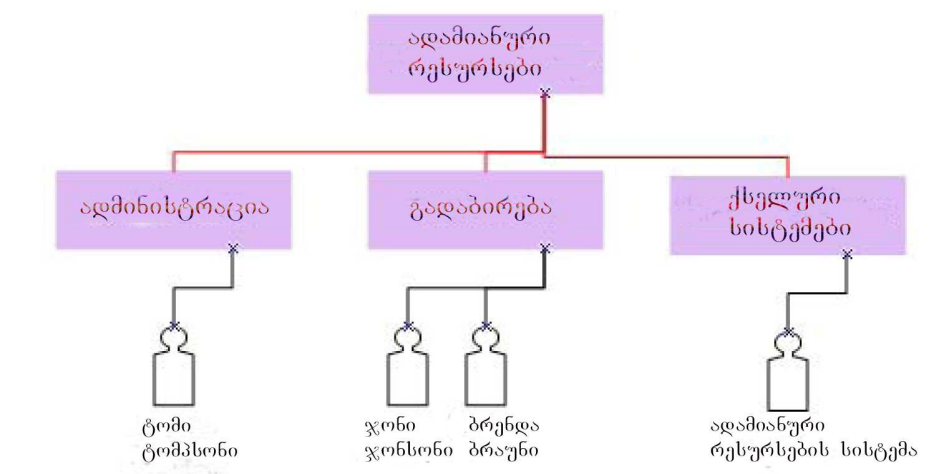
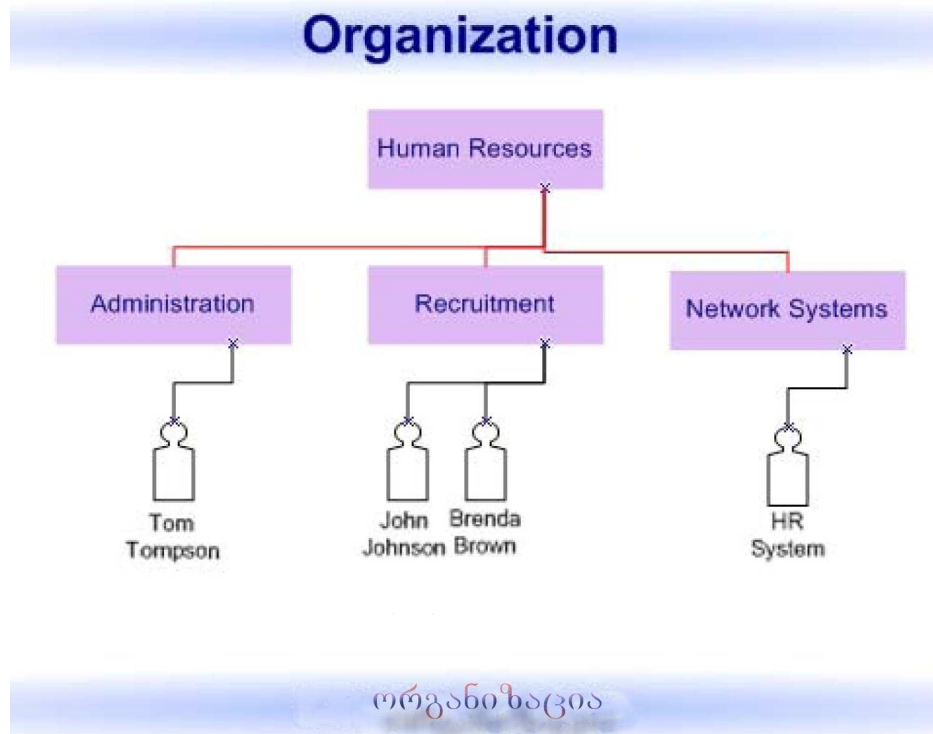
1. ინსტრუმენტების პანელიდან შეარჩიე **File > Save** (ფაილი > შენახვა).
2. ნახაზის პირველად შენახვისას გამოვა *Save As* (შეინახე როგორც) ფანჯარა. შეარჩიე ფაილის განთავსების ადგილი.
3. შეიყვანე ფაილის სასურველი სახელი.
4. დარწმუნდი, რომ **Drawing (*.vsd)** არის ნაჩვენები *Save as* ტიპის ველში. თუ ის არ არის ნაჩვენები, შეარჩიე ის ქვემოთ-ჩამოშლადი სიიდან.
5. დააწკაპუნე **Save**.

ბიზნეს პროცესის სრული მოდელი სათანადო დონეზე გაანალიზებისათვის მოითხოვს როგორც ორგანიზაციულ სტრუქტურას, ასევე პროცესის იერარქიას. ორგანიზაციულ სტრუქტურა აღწერს დეპარტამენტებს, როლებს (ე.ი. როლებს და აქტივებს) და რესურსებს, რომლებიც მონაწილეობენ მოდელში. პროცესის იერარქია არის ერთმანეთთან შეერთებული (დაკავშირებული) მოქმედებების მიმდევრობა, რაც წარმოადგენს სამუშაოს ნაკადს მოდელში.

შეგიძლიათ შექმნათ პროცესის მაღალი დონის სქემა ორგანიზაციულ სტრუქტურის შექმნის გარეშე. BPMN მოდელიორი აგრეთვე იძლევა საშუალებას შექმნათ თქვენი ორგანიზაციული სტრუქტურა და სწრაფად დაამატოთ რესურსები პროცესისა და მოქმედების დეტალებით შევსების მიზნით. ამ სახელმძღვანელოს მიზნებისათვის, ჩვენ დავიწყებთ ორგანიზაციული სტრუქტურისა და რესურსების შექმნით.

ორგანიზაციული სტრუქტურის აგება

სიმარტივისათვის, განსახილველი ორგანიზაცია შედგება ოთხი ერთეულისაგან: უმაღლესი დონის ერთეულს ჰქვია “ადამიანური რესურსები” (“Human Resources”), რომელიც წარმოადგენს მთავარ დეპარტამენტს, და სამი ქვე-ერთეული, რომლებსაც ჰქვიათ “ადმინისტრაცია”, “გადაბირება” და “ქსელური სისტემები” (“Administration”, “Recruitment”, and “Network Systems”). სახელმძღვანელოს ამ ნაწილში ჩვენ შევისწავლით ორგანიზაციის გვერდის (*Organization page*) შექმნას, ორგანიზაციული სტრუქტურის შექმნას და რესურსების განსაზღვრას ორგანიზაციის ერთეულებისათვის.



ორგანიზაციის გვერდის შექმნა

ჩვენი მოდელის სცენარის გვერდზე.

1. დააწკაპონე თავუნას მარჯვენა ღილაკი გვერდის ცარიელ არეზე.

- შეარჩიე **შექმენი ორგანიზაციის გვერდი (Create Organization Page)**. გვერდი შეიქმნება და სახელი დაერქმევა ავტომატურად. როდესაც ეს პროცედურა დამთავრდება, თქვენ იქნებით *ახალი ორგანიზაციის (new Organization)* გვერდზე.

ორგანიზაციისა და სცენარის გვერდებს შორის მოძრაობა

- მართვის სიმბოლოები (navigation tabs) განთავსებულია გვერდის ქვემოთ, ჰორიზონტალური გადახვევის ბარის მარცხნივ (to the left of the horizontal scroll bar).
- შეარჩიე სცენარის სიმბოლო (**Scenario tab**). ეს მიგიყვანთ სცენარის გვერდთან (to the *Scenario* page). თქვენ დაინახავთ სწორკუთხედს სახელწოდებით *ორგანიზაცია (Organization)*, ასევე სწორკუთხედს სახელწოდებით *რესურსები (Resources)*.
- დააწკაპუნეთ მარჯვენა დილაკი *ორგანიზაციის* სწორკუთხედზე (the *Organization* rectangle).
- შეარჩიე **გადადი ორგანიზაციაზე (Go To Organization)**. თქვენ დაბრუნდებით *ორგანიზაციის* გვერდზე (to the *Organization* page). (ორჯერ დააწკაპუნება *ორგანიზაციის* სწორკუთხედზე აგრეთვე მიგიყვანთ *ორგანიზაციის* გვერდთან).

ლაბორატორიული სამუშაო №2

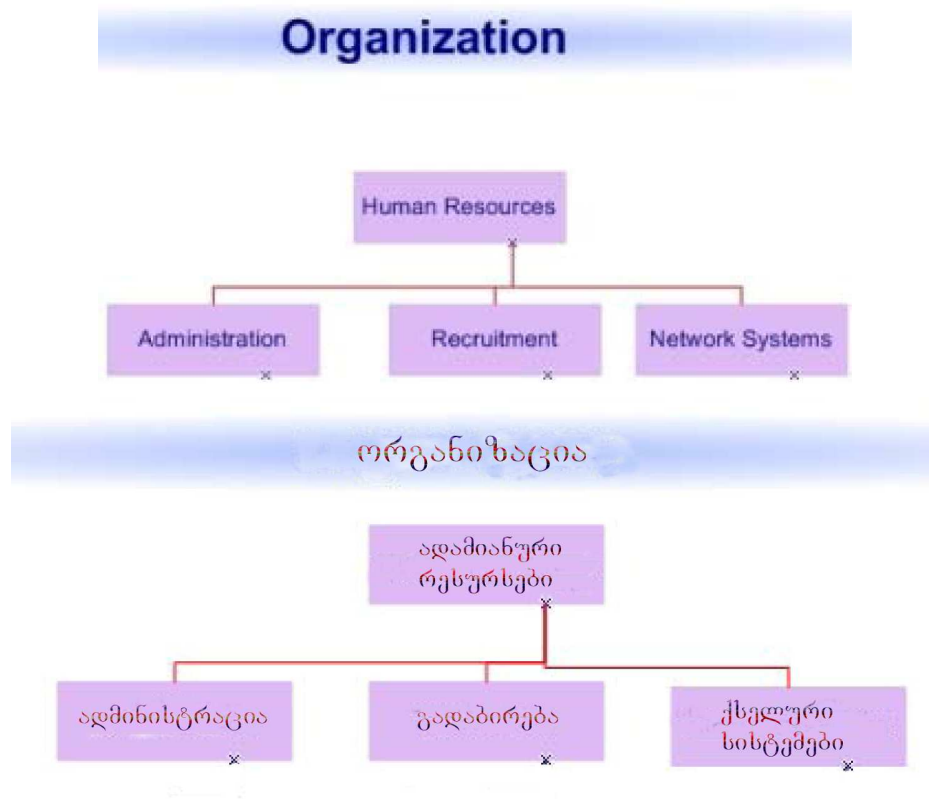
ორგანიზაციული სტრუქტურის შექმნა

1. ორგანიზაციის გვერდზე ორჯერ დააწკაპუნე გამოუცხადებელი უმაღლესი დონის ორგანიზაციის ერთეულზე სახელწოდებით საწარმო (**Enterprise**).
2. როდესაც *შემკვეთის თვისების რედაქტორი (Custom Property Editor)* გაიხსნება, გამოიყენე *საწარმო (Enterprise)* “Human Resources” (“ადამიანური რესურსები”) ორგანიზაციის სახელად. დააწკაპუნე OK. ეს ფორმა წარმოადგენს უმაღლესი დონის დეპარტამენტს ჩვენ მოდელში.
3. გადაათრე FS ორგანიზაციის (**FS organization**) ფორმა *მოწინავე სცენარის (Scenario Advanced)* ტრაფარეტიდან და ჩააგდე *ორგანიზაციის* გვერდში ქვემოთ და *ადამიანური რესურსების* დეპარტამენტის მარცხნივ. ამ ერთეულს დაარქვი “Administration” (“ადმინისტრაცია”).
4. ჩააგდე ორი სხვა FS ორგანიზაციის (**FS organization**) ფორმები გვერდის ფარგლებში და დაარქვი მათ “Recruitment” (“გადაბირება”) და “Network Systems” (“ქსელური სისტემები”).

შენიშვნა: თქვენ შეგიძლიათ ასევე გახსნათ *შემკვეთის თვისების რედაქტორი (Custom Property Editor)* თუ მარჯვენა ღილაკს დააწკაპუნებთ ფორმაზე და აირჩევთ თვისებებს (**Properties**).

ორგანიზაციის ერთეულების შეერთება

1. ორჯერ დააწკაპუნე “Administration”-ზე ორგანიზაციის გვერდზე *შემკვეთის თვისების რედაქტორის (the Custom Property Editor)* გასახსნელად.
2. შეარჩიე ღილაკი “...” ველში Parent და დააწკაპუნე.
3. შეარჩიე **ადამიანური რესურსები (Human Resources)** და დააწკაპუნე **Assign Organization** (მიანიჭე ორგანიზაცია). *ადამიანური რესურსები (Human Resources)* არის განსაზღვრული როგორც მშობელი *ადმინისტრაციის* ერთეულის (*Administration unit*). დააწკაპუნე OK.
4. გაიმეორე ეს ბიჯები *Recruitment (გადაბირება)* და *Network Systems (ქსელური სისტემების)* ორგანიზაციების *Human Resources (ადამიანური რესურსების)* ორგანიზაციასთან შესაერთებლად. თქვენი ორგანიზაციის გვერდი ახლა შეიცავს შემდეგ ფორმებს:



რესურსების გვერდის გახსნა

სანამ განვსაზღვრავთ ერთეულებს (რესურსებს) ამოცანების შესასრულებლად ჩვენ მოდელში, ჩვენ უნდა განვსაზღვროთ როლები (ე.ი. როლები ან აქტივები) (i.e. roles or assets) და ინდივიდუალური რესურსები მათი შექმნით *რესურსების გვერდზე* (on the *Resources* page).

1. შეარჩიე **რესურსების (Resources)** მართვის სიმბოლოები (navigation tabs) ფანჯრის ბოლოში.
2. *რესურსების გვერდი* (The *Resources* page) გამოჩნდება.

როლის შექმნა

ბიზნეს პროცესის მოდელში, მოქმედებების უმრავლესობა შეიძლება შესრულდეს ინდივიდუალური რესურსით ან **როლით** (ე.ი. როლები ან აქტივები) (i.e. roles or assets). როლების განსაზღვრა საშუალებას გვაძლევს ვიქონიოთ მოქმედებები შესრულებული რესურს გუბეებით (resource pools), იმის დაზუსტების

გარეშე თუ გუბის რომელი წევრი შეასრულებს ნამდვილად მოქმედებას. ყოველი როლი წარმოადგენს ერთ რესურს გუბეს (resource pool).

რესურს გუბის შესაქმნელად თქვენ უნდა შექმნათ როლი, ასევე ამ როლის ინდივიდუალური წევრები და შემდეგ შეუერთოთ (დაუკავშიროთ) წევრები ორგანიზაციის ერთეულს.

როლების დამატება

1. *თანამედროვე სცენარის (Scenario Advanced) ტრაფარეტიდან (From the Scenario Advanced stencil) გადაათრიე Role (როლი) ფორმა და ჩააგდეთ რესურსების გვერდზე.*
2. *შემკვეთის თვისების რედაქტორში (in the Custom Property Editor) როლის სახელად შეიყვანე “Servers + Software” (“სერვერები+პროგრამული უზრუნველყოფა”).*
3. დააწკაპუნე OK.
4. გაიმეორე 1-3 ბიჯები ორი ახალი როლების შესაქმნელად სახელებით “HR Manager” (“ადამიანური რესურსების მენეჯერი”) და “Administrative Assistant” (“ადმინისტრაციული თანამშემწე”).

რესურსების დამატება

იმის შემდეგ რაც შევქმენით სამი ახალი როლი, საჭიროა რესურსების დამატება და მათი არსებული ტიპებისათვის მინიჭება (assign them to the existing types).

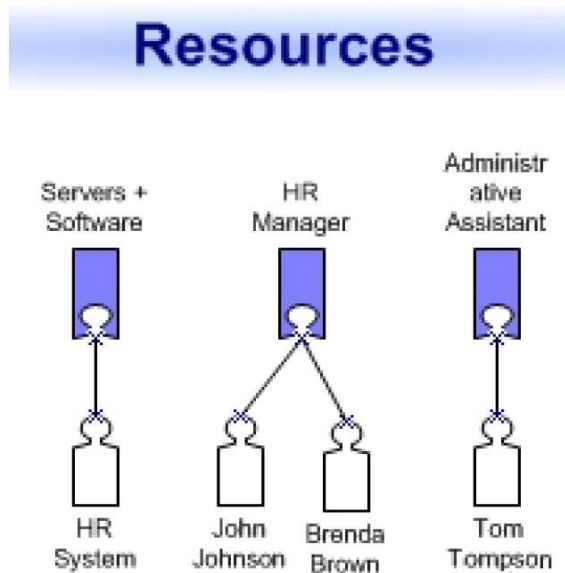
1. *სცენარის ტრაფარეტიდან (From the Scenario Advanced stencil) გადაათრიე **FS resource (FS რესურსის)** ფორმა და ჩააგდე ის რესურსების გვერდზე.*
2. *შემკვეთის თვისების რედაქტორში (in the Custom Property Editor) რესურსის სახელად შეიყვანე “HR System” (“ადამიანური რესურსების სისტემა”).*
3. დააწკაპუნე ღილაკზე “...”, რომელიც არის Org ვეილს შემდეგ (next to the Org field). გაიხსნება დიალოგური ყუთი *Organization Selection (ორგანიზაციის არჩევა)*. შესაძლებელი ორგანიზაციების სიიდან შეარჩიე **Network Systems (ქსელური სისტემები)**. დააწკაპუნე **Assign Organization (მიანიჭე ორგანიზაცია)**.
4. დააწკაპუნე *Roles (როლები)* მინდვის შემდეგი **Assign (მიანიჭე)** ღილაკი. გაიხსნება *Assign Roles (განსაზღვრე როლების)* დიალოგური ყუთი.
5. შეარჩიე **Servers + Software (სერვერები+პროგრამული უზრუნველყოფა)** შესაძლო როლების სიიდან და დააწკაპუნე **Assign (განსაზღვრე)**. მომხმარებელს შეუძლია აირჩიოს ერთზე მეტი როლი თუ საჭიროა.
6. დააწკაპუნე OK. *ადამიანური რესურსების სისტემა (HR System resource)* დაკავშირებულია *სერვერები+პროგრამული უზრუნველყოფა* როლთან (*to the Servers + Software Role*) და *ქსელური სისტემების ორგანიზაციასთან (the Network Systems organization)*.

7. გაიმეორე 1-6 ბიჯები ქვემოთ მოცემული ცხრილის გამოყენებით ახალი რესურსების დასამატებლად.

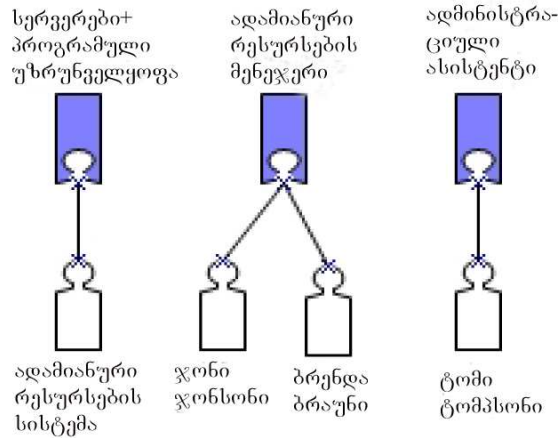
შენიშვნა: *მინიჭებული როლი (the Assigned Role)* და *მინიჭებული ორგანიზაცია (the Assigned Organization)* უნდა იყვნენ შეერთებული (დაკავშირებული) *რესურსის სახელთან (Resource Name)*.

რესურსის სახელი Resource Name	განსაზღვრული როლი Assigned Role	განსაზღვრული ორგანიზაცია Assigned Organization
ჯონი ჯონსონი John Johnson	ადამიანური რესურსების მენეჯერი HR Manager	გადაბირება Recruitment
ბრენდა ბრაუნი Brenda Brown	ადამიანური რესურსების მენეჯერი HR Manager	გადაბირება Recruitment
ტომი ტომპსონი Tom Tompson	ადმინისტრაციის ასისტენტი Administrative Assistant	ადმინისტრაცია Administration

თქვენი *რესურსების* გვერდი (*Resources page*) ეხლა უნდა ჰგავდეს ქვემოთ მოცემულ გრაფიკს. თქვენ შეიძლება მოაწესრიგოთ გვერდი შემდეგნაირად: 1) ავტოგანთავსების ფუნქცია (მარჯვნივ დააწკაპუნე და აირჩიე *Layout This Page (დაგეგმე ეს გვერდი)*), ან 2) განლაგების მოსაწესრიგებლად ფორმები გადაათრე ხელით.

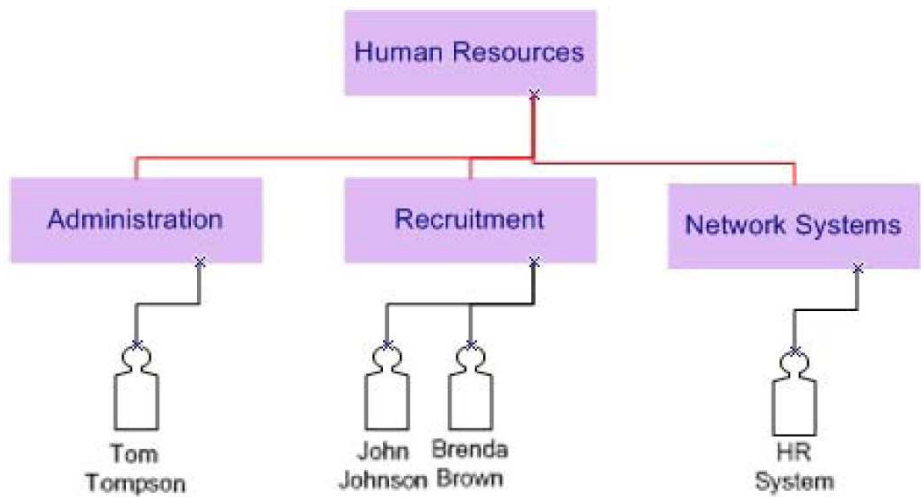


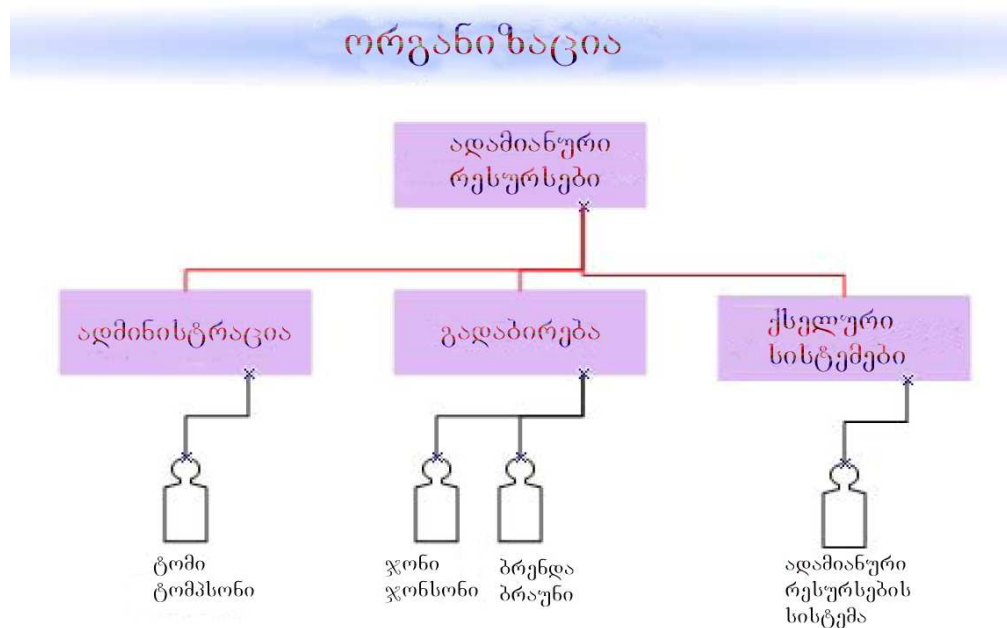
რესურსები



თქვენი ორგანიზაციის გვერდი (*Organization page*) ესლა უნდა შეიცავდეს ყველა ქვემოთ მოცემულ ფორმებს. თქვენ შეიძლება გინდოდეთ ავტო-განთავსების ფუნქციის გამოყენება (მარჯვნივ დააწკაპუნე და აირჩიე **დაგეგმე ეს გვერდი (Layout This Page)**), ან ფორმები გადაათრიე ხელით განლაგების მოსაწესრიგებლად:

Organization





ტელეფონის ნომრის და ი-მეილის დამატება რესურსზე

თვენ შეგიძლიათ შეიყვანოთ საკონტაქტო ინფორმაცია დაკავშირებული თქვენ რესურსებთან, როგორებიცაა ტელეფონის ნომერი და ი-მეილის მისამართი.

ეს შეიძლება გაკეთდეს *შემკვეთის თვისების რედაქტორის* (the *Custom Property Editor*) ფანჯარაში (ორჯერ დააწკაპუნეთ FS resource-ზე), შემდეგ გადადით ტელეფონის ნომრის და ი-მეილის მინდვრებში.

შეიყვანეთ ტელეფონის ნომერი და ი-მეილი. შეინახეთ თქვენი “დაყენებები” (settings) Apply-ს შერჩევით და ფანჯარის დახურვით.

ლაბორატორიული სამუშაო №3

პროცესის იერარქიის აგება

სახელმძღვანელოს ამ ნაწილში ჩვენ შევისწავლით პროცესის იერარქიის აგებას, რომელიც შედგება ბიზნეს პროცესისა და BPMN-ს ფორმებისაგან. ასევე შევისწავლით მათი ერთმანეთთან შეერთებით პროცესის ლოგიკური ნაკადის აგებას. აგრეთვე შევისწავლით პროცესებისა და BPMN-ს ფორმებისათვის გარკვეული თვისებების დამატებას ისე, რომ სამუშაოს ნაკადი იყოს ნათელი და ადვილი გასაგები.

Hire New Employees



ახალი თანამშრომლების დაქირავება



პროცესის იერარქიის მიმოხილვა

ახალი თანამშრომლების დაქირავების პროცესი გვიჩვენებს ახალი თანამშრომლის დაქირავებისათვის საჭირო მოქმედებებს. ბიზნეს პროცესის მოდელში, ყოველი ნაბიჯი (განსაზღვრავენ სამუშაოს აღწერის ადგილმდებარეობას, ამოწმებს სიზუსტეს, აცხადებს შიდად/გარეთად, დებულობს განცხადებას, ირჩევს კანდიდატურებს, იღებს ინტერვიუებს კანდიდატებისაგან, სთავაზობს თანამდებობას) წარმოდგენილია BPMN-ს სპეციფიური ფორმით.

დიდ მოდელში, ჩვეულებრივ უფრო მოხერხებულია BPMN-ს ფორმების ლოგიკური დაჯგუფება ცვლადი ქვე-პროცესების მიმდევრობად (სერიად). ეს იძლევა შესაძლებლობას, რომ პროცესი წარმოდგენილი იყოს მაღალ დონეზე სწრაფი ან დაწვრილებით განხილვისათვის.

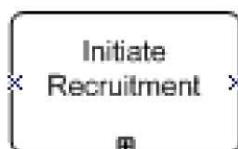
პროცესის ნაკადის უმაღლესი დონის შექმნა

განსახილველი დაქირავების პროცესი შეიცავს სამ ქვე-პროცესს სახელებით “Develop Job Description” (“სამუშაოს აღწერის დამუშავება”), “Initiate Recruitment” (“გადაბირების დაწყება”) და “Interview Candidates” (“ინტერვიუს აღება ცანდიდატებისაგან”).

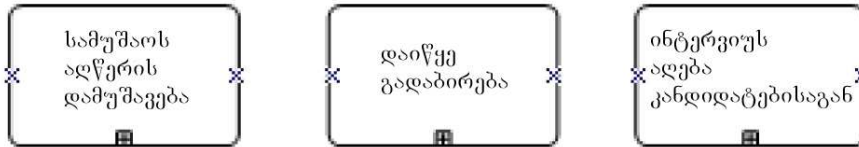
უმაღლესი დონის ფორმების შექმნა

1. გადადი *სცენარის* გვერდზე (Go to the *Scenario* page).
2. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) გადაათრეე პროცესის ფორმა (**Process** shape) სცენარის გვერდზე.
3. როდესაც *შემკვეთის თვისების რედაქტორი* (the *Custom Property Editor*) გაიხსნება, შეიყვანე “Hire New Employees” (“დაიქირავე ახალი თანამშრომლები”) როგორც პროცესის სახელი.
4. დააწკაპუნე **OK**. პროცესისათვის შექმნილია ცალკე გვერდი და პროცესის ფორმა სახელით *Hire New Employees* (*დაიქირავე ახალი თანამშრომლები*) იქნება დამატებული *სცენარის* გვერდზე.
5. ამ პროცესისათვის შექმნილი ახალი გვერდის სანახავად თაგუნას მარჯვენა ღილაკი დააწკაპუნე *Hire New Employees*-ზე (*დაიქირავე ახალი თანამშრომლები*).
6. აირჩიე გადადი პროცესზე (**Go To Process**). თქვენ გადახვალთ ცარიელ გვერდზე სახელით *Hire New Employees*-ზე (*დაიქირავე ახალი თანამშრომლები*). ამ გვერდს გამოვიყენებთ პროცესის ნაკადის დანარჩენი ნაწილის გამოსახვაზე.
7. *Hire New Employee* (*დაიქირავე ახალი თანამშრომლები*) გვერდს დაამატე სამი პროცესი. შეიყვანე სახელები “Develop Job Description” (“დაამუშავე სამუშაოს აღწერა”), “Initiate Recruitment” (“დაიწყე გადაბირება”) და “Interview Candidates” (“კანდიდატების ინტერვიურება”) შესაბამისად ამ ახალი პროცესებისათვის.
Hire New Employees (*დაიქირავე ახალი თანამშრომლები*) გვერდს ეხლა ექნება შემდეგი სახე:

Hire New Employees



ახალი თანამშრომლების დაქირავება



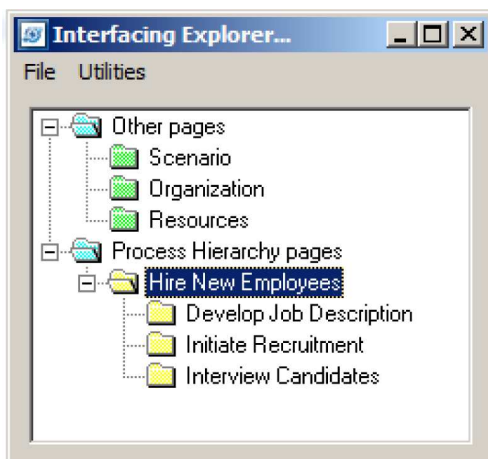
პროცესის გვერდებს შორის გადაადგილება

1. მარჯვენა ღილაკი დააწკაპუნე *Hire New Employee*-ის (დაიქირავე ახალი თანამშრომლები) გვერდზე და აირჩიე **Go To Parent** (გადადი მშობელზე). ეს გადაგიყვანთ *მშობელი*-ს გვერდზე (to the Parent page), რომელიც შეიცავს *Hire New Employee*-ის (დაიქირავე ახალი თანამშრომლები) ფორმას.
2. დაბრუნდით *Hire New Employee*-ის (დაიქირავე ახალი თანამშრომლები) გვერდზე **Hire New Employees**-ის (დაიქირავე ახალი თანამშრომლები) ფორმაზე ორჯერ დააწკაპუნებით.

ლაბორატორიული სამუშაო №4

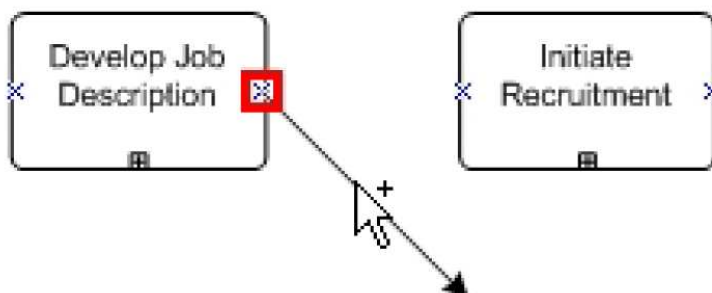
უმაღლესი დონის შეერთებების ჩასმა

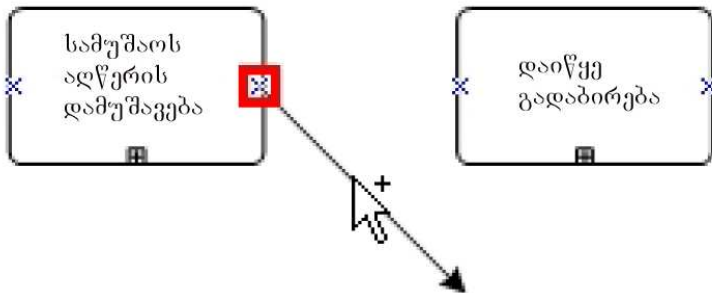
Interfacing menu-დან (*შეერთებების* მენიუ) აირჩიე **Interfacing Explorer** (*შეერთებების მკვლევარი*). როდესაც **Interfacing Explorer** ფანჯარა გაიხსნება, დაინახავთ პროცესის იერარქიას, როგორც ნაჩვენებია ქვემოთ.



პროცესებს შორის შეერთებებს (კავშირებს) ხშირად ეწოდება “პროცესებს შორის შეერთებები (კავშირები)” (“Inter-Process Links”), რომლებიც უნდა წარმოიქმნან განსაზღვრული ბიჯიდან საწყისი პროცესის შიგნით და დაუკავშირდნენ (შეუერთდეს) განსაზღვრულ ბიჯს დანიშნულების პროცესის შიგნით. რამდენადაც ჯერ კიდევ არ აგვიგია პროცესის შემადგენლობა ჩვენი ქვე-პროცესების გარეშე, ჯერ-ჯერობით უბრალოდ დავაკავშიროთ პროცესის ფორმები უმაღლეს დონეზე.

1. *Interfacing Explorer*-ში ორჯერ დააწკაპუნე **Hire New Employees**.
2. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) გადაათრიე **FS process link**-ის ფორმა *Hire New Employees*-ს გვერდზე და მიადგი (მიამაგრე) *Develop Job Description* (დაამუშავე სამუშაოს აღწერა) ფორმაზე მარცხნიდან მარჯვენა მიმართულებით (თავუნას განთავისუფლების გარეშე). შემდეგ გაანთავისუფლე თავუნა.





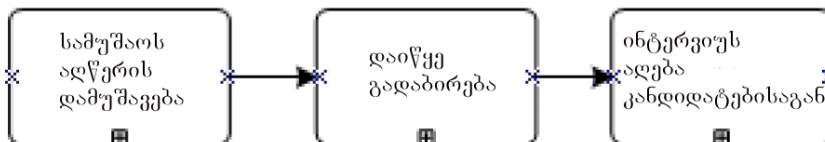
3. გამოვა შეერთება *Custom Property Editor* (*შემკვეთის თვისების რედაქტორი*). ჯერ-ჯერობით არაფერს შეცვალთ. დააწკაპუნე OK. დაიხურება ფანჯარა. ქვემოთ მოგვიანებით დავამატებთ *მასალებს (Materials)*.
4. მიამაგრე შეერთების დანიშნულების ისრის მარჯვნივ ბოლო *Initiate Recruitment* (*დაიწყო გადაბირების*) პროცესზე.
5. გაიმეორე 2 და 3 ბიჯები *Initiate Recruitment* (*დაიწყო გადაბირების*) პროცესის *Interview Candidates* (*ინტერვიუს აღება კანდიდატებისაგან*) პროცესთან შესაერთებლად.

Hire New Employees-ის გვერდს ეხლა უნდა ჰქონდეს დაახლოებით შემდეგი სახე:

Hire New Employees



ახალი თანამშრომლების დაქირავება



პროცესის ნაკადის გაწვევტა – სამუშაოს აღწერის დამუშავება (Develop Job Description)

პროცესის დამთავრების შემდეგ ნაკადი მოითხოვს გაწვევტოს ყოველი პროცესი სპეციალური ბიჯების სერიის საშუალებით. პირველად, განვიხილავთ პროცესს *Develop Job Description* (დაამუშავე სამუშაოს აღწერა). ვთქვათ გვინდა ამ პროცესში სამი მოქმედების მოდელირება: locating the job description (სამუშაოს აღწერის განთავსება), evaluating the job description (სამუშაოს აღწერის შეფასება) და modifying the job description (სამუშაოს აღწერის მოდიფიცირება). ეს მოქმედებები არიან წარმოდგენილი ამოცანებით.

პროცესი იწყება მაშინ, როდესაც Human Resources (ადამიანური რესურსები) მიიღებს ახალი თანამშრომლის შესახებ მოთხოვნას სხვა დეპარტამენტიდან. ეს მოთხოვნა ხდება გამშვები მექანიზმი, რომელიც იწყებს ყველა შემდეგ მოქმედებებს. ბიზნეს პროცესის მოდელში, დაწყების წერტილი ჩვეულებრივ არის წარმოდგენილი დაწყების ხდომილებით (start event). დაწყების ხდომილება (start event) გამოიყენება, როდესაც საჭიროა ახალი ინფორმაციის შეყვანის შემოტანა (to introduce new input), რომელიც სათავეს იღებს მოდელის გარეთ (სხვა დეპარტამენტიდან ჩვენ შემთხვევაში). დაწყების ხდომილებები, სხვა სიტყვებით, წარმოადგენენ შემომავალ მონაცემებს ან მოვლენებს, რომლებიც იწვევენ სამუშაოს ნაკადს პროცესის ფარგლებში.

ჩვენ მოდელში, ჩვენ შევქმნით დაწყების ხდომილებას სახელით “Receive notification” (“მიიღე შეტყობინება”), რომელიც უზრუნველყოფს ინფორმაციის შეყვანას, რასაც მოქმედებაში მოჰყავს მოდელის ყველა სხვა ამოცანა. შესაბამისად, დაწყების ხდომილებას არ შეუძლია მიიღოს შემომავალი შეერთებები მოდელის შიგნით სხვა მოქმედებებიდან.

მოგვეყავს შემდგომი გაგრძელება.

1. “Hire New Employees” (“დაიქირავე ახალი თანამშრომლები”) გვერდზე დააწკაპუნე მარჯვენა ღილაკი **Develop Job Description** (დაამუშავე სამუშაოს აღწერა) პროცესის ფორმაზე და ამოირჩიე **Go To Process** (გადადი პროცესზე). ეს გადაგვიყვანს ცარიელ ფურცელზე სახელწოდებით *Develop Job Description* (დაამუშავე სამუშაოს აღწერა).
2. BPMN-ს ტრაფარეტიდან (From the BPMN stencil) გადაათრიე ფორმა **დაწყება** (the **Start** shape) გვერდზე.
3. როდესაც *შემკვეთის თვისების რედაქტორი* (the *Custom Property Editor*) გაიხსნება, დაბეჭდე “Receive notification” (“მიიღე შეტყობინება”) როგორც ხდომილების სახელი. დააწკაპუნე OK. ფორმა დაემატება გვერდს.
4. მარჯვენა ღილაკი დააწკაპუნე ფორმაზე და შეარჩიე **Message** (შეტყობინება) დაწყება ხდომილების ტიპის მისათითებლად. *შეტყობინების* ტიპი (The *Message type*) მიუთითებს, რომ ზოგიერთი შეტყობინება იცვლება. ამ შემთხვევაში შეტყობინება არის სხვა დეპარტამენტიდან ახალი თანამშრომლის მოთხოვნა.

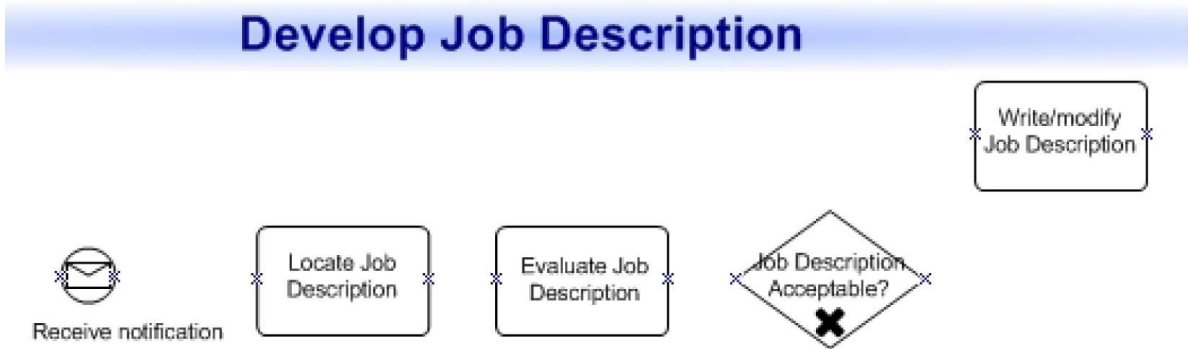
როდესაც საჭიროა წარმოვადგინოთ მოქმედება, რომელიც სრულდება ბიზნეს პროცესის მოდელში, ჩვენ ვიყენებთ ამოცანის ფორმას.

1. BPMN-ს ტრაფარეტიდან (From the *BPMN stencil*) გადაათრიე **Task** (**ამოცანა**) ფორმა და ჩააგდე გვერდში (onto the page).
2. როდესაც *შემკვეთის თვისების რედაქტორი* (the *Custom Property Editor*) გაიხსნება, შეიყვანე “Locate Job Description” (“განათავსე სამუშაოს აღწერა”) როგორც ამოცანის სახელი. დააწკაპუნე OK. ფორმა დაემატება გვერდს. შექმენი კიდევ ორი ამოცანა სახელებით “Evaluate Job Description” (“შეაფასე სამუშაოს აღწერა”) და “Write/Modify Job Description” (“დაწერე/მოდIFIციირება გაუკეთე სამუშაოს აღწერას”).

შემდეგი მოქმედება, რომლის მოდელირებაც ჩვენ გვინდა, არის სამუშაოს აღწერის ფაქტიური შეფასება. თუ სამუშაოს აღწერა არის ზუსტი, მაშინ ის შეიძლება სახალხოდ გამოცხადდეს. თუ არა, უნდა მოხდეს მისი მოდიფიცირება და გადაფასება. ეს წარმოადგენს გადაწყვეტილების კლასიკურ ნიმუშს, რომელიც ზოგჯერ მოიხსენიება როგორც **განსაკუთრებული მონაცემებზე დაფუძნებული გასასვლელი (Exclusive Data-Based Gateway)**, ან XOR (Xან) პირობა. ეს პირობა ქმნის გახლჩქმას სამუშაოს ნაკადში ურთიერთ გამომრიცხავი შედეგებით (with mutually exclusive outcomes).

1. BPMN-ს ტრაფარეტიდან (From the *BPMN stencil*) გადაათრიე ფორმა **Gateway** (**გასასვლელი**) და ჩააგდე ის გვერდში (onto the page).
2. როდესაც *შემკვეთის თვისების რედაქტორი* (the *Custom Property Editor*) გაიხსნება, დაბეჭდე “Job Description Acceptable?” (“სამუშაოს აღწერა მისაღებია?”) როგორც ფორმის სახელი. დააწკაპუნე OK. გასასვლელი (The gateway) დაემატება გვერდს.
3. მარჯვენა ღილაკი დააწკაპუნე ფორმაზე და შეარჩიე **Xor** გასასვლელის ტიპის მისათითებლად.

როდესაც დაამთავრებთ, თქვენი *დაამუშავე სამუშაოს აღწერა* (*Develop Job Description*) გვერდი დაახლოებით ასე უნდა გამოიყურებოდეს:



სამუშაოს აღწერის დამუშავება



BPMN–ს ფორმების შეერთება (Link the BPMN Shapes)

ეხლა, როდესაც ჩვენი პროცესის ყველა ბიჯი არის შექმნილი, ჩვენ უნდა შევაერთოთ ობიექტები ერთმანეთთან იმისათვის, რომ დავადგინოთ სამუშაოს ნაკადი ამ ბიჯებს შორის.

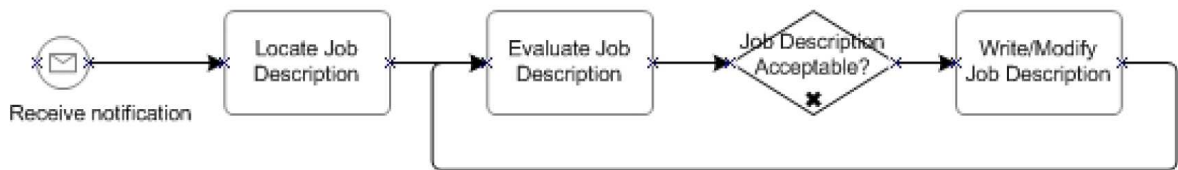
1. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) შეარჩიე ფორმა **FS process link (FS პროცესის შეერთება)** და მარცხენა ბოლო მიამაგრე (მიადგი) დაწყების ხდომილების შეერთების წყაროს გამომავალ შემაერთებელს. შეერთების *შემკვეთის თვისების რედაქტორი (Custom Property Editor)* გამოჩნდება.
2. ჯერ-ჯერობით არაფერს არ შეეცვლით. დააწკაპუნე OK ფანჯრის დასახურად. ქვემოთ მოგვიანებით დაუმატებთ *მასალებს (Materials)*.
3. მიამაგრე (მიადგი) შეერთების დანიშნულების (მარჯვენა) ბოლო შეყვანის დამაკავშირებელს (შემაერთებელს) (to the input connector) *განათავსე სამუშაოს აღწერის (the Locate Job Description)* ამოცანაზე.

ქვემოთ მოყვანილი ცხრილის მონაცემების გამოყენებით, თქვენ ეხლა შეგიძლიათ ჩასვათ (შეიტანოთ) დარჩენილი მოთხოვნილი შეერთებები (კავშირები). შევნიშნოთ, რომ შეერთების წყარო არის ფორმა რომლიდანაც შეერთება იწყება. შეერთების დანიშნულება არის ფორმა, რომელზედაც ისრის თავი არის მიდგმული.

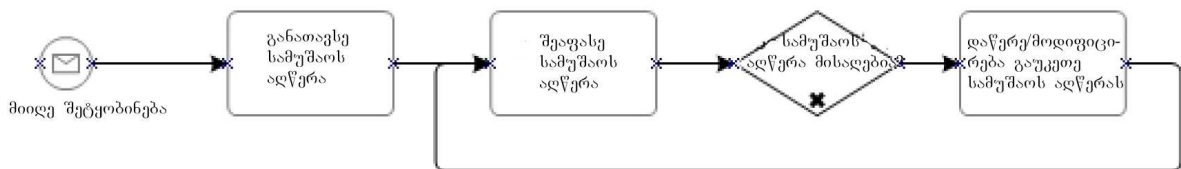
წყარო Source	დანიშნულება Destination
ამოცანა “განათავსე სამუშაოს აღწერა” “Locate Job Description” task	ამოცანა “შეაფასე სამუშაოს აღწერა” “Evaluate Job Description” task
ამოცანა “შეაფასე სამუშაოს აღწერა” “Evaluate Job Description” task	გასასვლელი “სამუშაოს აღწერა მისაღებია?” “Job Description Acceptable?” gateway
გასასვლელი “სამუშაოს აღწერა მისაღებია?” “Job Description Acceptable?” gateway	ამოცანა “დაწერე/მოდდიფიცირება გაუკეთე სამუშაოს აღწერას” “Write/Modify Job Description” task
ამოცანა “დაწერე/მოდდიფიცირება გაუკეთე სამუშაოს აღწერას” “Write/Modify Job Description” task	ამოცანა “შეაფასე სამუშაოს აღწერა” “Evaluate Job Description” task

როდესაც დაამთავრებთ, შეიძლება მოისურვოთ განლაგების მოწესრიგება. მარჯვენა ღილაკი დააწკაპუნეთ გვერდის ცარიელ ადგილას და აირჩიეთ **Layout this page** (დაგეგმე ეს გვერდი). თქვენი პროცესი ეხლა უნდა გამოიყურებოდეს ასე:

Develop Job Description



სამუშაოს აღწერის დამუშავება



ლაბორატორიული სამუშაო №5

პროცესის ნაკადის გაწყვეტა – გადაბირების დაწყება (Break Down the Process Flow - Initiate Recruitment)

ამ პუნქტში დაიწყო გადაბირების პროცესისათვის (Initiate Recruitment process) ჩვენ დავამატებთ ბიჯებს. *Initiate Recruitment* (დაიწყო გადაბირება) გვერდზე გადასასვლელად მენიუდან აირჩიე ოფციები **Edit > Go To > Initiate Recruitment**.

დავუშვათ, რომ ამ პროცესის მოდელირება გადავწყვიტეთ შემდგენიარად: მდივანი (ტომი ტომპსონი) ღებულობს სამუშაოს აღწერას. ის ავრცელებს განცხადებას კომპანიის თანამშრომლებს შორის და აგზავნის აღწერას კომპანიის გარეთ, სამუშაოს საძებნ ვებ გვერდებზე. სამუშაოზე მიღების შემოსული განცხადებებიდან კვალიფიციური კანდიდატები შეირჩევიან და დაეგზავნებათ ინტერვიუს ჩატარების გრაფიკი.

1. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) გადაათრე სამი ამოცანის ფორმა *Initiate Recruitment* (დაიწყო გადაბირება) გვერდზე და დაარქვი მათ სახელები “Distribute Internally” (“გაანაწილე შიგნით”), “Distribute Externally” (“გაანაწილე გარეთ”) და “Select Candidates for Interview” (“შეარჩიე კანდიდატურები ინტერვიუსათვის”).

როგორც კი სამუშაოს აღწერა გაიგზავნება, სამუშაოს მაძიებლები გამოაგზავნიან თავიანთ სამუშაოზე მიღების განცხადებებს კომპანიაში. რამდენადაც სამუშაოზე მიღების განცხადებების მიღების აქტი არის უფრო მეტად ხდომილება, ვიდრე ამოცანა, ჩვენ ამას გაუკეთებთ მოდელირებას შეტყობინების (Message) ტიპის შუალედური ხდომილების (an Intermediate Event) გამოყენებით.

1. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) გადაათრე **Intermediate** (შუალედური) ფორმა *Initiate Recruitment* (დაიწყო გადაბირების) გვერდზე.
2. როდესაც *შემკვეთის თვისების რედაქტორი* (the *Custom Property Editor*) გაიხსნება, შეიყვანე “Receive Job Applications” (“მიიღე სამუშაოზე მიღების განცხადებები”) როგორც ახალი ფორმის სახელი. დააწკაპუნე OK.
3. მარჯვენა ღილაკი დააწკაპუნე ხდომილებაზე და შეარჩიე **Message** (შეტყობინება) შუალედური ხდომილების ტიპის საჩვენებლად.

როდესაც მდივანი აგზავნის სამუშაოს აღწერას, მას შეუძლია ან არ შეუძლია გააგზავნოს ის ორივეგან კომპანიის შიგნითაც და გარეთაც, თუმცა ამ ამოცანებიდან მინიმუმ ერთი მაიც არის შესრულებული ნორმალური გადაბირების პროცესის დროს. სამუშაოს ნაკადის ეს გახლეჩვა შეიძლება წარმოდგენილი იყოს გასასვლელის სხვადასხვა ტიპით; მაგალითად, შემცველი გასასვლელი (Inclusive Gateway), რომელსაც ასევე ჰქვია ან (OR) პირობა. ამ გზით ჩვენ ვაჩვენებთ, რომ სამუშაო შესაძლებელია იყოს გაგზავნილი კომპანიის გარეთ, შიგნით ან ორივეგან.

1. BPMN-ს ტრაფარეტიდან (From the *BPMN* stencil) გადაათრე **Gateway** (გასასვლელი) ფორმა *Initiate Recruitment* (დაიწყო გადაბირება) გვერდზე.

- როდესაც *შემკვეთის თვისების რედაქტორი* (the Custom Property Editor) გაიხსნება, შეიყვანე “Distribute JD Internally and/or Externally” (“გაანაწილე სამუშაოს აღწერა შიგნით და/ან გარეთ”) როგორც ახალი გასასვლელის სახელი. დააწკაპუნე OK.
- მარჯვენა ღილაკი დააწკაპუნე ფორმაზე და აირჩიე **Or (ან)** გასასვლელის ტიპის მისათითებლად.

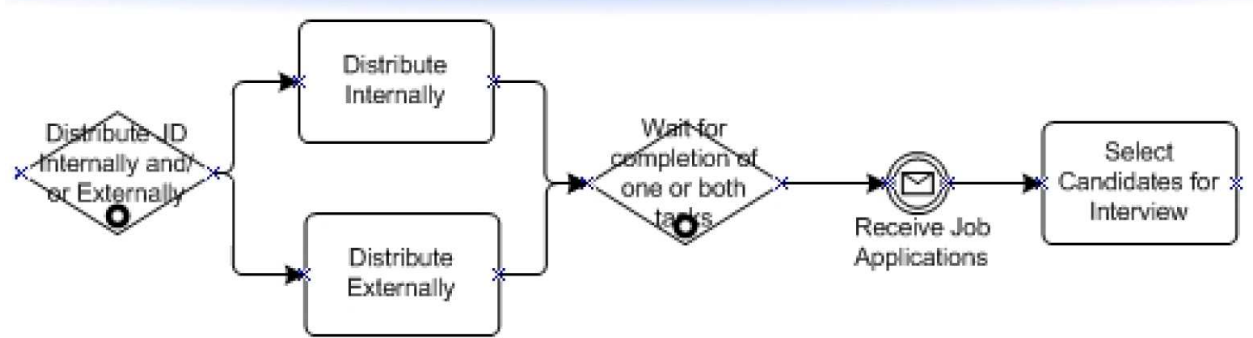
აღსანიშნავია ორი მომენტი:

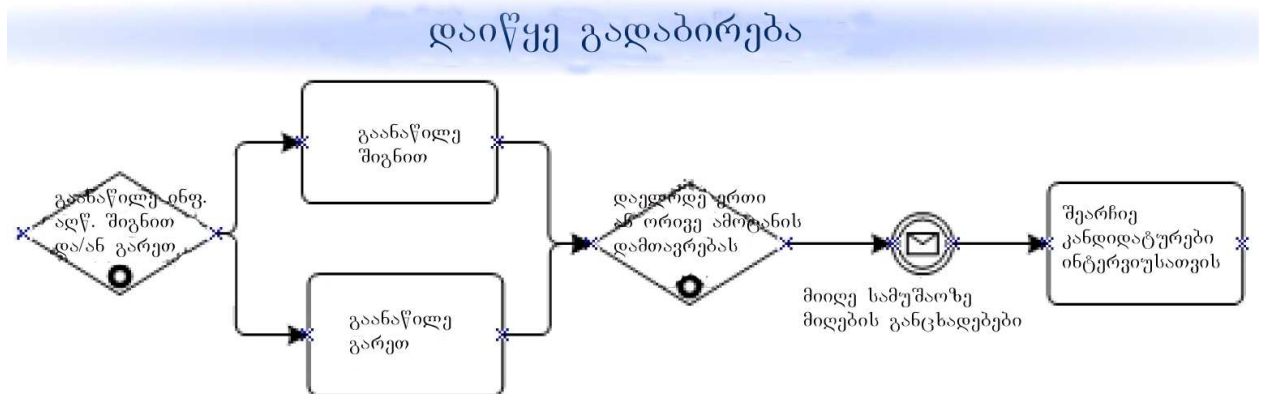
- აქამდე ჩვენ ვქმნიდით პროცესის მოდელს მიმდევრობით. ეს ნიშნავს, რომ ერთი მოქმედება მთავრდება სანამ მეორე დაიწყება. სამუშაოს აღწერის გაგზავნის ამოცანის გაყოფით შიდა და გარეთად, ჩვენ შემოვიტანეთ შესაძლო პარალელური მოქმედებების მაგალითი.
- ბიზნეს პროცესის მოდელში, როდესაც სამუშაოს ნაკადი იყოფა პარალელურ მოქმედებებს შორის, ჩვენ ასევე უნდა განვსაზღვროთ, უნდა კვლავ შეერწყას ნაკადი თუ არა. გასასვლელები ასევე გამოიყენება სამუშაო ნაკადების შესარწყმელად სანამ ისინი არიან იგივე ტიპის, რომელიც იყო გამოყენებული თავიდან ნაკადის გასახლეჩად.

ბოლო ბიჯი *Initiate Recruitment* (დაიწყე გადაბირება) ქვე-პროცესში არის სამუშაოს ნაკადის შერწყმა იმის შემდეგ, რაც სამუშაოს აღწერა იყო დაგზავნილი, სამუშაოზე მიღების განცხადებების მიღების წინ.

- BPMN-ს ტრაფარეტიდან (From the BPMN stencil) გადაათრიე **Gateway** (გასასვლელის) ფორმა *Initiate Recruitment* (დაიწყე გადაბირება) გვერდზე.
- როდესაც *შემკვეთის თვისების რედაქტორი* (the Custom Property Editor) გაიხსნება, შეიყვანე “Wait for completion of one or both tasks” (“დაელოდა ერთი ან ორივე ამოცანის დამთავრებას”), როგორც გასასვლელის (gateway) სახელი. დააწკაპუნე OK.
- მარჯვენა ღილაკი დააწკაპუნე ფორმაზე და შეარჩიე **Or (ან)** გასასვლელის (gateway) ტიპის საჩვენებლად.
- მიჰყევი ინსტრუქციას წინა გვერდიდან BPMN-ს ფორმების შესაერთებლად ქვემოთ მოცემული დიაგრამის შესაბამისად:

Initiate Recruitment





პროცესის ნაკადის გაწყვეტა – ინტერვიუს აღება კანდიდატებისაგან (Break Down the Process Flow - Interview Candidates)

გავაკეთოთ კიდევ ერთი პროცესის მოდელირება. შექმენით ქვემოთ ილუსტრირებული ფორმები შესაბამისი სახელებით და ტიპებით. გამოიყენეთ წინა ბიჯები, რომლებიც დაგეხმარებიან ფორმების შექმნაში.

შენიშვნა. ფორმების სახელებია: Conduct Interview (ჩაატარე ინტერვიუ), Accept candidate? (მიიღე კანდიდატი?), Offer Position to Candidate (შესთავაზე თანამდებობა კანდიდატს).



კანდიდატების ინტერვიუება

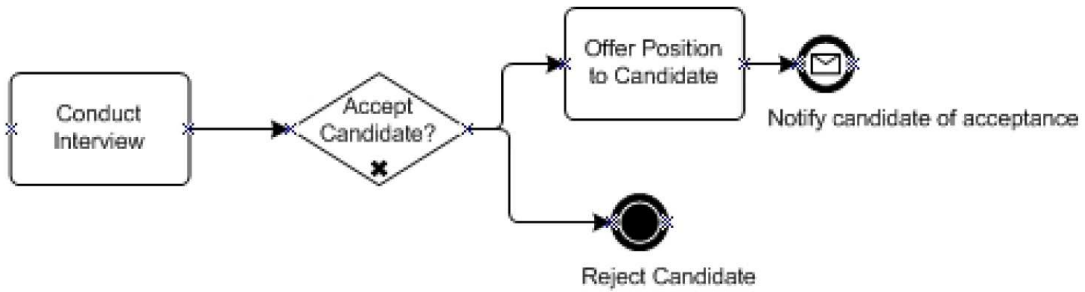


აქ არის ორი ბოლო ფორმა ამ პროცესზე დასამატებლად. თუ კანდიდატი მიღებულია ინტერვიუს შემდეგ, მაშინ დაქირავების მენეჯერი კანდიდატს შესთავაზებს პოზიციას (თანამდებობას). თუმცა, თუ კანდიდატი ამ სამუშაოსათვის შეუფერებელია, მაშინ პროცესი დამთავრდება დაუყოვნებლად. ეს მოვლენები წარმოდგენილი იქნებიან დამთავრება ხდომილებების საშუალებით.

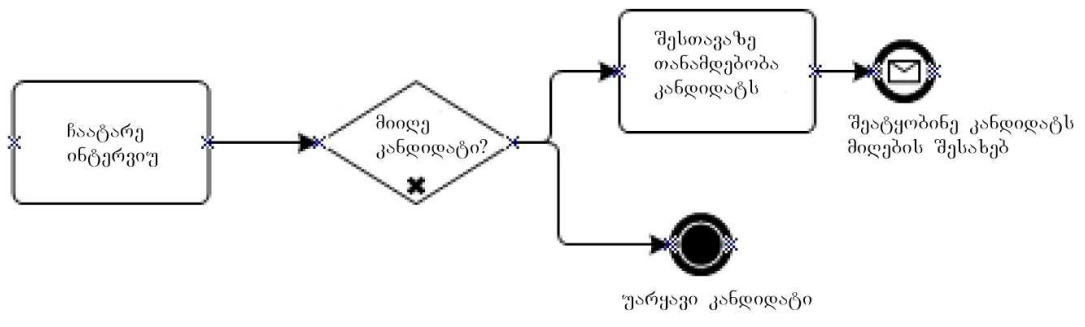
დამთავრება ხდომილებები გამოიყენება სამუშაოს კონკრეტული ნაკადის საბოლოო შეწყვეტის აღსანიშნავად. ჩვენ მოდელში, დამთავრების ხდომილება მიუთითებს ახალი თანამშრომლების დაქირავების პროცესის დამთავრებას. *შეტყობინების (Message)* ტიპის დამთავრების ფორმა წარმოადგენს მიღების შეტყობინებას, რომელიც კლიენტისათვის არის გაგზავნილი. *შეწყვეტის (Terminate)* ტიპის დამთავრება ფორმა წარმოადგენს კანდიდატის აურყოფას, რადგან დაქირავების პროცესი არ გაგრძელდება ამ წერტილის შემდეგ.

1. შექმენი ორი **End (დამთავრება)** ფორმა სახელებით “Notify candidate of acceptance” (“შეტყობინე კანდიდატს მიღების შესახებ”) და “Reject Candidate” (“უარყავი კანდიდატი”).
2. მარჯვენა ღილაკი დააწკაპუნე “უარყავი კანდიდატი” (“Reject Candidate”) ფორმაზე და შეარჩიე **Terminate (შეწყვეტა)** ხდომილების ტიპის მისათითებლად. შეარჩიე **Message (შეტყობინება)** ტიპი დარჩენილი დამთავრების (end) ფორმისათვის.
3. მიჰყევი ინსტრუქციებს წინა გვერდიდან BPMN-ს ფორმების შესაერთებლად ქვემოთ მოცემული დიაგრამის შესაბამისად:

Interview Candidates



კანდიდატების ინტერვიუება



ჩვენი მოდელისათვის ჩვენ ესლა გვაქვს დამთავრებული პროცესების გაწყვეტა (the breakdown of processes).

ლაბორატორიული სამუშაო №6

შეტანის და გამოტანის პროცესების მიმოხილვა (Review Process Inputs and Outputs)

პროცესის სქემაზე დატანის დასამთავრებლად ჩვენ გვჭირდება სამუშაოს ნაკადის წარმოდგენა მოდელის საშუალებით. ამისათვის, ჩვენ გამოვიყენებთ *მასალებს (Materials)*, რომლებიც წარმოადგენენ დოკუმენტებს, მონაცემების ნაწილს ან ობიექტებს, გამოყენებულს სამუშაოს ნაკადის ფარგლებში. ისინი ასევე შეიძლება იყვნენ მოხსენიებული, როგორც BPMN-ს ფორმების შეტანები და/ან გამოტანები (the inputs and/or outputs of BPMN shapes).

BPMN-ის ფორმის ყოველი შეტანა ან გამოტანა (პროდუქცია) უნდა იყოს წარმოდგენილი პროცესის შეერთებაზე მიდგმული (attached) მასალის საშუალებით. შეერთების ამ ფორმას აქვს მიმართულება: ფორმასთან მისი მიდგმის გზა (განხორციელება) მიუთითებს ბიზნეს პროცესის მოდელში სამუშაოს ნაკადის მიმართულებაზე.

პირველად განვიხილოთ ჩვენ მოდელში სხვადასხვა პროცესების შეტანები ან გამოტანები (პროდუქციები). შემდეგი ცხრილი წარმოგვიდგენს ამას. (თქვენ შეიძლება დაგჭირდეთ ლაბორატორიულ სამუშაო №4-ში აღწერილი შემდეგი ბიჯის მიმოხილვა “გაწყვიტე პროცესის ნაკადი – დაამუშავე სამუშაოს აღწერა” (“Break down the process flow - Develop Job Description”).

პროცესი Process	შეტანა Input	გამოტანა Output
დაამუშავე სამუშაოს აღწერა Develop Job Description	მოთხოვნა თანამშრომლებზე Request for Employees	სამუშაოს აღწერა Job Description
გადაბირების დაწყება Initiate Recruitment	სამუშაოს აღწერა Job Description	სამუშაოზე მიღების განცხადებები Job Application
კანდიდატების ინტერვიუება Interview Candidates	სამუშაოზე მიღების განცხადებები Job Application	მიღების შეტყობინება Acceptance Notification

სამუშაოს მოდელი ჩვენ ნაკადში იწყება დაქირავების მოთხოვნის (HR) (Hire request) შეტყობინების მიღებით, ანუ რომელიმე დეპარტამენტი მოითხოვს ახალ თანამშრომელს და ამრიგად, შეგვიძლია ვთქვათ, რომ *თანამშრომლების მოთხოვნა (Request for Employees)* არის შეტანა (input) *დაამუშავე სამუშაოს აღწერა (Develop Job Description)* პროცესისათვის (ასევე *ახალი თანამშრომლების დაქირავებისათვის (Hire New Employees)*), და სამუშაოს აღწერა (job description) არის გამოტანა (პროდუქცია) (output) ამ პროცესის. სამუშაოს აღწერა (job description) აგრეთვე არის შეტანა (input) *დაიწყე გადაბირება (Initiate Recruitment)* პროცესისათვის, რომელიც ანაწილებს სამუშაოს აღწერებს და იღებს სამუშაოზე მიღების განცხადებებს. სამუშაოზე მიღების ეს განცხადებები არიან *დაიწყე გადაბირება (Initiate Recruitment)*

პროცესისათვის გამოტანა (პროდუქცია) (output) და შეტანა (input) პროცესისათვის კანდიდატების ინტერვიუება (Interview Candidates). საბოლოოდ, პროცესის ბოლოს (end), შეიძლება იყოს გაგზავნილი ორი შეტყობინება (messages), მიღების და უარყოფის შეტყობინებები. თუმცა, რადგან უარყოფის შეტყობინებას მიყვავართ შეწყვეტის (Terminate) დამთავრების ხდომილებასთან (end event) (რადგან პროცესი არ და ვერ გაგრძელდება ამ წერტილის შემდეგ), მხოლოდ პროცესის კანდიდატების ინტერვიუება (Interview Candidates) გამოტანა (პროდუქცია) (output) არის მისაღები შეტყობინება.

მასალების გვერდის შექმნა (Create the Materials Page)

1. გადადი სცენარის გვერდზე (Go to the Scenario page).
2. მარჯვენა ღილაკი დააწკაპუნე და **Create Material Page** (შეარჩიე მასალების გვერდი) გამოსული (pop-up) მენიუდან.
3. ახალი მასალების გვერდი (The new Materials page) იქნება ნახევნები.

მასალების გვერდზე მასალების დამატება (Add Materials to the Materials Page)

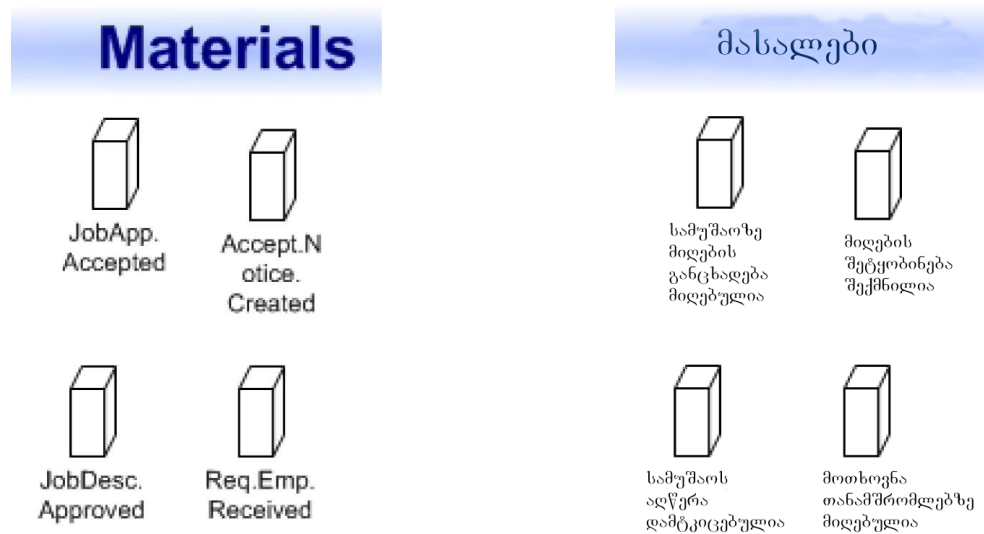
ერთი და იგივე მასალას შეიძლება ჰქონდეს მრავალი მდგომარეობა, რაც ნიშნავს, რომ ერთ მასალას შეიძლება ჰქონდეს მრავალჯერადი გადასვლები (ცვლილებები) (multiple transitions) პროცესის შიგნით. ჩვენ შევქმნით მასალებს, რომლებიც გამოიყენებიან ჩვენ პროცესში, მათ ყველა შესაძლო მდგომარეობაში.

1. სცენარის ტრაფარეტიდან (From the Scenario Advanced stencil) გადაათრიე **FS material (FS მასალა)** ფორმა და ჩააგდე მასალების გვერდში (onto the Materials page).
2. შეიტანე “JobDesc” (“სამუშაოს აღწერა”) სახელის ველში (in the Name field).
3. შეარჩიე **Form (ფორმა)** ტიპის ქვემოთ ჩამოშლადი სიიდან (Select **Form** from the Type drop-down list).
4. შეიტანე “Approved” (“დამტკიცებულია”) ველში State (მდგომარეობა) (Enter “Approved” in the State field).
5. დააწკაპუნე OK.
6. გაიმეორე 1-5 ბიჯები ქვემოთ მოცემული ცხრილის გამოყენებით ახალი მასალის ობიექტების შესაქმნელად (Repeat steps 1-5 using the table below to create new material objects).

მასალის სახელი Material Name	მასალის ტიპი Material Type	მასალის მდგომარეობა Material State
მოსთხოვნა თანამშრომლებზე Req.Emp	ფორმა (Form)	მიღებული (გამოგზავნილი) (Received)

სამუშაოზე განცხადებები Job App.	მიღების ფორმა (Form)	მიღებული (დათანხმებით) (Accepted)
მიღების შეტყობინება Accept Notice	ფორმა (Form)	შექმნილი (Created)

შენიშვნა: მასალის ტიპები არის შეზღუდული ფორმებით (*Forms*) და ნაწილებით (*Parts*). ამ სახელმძღვანელოს მიზნებისათვის, მასალები განხილულია როგორც ფორმები, რადგანაც ისინი აკეთებენ მითითებებს სხვადასხვა დოკუმენტებზე.

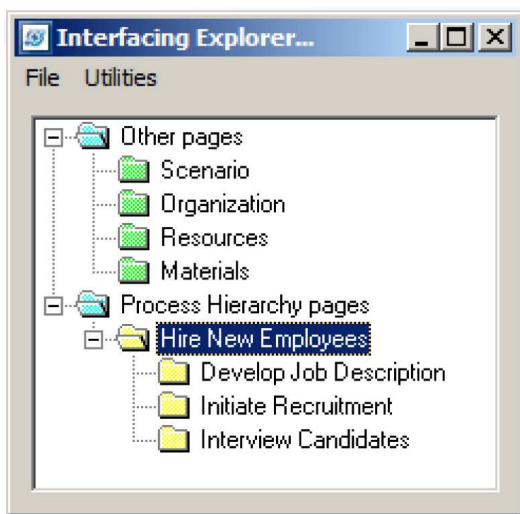


თუ მასალის მდგომარეობა არ არის ნაჩვენები, შეარჩიე **Interfacing** (ინტერფეისის გაკეთება) მთავარი ინსტრუმენტების პანელიდან და შემდეგ შეარჩიე **Interfacing Options... > Material State > Material State visible on material page** (ინტერფეისის გაკეთების ოფციები... > მასალის მდგომარეობა > მასალის მდგომარეობა ხილულია მასალის გვერდზე). შეარჩიე **Update State** (განაახლე მდგომარეობა) ცვლილებების განსახორციელებლად და დააწკაპუნე **OK**.

ლაბორატორიული სამუშაო №7

პროცესებს შორის შეერთებების (კავშირების) განსაზღვრა (Define Inter-Process Links)

ინტერფეისის გაკეთების მენიუდან (From the *Interfacing* menu) შეარჩიე **Interfacing Explorer** (ინტერფეისის გაკეთების მკვლევარი). როდესაც *Interfacing Explorer*-ის (ინტერფეისის გაკეთების მკვლევარის) ფანჯარა გაიხსნება, დაინახავთ პროცესის იერარქიას, როგორც ნახვენებია ქვემოთ.



პროცესებს შორის შეერთებებს (კავშირებს) ხშირად ეძახიან “პროცესებს შორის შეერთებებს (კავშირებს)” (“Inter-Process Links”), რომელიც უნდა დაიწყოს განსაზღვრული ბიჯიდან წყარო (საწყისი) პროცესის შიგნით (within the source process) და შეუერთდეს (დაუკავშირდეს) განსაზღვრულ ბიჯს დანიშნულების პროცესის საზღვრებში (within the destination process). ესლა განსაზღვროთ წყარო (საწყისი) და დანიშნულების ბიჯები ჩვენ პროცესებს შორის შეერთებებისათვის (კავშირებისათვის).

1. *Interfacing Explorer*-ში (ინტერფეისის გაკეთების მკვლევარში) ორჯერ დააწკაპუნე **Hire New Employees** (დაიქირავე ახალი თანამშრომლები) შესაბამის გვერდზე გადასასვლელად.
2. ორჯერ დააწკაპუნე პროცეს *Develop Job Description* (დაამუშავე სამუშაოს აღწერა) და პროცეს *Initiate Recruitment* (დაიწყე გადამირება) შორის შეერთებაზე (link).
3. შეერთება *Custom Property Editor* (შემკვეთის თვისების რედაქტორი) გამოჩნდება.
4. დააწკაპუნე **Assign** (მინიჭება). მასალის შერჩევის (*Material Selection*) ფანჯარა გამოვა.

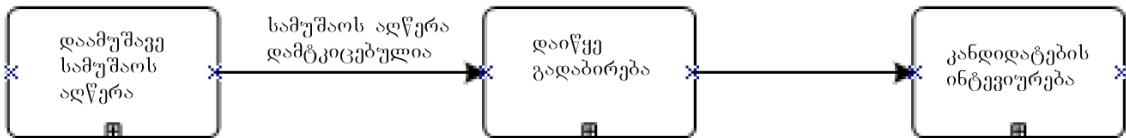
5. სიდან შეარჩიე **JobDesc; state: Approved** (სამუშაოს აღწერა; მდგომარეობა: დამტკიცდა) და დააწკაპუნე **Assign** (მინიჭება). დააწკაპუნე **Apply** (გამოიყენე) ცვლილებების განსახორციელებლად.
6. შეარჩიე ღილაკი **Select Source Low Link** (შეარჩიე წყაროს ქვედა შეერთება (კავშირი)).
7. როდესაც ფანჯარა *Source Link Selection* (წყაროს შეერთების შერჩევა) გაიხსნება, შეარჩიე გასასვლელი **Job Description Acceptable?** (სამუშაოს აღწერა მისაღებია?) და დააწკაპუნე **OK**.
8. ფანჯარა *Destination Link Selection* (დანიშნულების შეერთების შერჩევა) გაიხსნება ავტომატურად. შეარჩიე გასასვლელი **Distribute JD internally and/or Externally** (გაანაწილე ინფ. აღწ. შიგნით და/ან გარეთ) და დააწკაპუნე **OK**.
9. დააწკაპუნე **OK** შეერთების *Custom Property Editor* (შემკვეთის თვისების რედაქტორის) დასახურად.

გვერდი *Hire New Employees* (დაიქირავე ახალი თანამშრომლები) ეხლა დაახლოებით უნდა გამოიყურებოდეს ასე:

Hire New Employees



დაიქირავე ახალი თანამშრომლები

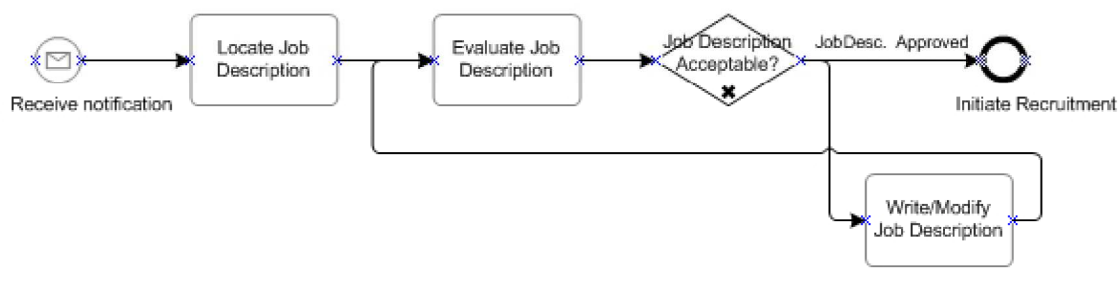


თუ მასალის მდგომარეობა არ არის ნაჩვენები, შეარჩიე **Interfacing** (ინტერფეისის გაკეთება) მთავარი ინსტრუმენტების პანელიდან და შემდეგ შეარჩიე **Interfacing Options... > Material State > Material State visible on links** (ინტერფეისის გაკეთების ოფციები...> მასალის მდგომარეობა > მასალის მდგომარეობა ხილული (რომელიც ჩანს) შეერთებებზე). შეარჩიე **Update State** (განაახლე მდგომარეობა) ცვლილებების შესასრულებლად და დააწკაპუნე **OK**.

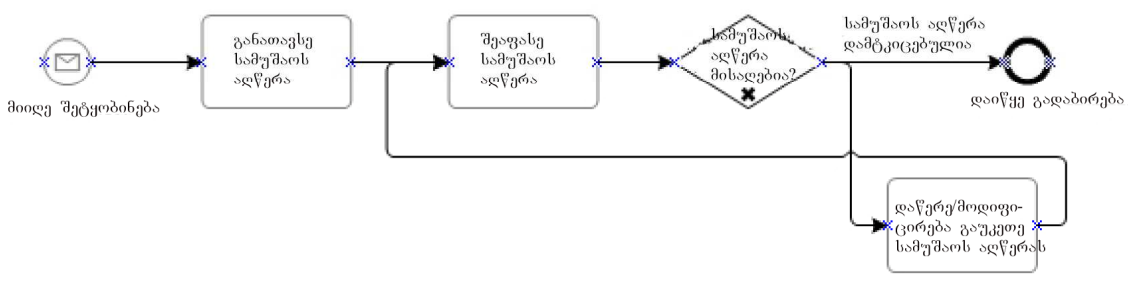
თუ თქვენ გახსნით *Develop Job Description* (დაამუშავე სამუშაოს აღწერა) გვერდს, დაინახავთ, რომ დაბალი დონის შეერთება (კავშირი) (low-level link) ავტომატურად იყო ჩასმული იგივე სახელით (და სხვა თვისებებით), როგორც ჩვენი უმაღლესი

დონის შეერთება (კავშირი). თქვენ შეგიძლიათ გადაათრიოთ დამთავრება ხდომილება (the end event) მათ სასურველ ადგილას თქვენ გვერდზე ან შეგიძლიათ დააწკაპუნოთ მარჯვენა ღილაკი და შეარჩიოთ *Layout This Page* (ლაგეგმე ეს გვერდი).

Develop Job Description

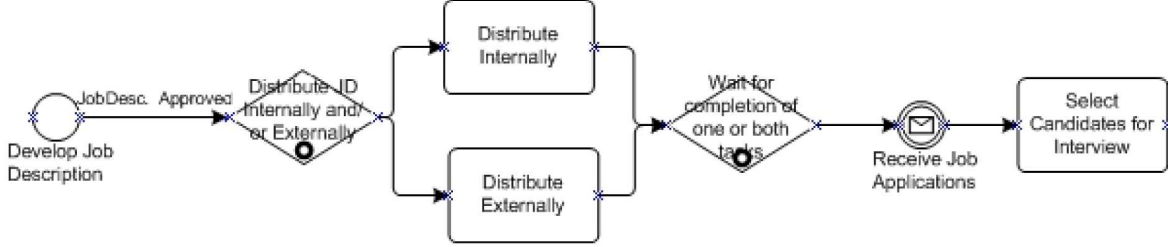


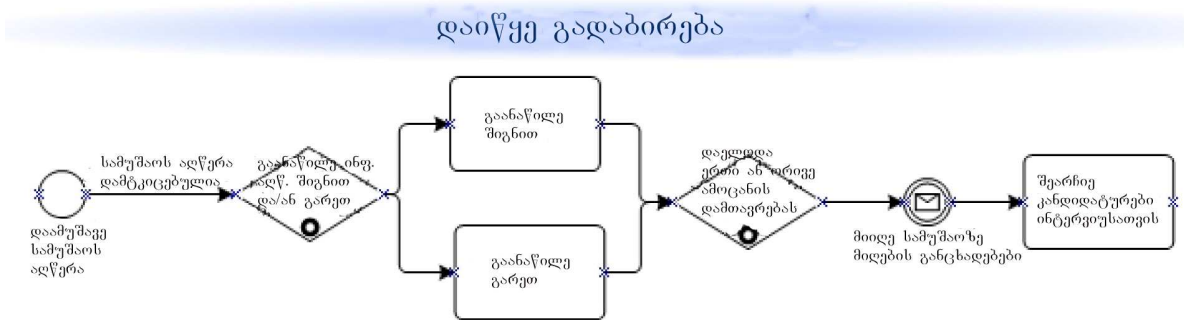
დაამუშავე სამუშაოს აღწერა



დაბალი დონის შეერთება (კავშირი) (A low-level link) აგრეთვე დამატებულია პროცესზე *Initiate Recruitment* (დაიწყო გადაბირება).

Initiate Recruitment





დარჩენილი შეერთებების (კავშირების) განსაზღვრა (Define the Remaining Links)

ეხლა ჩვენ გავაკეთებთ იგივე ბიჯებს რაც წინა გვერდზეა მოცემული პროცესების *Initiate Recruitment* (დაიწვე გადაბირება) და *Interview Candidates* (კანდიდატების ინტერვიუება) შესაერთებლად (დასაკავშირებლად).

1. *Hire New Employees* (დაიქირავე ახალი თანამშრომლები) გვერდზე ორჯერ დააწკაპუნე პროცესებს შორის *Initiate Recruitment* (დაიწვე გადაბირება) და *Interview Candidates* (კანდიდატების ინტერვიუება) შეერთება (კავშირი).
2. შეერთება *შემკვეთის თვისების რედაქტორი* (*Custom Property Editor*) გამოჩნდება.
3. დააწკაპუნე **Assign** (მინიჭება). *მასალის შერჩევის* (*Material Selection*) ფანჯარა გამოვა.
4. სიიდან შეარჩიე **JobApp; state: Accepted** (სამუშაოს აღწერა; მდგომარეობა: დამტკიცდა) და დააწკაპუნე **Assign** (მინიჭება). დააწკაპუნე **Apply** (გამოიყენე) ცვლილებების განსახორციელებლად.
5. შეარჩიე ლილაკი **Select Source Low Link** (შეარჩიე წყაროს დაბალი შეერთება (კავშირი)).
6. როდესაც ფანჯარა *Source Link Selection* (წყაროს შეერთების შერჩევა) გაიხსნება, შეარჩიე ამოცანა **Select Candidates for Interview** (შეარჩიე კანდიდატები ინტერვიუსათვის) და დააწკაპუნე **OK**.
7. ფანჯარა *Destination Link Selection* (დანიშნულების შეერთების შერჩევა) გაიხსნება ავტომატურად. შეარჩიე ამოცანა **Conduct Interview** (წაიყვანე ინტერვიუ) და დააწკაპუნე **OK**.
8. დააწკაპუნე **OK** შეერთების *Custom Property Editor* (*შემკვეთის თვისების რედაქტორის*) დასახურად.

Hire New Employees (დაიქირავე ახალი თანამშრომლები) გვერდს ეხლა ექნება შემდეგი სახე:

Hire New Employees

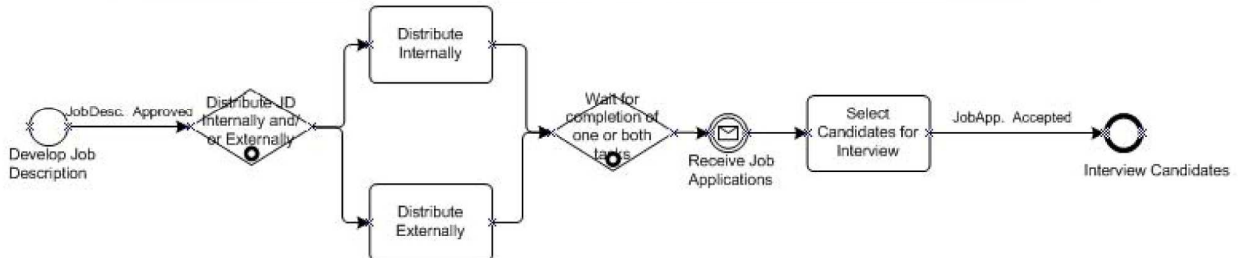


დაიქირავე ახალი თანამშრომლები

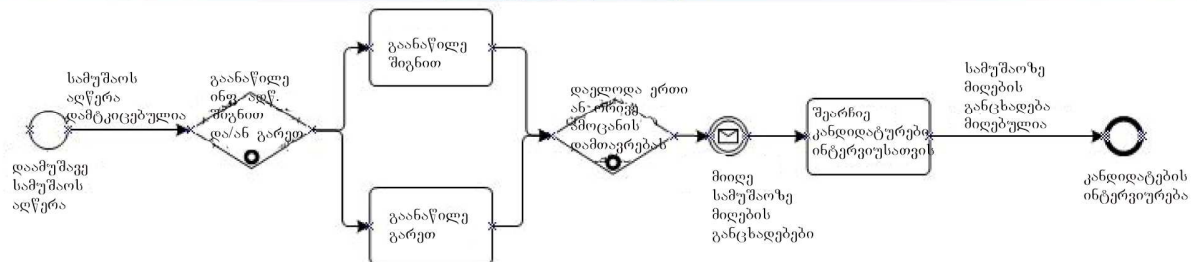


გვერდი *Initiate Recruitment* (დაიწყო გადამიღება) ეხლა დაახლოებით ასე უნდა გამოიყურებოდეს:

Initiate Recruitment

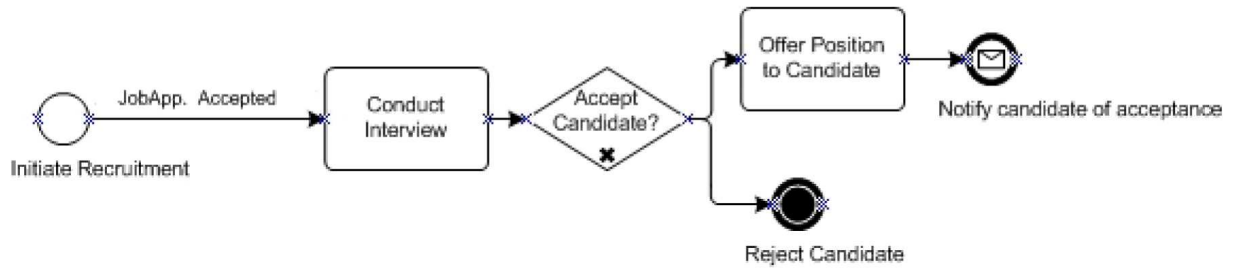


დაიწყო გადამიღება

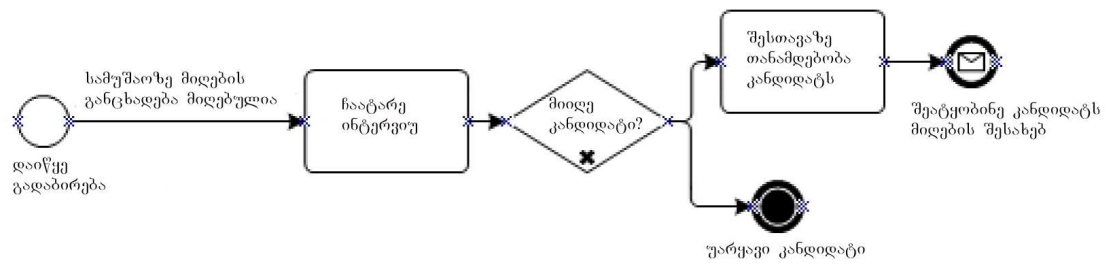


გვერდი *Interview Candidates* (კანდიდატების ინტერვიუება) ეხლა დაახლოებით ასე უნდა გამოიყურებოდეს:

Interview Candidates



კანდიდატების ინტერვიუება



ლაბორატორიული სამუშაო №8

დარჩენილი მასალების შექმნა და მინიჭება (Create and Assign the Remaining Materials)

ჩვენ დავამატეთ შეტანები და გამოტანები (პროდუქციები) პროცესის უმაღლეს დონეს. ესეა, ჩვენ აგრეთვე უნდა დავიტანოთ ნახაზზე მასალების ნაკადი ცალკეულ ამოცანებსა და ხდომილებებს შორის. ამის გაკეთებით, ჩვენ გავივლით ყოველ პროცესს და განვსაზღვრავთ რომელი მასალები იყვნენ შეცვლილი და მათ სხვადასხვა მდგომარეობებს.

1. გახსენი *Develop Job Description* (დაამუშავე სამუშაოს აღწერა) გვერდი და ორჯერ დააწკაპუნე შეერთება (კავშირი) (the link) შემდეგ ამოცანებს შორის *Locate Job Description* (განათავსე სამუშაოს აღწერა) და *Evaluate Job Description* (შეაფასე სამუშაოს აღწერა).
2. *Material Name* (მასლის სახელის) ველში დააწკაპუნე მინიჭება (**Assign**).
3. რამდენადაც ჩვენ ჯერ კიდევ არ შეგვიქმნია პროცესში გამოყენებული მასალები შესაბამისი მდგომარეობებით, ჩვენ მათ შევქმნით იმის მიხედვით თუ როგორ გავაგრძელებთ. გამოსულ *Material Selection* (მასლის შერჩევა) ფორმაზე დააწკაპუნე **Create New** (შექმენი ახალი).
4. სახელის (Name) ველში შეიყვანე “JobDesc” (“სამუშაოს აღწერა”). მინდვრიდან ტიპი (from the *Type* field) შეარჩიე **form** (ფორმა) და შეიყვანე “Created” (“შექმნილი”) როგორც მდგომარეობა (State). დააწკაპუნე OK.
5. შეარჩიე ახლად შექმნილი მასალა და დააწკაპუნე **Assign** (მინიჭება). დააწკაპუნე OK *შემკვეთის თვისების რედაქტორის* (the *Custom Property Editor*) დასახურად.
6. გაიმეორე ბიჯები 1-5 შემდეგი შეერთებებისათვის (კავშირებისათვის) (to the following links) შესაბამისი მასალების შესაქმნელად და/ან მისანიჭებლად. გვახსოვდეს, რომ ყველა მასალა იქნება **form** (ფორმა) ტიპის. **შენიშვნა:** თუ მასალის საჭირო სახელი უკვე არსებობს გამოსულ *Material Selection* (მასლის შერჩევა) ფორმაზე ახლის შექმნა არ არის საჭირო; აირჩიე არსებული და მინიჭება ის შერჩეულ კავშირს.

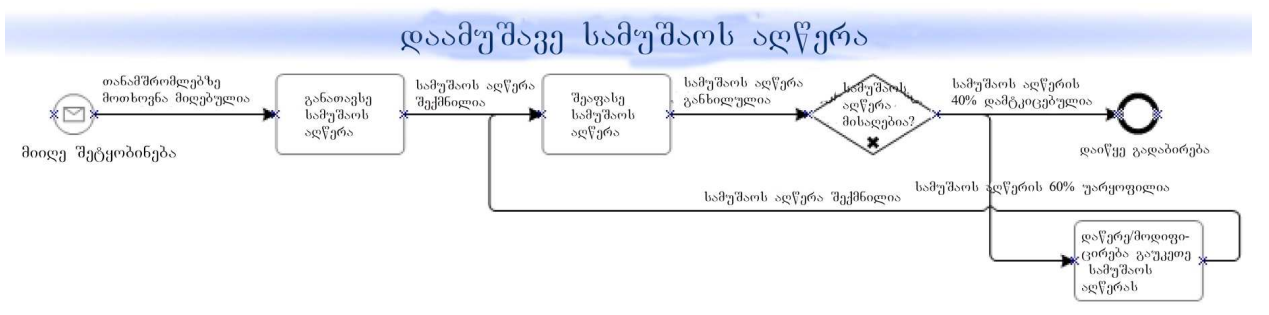
პროცესი Process	წყარო Source	დანიშნულება Destination	მასალა Material	მდგომარეობა State
დაამუშავე სამუშაოს აღწერა Develop Job Description	შეაფასე სამუშაოს აღწერა Evaluate Job Description	სამუშაოს აღწერა მისაღებია? Job Description Acceptable?	სამუშაოს აღწერა JobDesc	განხილული Reviewed
	სამუშაოს აღწერა მისაღებია? Job Description Acceptable?	დაწერე/მოდულირება გაუკეთე სამუშაოს აღწერას Write/Modify Job	სამუშაოს აღწერა JobDesc	უარყოფილი Rejected

		Description		
	დაწერე/მოდირება გაუკეთე სამუშაოს აღწერას Write/Modify Job Description	შეაფასე სამუშაოს აღწერა Evaluate Job Description	სამუშაოს აღწერა JobDesc	შექმნილია Created
	მიიღე შეტყობინება Receive notification	განათავსე სამუშაოს აღწერა Locate Job Description	მოთხოვნა თანამშრომლებ ზე Req.Emp	მიღებულია Received
დაიწყო გადაბირება Initiate Recruitment	გაანაწილე სამუშაოს აღწერა Distribute Job Description	გაანაწილე შიგნით Distribute Internally	სამუშაოს აღწერა JobDesc	დამტკიცებუ ლი Approved
	გაანაწილე სამუშაოს აღწერა Distribute Job Description	გაანაწილე გარეთ Distribute Externally	სამუშაოს აღწერა JobDesc	დამტკიცებუ ლი Approved
	მიიღე სამუშაოს აღწერა Receive Job Description	შეარჩიე კანდიდატები ინტერვიუსათვის Select Candidates for Interview	სამუშაოზე მიღების განცხადებები JobApp	მიღებულია Received
კანდიდატები ს ინტერვიურე ბა Interview Candidates	მიიღებთ კანდიდატს? Accept Candidate?	უარყავი კანდიდატი Reject Candidate	უარყოფის შეტყობინება Reject.Notice	შექმნილი Created
	შესთავაზე თანამებობა კანდიდატს? Offer Position to Candidate?	შეატყობინე კანდიდატს მიღების შესახებ Notify Candidate of Acceptance	მიღების შეტყობინება Accept.Notice	შექმნილი Created

თუმცა, გასასვლელი (gateway) არის განსაკუთრებული შემთხვევა. გავიხსენოთ, რომ განსაკუთრებულ გასასვლელს “Job Description Acceptable?” (“სამუშაოს აღწერა მისაღებია?”) აქვს ორი შესაძლო შედეგი: 1) ან სამუშაოს აღწერა მოითხოვს მოდიფიკაციას, ან 2) სამუშაოს აღწერა მისაღებია. რამდენადაც განსაკუთრებული გასასვლელის შედეგები არიან ურთიერთ გამომრიცხავი, ჩვენ ვიყენებთ ერთზე ნაკლებ ალბათობების მნიშვნელობებს იმის განსასაზღვრავად თუ რომელი პოტენციალური შედეგი მოხდება ფაქტიურად მაშინ, როდესაც მოქმედება შესრულდება. ერთი გასასვლელისათვის (gateway) ყველა შედეგის შეერთების (კავშირების) ალბათობების ჯამი უნდა იყოს ერთის ტოლი.

1. გადადი *სამუშაოს აღწერის დამუშავება (Develop Job Description)* პროცესის გვერდზე.
2. ორჯერ დააწკაპუნე შეერთებაზე, რომელიც აკავშირებს ფორმებს *Job Description Acceptable? (სამუშაოს აღწერა მისაღებია?)* და *Write/Modify Job Description (დაწერე/მოდიფიცირება გაუკეთე სამუშაოს აღწერას)*.
3. როდესაც *შემკვეთის თვისების რედაქტორი (Custom Property Editor)* გაიხსნება, ჩაბეჭდე მნიშვნელობა “60.00” *ალბათობის (Probability)* ველში და დააწკაპუნე **OK**.
4. სხვა შეერთებისათვის, რომელიც იწყება *Job Description Acceptable? (სამუშაოს აღწერა მისაღებია?)* გასასვლელიდან, შეცვალე მისი ალბათობა “40.00”-ზე.

პროცესის გვერდზე *Develop Job Description (სამუშაოს აღწერის დამუშავება)* გასასვლელისათვის (for the gateway) გამოსასვლელი შეერთებები ეხლა ნაჩვენებია ალბათობების მნიშვნელობებით 60% და 40%, რომლებიც მიუთითებენ, რომ ყველა შემთხვევების 60%-ში სამუშაოს აღწერა მოითხოვს რაღაც მოდიფიკაციას.



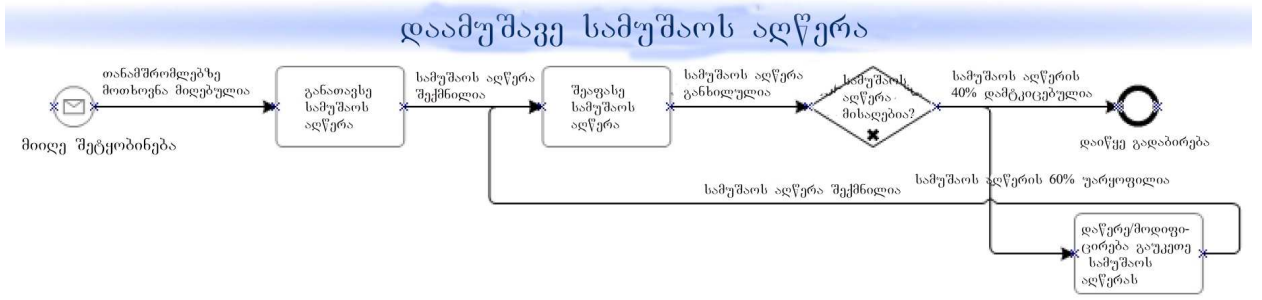
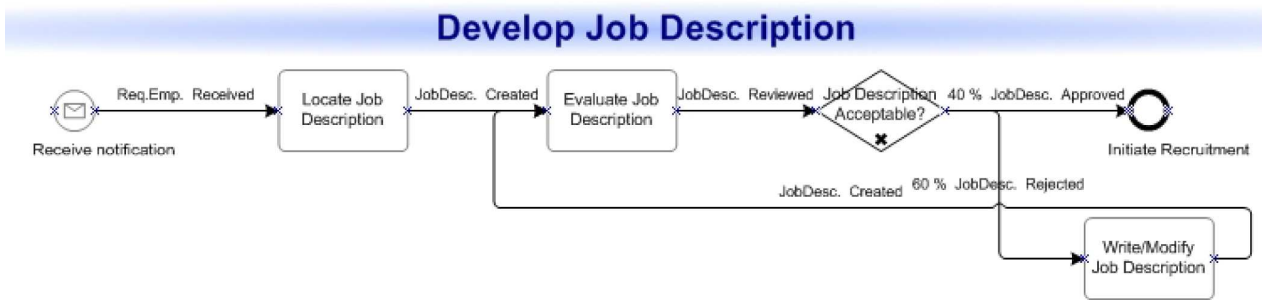
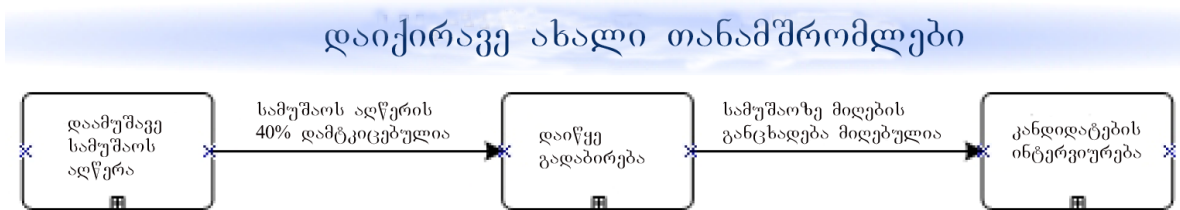
შენიშვნა: ალბათობების ახალი მნიშვნელობები წარმოიქმნა (გამოჩნდა) ასევე უმაღლესი დონის შეერთებებზე.



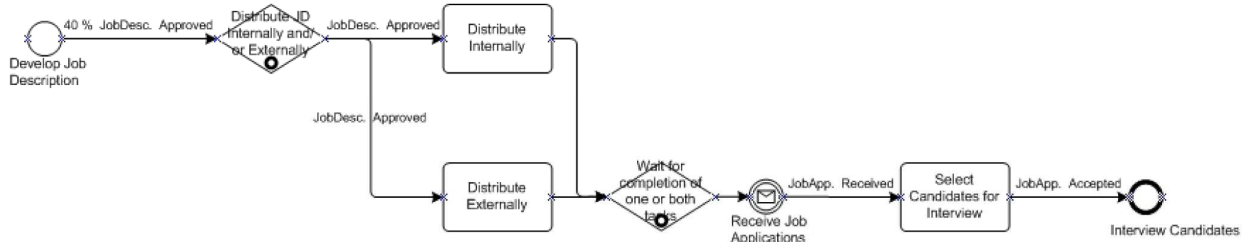
ლაბორატორიული სამუშაო №10

პროცესის დამთავრებული სქემის შემოწმება (Verify Completed Process Map)

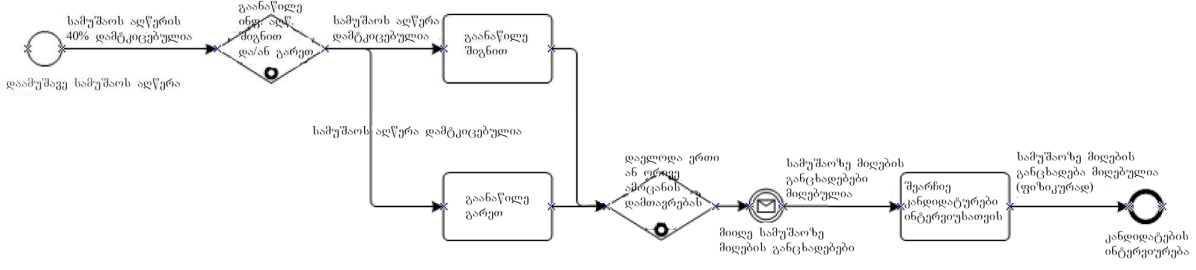
ქვემოთ მოყვანილი ილუსტრაციები აჩვენებენ, როგორ უნდა გამოიყურებოდნენ თქვენი პროცესის გვერდები, როდესაც გამოიყენება *ლაგეგმე ეს გვერდი (Layout This Page)* ფუნქცია. შეგიძლიათ თავისუფლად (თქვენი შესვლად) დაარეგულიროთ შეერთებები და ფორმები განლაგების ოპტიმიზაციისათვის.



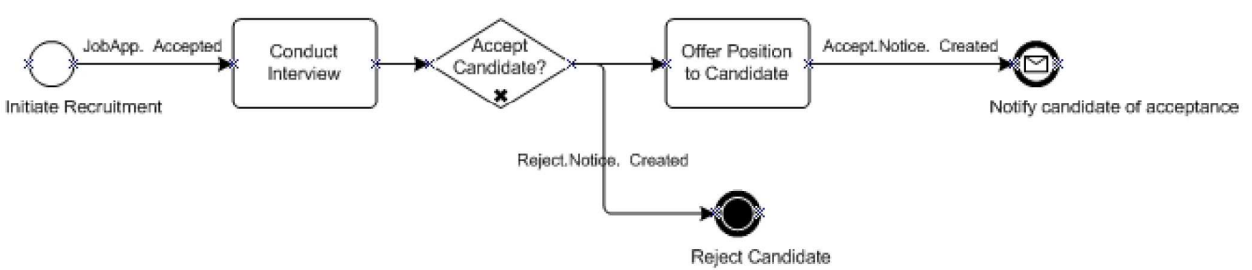
Initiate Recruitment



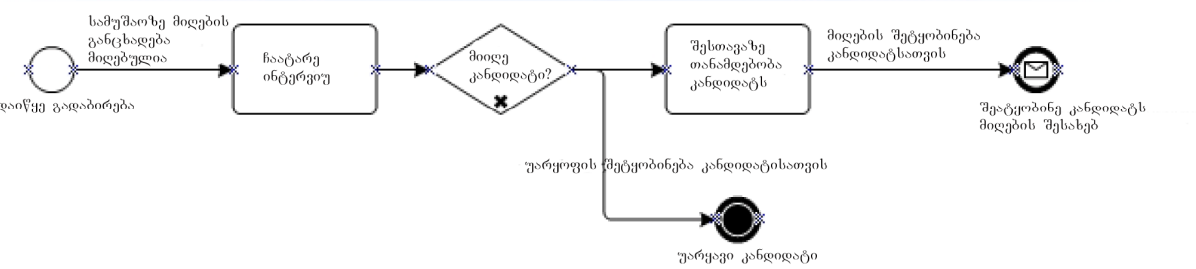
დაიწყო გადაბირება



Interview Candidates

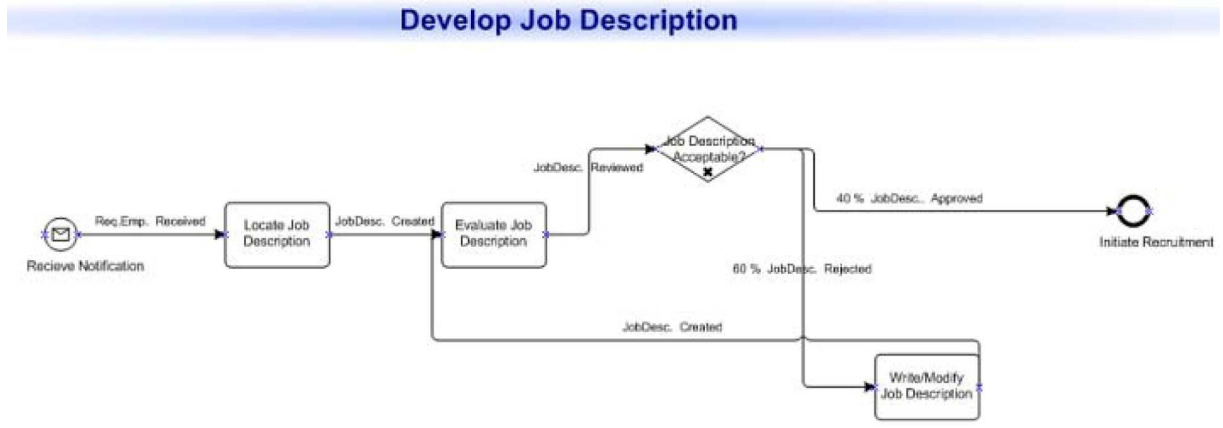


კანდიდატების ინტერვიუება

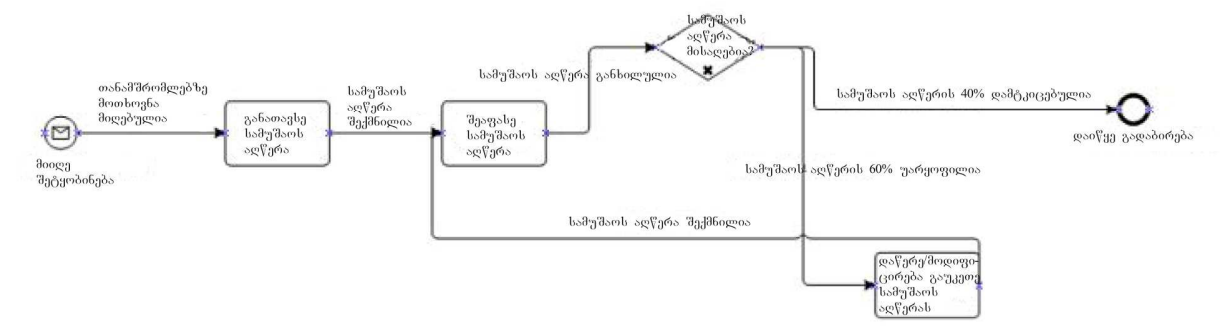


ზოგჯერ, როდესაც თქვენ აირჩევთ *Layout This Page* (დაგეგმე ეს გვერდი), ბლოკებმა შეიძლება ერთმანეთი გადაფარონ თქვენი პროცესის სქემაზე. იმისათვის, რომ მისცეთ თქვენ სქემას უფრო მეტად მოწესრიგებული (ორგანიზებული) სახე, თქვენ შეგიძლიათ ხელით გადაიტანოთ ობიექტებს შორის შეერთებები უბრალო

დაწკაპუნებით შეერთებაზე და მისი გადთრევით ურთერთ გადაფარვის ბლოკებიდან. მაგალითი იხილეთ ქვემოთ:



დაამუშავე სამუშაოს აღწერა



პროცესისათვის თვისებების მინიჭება (Assign Process Properties)

ყოველ პროცესს უნდა ჰქავდეს მფლობელი. მფლობელი არის ორივე: ან როლი ან რესურსი, რომელიც პასუხისმგებელია პროცესის უმრავლეს ან ყველა მოქმედებებზე. მრავალჯერადი როლები და/ან რესურსები შეიძლება იყვნენ დადგენილი პროცესის მფლობელებად.

პროცესის მფლობელის დასადგანად შემდეგი პროცედურა უშვებს, რომ თქვენ გაქვთ განსაზღვრული თქვენი როლებიც და რესურსებიც. ამ სახელმძღვანელოს მიზნებიდან გამომდინარე, ჩვენ პროცესის მფლობელებს განვსაზღვრავთ მას შემდეგ, რაც აგებული გვექნება ჩვენი პროცესი; ეს ინფორმაცია შეიძლება იყოს აგრეთვე განსაზღვრული ჩვენი მოდელის შექმნისას.

1. გადადი სცენარის გვერდზე (to the Scenario page).

2. მარჯვენა ღილაკის დაწკაპუნება პროცესის ფორმაზე **Hire New Employees** (დაიქირავე ახალი თანამშრომლები) და შეარჩიე თვისებები (**Properties**) რომ გახსნა *Custom Property Editor* (*შემკვეთის თვისების რედაქტორი*).
3. აირჩიე ფანჯარა *Basic* (*ძირითადი*) და დააწკაპუნე **Assign** (**მიანიჭე**) *Owners* (*მფლობელი*) მინდვრის მარჯვნივ.
4. *Owner Selection* (*მფლობელის შერჩევა*) ფანჯარაში, შეარჩიე **Servers + Software** (**სერვერები+პროგრამული უზრუნველყოფა**) და დააწკაპუნე **Assign** (**მიანიჭე**).
5. დააწკაპუნე **OK** ცვლილებების შესანახად და დასურე ფანჯარა.

გადადი *Hire New Employees* (დაიქირავე ახალი თანამშრომლები) გვერდზე და მიანიჭე შესაბამისი მფლობელები შემდეგ პროცესებს:

პროცესი Process	მფლობელი Owner
დაამუშავე სამუშაოს აღწერა Develop Job Description	ადმინისტრაციული დამხმარე ადამიანური რესურსების მენეჯერი Administrative Assistant HR Manager
დაიწყო გადაბირება Initiate Recruitment	ადამიანური რესურსების მენეჯერი HR Manager
კანდიდატების ინტერვიუება Interview Candidates	ადამიანური რესურსების მენეჯერი HR Manager

ლაბორატორიული სამუშაო №11

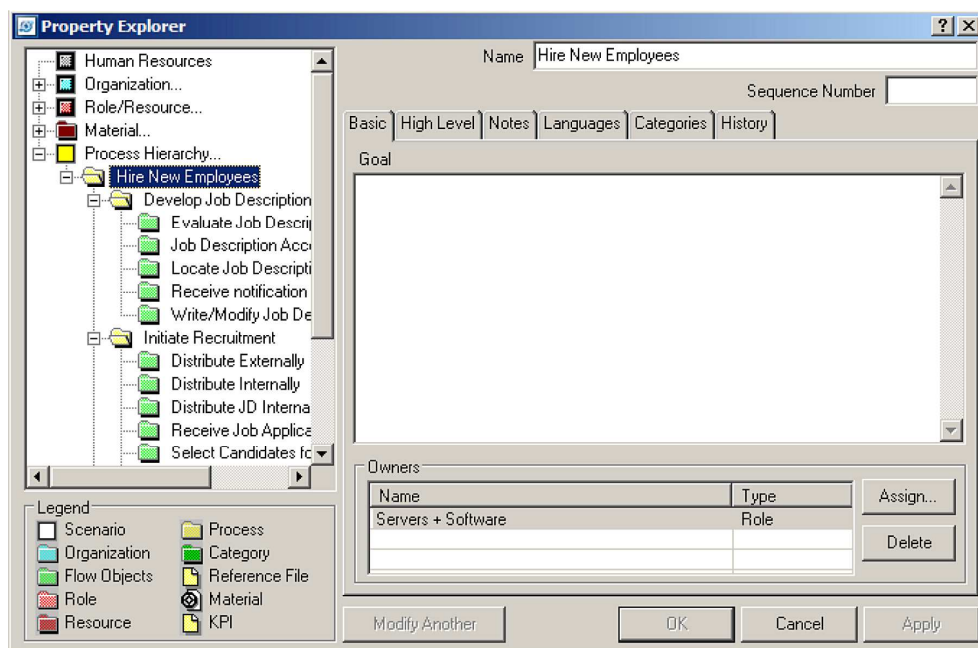
BPMN–ს ფორმების შემსრულებლების დანიშვნა (Assign Performers to BPMN Shapes)

მიმდინარე მოდელს უნდა აგრეთვე დავამატოთ რესურსის სახელი, რომელიც ასრულებს ყოველ მოქმედებას.

1. პროცესში *Develop Job Description* (დაამუშავე სამუშაოს აღწერა) ამოცანისათვის **Locate Job Description** (განათავსე სამუშაოს აღწერა) გახსენი *Custom Property Editor* (შემკვეთის თვისების რედაქტორი).
2. დააწკაპუნე **Assign** (მინიჭება) მიღვრის *Performers* (შემსრულებლები) მარჯვნივ.
3. ფანჯარაში *Performer Selection* (შემსრულებლის შერჩევა) შეარჩიე *Select Resources* (შეარჩიე რესურსები) და შემდეგ სტრიქონი **Tom Tompson** (ტომი ტომპსონი).
4. დააწკაპუნე **Assign** (მინიჭება). ეს თქვენ დაგაბრუნებთ *შემკვეთის თვისების რედაქტორთან* (*Custom Property Editor*). რესურსი *ტომი ტომპსონი* (*Tom Tompson*) გამოჩნდება ველში *Performers* (შემსრულებლები).
5. დააწკაპუნე **OK** იმისათვის, რომ დაადასტუროთ როლის/რესურსის დანიშნულება (the role/resource assignment) და დახურე ფანჯარა.

იმისათვის, რომ მივანიჭოთ დარჩენილი მითითებები შესაბამის ფორმებს, ჩვენ გამოვიყენებთ the Property Explorer-ს (თვისების მკვლევარს), რომელიც არის ძალიან სწრაფი მეთოდი. ხილული ინსტრუმენტების პანელიდან შეარჩიეთ **Interfacing >**

Property Explorer (ურთიერთ კავშირი > თვისების მკვლევარი). *Custom Property Editor* (შემკვეთის თვისების რედაქტორი) იქნება ნაჩვენები.



ამ ფანჯრიდან თქვენ შეგიძლიათ შექმნათ ან რედაქტირება გაუკეთოთ BPMN მოდელიორის ნებისმიერ ობიექტს.

1. *Hierarchy Explorer*–დან (იერარქიის მკვლევარიდან) (ფანჯარა ზედა მარცხენა კუთხეში), გადადით ამოცანაზე **Evaluate Job Description** (**შეაფასე სამუშაოს აღწერა**) (*Process Hierarchy > Hire New Employees > Develop Job Description*).
2. დააწკაპუნე **Assign** (მინიჭება) *Performers* (*შემსრულებლები*) მინდვრის მარჯვნივ.

Performer Selection (*შემსრულებლის არჩევა*) ფანჯარაში დააწკაპუნე *Select Resource* (*შეარჩიე რესურსი*) ოფცია. აირჩიე რესურსი **Brenda Brown** (ბრენდა ბრაუნი). დააწკაპუნე **Assign** (მინიჭება). თუ საჭიროა თქვენ შეგიძლიათ მიანიჭოთ სხვა როლი ან რესურსი მესამე ბიჯის გამოყენებით. ეს შეიძლება გააკეთდეს **Assign**-ის (მინიჭება) ახლიდან არჩევით.

3. რესურსი იქნება მინიჭებული ამოცანისათვის. დააწკაპუნე **Apply** (გამოყენება) ცვლილებების შესასრულებლად.

მიანიჭე შემდეგი რესურსები დარჩენილ ფორმებს. ქვემოთ მოყვანილ ცხრილში მოცემულია საჭირო ინფორმაცია.

შენიშვნა: თქვენ აგრეთვე შეგიძლიათ მიანიჭოთ როლები ფორმებს.

პროცესი Process	BPMN-ს ფორმა BPMN Shape	რესურსის სახელი Resource Name
დაამუშავე სამუშაოს აღწერა Develop Job Description	სამუშაოს აღწერა მისაღებია? Job Description acceptable?	ბრანდა ბრაუნი Brenda Brown
	მიიღე შეტყობინება Receive Notification	ბრანდა ბრაუნი Brenda Brown
	დაწერე/მოდიფიცირება გაუკეთე სამუშაოს აღწერას Write/Modify Job Description	ტომი ტომპსონი Tom Tompson
დაიწყო გადაბირება Initiate Recruitment	გაანაწილე სამუშაოს აღწერა Distribute Job Description	ტომი ტომპსონი Tom Tompson
	გაანაწილე შიგნით Distribute Internally	ტომი ტომპსონი Tom Tompson
	გაანაწილე გარეთ Distribute Externally	ტომი ტომპსონი Tom Tompson
	დაელოდა ერთი ან ორივე ამოცანის დამთავრებას Wait for completion of one or both tasks	ტომი ტომპსონი Tom Tompson
	მიიღე სამუშაოზე მიღების განცხადებები Receive Job Applications	ადამიანური რესურსების სისტემა HR System
	შეარჩიე კანდიდატურები ინტერვიუსათვის Select Candidates for Interview	ბრენდა ბრაუნი Brenda Brown
კანდიდატების ინტერვიუება	წაიყვანე ინტერვიუ	ბრენდა ბრაუნი

Interview Candidates	Conduct Interview	Brenda Brown
	მიიღებთ კანდიდატს? Accept Candidate?	ჯონი ჯონსონი John Johnson
	შესთავაზე თანამებობა კანდიდატს? Offer Position to Candidate?	ჯონი ჯონსონი John Johnson
	შეატყობინე კანდიდატს მიღების შესახებ Notify Candidate of Acceptance	ტომი ტომპსონი Tom Tompson
	უარყავი კანდიდატი Reject Candidate	ტომი ტომპსონი Tom Tompson

ლაბორატორიული სამუშაო №12

ხილული თვისებების შექმნა (Make Properties Visible)

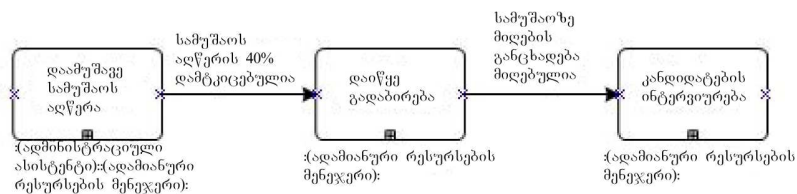
შენიშვნა: შემსრულებლები, რომლებიც თქვენ მხოლოდ განსაზღვრეთ, არ გამოჩნდებიან პროცესის გვერდებზე. ეს იმიტომ, რომ საჭიროა ეს თვისებები გახადოთ თვალსაჩინო.

1. მარჯვენა დააწკაპუნე **Develop Job Description** (დაამუშავე სამუშაოს აღწერა) გვერდზე.
2. აირჩიე ოფცია **Select Visible Properties** (აირჩიე თვალსაჩინო თვისებები).
3. როდესაც **Properties Display Selection** (თვისებების ჩვენების არჩევა) ფანჯარა გაიხსნება, აირჩიე თვისებები **Activity_Performer** (მოქმედება-შემსრულებელი) და **Process_Owner** (პროცესი-მფლობელი).
4. დააწკაპუნე **OK**. რესურსის და/ან როლის სახელი, რომელიც ასრულებს მოქმედებას, ეხლა იქნება ნაჩვენები შესაბამისი ფორმის ქვემოთ. თქვენ პროცესებს ეხლა ექნებათ შემდეგნაირი სახე:

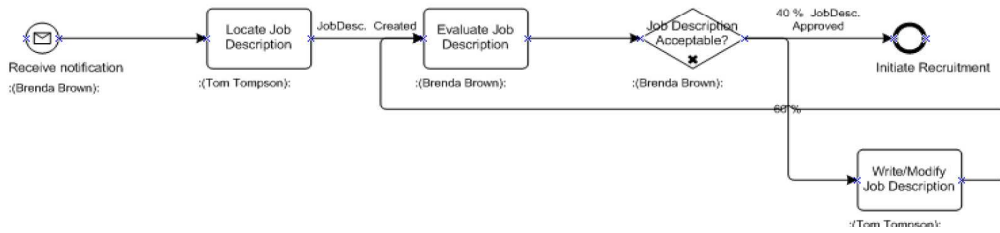
Hire New Employees



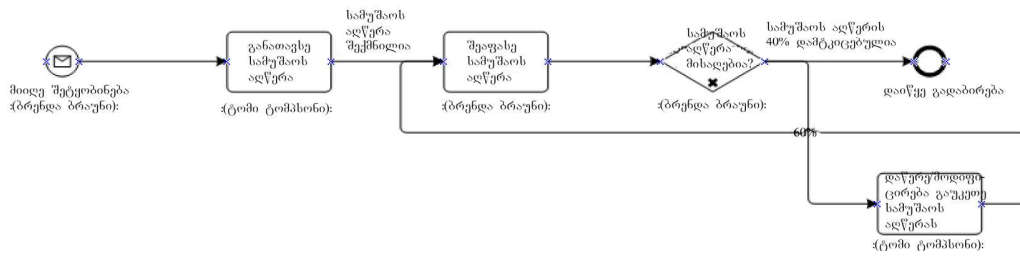
დაიქირავე ახალი თანამშრომლები



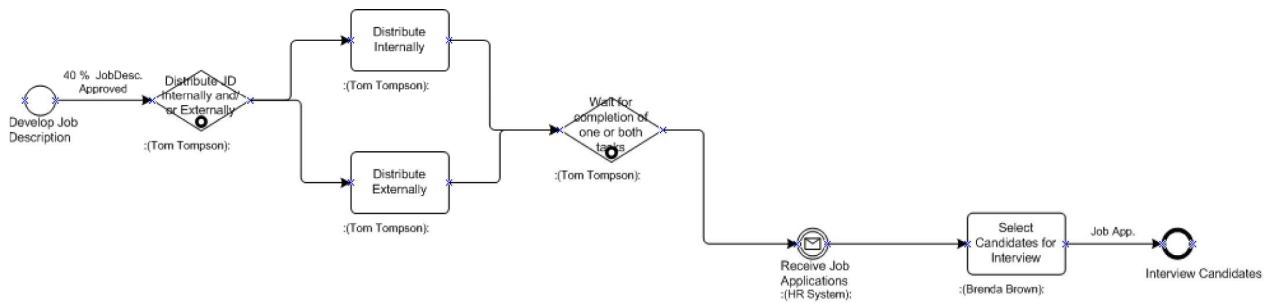
Develop Job Description



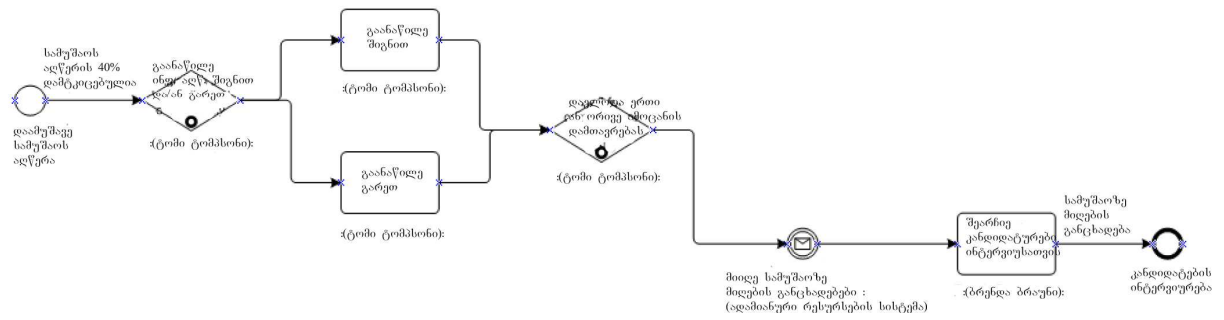
დაამუშავე სამუშაოს აღწერა



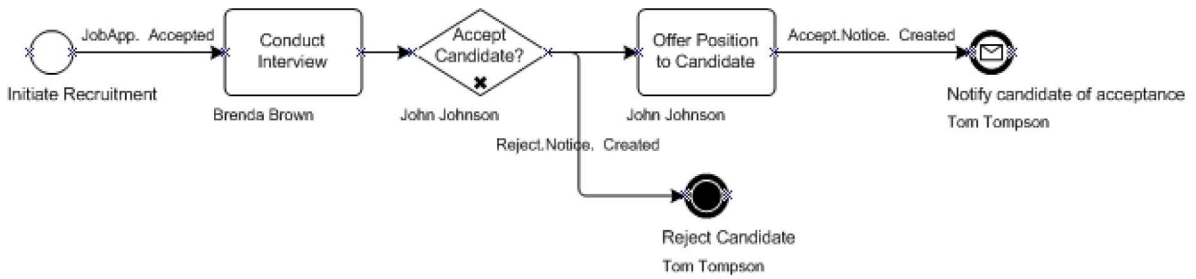
Initiate Recruitment



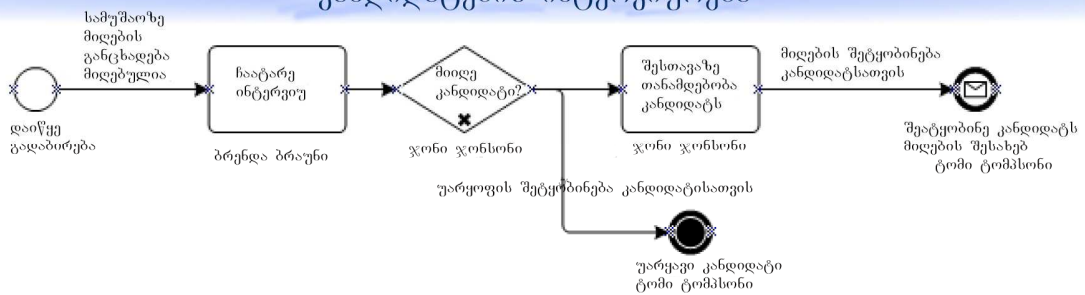
დაიწვე გადაბირება



Interview Candidates



კანდიდატების ინტერვიუება

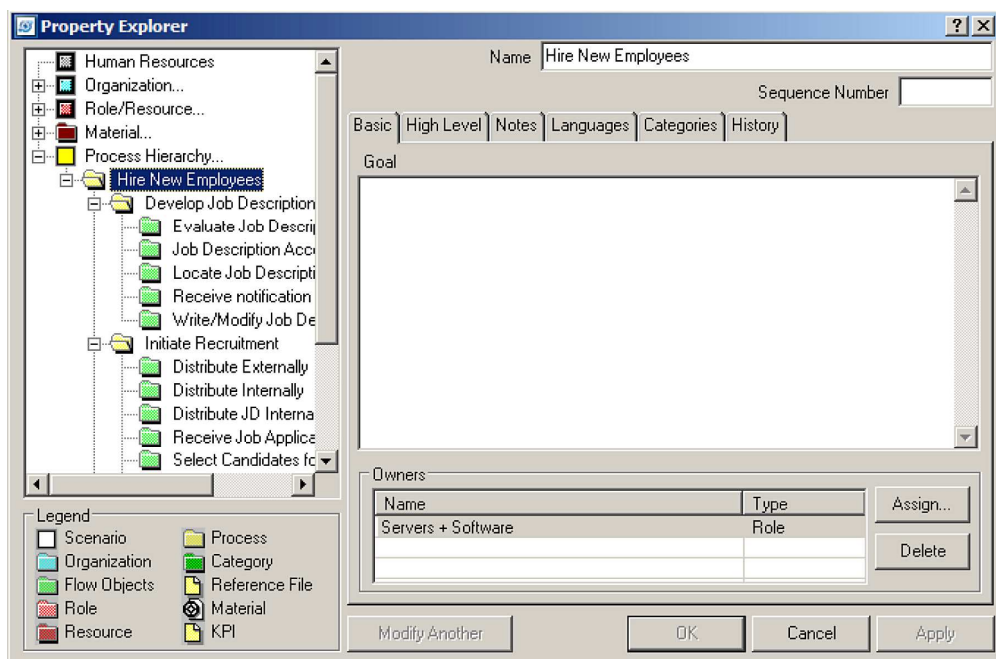


ლაბორატორიული სამუშაო №13

პროცესისათვის მუშაობის ძირითადი ინდიკატორის მინიჭება (Assign KPI's to a Process)

როგორც წესი, მუშაობის ძირითადი ინდიკატორები (KPI) არიან დროის, ღირებულების ან ხარისხის მიზნები, რომელთა შესრულება განსაზღვრავს მოცემული პროცესის მუშაობას. თქვენ შეგიძლიათ შექმნათ მუშაობის ძირითადი ინდიკატორების გვერდი *Property Explorer*-ის (*თვისების მკვლევარის*) გამოყენებით. ინსტრუმენტების პანელიდან აირჩიეთ **Interfacing > Property Explorer** (ურთიერთ კავშირი > თვისების მკვლევარი). ნახვენები იქნება *Property Explorer*-ის (*თვისების მკვლევარის*) ფანჯარა.

შენიშვნა: Key Performance Indicator (KPI) – მუშაობის ძირითადი ინდიკატორი



ჩვენ ეხლა შევქმნით მუშაობის ძირითადი ინდიკატორებს და მათ უშუალოდ დავამატებთ შესაბამის პროცესებს.

1. *Hierarchy Explorer*-დან (იერარქიის მკვლევარიდან) (ფანჯარა ზედა მარცხენა კუთხეში), აირჩიე **Hire New Employees** (დაიკირავე ახალი თანამშრომლები).
2. აირჩიე **High Level** (მაღალი დონის) სიმბოლო (tab). დარწმუნდი, რომ *K.P.I.* (მუშაობის ძირითადი ინდიკატორი) ფანჯარა გაიხსნა.
3. დააწკაპუნე **Add** (დაამატე). შეტყობინება *No KPI page was found. Do you wish to create one?* (არ მუშაობის ძირითადი ინდიკატორი იყო ნაპოვნი. გსურთ შექმნათ ის?) გვერდი იქნება ნაპოვნი. აირჩიე **Yes** (დიახ).

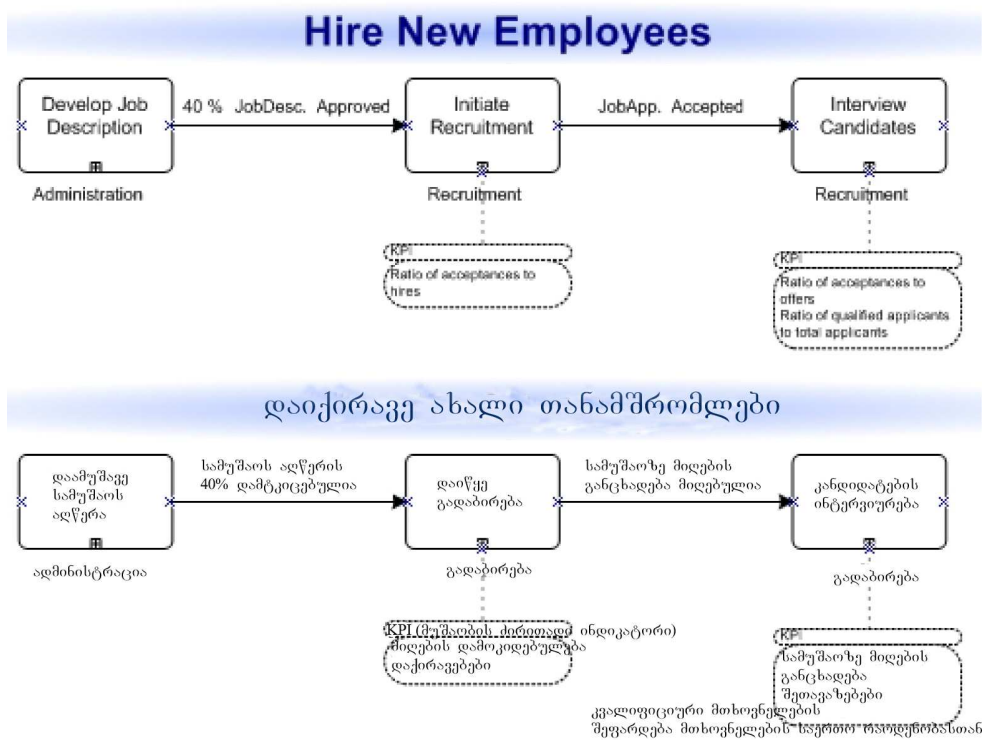
4. აირჩიე **Create New...** (**შექმენი ახალი**) ფანჯარაში *Key Performance Indicator* (მუშაობის ძირითადი ინდიკატორი).
5. შეიყვანე “Average days to fill open positions” (“დღეების საშუალო რაოდენობა ღია პოზიციების შესავსებად”) ველში *KPI Name* (მუშაობის ძირითადი ინდიკატორის სახელი).
6. *Target* (მიზანი) ველში შეიყვანე “30”. შეიყვანე “0” ველში *Start Value* (საწყისი მნიშვნელობა) და “Day” (დღე) ველში *Unit* (ერთეული). ეს ნიშნავს, რომ დაახლოებით 30 დღეა საჭირო საშუალო პოზიციის შესავსებად. აირჩიე **OK**.
7. აირჩიე ახლად შექმნილი KPI (მუშაობის ძირითადი ინდიკატორი) და დააწკაპუნე **Assign** (მიანიჭე). KPI (მუშაობის ძირითადი ინდიკატორი) არ იქნება მინიჭებული *Hire New Employees*–თვის (დაიქირავე ახალი თანამშრომლებსათვის). დააწკაპუნე **Apply** (გამოიყენე) ცვლილებების განსახორციელებლად.
8. ქვემოთ, ცხრილში მოცემული ინფორმაცია გამოიყენე კიდევ სამი KPIs (მუშაობის ძირითადი ინდიკატორის) შესაქმნელად შესაბამისი პროცესებისათვის:

პროცესი Process	მუშაობის ძირითადი ინდიკატორის სახელი KPI Name	მიზანი Target	საწყისი მნიშვნელობა Start Value	ერთეული Unit
დაიწყე გადაბირება Initiate Recruitment	მიღებების დამოკიდებულება დაქირავებებისათვის Ratio of acceptances to hires	3:1	დატოვეთ ცარიელი	დატოვეთ ცარიელი
კანდიდატების ინტერვიუება Interview Candidates	შეთავაზებული მიღებების დამოკიდებულება Ratio of acceptances to offers	1:1	1:2	დატოვეთ ცარიელი
კანდიდატების ინტერვიუება Interview Candidates	კვალიფიციური პრეტენდენტების შეფარდება პრეტენდენტების საერთო რაოდენობასთან Ratio of qualified applicants to total applicants	1:3	1:5	დატოვეთ ცარიელი

განცალკევებულ ტექსტურ ყუთში თვისებების ჩვენება (Displaying Properties in a Separate Text Box)

FS Property (FS თვისება) ფორმა *სცენარის თანამედროვე* ტრაფარეტზე (on the *Scenario Advanced stencil*) შეიძლება იყოს გამოყენებული BPMN-ს ფორმების განსაზღვრული თვისებების საჩვენებლად. **FS Property (FS თვისება)** ფორმა შეიძლება იყოს გადაადგილებული რომელიმე სხვა ფორმისაგან დამოუკიდებლად.

1. გადადი გვერდზე *Scenario (სცენარის)* და გადაათრე **FS Property (FS თვისება)** ფორმა *Hire New Employees (დაიქირავე ახალი თანამშრომლების)* ფორმაზე. *Property Selection (თვისების არჩევის)* ფანჯარა გაიხსნება.
2. *Property Selection (თვისების არჩევის)* ფანჯარა ყოველთვის ჩამოთვლის განსაზღვრულ თვისებებს გამოუცხადებლად. დამატებითი თვისებები დაემატება სიას როგორც კი შექმნილ თვისებებს მოცემული პროცესისათვის ან მოქმედების ფორმისათვის. აირჩიე თვისება **KPI (მუშაობის ძირითადი ინდიკატორი)** და დააწკაპუნე **Assign (მიანიჭე)**.
3. *Hire New Employees (დაიქირავე ახალი თანამშრომლების)* ფანჯარაში გაიმეორე 1 და 2 ბიჯები პროცესების *Initiate Recruitment (დაიწვე გადაბირების)* და *Interview Candidates (კანდიდატების ინტერვიუება)* KPI-ს (მუშაობის ძირითადი ინდიკატორის) მნიშვნელობების საჩვენებლად.

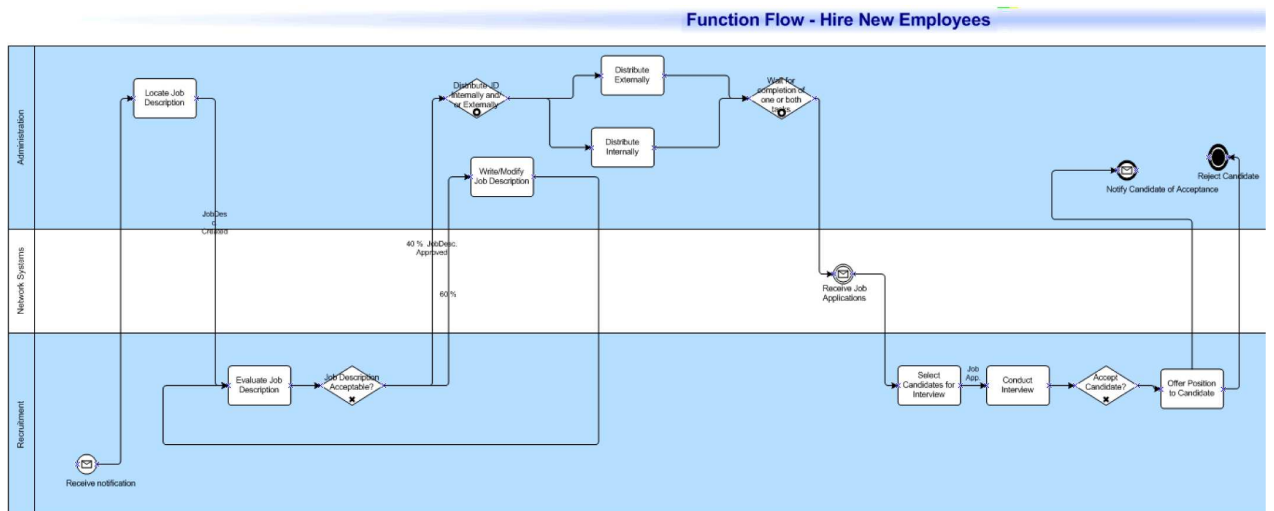


ფუნქციის ნაკადის გვერდი შექმნა (წარმოება) (Generate the Function Flow Page)

ეხლა, როდესაც პროცესის იერარქია ჩვენი მოდელისათვის არის სრული, ჩვენ უნდა შევქმნათ *ფუნქციის ნაკადის (Function Flow)* გვერდი, რომელიც ასახავს დამთავრებულ მოდელს.

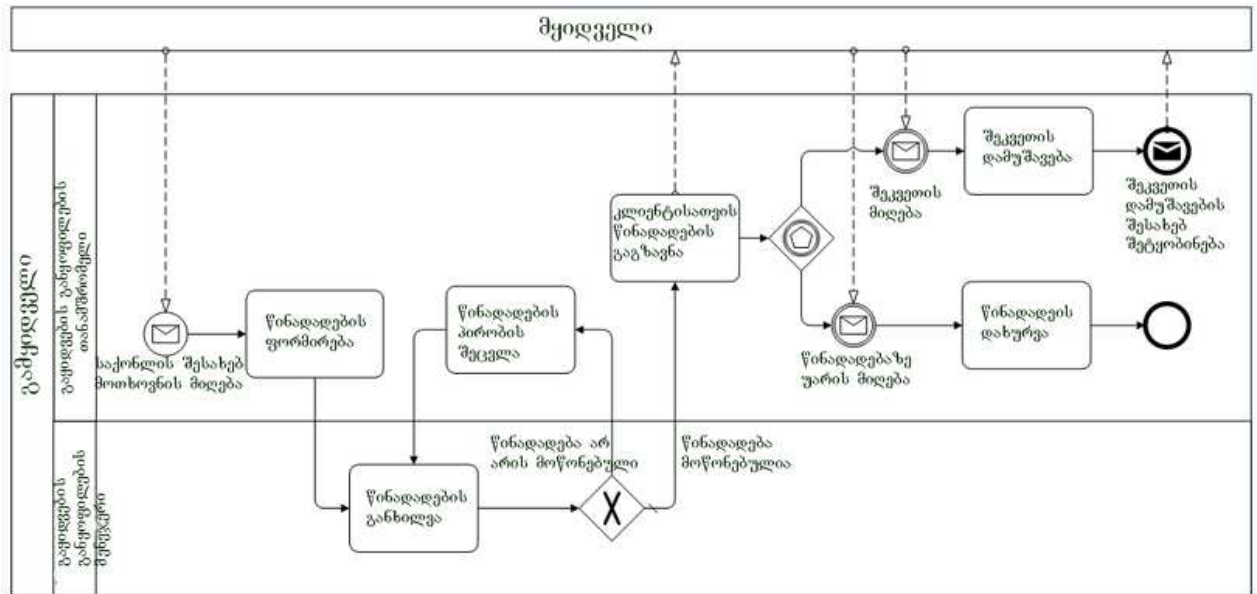
გადადი *Scenario (სცენარი)* გვერდზე. მარჯვენა ღილაკი დააწკაპუნე და აირჩიე **Generate Flow Page (შექმენი ნაკადის გვერდი)**. ამოსავალ (საწყის) წერტილად გამოიყენე **Hire New Employees (დაიქირავე ახალი თანამშრომლები)**. საბოლოოდ გვერდი უნდა გამოიყურებოდეს ქვემოთ ნაჩვენები დიაგრამის მსგავსად.

თქვენ შეამჩნევთ, რომ ფორმები ეხლა განლაგებულია “პროცესის მონაწილეებში” (in “Swimlane”) დანიშნული მფლობელების და შემსრულებლების შესაბამისად. პროცესის კავშირები, რომლებიც კვეთენ პროცესის მონაწილეებს შორის (between Swimlane) საზღვრებს, მიუთითებენ პრობლემის გადასწვევად საჭირო დროზე (hand-offs) ორგანიზაციის ერთეულებს შორის.



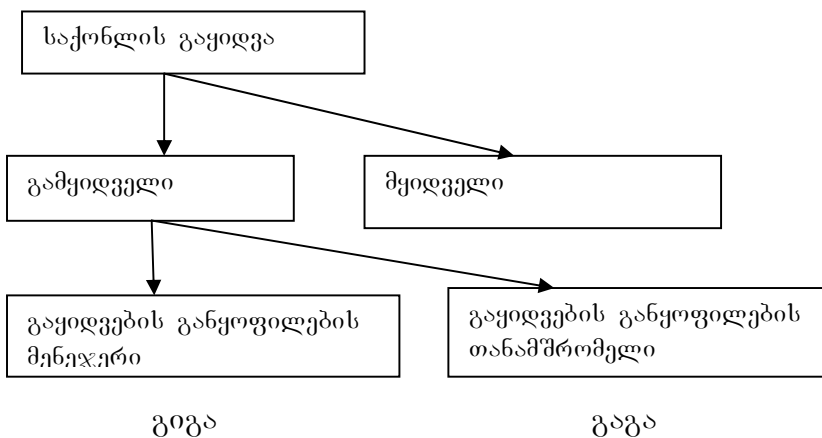
ლაბორატორიული სამუშაო №14

წინა ლაბორატორიულ სამუშაოებში ნასწავლი მასალის გამოყენებით ააგე შემდეგი ბიზნეს პროცესის მოდელი.



დახმარება:

- 1) სახელის დარქმევა
- 2) ორგანიზაციული სტრუქტურის აგება

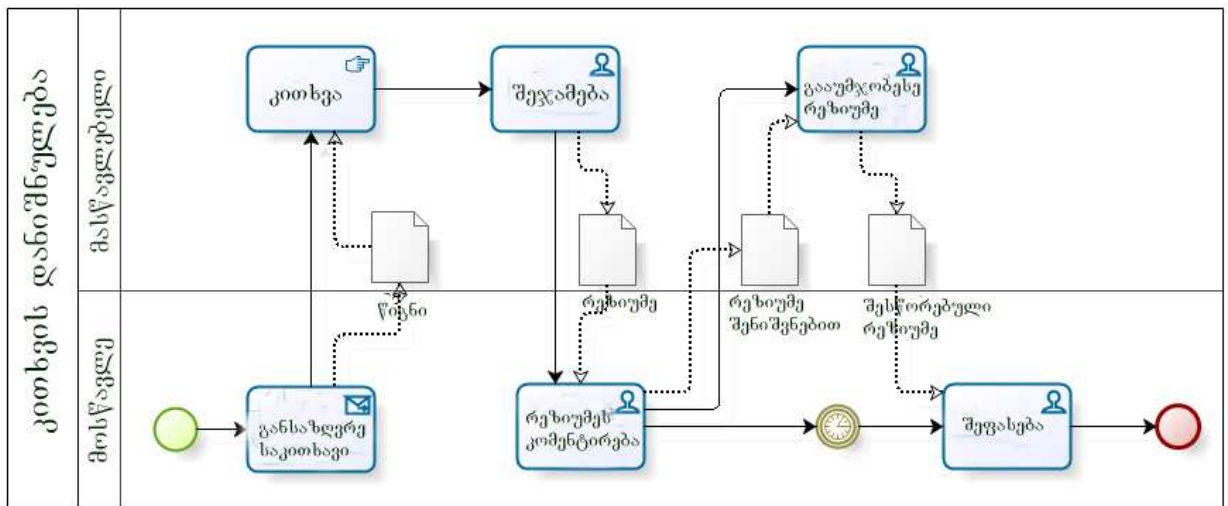
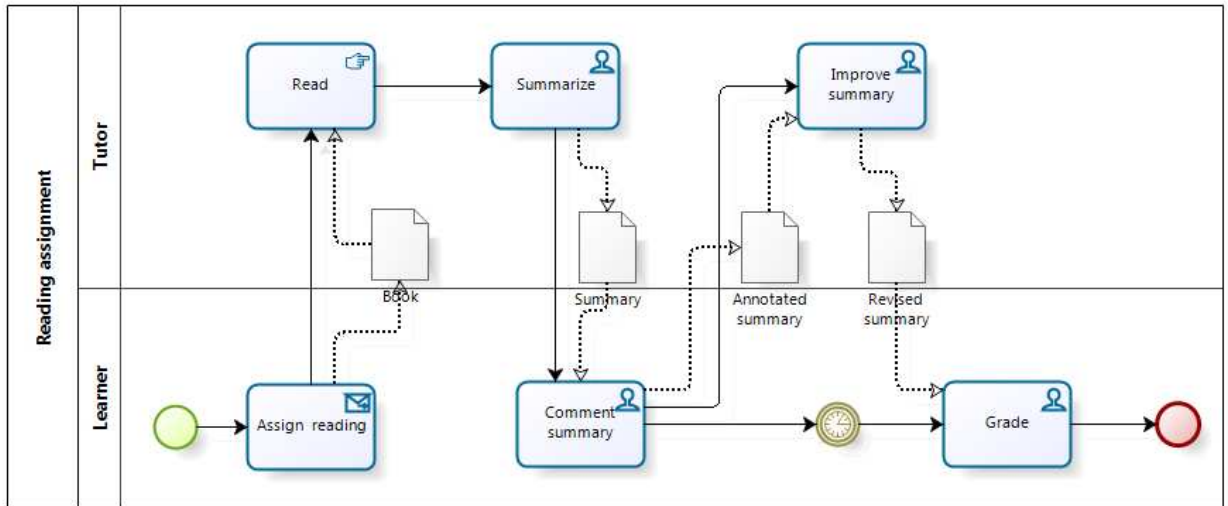


- 3) როლების და რესურსების შექმნა (**როლები:** გაყიდვების მენეჯერი, გაყიდვების თანამშრომელი. **რესურსები:** მენეჯერი-გიგა, თანამშრომელი-გაგა)
- 4) პროცესის ნაკადის უმაღლესი დონის შექმნა
- 5) მასალების გვერდის შექმნა (მასალები საჭიროა ნაკადებისათვის სახელების და ტიპების მისანიჭებლად)
- 6) სიბრტითი სქემის (ნახაზის) შექმნა

- 7) პროცესისათვის თვისებების მინიჭება
- 8) BPMN–ს ფორმების შემსრულებლების დანიშვნა
- 9) ხილული თვისებების შექმნა
- 10) ფუნქციის ნაკადის გვერდი შექმნა

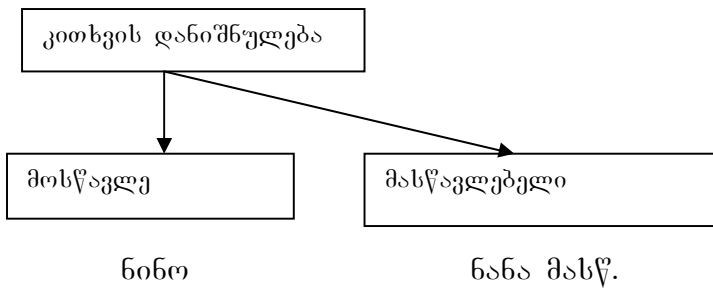
ლაბორატორიული სამუშაო №15

წინა ლაბორატორიულ სამუშაოებში ნასწავლი მასალის გამოყენებით ააგე შემდეგი ბიზნეს პროცესის მოდელი.



დახმარება:

- 1) სახელის დარქმევა
- 2) ორგანიზაციული სტრუქტურის აგება



- 3) როლების და რესურსების შექმნა (**როლები:** გაყიდვების მენეჯერი, გაყიდვების თანამშრომელი. **რესურსები:** მენეჯერი-გიგა, თანამშრომელი-გაგა)
- 4) პროცესის ნაკადის უმაღლესი დონის შექმნა
- 5) მასალების გვერდის შექმნა (მასალები საჭიროა ნაკადებისათვის სახელების და ტიპების მისანიჭებლად)
- 6) რეფერირების (reference) გვერდის შექმნა და სქემაზე დამატება
- 7) სიბრტითი სქემის (ნახაზის) შექმნა
- 8) პროცესისათვის თვისებების მინიჭება
- 9) BPMN–ს ფორმების შემსრულებლების დანიშვნა
- 10) ხილული თვისებების შექმნა
- 11) ფუნქციის ნაკადის გვერდი შექმნა
- 12) მატერიალური ნაკადების ფორმების დამატება

ლიტერატურა

1. Business Process Model and Notation (BPMN), *Version 1.2* OMG. <http://www.omg.org/spec/BPMN/1.2>
2. Tutorial BPMN Modeler for Visio. Interfacing Technology Corporation.
3. Business Process Modeling Notation, BPMN. <http://ru.wikipedia.org/wiki/BPMN>.
4. BPMN 1.2 tutorial. http://edutechwiki.unige.ch/en/BPMN_1.2_tutorial.