

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Využitie BPMN v Windows Workflow Foundation

Using BPMN in Windows Workflow Foundation

Zadání diplomové práce

| | |
|--------------------|--|
| Student: | Bc. Martin Bielik |
| Studijní program: | N2647 Informační a komunikační technologie |
| Studijní obor: | 2612T025 Informatika a výpočetní technika |
| Téma: | Využití BPMN ve Windows Workflow Foundation Using BPMN in Windows Workflow Foundation |
| Jazyk vypracování: | čeština |

Zásady pro vypracování:

Business Process Model and Notation (BPMN) je standardní grafickou notací pro modelování podnikových procesů. Procesní diagram je velmi podobný diagramu aktivit z UML. Jde o rozšířený standard v této oblasti. Windows Workflow Foundation (WF) je infrastruktura pro vytváření a spouštění workflow. Integrální součástí je běhové prostředí, které je schopno provádět definované workflow na platformě .NET. Workflow v této technologii je možné definovat pomocí třech proprietárních typů diagramů (aktivitní, stavový a vývojový diagram). Hlavním cílem práce bude vytvořit komponentu, která umožní spouštět workflow definovaný v notaci BPMN v prostředí Windows Workflow Foundation.

Cíle práce lze shrnout v těchto bodech:

1. Seznamte se s technologií Windows Workflow Foundation a notací BPMN.
2. Prozkoumejte existující workflow systémy používající notaci BPMN.
3. Vytvořte komponentu, která primárně umožní vytvořit workflow v notaci BPMN a spouštět takto vytvořené instance na běhovém prostředí WF.
4. Realizujte vhodnou množinu podpůrných funkcí, jako je poskytování informací o probíhajících procesech, perzistentní uložení těchto procesů, správa verzí jednotlivých instancí a podobně.
5. Analyzujte a případně realizujte další vhodná rozšíření jako je import-export vytvořeného modelu pro existující systémy používající BPMN, převod modelů z WF do notace BPMN nebo integrace vytvořené komponenty do nástroje Visual Studio.

Řešení bude využívat platformu .NET, předpokládá se použití programovacího jazyka C#.

Seznam doporučené odborné literatury:

- [1] White, Bayer: Pro WF 4.5. Apress, 2013, ISBN 978-1-4302-4383-0, DOI 10.1007/978-1-4302-4384-7
- [2] Object Management Group: Business Process Model And Notation. URL: <http://www.omg.org/spec/BPMN/>

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Marek Běhálek, Ph.D.**

Datum zadání: 01.09.2017

Datum odevzdání: 30.04.2018




doc. Ing. Jan Platoš, Ph.D.
vedoucí katedry

prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 20. apríla 2018


.....

Souhlasím se zveřejněním této diplomové práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v magisterských programech VŠB-TU Ostrava.

V Ostravě 20. apríla 2018


.....

Rád by som na tomto mieste poďakoval Ing. Marek Běhálek, Ph.D. a všetkým, ktorý mi s prácou pomohli, pretože bez nich by táto práca nevznikla.

Abstrakt

Modelovanie business workflow je v dnešnej dobe neodmysliteľnou súčasťou každého podniku. Business workflow sa využívajú pre znázornenie chodu podniku, či priamo spúšťajú chod podniku. Aktuálne sú dostupné dve rozsiahle notácie vykonávajúce tvorbu workflow - BPMN a WF. Dobrým nápadom by teda bolo preskúmať možnosti konverzií medzi nimi. Tým sa zaoberá táto diplomová práca. Začiatok práce bude zameraný na technológie a ich možnosti. Ďalej predostriem požiadavky a problematiku prevodu. Praktická časť práce je zameraná na postup riešenia, ako aj na hlavný algoritmus prevodu. Diplomová práca ďalej hovorí o nástrojoch, ktoré majú podobnú funkcionality a kde porovnam svoje výsledky s ich výsledkami. Spomenuté sú taktiež možné rozšírenia nástroja, ako aj obtiažnosť týchto rozšírení. V závere je zhodnotený prínos práce a splniteľnosť podmienok.

Kľúčové slová: workflow, diagram, BPMN, element, migrácia, cyklus, paralelizmus

Abstract

Modeling of business workflow is nowadays an essential part of every business. Business workflow is used to illustrate the operation of the business or to directly start the operation of the business. Currently, there are available two extensive notations which carry out the creation of workflow - BPMN and WF. Thus, a good idea would be to explore the possibilities of conversions between them. The diploma thesis deals with this subject. The beginning of the work will be focused on technologies and their possibilities. Further, I will outline the requirements and the issue of the transfer. The practical part of the work is focused on the solution process as well as on the main transfer algorithm. The diploma thesis further mentions tools that have a similar functionality and where I compare my results with their results. There are also mentioned possible extensions of the tool as well as the difficulty of these extensions. In the conclusion, benefits of the work and the fulfillment of conditions are evaluated.

Key Words: workflow, diagram, BPMN, element, migration, cycle, parallelism

Obsah

| | |
|--|-----------|
| Seznam použitých zkratk a symbolů | 10 |
| Zoznam obrázkov | 11 |
| Zoznam tabuliek | 13 |
| Seznam výpisů zdrojového kódu | 14 |
| 1 Úvod | 15 |
| 2 Workflow | 16 |
| 2.1 Workflow Management System(WfMS) | 16 |
| 2.2 Workflow application system | 18 |
| 2.3 Workflow engine | 18 |
| 2.4 Business Process Model and Notation(BPMN) | 19 |
| 2.5 Windows Workflow Foundation | 22 |
| 2.6 Rozdiely medzi notáciami BPMN a WF | 26 |
| 3 Detailnejší prehlád Business Process Model and Notation | 28 |
| 3.1 Aktivity | 28 |
| 3.2 Udalosti | 29 |
| 3.3 Brány | 30 |
| 3.4 Konektory | 31 |
| 4 Detailnejší prehlád Windows Workflow Foundation | 33 |
| 4.1 Sekvenčný workflow | 33 |
| 4.2 State Machine | 34 |
| 4.3 Flow Chart | 35 |
| 5 Analýza problémov migrácie | 36 |
| 6 Špecifikácia požiadavkov | 39 |
| 6.1 Účel | 39 |
| 6.2 Rozsah projektu a funkcie systému | 39 |
| 6.3 Kontext systému | 39 |
| 6.4 Prehlád funkcií | 40 |
| 6.5 Rozhrania | 40 |
| 6.6 Jednotlivé špecifické požiadavky rozpísané do bodov | 40 |

| | | |
|-----------|--|-----------|
| 7 | Analýza požiadaviek a návrh riešenia | 42 |
| 7.1 | Popis algoritmu prevodu | 42 |
| 8 | Implementácia a Testovanie | 49 |
| 8.1 | Book lending process | 49 |
| 8.2 | Shipment Process of a Hardware Retailer | 51 |
| 9 | Možnosti prevodu a obmedzenia | 54 |
| 10 | Existujúce nástroje a ich porovnanie s naimplementovaným riešením | 56 |
| 10.1 | Príklad 1 | 56 |
| 10.2 | Príklad 2 | 58 |
| 11 | Záver | 61 |
| 11.1 | Možnosti rozšírenia | 61 |
| | Literatura | 63 |

Seznam použitých zkratek a symbolů

| | |
|--------|--|
| BPMN | – Business Process Model and Notation |
| BPMI | – Business Process Management Initiative |
| OMG | – Object Management Group |
| WfMS | – Workflow Management System |
| BPA | – Business Process Automation |
| DDAF | – Department of Defense Architecture Framework |
| PMC | – Process Modeling Conformance |
| BPD | – Business Process Definition |
| WF | – Windows Workflow Foundation |
| ISV | – Independent Software Vendor |
| WF API | – WIndows Workflow Foundation Application Programming Inter- face |
| XAML | – Extensible Application Markup Language |
| BPM | – Business Process Management |
| WOE | – Workflow Orchestration Engines |
| REST | – Representational State Transfer |
| JVM | – Java virtual machine |
| RAM | – Random-access memory |
| WCF | – Windows Communication Foundation |

Zoznam obrázkov

| | | |
|----|---|----|
| 1 | Logo BPMN a logo OMG[3] | 20 |
| 2 | Typy workflow v BPMN notácií[2] | 21 |
| 3 | Ukážka elementov v BPMN workflow | 22 |
| 4 | Logo Microsoft Windows Workflow Foundation | 23 |
| 5 | Ukážka control flow | 25 |
| 6 | Ukážka Flow Chart | 25 |
| 7 | Ukážka State Machine | 26 |
| 8 | Základné elementy BPMN | 28 |
| 9 | Základné druhy aktivít [1] | 28 |
| 10 | Ukážka typov aktivít | 29 |
| 11 | Zložený a rozložený podproces | 29 |
| 12 | Základné typy udalostí | 29 |
| 13 | Rozšírené typy udalostí | 30 |
| 14 | Základné typy brán | 30 |
| 15 | Prehľad spojovačov | 31 |
| 16 | Sekvenčný a podmienený tok | 31 |
| 17 | Vnútna štruktúra WF | 33 |
| 18 | Vizuálne znázornenie vzniknutého problému pri tvorbe workflow | 38 |
| 19 | Kontext systému | 39 |
| 20 | User Interface migračného nástroja | 40 |
| 21 | Príklad zložitého poddiagramu v notácií BPMN | 43 |
| 22 | Triedny diagram zahrňujúci štruktúru Generického modelu | 44 |
| 23 | Spracovávanie poddiagramov popri interakcií s užívateľom | 46 |
| 24 | Algoritmus prevodu workflow | 47 |
| 25 | Book lending process v notácií BPMN | 49 |
| 26 | Vnútna špecifikácia Update database elementu | 50 |
| 27 | Výsledok prevodu nástroja do notácie WF | 50 |
| 28 | Vnútna štruktúra poddiagramu | 51 |
| 29 | Shipment Process of a Hardware Retailer v notácií BPMN | 51 |
| 30 | Výsledný workflow po migrácií | 52 |
| 31 | Výsledný poddiagram po migrácií | 53 |
| 32 | Základný workflow v notácií BPMN | 56 |
| 33 | Výsledný základný workflow po migrácií v nástroji HeadSight | 57 |
| 34 | Výsledný základný workflow po migrácií mojím nástrojom | 57 |
| 35 | Výsledný základný poddiagram po migrácií mojím nástrojom | 58 |
| 36 | Zložitejší workflow v notácií BPMN | 58 |
| 37 | Výsledný zložitejší workflow po migrácií HeadSight | 59 |

| | | |
|----|---|----|
| 38 | Výsledný zložitější workflow po migracích mojím nástrojom | 59 |
|----|---|----|

Zoznam tabuliek

| | | |
|---|--|----|
| 1 | Prvky Sekvenčného workflow | 34 |
| 2 | Prvky State Machine workflow | 34 |
| 3 | Prvky Flow Chart workflow | 35 |

Seznam výpisů zdrojového kódu

| | | |
|---|--|----|
| 1 | Metóda detekujúca cyklenie workflow | 37 |
| 2 | Nález jedného stavu v notácií BPMN | 43 |
| 3 | Detekcia paralelizmu v generickom modeli | 45 |

1 Úvod

V dnešnej dobe sa ani medzinárodné korporácie nezaobídu bez business workflow. Slúži na zvýšenie efektivity riadenia výroby, nasadenia či samotný chod podniku. Mojou motiváciou v tejto práci bolo zvýšenie možností rastu efektivity podnikov. Microsoft technológia ma zaujala už od začiatku môjho štúdia na univerzite, preto chcem preskúmať workflow na platformách Microsoft. Najznámejšia notácia, ktorá sa aktuálne používa je Business Process Model and Notation (BPMN). Dokonca aj Microsoft Corporation vyvíja svoju vlastnú notáciu Windows Workflow Foundation (WF), ktorá sa značne líši od BPMN.

Táto diplomová práca sa zaoberá možnosťami konverzie, ako aj konverziou samotnou z notácie BPMN do notácie WF. V úvode práce sa opíše workflow samotný ako aj Workflow Management System(WfMS), workflow application system či workflow engine. Neskôr budú opísané dané technológie a ich možnosti. Podrobne sa charakterizujú všetky elementy, s ktorými je možné stretnúť sa v notáciách BPMN ako aj WF. Následne sa definujú špecifické požiadavky kladené na nástroj. Neskôr budú predstavené možné problémy migrácie, ich riešenia a priložená bude aj ukážka implementácie v podobe kódu. Zdôraznené budú aj nástroje, ktoré majú podobnú funkciu a porovnajú sa testovaním na jednotlivých príkladoch v kapitole 13. V závere tejto práce sa zhodnotí prínos získaných skúseností a opíše splniteľnosť podmienok. Doplňa sa aj body možného rozšírenia aplikácie pre tých, ktorí by chceli s prácou pokračovať.

2 Workflow

Moderná história workflow siaha až k Frederickovi Taylorovi a Henrymu Ganttovi, ktorý tento termín použili ako prví a to v roku 1921.[21]

Vo všeobecnosti môže byť workflow považovaný za tok procesov alebo úloh, ktoré prinášajú určitý výsledok. workflow sa často zaoberajú dokumentmi, rôznymi procesmi a môžu nadobudnúť funkciu duplikácie dokumentov, ktoré spracúvajú. Workflow môžu byť manuálne s papierovými dokumentmi, ktoré sa posielajú medzi ľuďmi pracujúcimi na inom pracovisku v kancelárii. Každá osoba vystupuje ako objekt podľa definície úlohy a určených pravidiel.

Práca sa zaoberá workflow ako softvérovým systémom, ktorý definuje činnosti a pravidlá, ktoré sa používajú v súčasnej praxi:

- pravidlá môžu byť vyjadrené deklaratívne alebo v kóde
- činnosti môžu byť úplne automatizované alebo môžu zahŕňať interakciu človeka
- workflow môžu byť distribuované medzi viacerými počítačmi v rôznych lokáciách
- workflow sú typicky reprezentované graficky

Moderné organizácie využívajú workflow na koordináciu úloh medzi ľuďmi a synchronizáciu dát medzi systémami s konečným cieľom zlepšiť organizačnú efektivitu, schopnosť reagovať a prinášať zisk.

Workflow sa môže chápať ako postup, ktorý hovorí ako práca plynie z jedného stupňa do druhého či už prostredníctvom kolegu, nástroja alebo iného procesu. Úplný workflow je možné vykonať samostatne (napríklad písanie, editovanie a publikovanie blogového príspevku) alebo môže zahŕňať viacero účastníkov(napríklad fakturáciu klienta).[1]

2.1 Workflow Management System(WfMS)

Riadenie workflow je koordinácia úloh, ktoré tvoria prácu organizácie. Týmto “workflow“ sa rozumie postupnosť úloh, ktoré sú súčasťou väčšej úlohy a niekedy sú synonymom pre “obchodný proces“. Cieľom workflow je dosiahnuť určitý výsledok a účelom riadenia workflow je dosiahnuť lepšie výsledky podľa zadaných cieľov. Riadenie pracovného toku je v prvom rade riadiacou disciplínou zameranou na štruktúru práce v rámci nejakej organizácie a ako tímy spolupracujú na dokončení tejto práce. Správa workflow v zásade nevyžaduje softvér, ale v praxi sa softvérové nástroje používajú na sledovanie práce a automatizáciu jej častí.

Riadenie workflow sa zaoberá prácou, ktorú ľudia robia, ale úlohy môžu byť tiež automatizované a vykonávané IT systémami - počítačovým softvérom. Riadenie toku workflow môže preto zahŕňať integráciu systémov IT s cieľom zdieľať údaje medzi workflow a inými systémami IT v rámci organizácie.[22]

Medzi najvýznamnejšie WfMS patrí:

- Apache ODE
- Bonita BPM
- Camuda BPM
- IBM BPM
- jBPM
- Pyrus
- Windows Workflow Foundation(WF)

Z praktického hľadiska som si vybral Camuda BPM pre tvorbu príkladov v notácii BPMN a samozrejme WF ako cieľový WfMS.

WfMS je softvér, ktorý podporuje riadenie workflow automatizáciou častí koordinácie riadenia, ktoré sú spoločné pre rôzne pracovné postupy. V prípade prijímania zamestnancov WFMS spravuje informácie o každom procese prijímania a dokumentoch, ako je napríklad životopis kandidáta, a koordinuje pridelenie úloh pre pracovníkov na prijímanie pracovných miest, anketárov a ostatných pracovníkov podľa preddefinovaného pracovného postupu. WfMS má byť univerzálnym nástrojom, ktorý môže potenciálne podporovať akýkoľvek pracovný tok.[22] WfMS zvyčajne poskytujú nasledujúce funkcie:

- Grafický návrhár procesov pre špecifikáciu pracovného postupu, prípadne s použitím normy BPMN
- Tvorba foriem pre vytváranie užívateľského rozhrania pre zadávanie štruktúrovaných dát workflow
- Integrácia dokumentov na odovzdanie neštruktúrovaných údajov do súborov
- “Doručená pošta“, ktorú ľudia môžu použiť na objavovanie úloh
- Upozornenia, ktoré informujú účastníkov o zmenách v pracovných tokoch, napríklad pri priradení úlohy
- Prístupové povolenia, ktoré možno nastaviť, aby určili, kto môže čo robiť
- Reporting, ktorý poskytuje prehľad o prebiehajúcej a dokončenej práci
- Splnenie integračných bodov potrebných pre integráciu systémov s workflow

2.2 Workflow application system

Aplikácia workflow je softvérová aplikácia, ktorá automatizuje, aspoň do určitej miery, proces alebo procesy. Procesy zvyčajne súvisia s podnikaním, ale môžu to byť akékoľvek procesy, ktoré vyžadujú automatizáciu sérií krokov prostredníctvom softvéru. Niektoré kroky procesu môžu vyžadovať zásah človeka, napríklad schválenie alebo vývoj vlastného textu, ale funkcie, ktoré môžu byť automatizované, by mali byť spracované aplikáciou. Pokročilé aplikácie umožňujú používateľom zaviesť do operácie nové komponenty.

Zvážme napríklad nákupnú objednávku, ktorá sa presunie cez rôzne oddelenia na autorizáciu a prípadný nákup. Objednávku možno presunúť z oddelenia na oddelenie na schválenie automaticky. Po získaní všetkých povolení sa žiadateľ o objednávku oznámi a dostane povolenie. Proces workflow môže zahŕňať neustálu zmenu a aktualizáciu. Napríklad bežný schvaľovač objednávok môže byť na dovolenke, v takom prípade žiadosť bude vyžadovať súhlas od náhradných schvaľovateľov.[23]

2.3 Workflow engine

Workflow engine je softvérová aplikácia, ktorá riadi obchodné procesy. Je kľúčovou súčasťou technológie workflow a zvyčajne využíva databázový server.

Technológia workflow riadi a monitoruje stav činností v workflow, ako je napríklad spracovanie a schválenie formulára žiadosti o úver a určuje, na ktorú novú aktivitu sa prechádza podľa definovaných procesov (pracovných tokov).[24] Činnosti môžu byť čokoľvek, od uloženia formulára žiadosti v systéme správy dokumentov k odosielaniu upomienkových e-mailov používateľom alebo eskalácii položiek po splatnosti do manažmentu. Technológia workflow uľahčuje tok informácií, úloh a udalostí. Workflow engine môžu byť tiež označované ako Workflow Orchestration Engines(WOE).[25]

Workflow engine majú prevažne tri funkcie:

- Overenie aktuálneho stavu: Skontroluje, či je príkaz platný pri vykonávaní úlohy
- Určenie autority používateľov: Skontroluje, či je aktuálny používateľ oprávnený vykonať úlohu
- Spúšťanie podmieneného skriptu: Po úspešnom absolvovaní predchádzajúcich dvoch krokov začne workflow engine vyhodnocovať stavový skript, v ktorom sa vykonávajú tieto dva procesy
 - Ak je podmienka pravdivá, workflow vykoná úlohu a ak sa vykonanie úspešne dokončí, vráti sa úspech
 - Ak podmienka nieje pravdivá, hlási chybu spúšťania a vrátenia zmeny[26]

Workflow engine na spracovanie workflow je základnou technikou pre softvér na pridelovanie úloh, ako je napríklad riadenie obchodných procesov, v rámci ktorého workflow engine prideluje

úlohy rôznym vykonávateľom pri komunikácii údajov medzi účastníkmi. Workflow engine na spracovanie workflow môže vykonávať ľubovoľnú postupnosť krokov, napríklad analýzu údajov o zdravotnej starostlivosti.[27]

2.3.1 Camuda engine

Camuda Workflow Engine automatizuje procesné diagramy BPMN 2.0 pre orchestráciu (mikro-) služby, toky úloh človeka alebo oboje. Workflow engine vykonáva väčšinu symbolov definovaných v štandarde BPMN 2.0. Prístup k nástroju Workflow engine sa môže získať prostredníctvom Representational State Transfer(REST), aby sa mohla spustiť procesná inštancia či dokončiť úloha. Workflow Engine vyžaduje menej ako 3 MB Random-access memory(RAM), môže byť spustený v ľubovoľnom Java Virtual Machine(JVM) a je dodávaný s rozšírenou integráciou pre rôzne Java runtime kontajnery.

2.3.2 WF engine

WF engine umožňuje spúšťanie workflow ako aj prvkov samotných. Prístup k nástroju WF engine je možný aj prostredníctvom Windows Communication Foundation(WCF) Workflow Service Application. Workflow engine základnej WF konzolovej aplikácie vyžaduje 9 MB RAM a môže byť spustený na akomkoľvek prostredí s podporou minimálne .NET framework 4.0. [28]

2.4 Business Process Model and Notation(BPMN)

BPMN bol vyvinutý Business Process Management Initiative (BPMI). Prvú verzia bola vydaná v marci 2007. V júni 2008 sa BPMI spojilo s Object Management Group (OMG). Špecifikačný dokument bol vydaný OMG vo februári 2008. Druhá verzia BPMN bola vyvinutá v roku 2011 a aktuálna verzia špecifikácie bola vydaná v decembri 2013. V nasledujúcich pár rokoch sa BPMN stal štandardom pre modelovanie business workflow . Časť úspechu je aj dôsledkom toho, že BPMN ponúka známy pohľad obchodným analytikom a súčasne poskytuje výkonný model obchodných workflow . Modely BPMN 2.0 sa dajú použiť na komunikáciu a výmenu obchodných workflow , ako aj na skutočnú implementáciu workflow . BPMN bol založený na metóde vývojových workflow , ktorá je veľmi podobná workflow aktivít z Unified Modeling Language (UML). [2]

Cieľom BPMN je podporovať riadenie obchodných workflow pre technických aj podnikových užívateľov. Poskytne intuitívny zápis podnikovým používateľom a zároveň je schopný reprezentovať zložitú sémantiku workflow. BPMN je primárne určený pre obchodných analytikov, ktorí vytvárajú a zdokonaľujú procesy, technickí vývojári zodpovední za ich implementáciu a obchodní manažéri, ktorý ich monitorujú a riadia. V dôsledku toho BPMN slúži ako spoločný jazyk, čím prekonáva komunikačný problém, ktorý sa často vyskytuje medzi návrhom a implementáciou podnikových procesov. Štandardný BPMN poskytne podnikom schopnosť porozumieť ich inter-

ným obchodným postupom v grafickej notácii. Okrem toho grafická notácia uľahčí pochopenie spolupráce v oblasti výkonnosti a obchodných transakcií medzi organizáciami [3].

BPMN workflow sa štandardne skladajú zo širokej škály elementov. Súčasná inkarnácia BPMN pozostáva z 52 graficky odlišiteľných elementov: 41 objektov toku, 6 spájajúcich objektov, 2 zoskupovacích objektov a 3 artefaktov. Túto škálu som pre lepšie pochopenie rozdelil do dvoch skupín:

- jednoduché
- zložité

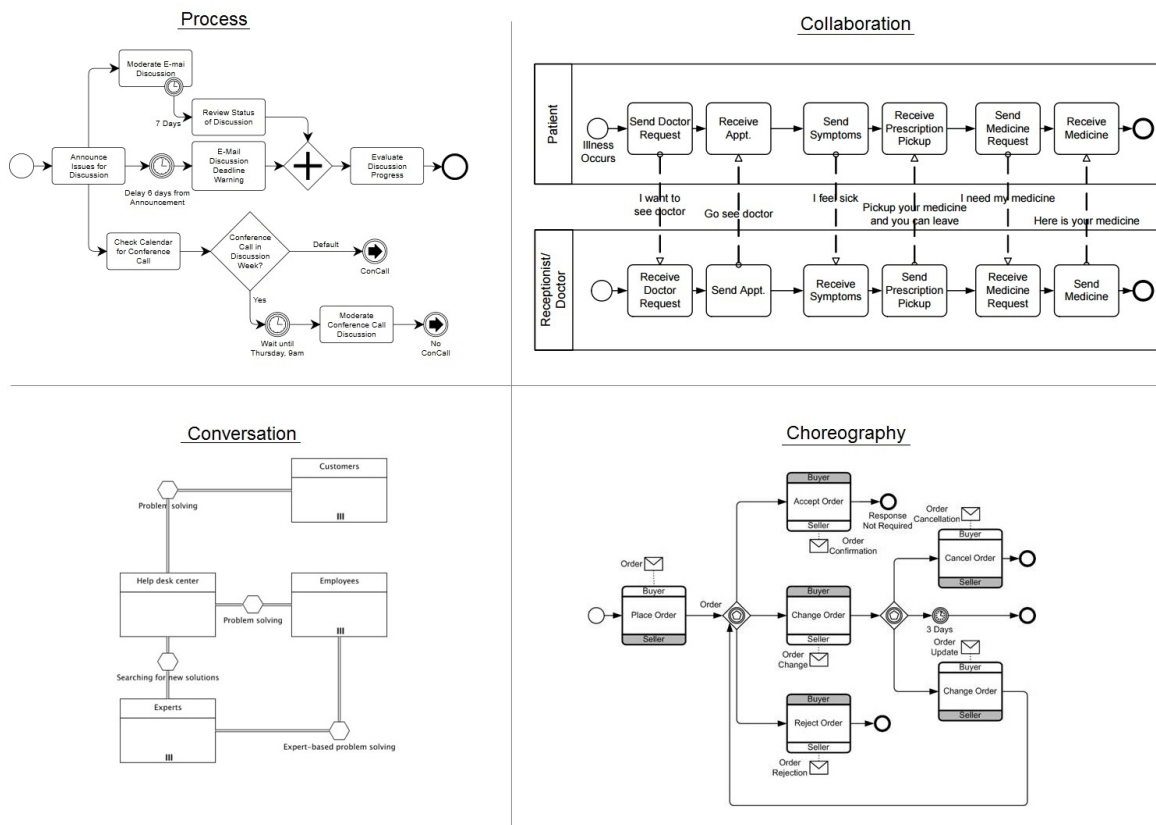


Obr. 1: Logo BPMN a logo OMG[3]

Práca sa na začiatku zaoberala jednoduchými elementami. Neskôr však bola absencia zložitých elementov značne viditeľná a preto sa museli začleniť už do prvej fáze vývoja migračného nástroja.

V BPMN sa nachádzajú 4 typy workflow:

- spolupracovné workflow obsahujúce pooly a tok správ
- konverzačné workflow obsahujúce pooly, konverzácie a konverzačné linky
- choreografické workflow obsahujúce interakcie medzi procesmi a tokmi správ



Obr. 2: Typy workflow v BPMN notácii[2]

Ako alternatíva k úplnému pochopeniu modelovania workflow sú definované tri podtriedy:

- opisné
- analytické
- spoločný spustiteľný súbor

Opisné podtriedy workflow Opisné metódy sa zaoberajú viditeľnými prvkami a atribútmi používanými pri modelovaní na vysokej úrovni. To by malo byť pohodlné pre analytikov, ktorí použili nástroje Business Process Automation (BPA) Flow Charting.[4]

Analytické podtriedy workflow Analytické metódy obsahujú všetky popisné elementy a celkove približne polovicu konštruktov z plnej triedy konformácií modelovania workflow. Sú založené na skúsenostiach získaných pri analýze užívateľských vzorov v Department of Defense Architecture Framework (DDAF) a plánovaných štandardizovaniach tohoto rámcu.[4]

Spoločný spustiteľný súbor Spoločný spustiteľný súbor sa sústreďuje na to, čo je potreba aby sa mohol spustiť daný workflow.[4]

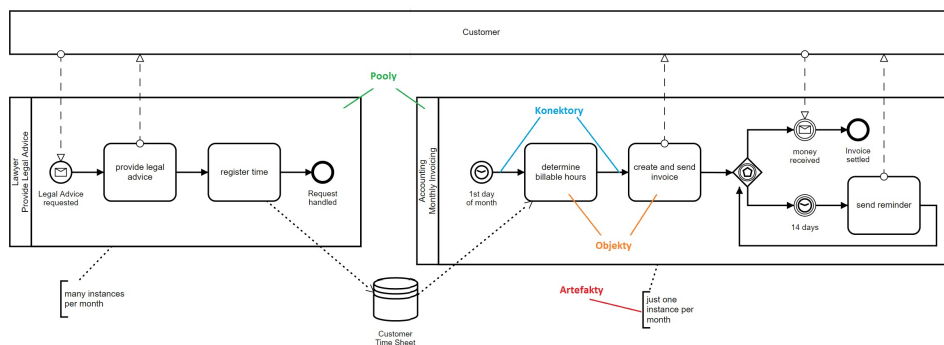
Opisné aj analytické podtriedy sú zamerané na viditeľné elementy a minimálnu škálu pomocných prvkov ako napr. atribútov. Elementy a atribúty, ktoré nie sú v týchto troch podtriedach, sú obsiahnuté v Process Modeling Conformance (PMC).

Business Process Diagram (BPD) pozostáva zo súboru grafických prvkov. Tieto prvky umožňujú ľahký rozvoj jednoduchých workflow, ktoré používajú obchodní analytici (napr. vývojový workflow).

Prvky boli vybrané tak, aby boli odlišiteľné od seba a využívali známe tvary väčšine modelárov. Napríklad aktivity sú obdĺžniky a rozhodnutia sú diamanty. To by malo zdôrazniť, že jedným z vodičov pre vývoj BPMN je vytvoriť jednoduchý mechanizmus na vytváranie modelov obchodných workflow a súčasne na zvládnutie zložitosti workflow. Prístup k riešeniu týchto dvoch protichodných požiadaviek bolo zorganizovať grafické aspekty notácie do špecifických kategórií. Toto poskytuje malý súbor kategórií notácie tak, aby čitateľ Business Process Definition (BPD) mohol ľahko rozpoznať základné typy prvkov a pochopiť workflow.[5]

V BPMN sa nachádzajú štyri základné kategórie elementov:

- objekty
- konektory
- pooly
- artefakty



Obr. 3: Ukážka elementov v BPMN workflow

2.5 Windows Workflow Foundation

WF bol poprvý krát vydaný s .NET rámcem 3.0 v roku 2006 a následne aktualizovaný s .NET verziami. Spolu s verziou .NET 3.5 boli tieto verzie užitočné najmä pre nezávislých dodávateľov softvéru Independent Software Vendor (ISV), ale nestali sa hlavnou technológiou podnikových vývojárov. Vo verzií .NET 4.0 sa jej tvorcovia snažia práve toto zmeniť. Framework .NET 4.0 sa snaží prvky v workflow reprezentovať ako triedy, čím značne zjednodušuje prístup k prispôbeniu potrebám vývojára. Hlavným cieľom tejto verzie je urobiť WF štandardnou časťou programovacích workflow pre všetkých .NET vývojárov.

WF je rámec, ktorý pomáha vytvárať a spúšťať workflow aplikácie pod Windows platformou. Rámec WF je prepojený s technológiou .NET ako aj WCF (Windows Communication Foundation) a WPF (Windows Presentation Foundation). Windows Workflow Foundation Application

Programming Interface (WF API) obsahuje podporu C#, Visual Basic, špeciálny workflow kompilér, workflow krokovaciu podporu a workflow dizajnér. WF môže byť vytvorené len pomocou kódu (najviac využívaný proces v tomto migračnom nástroji), alebo pomocou Extensible Application Markup Language (XAML) značiek. Dizajnér a model WF sú rozširiteľné a dovoľujú vývojárom vytvorenie používateľských aktivít, ktoré zahŕňujú konkrétne funkcie workflow. Aktivity môžu mať štandardnú funkcionálnu, alebo užívateľskú implementáciu.[11]



Obr. 4: Logo Microsoft Windows Workflow Foundation

Windows Workflow Foundation pozostáva z:

- aktivity
- workflow návrh
- základné knižnice aktivít
- workflow (Workflow Instance)
- bežiacie jadro (Workflow Runtime)
- bežiacie služby
- hosťujúci proces

Aktivity(Activities) sú pracovné jednotky, ktoré vykonávajú určitú činnosť ako napr. prevedenie užívateľského či systémového kódu, komunikácie, udalosti, logické operácie a iné.

Workflow návrh(Workflow Designer) slúži ako grafické rozhranie pre tvorbu dizajnu workflow. V návrhu workflow je možné umiestňovať elementy workflow a meniť ich veľkosť. Workflow návrh je vytvorený pomocou WPF. To zvyšuje skúsenosti a výkonnosť dizajnérov aj pri veľkých a zložitých workflow.

Základné knižnice aktivít(Base Activity Library) je implicitne podporovaná nástrojom Visual Studio. Tieto aktivity slúžia k vytvoreniu workflow.

Workflow(Workflow Instance) obsahuje zhluk aktivít či ostatných prvkov, ktoré tvoria celok procesu workflow. Workflow instancie je vytvorená aj udržiavaná workflow bežiacim jadrom.

Pracovné toky sú konštruované ako súbor činností, ktoré vykoná **bežiacie jadro(Workflow Runtime)**. Riadiaci modul musí bežať v hostiteľskej aplikácii. Toto jadro taktiež pridáva možnosť používať služby, ako napríklad mechanizmus na komunikáciu so softvérom mimo workflow. Bežiacie jadro vykonáva každý workflow a riadi stav workflow počas celej jeho životnosti. Runtime WF je však knižnica, a preto musí byť spustená v niektorom hostiteľskom procese. Namiesto poskytovania jediného požadovaného hostiteľa môže systém Windows WF umožniť, aby bežiacie jadro (a akékoľvek pracovné postupy, ktoré vykonával) boli umiestnené v takmer ľubovoľnom procese Windows, od jednoduchej konzoly alebo aplikácie Windows Forms až po komplexný server navrhnutý s workflow.

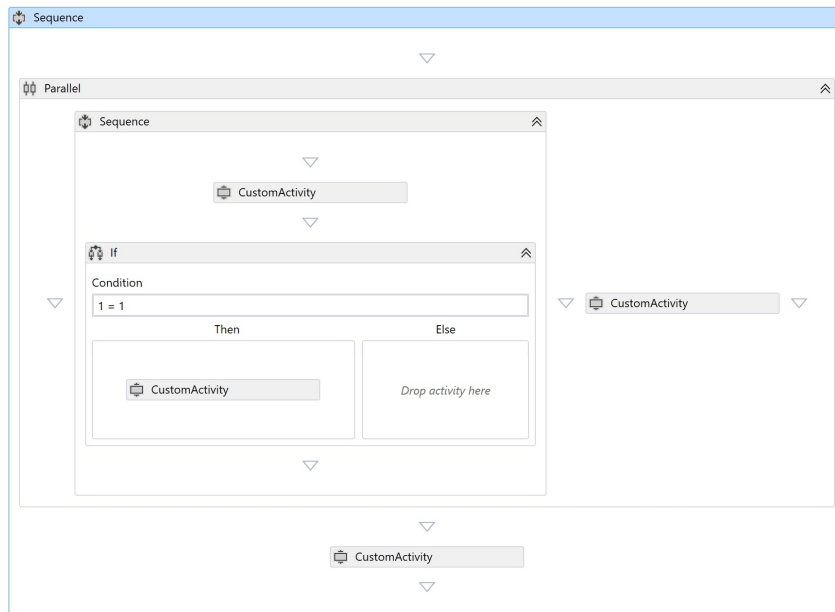
Bežiacie služby(Runtime Services) je skupina služieb, ktorá dopĺňa workflow aktivity. Medzi tieto služby môžeme zaradiť sledovanie, spracovanie vlákien, lokálnu komunikáciu, perzistenciu a iné. Tieto služby ako napríklad perzistencia, umožňujú uložiť instanciu WF do externého úložného priestoru, ako je databáza alebo XML súbor.

Hostujúci proces(Host Process) je aplikácia, ktorá hostuje bežiacie jadro a vykonáva ostatné workflow procesy. Bežiacie služby pracujú v hostujúcom procese a poskytujú dodatočnú funkčnosť bežiacemu jadru pri riadení vykonávania workflow.[12]

Typy WF workflow by sa dali rozdeliť do troch základných skupín:

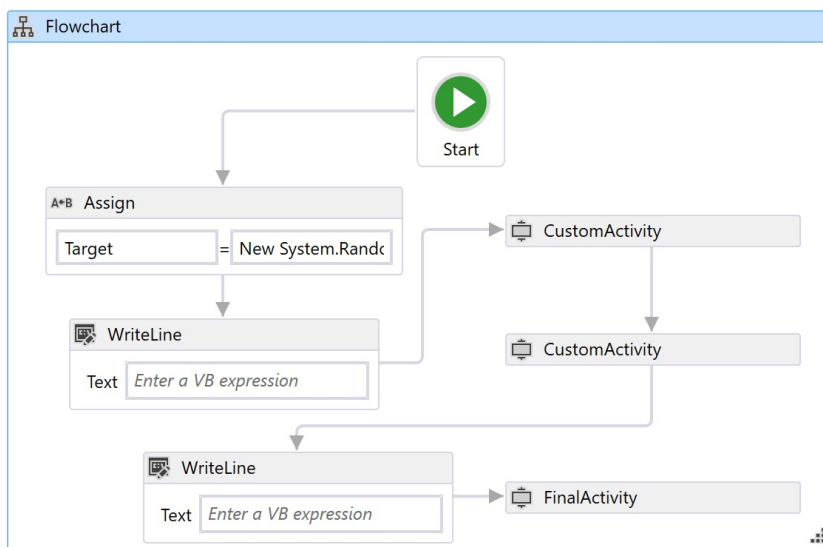
- control flow
- Flow Chart
- State Machine

Control flow v .NET frameworku 4.6.1 poskytuje niekoľko činností na riadenie toku vykonávania v workflow. Niektoré z týchto aktivít (napríklad Switch a If) implementujú štruktúry riadenia toku podobné tým, ktoré sa nachádzajú v programovacích prostrediach, ako je Visual C#, zatiaľ čo iné (napríklad Pick) modelujú nové programové štruktúry.[13]



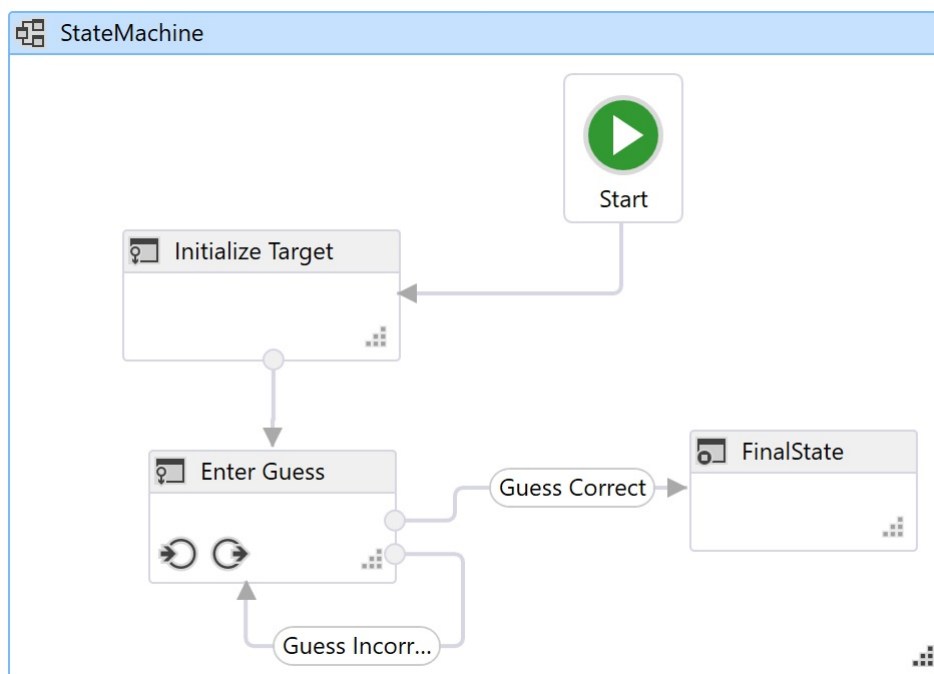
Obr. 5: Ukážka control flow

Flow Chart aktivita je súbor činností, ktoré sa majú vykonať. Vývojové schémy obsahujú aj prvky riadenia toku, ako FlowDecision a FlowSwitch <T>, ktoré riadia vykonávanie medzi obsiahnutými aktivitami na základe hodnôt premenných.[13]



Obr. 6: Ukážka Flow Chart

State Machine workflow poskytujú štýl modelovania, pomocou ktorého môžete modelovať svoj workflow spôsobom riadeným udalosťami. Aktivita State Machine obsahuje stavy a prechody, ktoré tvoria logiku State Machine, a môžu byť použité všade, kde môže byť aktivita použitá.[13]



Obr. 7: Ukážka State Machine

2.6 Rozdiely medzi notáciami BPMN a WF

Prvá vec, ktorú treba mať na pamäti je, že WF nie je nástrojom Business Process Management (BPM), ale skôr rámec, ktorý možno využiť na vytvorenie spoľahlivého a flexibilného súboru obchodných tokov. Ak sa hľadá riešenie, ktoré poskytne veľa vstavaných funkcií, kde nebude nutné robiť žiadne programovanie, ale skôr len dať tvary na list - WF nie je náš nástroj. Poskytuje však veľkú flexibilitu pri definovaní workflow. Súčasná spoločnosť, ktoré disponujú tímom softvérových vývojárov s aspoň jedným vedúcim vývojárom, ktorý vie o BPM podnikaní, môžu stavať vlastné BPM pomocou WF namiesto toho, aby si kúpili kompletnú sadu BPM.

WF je časovo náročné. Navrhnuť a implementovať pracovné postupy pomocou nástroja ako Process Maker, Bonita Software a podobne, je oveľa jednoduchšie, ako vyvinúť niečo podobné pomocou WF od začiatku. Preto, tento migračný nástroj slúži na prevedenie BPMN do WF a nie vytváranie BPMN procesu na platforme .NET od začiatku až do konca.

Potenciálne výhody WF:

- veľmi dobrý výkon vo WF 4.0
- ľahko distribuovateľný v rámci farmy webových serverov

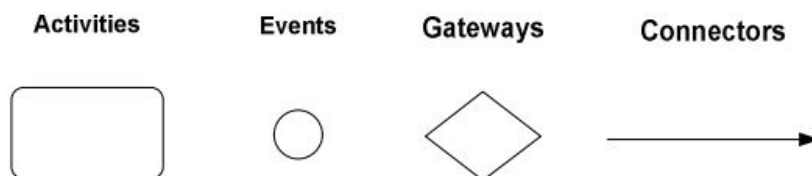
- flexibilita, ktorá dáva silu dosiahnúť a prekonať okrajové prípady workflow

Cieľom práce je aj snaha integrovať s .NET a WF je asi jedinou možnou voľbou. Samozrejme existujú aj iné nástroje, avšak neintegrujú sa tak dobre ako WF.

3 Detailnejší prehľad Business Process Model and Notation

Tato kapitola sa venuje charakterizácií BPMN elementov, keďže diplomová práca sa v globále zameriava na preklad konkrétnych prvkov. Detailne opísané budú 4 základne typy elementov BPMN a to konkrétne:

- aktivity
- udalosti
- brány
- konektory



Obr. 8: Základné elementy BPMN

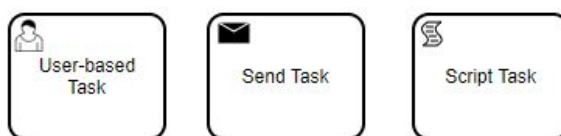
3.1 Aktivity

Aktivita je práca, ktorá je vykonávaná v busines procese. Aktivita môže byť atomická - vykonáva len jednoduchú úlohu, alebo zložitá - skladá sa z dvoch a viac aktivít. Typy aktivít v procesnom modeli sú podproces a úloha. Aktivity sú zaoblené obdĺžniky. Môžu sa vykonať raz alebo môžu mať vnútorne definované slučky.[6]



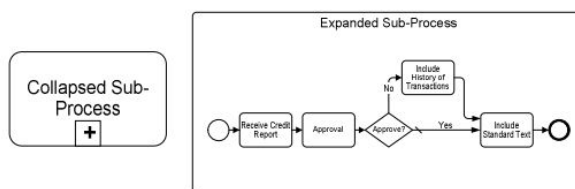
Obr. 9: Základné druhy aktivít [1]

Úlohy sú atomické aktivity, ktoré sú zahrnuté v procesnom modeli. Ten sa využíva všade tam, kde sa práca nedá rozdeliť na menšie diely, tým pádom získavame najjemnejší level procesného detailného modelu. Existujú špecializované typy úloh pre odosielanie a prijímanie alebo vytvorené užívateľom samotným - užívateľsky založené úlohy. Značky alebo ikony sú pridávané do úloh na identifikovanie typu špecializácie.[6]



Obr. 10: Ukážka typov aktivít

Podprocesy dovoľujú hierarchickú štruktúru BPMN workflow. Podprocesy sú zložené aktivity, ktoré sú súčasťou procesného modelu. Základnou vlastnosťou podprocesu je ich rozložiteľnosť na jemnejšiu úroveň detailu. Uzavreté podprocesy nie sú viditeľné v workflow. Znak ‘+’ indikuje, že sa jedná o uzavretý podproces a má jemnejšiu úroveň detailu v externom workflow. Pre otvorené podprocesy platí, že detail je prítomný priamo v základnej úrovni workflow.[6]



Obr. 11: Zložený a rozložený podproces

3.2 Udalosti

Udalosti sú značené ako kruhy. V BPMN workflow indikujú, že sa niečo stalo. Tieto udalosti zmenia tok business workflow a zvyčajne majú spúšťač alebo výsledok. Môžu odštartovať, prerušiť alebo ukončiť beh workflow. Typ okraju opisuje o aký druh udalosti ide.[6] V BPMN workflow sú tri základné druhy udalostí:

- štart
- prerušenie
- koniec

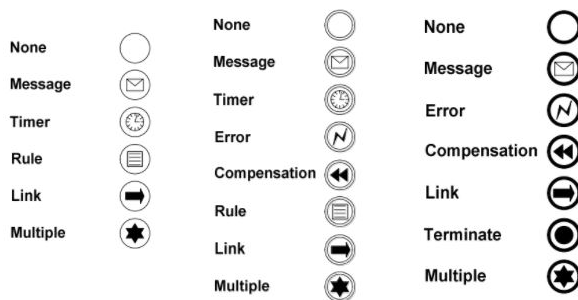


Obr. 12: Základné typy udalostí

Štart udalosti indikuje kde má proces začať. V BPMN sú rôzne spúšťače, ktoré prebehnú za určitých okolností a spustia proces. Žiadna z týchto udalostí nie je použitá na odštartovanie podprocesu.[9]

Prerušovacie udalosti sa spúšťajú po štarte procesu a zároveň pred skončením procesu. Môžu byť vložené v normálnom prúde procesu alebo pripojené na okraji aktivity či podprocesu. V normálnom toku tieto udalosti reprezentujú normálnu operáciu procesu. Môžu reprezentovať výsledok udalosti alebo vytvorenie udalosti. Udalosti pripojené na okraji aktivity či podprocesu sa používajú na spracovanie chýb, spracovanie výnimiek a odškodnenie.

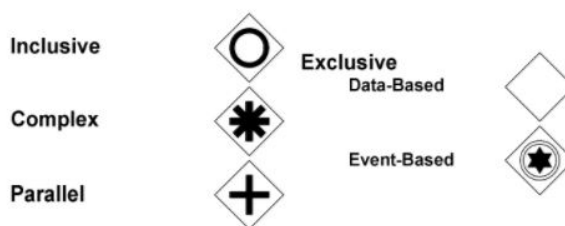
Udalosti indikujúce **koniec** workflow sa označujú plným obrysom. V BPMN sa nachádzajú rôzne “výsledky”, ktoré identifikujú špecifické okolnosti ukončenia workflow. [6]



Obr. 13: Rozšírené typy udalostí

3.3 Brány

Všetky typy brán majú tvar diamantu. Tento diamant reprezentuje v procese miesto, kde treba riadiť tok. Brány sú modelovacie prvky, ktoré sa používajú na riadenie postupnosti tokov. Tieto toky sa vzájomne ovplyvňujú a vylučujú. Rôzne značky v BPMN bránach indikujú iné typy chovania. [6]



Obr. 14: Základné typy brán

Exkluzívne brány (rozhodnutia) sú miesta v procese, pri ktorých postupnosť tokov môže mať dva a viac alternatívnych ciest. Len jedna z možných ciest sa prevedie. V BPMN sa nachádzajú dva typy rozhodovacích mechanizmov:

- na základe dát,
- na základe udalostí.

Brány založené **na dátovom rozhodovacom mechanizme** sa v workflow používajú najčastejšie. Môžu sa v BPMN nachádzať s znakom “X” alebo bez neho.

Brány založené **na udalostnom rozhodovacom mechanizme** sa rozhodujú podľa udalostí, ktoré vedú z brány. Prvá splnená podmienka prevezme tok aplikácie.[6]

Inkluzívne brány sú značené “O” a identifikujú typ brány, pri ktorom všetky vstupné podmienky musia byť splnené, aby bolo možné pokračovať. Najčastejšie sa vyskytujú dve za sebou, ktoré tok rozdeľujú a následne ho zlúčia.[10]

Komplexné brány sú rozhodovacie brány, kde môže byť nadefinované pokročilejšie chovanie podmienky. Táto pokročilá podmienka môže byť nadefinovaná v rozdeľovacom aj zlučovacom chovaní. Sú značené hviezdou uprostred diamantu. [6]

Paralelné brány sa značia “+” znakom uprostred diamantu. Tieto brány slúžia na spracovanie paralelného procesu. Používajú sa aj na zosynchronizovanie paralelných ciest.

3.4 Konektory

Konektory v BPMN slúžia na prepojenie udalostí, aktivít či brán. V BPMN sú známe tri základné typy:

- sekvenčný tok,
- správový tok
- asociácia.



Obr. 15: Prehľad spojovačov

Sekvenčný tok sa ďalej delí na základný a podmienkový.

Základný sekvenčný tok sa používa na zobrazenie postupnosti aktivít v procese. Zdroj sekvenčného toku musí byť buď udalosť, aktivita alebo brána. Sekvenčný tok nesmie prechádzať cez hranice podprocesu alebo hranice poolu.

Podmienkový sekvenčný tok je značený malým diamantom. Môže mať definovanú podmienku po opustení aktivity. Takáto aktivita musí mať najmenej dva podmienkové sekvenčné toky.[7]



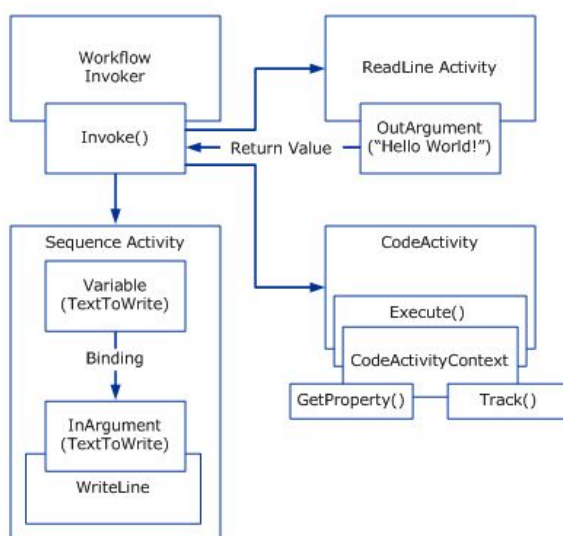
Obr. 16: Sekvenčný a podmienený tok

Správový tok sa používa na zobrazenie správ medzi dvoma entitami, tieto dve entity musia byť pripravené dané správy poslať alebo ich obdržať. Tok správ nie je možný medzi dvoma objektami v jednom pooli. Naproti tomu dva pooly vystupujú ako entity, medzi ktorými je tok správ povolený.

Asociácia sa používa pre priradenie dát, informácií a artefaktov s objektami v procese. Textové anotácie môžu byť priradené k objektom ako asociácia.[8]

4 Detailnejší prehľad Windows Workflow Foundation

Každý pracovný postup Windows Workflow Foundation obsahuje množstvo aktivít, z ktorých každá vykonáva niektoré aspekty funkcie tohto pracovného postupu. WF slúži ako kontajner pre tieto činnosti, ktorý poskytuje spôsob kontroly ich životných cyklov a poradie vykonávania. WF sa usiluje podporovať systémové workflow a ľudské pracovné postupy jednotným spôsobom. Služba WF poskytuje tri vstavané workflow: sekvenčné postupy, ktoré umožňujú vykonávať činnosti v preddefinovanom vzore, postupy State Machine schopné reagovať na vonkajšie udalosti, ktoré sa vyskytujú a postupy Flow Chart, ktoré priamo spúšťajú dané aktivity bez vonkajšieho obalu (napr. Aktivita v State Machine musí byť obalená stavom). Všetky tri postupy sa spoliehajú na rovnaké prostredie behu a môžu používať rovnaké aktivity. [14]



Obr. 17: Vnútoraná štruktúra WF

Sekvenčný prístup je prirodzene vhodný pre pracovný postup systému, zatiaľ čo State Machine a Flow Chart poskytujú spôsob, ako modelovať voľnejšie definovanú povahu pracovného toku. Jediný workflow môže kombinovať prvky všetkých štýlov, čo umožňuje kombináciu týchto štýlov.[15]

4.1 Sekvenčný workflow

Sekvenčný workflow sa vizuálne podobá na vývojový workflow. Sekvenčný workflow je výhodné použiť v podnikových procesoch kde je dôležité spracovať dáta, poslať správy a zapisovať výsledky. V sekvenčnom workflow užívateľ môže ovplyvňovať proces, avšak používanie tohoto typu nie je užívateľsky priateľské a preto by tento typ nemal byť primárne používaný na interakciu s užívateľom.[16]

Tabuľka 1: Prvky Sekvenčného workflow

| Akcia | Funkcia akcie |
|------------------|--|
| IfElse | Vykonáva činnosti obsiahnuté v dvoch alebo viacerých možných cestách na základe toho, či je splnená podmienka. |
| While | Opakovane vykoná jednu alebo viac aktivít, ak je podmienka splnená. |
| Sequence | Spravuje skupinu aktivít naraz v určenom poradí. |
| Parallel | Spravuje dve alebo viac sekvencií aktivít paralelne a čaká na dokončenie všetkých sekvencií pred pokračovaním. |
| Code | Spravuje definovaný počet kódov. |
| Listen | Čaká na konkrétnu udalosť a po vykonaní tejto udalosti vykoná jednu alebo viac činností. |
| Delay | Pozastavuje vykonanie pracovného toku na určitý čas. |
| InvokeMethod | Volá metódu v objekte, ktorý sa nachádza v aplikácii, ale je mimo pracovného toku. |
| EventSink | Čaká na volanie z inej metódy, ktorá sa nachádza v aplikácii, ale je mimo workflow. |
| InvokeWorkflow | Spôsobuje spustenie iného workflow. |
| InvokeWebService | Zavolá webovú službu. |
| Terminate | Ukončí vykonávanie workflow. |

4.2 State Machine

State Machine workflow poskytuje súbor stavov. Workflow začína v počiatočnom stave a končí, keď dosiahne konečný stav. Prechody medzi stavmi definujú správanie. Vo všeobecnosti State Machine workflow reagujú na udalosti. Výskyt udalosti spôsobí, že workflow prechádza do iného stavu.[17]

Tabuľka 2: Prvky State Machine workflow

| Akcia | Funkcia akcie |
|--------------|--|
| StateMachine | je prvok pre vytvorenie State Machine ako workflow a jeho následné vloženie do štruktúry workflow. |
| State | je základný element State Machine a reprezentuje stav, do ktorého workflow môže dospieť alebo z neho vychádzať. Zároveň State obsahuje tzn.: Entry,Exit - Tieto dva prvky reprezentujú akcie, ktoré nastanú pri vstupe / výstupe zo stavu. |
| FinalState | je finálny alias ukončovací prvok workflow. V State Machine sa musí nachádzať minimálne jeden takýto prvok, inak State Machine nie je validný. |

4.3 Flow Chart

Aktivita vývojového workflow je činnosť, ktorá obsahuje súbor činností, ktoré sa majú vykonať. Vývojové schémy obsahujú aj prvky riadenia toku, ako FlowDecision a FlowSwitch <T>, ktoré riadia vykonávanie medzi obsahnutými aktivitami na základe hodnôt premenných. Vývojový workflow slúži skôr ako pomôcka pre určenie toku aktivít. V State Machine je možné akciu pridať do vstup / výstup funkcie stavu, pri Flow Chart sa tieto aktivity priamo vkladajú do workflow bez pomocných prvkov.[18]

Tabuľka 3: Prvky Flow Chart workflow

| Akcia | Funkcia akcie |
|--------------|--|
| FlowChart | ako u State Machine má funkciu vytvorenia workflow a zaradenia do štruktúry workflow. |
| FlowDecision | je špecializovaná metóda FlowNode, ktorá poskytuje možnosť modelovať podmienený uzol s dvomi výsledkami. |
| FlowSwitch | je špecializovaný FlowNode, ktorý umožňuje modelovanie konštrukcie prepínača s jedným výrazom typu definovaným v špecifikátore typu aktivity a jediným výsledkom pre každý prípad. |

5 Analýza problémov migrácie

Táto kapitola sa zaoberá problémami a výzvami súvisiacimi s tvorbou migračného nástroja. Väčšina problémov bola zistená už pri prvotnej investigácii ale keďže BPMN je zložitá notácia mnoho výziev tvoriace problémy sa pridružilo postupom času.

Na začiatku zisťovania sa vyskytovali problémami rôznych vydavateľov platformy BPMM, vznikali nedorozumenia ohľadne workflow samotných, ako aj prvkov použitých v jednotlivých workflow. Po čase sa našiel jednotný postup ako rozoznávať BPMN workflow a jednotlivé konštrukcie použité vo workflow sa zaradili do dvoch skupín: jednoduché a zložené. Prvotne sa začalo pracovať s jednoduchou skupinou prvkov, ale postupom času sa objavovalo stále viac a viac zložitých prvkoch v príkladoch workflow. V dôsledku toho sa tieto dve skupiny začali spracovávať paralelne. Vo všeobecnosti, sa dá každý zložitejší prvok prepísať na viac jednoduchých. Tým pádom by stačilo spraviť maper medzi jednoduchými prvkami a WF prvkami a následne len každý zložitejší prvok rozložiť na jednoduché prvky WF. Teória však sklamala, keďže sa vyskytli príklady, kde sa jeden prvok choval odlišne pri rôznych stavoch.

Prvý problém nastal pri rozložení BPMN workflow za použitia BPMN.Sharp knižnice, pričom nebola možná žiadna modifikácia atribútov v dôsledku objektov v read-only móde. Read-only mód bol prelomený cez iný prístup k týmto premenným. V podstate sa nedal zmeniť objekt samotný ako taký, ale zistilo sa že najhlbší element hodnoty má nastavený private mód. Tým pádom bolo možné aspoň v jadre tento objekt zmeniť.

Ďalším problémom, ktorý bolo nutné vyriešiť bolo načítanie BPMN workflow a jeho následné spracovanie. Keďže BPMN workflow môžu obsahovať poddiagramy, bolo potrebné vymyslieť algoritmus, ktorým sa tieto poddiagramy odstránia a stanú sa súčasťou jedného veľkého diagramu. V tomto kroku boli odstránené zbytočné cesty, element poddiagramu a vstupno/výstupné elementy. Zároveň priebeh tohto mazania musel prebiehať rekurzívne, aby bolo možné spracovať všetky úrovne poddiagramov súčasne.

Popri vytváraní príkladov a ich spracovávaní došlo k množstvu prípadom, kedy sa workflow cyklil. Niektoré z týchto cyklov sa dali previesť na WF, avšak našli sa aj také, ktoré prevedenie nedovoľovali. Úlohou pri tomto probléme bolo identifikovať dané workflow pomocou algoritmu a následne podniknúť kroky k ich migrácii, alebo uviesť užívateľa o ich nemožnosti migrácie.

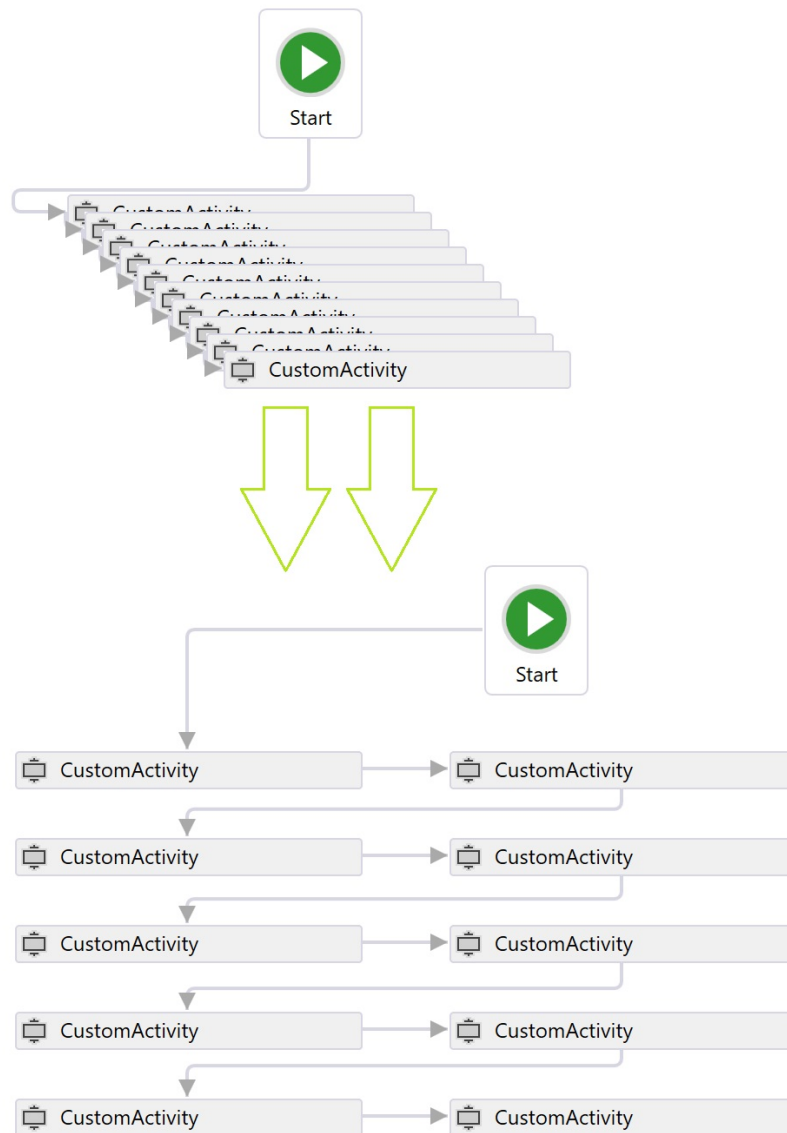
```
public bool DetectCycleParticipate(GenericModel input, List<GenericModel> way)
{
    if (way.Any(x => x.Name == input.Name && x.ExternId == input.ExternId))
        return false;
    way.Add(input);
    // iterate all transitions from selected state and run new instance
    foreach (var item in input.Transitions){
        List<GenericModel> newWay = way.ConvertAll(s => GenericModel.DeepClone(
            s));
        if (!DetectCycleParticipate(item.TargetState, newWay))
            return false;
    }
    return true;
}
```

Výpis 1: Metóda detekujúca cyklenie workflow

Pri predošlom probléme cyklenia workflow, sa narazilo aj na problém s paralelizáciou. Tzn. workflow môže obsahovať paralelné cesty len v prípade, že tieto cesty končia v spoločnom elemente. Úlohou bolo znovu vymyslieť algoritmus na detekciu takýchto prípadov a ich následné riešenie.

Ďalším z rady problémov bolo vymyslieť riešenie, akým sa bude workflow prevádzať z formy objektu na formu XAML. Avšak pri tomto probléme sa naskytovala rada riešení. Vybrať z nich jedno validné vyžadovalo mnoho testovania.

Pri exportovaní objektu na XAML súbor došlo k problému umiestnenia elementov. Tzn. všetky elementy boli na jednom mieste a nebolo možné rozoznať čo je element a čo je prechod. Úlohou teda bolo vymyslieť algoritmus, ktorým sa budú usporiadať elementy a cesty.[20]



Obr. 18: Vizuálne znázornenie vzniknutého problému pri tvorbe workflow

Jedným z posledných problémov, s ktorými sa bolo nutné vysporiadať, bolo ukazovanie užívateľovi, kde presne sa jeho workflow nachádza. Úlohou bolo buď vymyslieť vlastné riešenie alebo nájsť a implementovať stávajúce riešenie, ktoré by plne vyhovovalo požiadavkom.

6 Špecifikácia požiadavkov

6.1 Účel

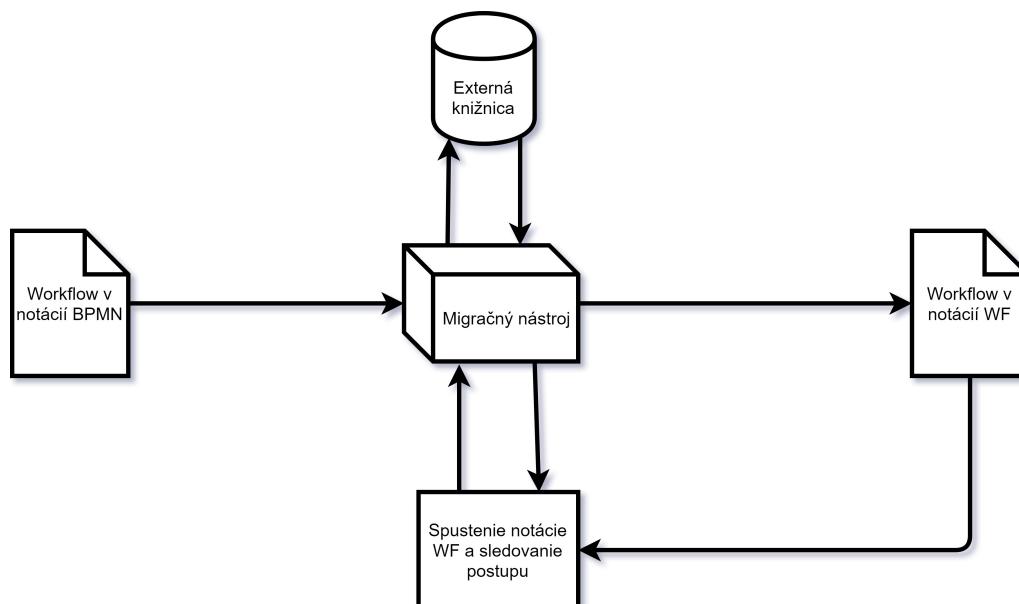
Špecifikácia migračného nástroja opisuje prevedenie BPMN workflow na platformu WF. Nástroj slúži pre učiteľský zbor, ktorý ho bude naďalej dopĺňať a rozširovať pre účely testovania a používania v business prípadoch. Výsledkom migrácie bude zákazníkov BPMN workflow prevedený do WF notácie s zachovanou pôvodnou ideou diagramu. Naprogramovaná verzia bude obsahovať základný prevod notácií a možnosť určenia poddiagramu.

6.2 Rozsah projektu a funkcie systému

Migračný nástroj bude obsahovať možnosť načítania užívateľského súboru a následne jeho spracovanie. Užívateľovi systém umožní vybrať z možných poddiagramov. Po nadeinovaní nástroj uloží BPMN notáciu vo WF notácií. Nástroj bude schopný vyhodnotiť a overiť správnosť prevodu. Úlohou nástroja je korektný a pevný podklad pre ďalší vývoj a rozšírenie. Do rozsahov projektu patrí aj návrh rozšírení pre systém.

6.3 Kontext systému

Migračný nástroj predstavuje službu vykonávajúcu užívateľov požiadavok na migráciu workflow. Užívateľ používa BPMN súbor pre vloženie a očakáva WF notáciu prevedeného súboru. S nástrojom pracuje učiteľský zbor, ktorý testuje možnosti notácie WF a rozširuje jeho možnosti a funkcionality. Dôraz je kladený najmä na zachovanie podobnosti schémy medzi notáciami BPMN a WF.



Obr. 19: Kontext systému

6.4 Prehľad funkcií

Užívateľ migračného nástroja bude schopný upravovať typy poddiagramov v notácii WF vlastného diagramu z notácie BPMN. Funkcie nástroja sú interpretované nasledujúcim obrázkom.

6.5 Rozhrania

Hardverové rozhranie

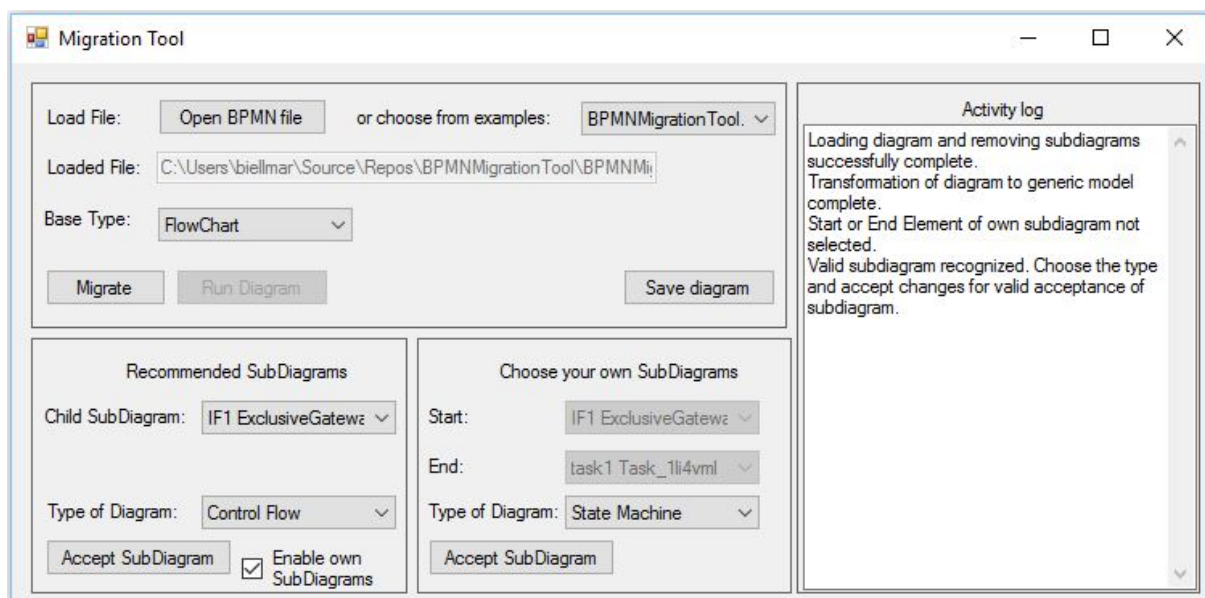
Nástroj bude možné spustiť na systéme Windows s podporou .Net minimálne 4.5.2.

Softvérové rozhranie

Migračný nástroj bude reprezentovaný ako Windows Form aplikácia.

Užívateľské rozhranie

Rozhranie bude obsahovať jednoduchú štruktúru s dostatkom informácií poskytovaných užívateľovi.



Obr. 20: User Interface migračného nástroja

6.6 Jednotlivé špecifické požiadavky rozpísané do bodov

6.6.1 Vytvorenie workflow na základe BPMN diagramu v prostredí WF

Cielom požiadavku na vytvorenie workflow vyplývajúceho z BPMN do WF je previesť notáciu workflow. Tým bude zaručený prevod z BPMN do WF. Úlohy potrebné na splnenie tejto požiadavky zahrňovali riešenie problémov cyklov v workflow či paralelizmus workflow. Taktiež jednou z úloh bolo vyriešiť zachovanie schémy workflow po jeho vytvorení v notácii WF.

6.6.2 Informácia o prebiehajúcich procesoch

Cielom požiadavku je vytvorenie podprogramu, ktorý informuje užívateľa o prebiehajúcich procesoch. Podprogram bude umožňovať spúšťanie diagramu a následne monitorovať aktuálny stav.

6.6.3 Perzistentné uloženie workflow

Perzistentným uložením workflow je myslený XAML súbor s obsahujúcou štruktúrou WF. Po migrácií si užívateľ môže danú instanciu workflow uložiť do súboru.

6.6.4 Správa verzii jednotlivých instancií

Perzistentné uloženie má zahrňovať aj správu verzií vzniknutých počas migrácie workflow.

6.6.5 Import-export

Pri vkladaní program očakáva validný súbor s príponou BPMN, následne po spracovaní výstup obsahuje diagram uložený v XAML súbore s notáciou WF.

7 Analýza požiadaviek a návrh riešenia

V analýze požiadavkov a návrhu riešenia sa predovšetkým budem venovať investigácií požiadavkov a návrhu riešenia. Popíšem aj algoritmus prevodu.

Prvou a najzákladnejšou požiadavkou bol prevod **BPMN workflow do WF notácie**. Bude sa teda vyvíjať nástroj, ktorý postupne načíta BPMN workflow a skonštruuje tzn. “Generický Model”, ktorý následne budeme prevádzať na užívateľom požadovaný typ workflow.

Pri investigácií požiadavky na **prehľad o behu workflow** sa narazilo na problém v platforme WF. Zistilo sa, že aktuálne WF neobsahuje žiaden modul na zobrazenie aktuálneho behu workflow. Jediná funkcionálna, ktorá je blízka tejto požiadavke je debugovanie počas behu. Došlo preto k voľbe zobrazenia behu v notácii BPMN. Taktiež pre užívateľa bude prívetivejšie rozpoznať stav podľa jeho BPMN notácie ako podľa WF notácie.

Perzistentné uloženie workflow zahrňovalo úpravu výsledného reťazca znakov a jeho následné uloženie do súboru. V tejto požiadavke sa vyskytli problémy pri načítaní workflow do nástroja Visual Studio. Investigáciou som zistil, že problém sa vyskytuje prevažne pri prezentačných prvkoch. Po doplnení potrebných tried a namespace bol problém vyriešený.

Po zistení úloh potrebných na **ukladanie verzií instancií prevodu** sa rozhodlo tento požiadavok vynechať z dôvodu nepotrebnosti danej informácie. Proces nastavovania a interakcie s užívateľom väčšinou nie je natoľko rozsiahly, aby bolo nutné ukladať jednotlivé verzie.

Popri vyvíjaní nástroja sa narazilo aj na validné druhy BPMN workflow, ktoré nie je možné previesť do WF notácie. Tento problém sa riešil pomocou **Import-Export** požiadavky. Pri importe sa validuje daný BPMN workflow voči podmienkam prevodu a pri Exporte sa kontroluje správnosť WF diagramu tzn. “Experimentálnym spustením WF diagramu”.

7.1 Popis algoritmu prevodu

Popis algoritmu prevodu

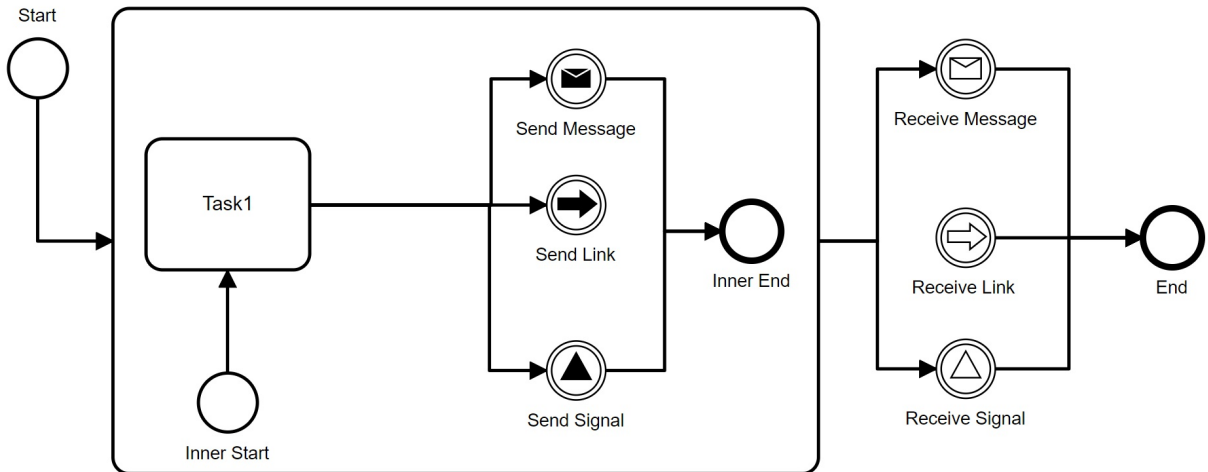
1. Načítanie modelu

Vstup: Cesta k BPMN súboru

Výstup: BPMN model

- (a) Načítanie BPMN do modelu - Po investigácií možností BPMN diagram sa narazilo na dva druhy poddiagramov. Tieto poddiagramami bolo nutné rozložiť do jedného veľkého diagramu
- (b) Odstránenie subdiagramov typu CallActivity
- (c) Odstránenie subdiagramov typu SubDiagram
- (d) Korekcia správových tokov medzi najvyššou úrovňou entít (pool, swimlanes, ...)
- (e) Korekcia participants elementov

Nasledujúci príklad demonštruje väčšinu možností poddiagramov a ich prepojení s rodičovským diagramom. Tento diagram nie je možné previesť v nástroji z dôvodu viacerých koncov paralelizácie. Avšak je možné ho spracovať do generického modelu a pripraviť ho pre možné rozšírenie nástroja o spracovanie takéhoto druhu paralelizácie.



Obr. 21: Príklad zložitého poddiagramu v notácii BPMN

Kód v nasledujúcej ukážke demonštruje vyhľadanie elementu v BPMN notácii. Najskôr sa vyhledá konektor podľa identifikátora jeho začiatočného stavu, následne sa v konektore zistí identifikátor stavu do ktorého smeruje. Na koniec sa podľa identifikátora stavu vyhledá stav samotný.

```
// get sequence element
var sequenceFlow = allElements.Where(x => x.TypeName
    .Equals(BPMNTypes.sequenceFlow.ToString()) &&
    x.Attributes.Any(y => y.Key.Equals(BPMNAttributes.id.ToString())
        ) && y.Value.Equals(outElementId)).FirstOrDefault();
// get its target id
var targetRefId = sequenceFlow.Attributes
    .Where(x => x.Key.Equals(BPMNAttributes.targetRef.ToString()))
    .Select(y => y.Value).FirstOrDefault();
// get element by targedRefId
var targetRef = allElements.Where(x => x.Attributes
    .Any(y => y.Key.Equals(BPMNAttributes.id.ToString()) &&
    y.Value.Equals(targetRefId))).FirstOrDefault();
```

Výpis 2: Nález jedného stavu v notácii BPMN

3. Nadefinovanie podmienok prevodu

Vstup: Generický model

Výstup: Užívateľom nadefinované podmienky prevodu

Počas spracovávania cyklov či paralelizov algoritmus vždy najskôr spracuje poddiagram

- (a) Detekcia paralelizmov a validácia paralelizmov
 - i. Algoritmus sa presunie na začiatkový prvok generického modelu a začne prechádzať všetkých jeho potomkov. Akonáhle má jeden generický model viac ako jedného potomka – jedná sa buď o podmienkovú klauzulu, alebo o paralelizmus.
 - ii. Ak je model označený ako podmienkový, algoritmus pokračuje ďalej.
 - iii. Ak je model označený ako stav, algoritmus prechádza všetky možné cesty vychádzajúce z vybraného uzla až ku koncom diagramu. Akonáhle sú vyhladané všetky cesty, algoritmus sa zastaví a začne hľadať spoločné kombinácie v cestách.
 - iv. Ak systém nenájde ani jednu zhodu v cestách, nie je možné daný diagram ďalej spracovávať. Paralelizmus musí začínať aj končiť v spoločnom bode (definícia prvku Parallel v notácii WF).
 - v. Ak nájde zhodu, označí koncový stav tejto zhody a tým ho prepojí s štartovacím stavom paralelizácie.
- (b) Detekcia cyklov

Algoritmus vyhľadáva cykly pomocou kontrolovania všetkých možných ciest. Akonáhle narazí na stav, ktorý už je zahrnutý v ceste, Algoritmus obsahuje cyklus. V takomto prípade je možné použiť poddiagram s typom Flow Chart alebo State Machine. V type ControlFlow sa nachádza prvok While, tento prvok by relatívne mohol byť použitý na integráciu cyklov v tomto type, avšak po investigácii sa zistilo, že jeho použitie by bolo veľmi obmedzené vzhľadom na to, že tento prvok musí cykliť len určitú skupinu prvkov stále dookola a nedovoľuje vystúpenie z cyklu iným stavom.
- (c) Interakcia s užívateľom

Užívateľ z ponuky nadefinuje aký typ diagramu sa má použiť pre jednotlivé poddiagramy, ako aj pre diagram samotný. Následne sa spustí algoritmus, ktorý “Odkrojí” daný poddiagram a následne ho uloží do novo vytvoreného generického modelu ako SubDiagram premenná. Po uložení poddiagramu do premennej sa nastaví premenná SubDiagramType na užívateľom zvolenú hodnotu. Akonáhle máme takto nadefinovaný nový model, prepoja sa “Odkrojené” cesty s novo vzniknutým modelom.

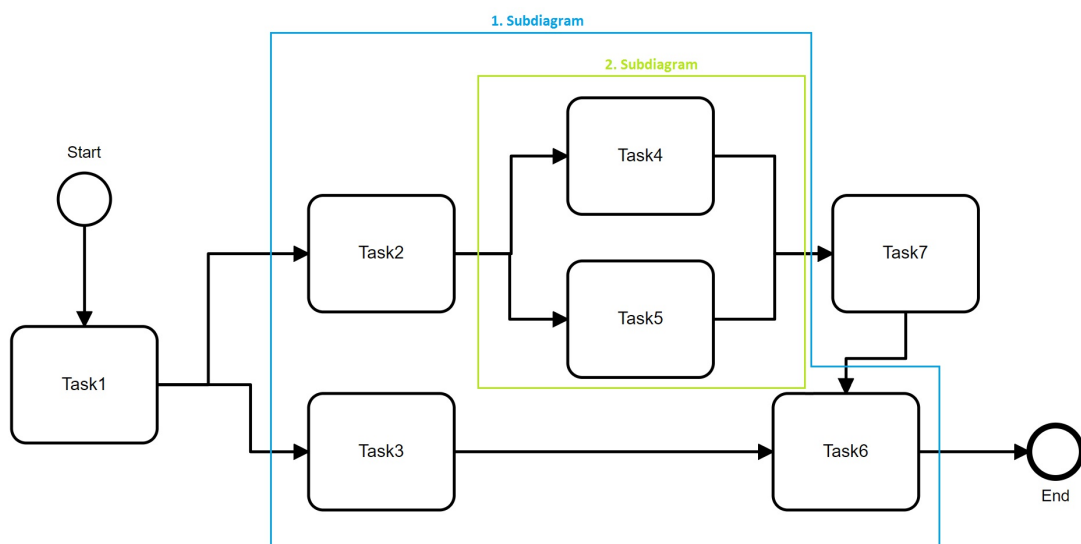
```
public Result DetectParallelism(GenericModel input)
{
    if (input.Checked){
        return new Result(true, "BPMN diagram can be migrated to
            Windows Workflow Flow Chart");
    }
}
```

```

}
input.Checked = true;
if (input.Transitions.Count > 1 &&
    (input.TypeOfModel != BPMNTypes.exclusiveGateway ||
     input.TypeOfModel != BPMNTypes.inclusiveGateway ||
     input.TypeOfModel != BPMNTypes.parallelGateway) &&
    input.SubDiagram == null){
    return new Result(false, "Parallelism is not allowed in
        selected Base type, change base type to Control Flow or
        include subdiagrams.");
}
// if type == gateway -> multiple transitions is presented
// otherwise -> just one
foreach (var transition in input.Transitions){
    var result = DetectParallelism(transition.TargetState);
    if (!result.Success)
        return result;
}
return new Result(true, "BPMN diagram can be migrated to Windows
    Workflow Flow Chart");
}

```

Výpis 3: Detekcia paralelizmu v generickom modeli



Obr. 23: Spracovávanie poddiagramov popri interakcií s užívateľom

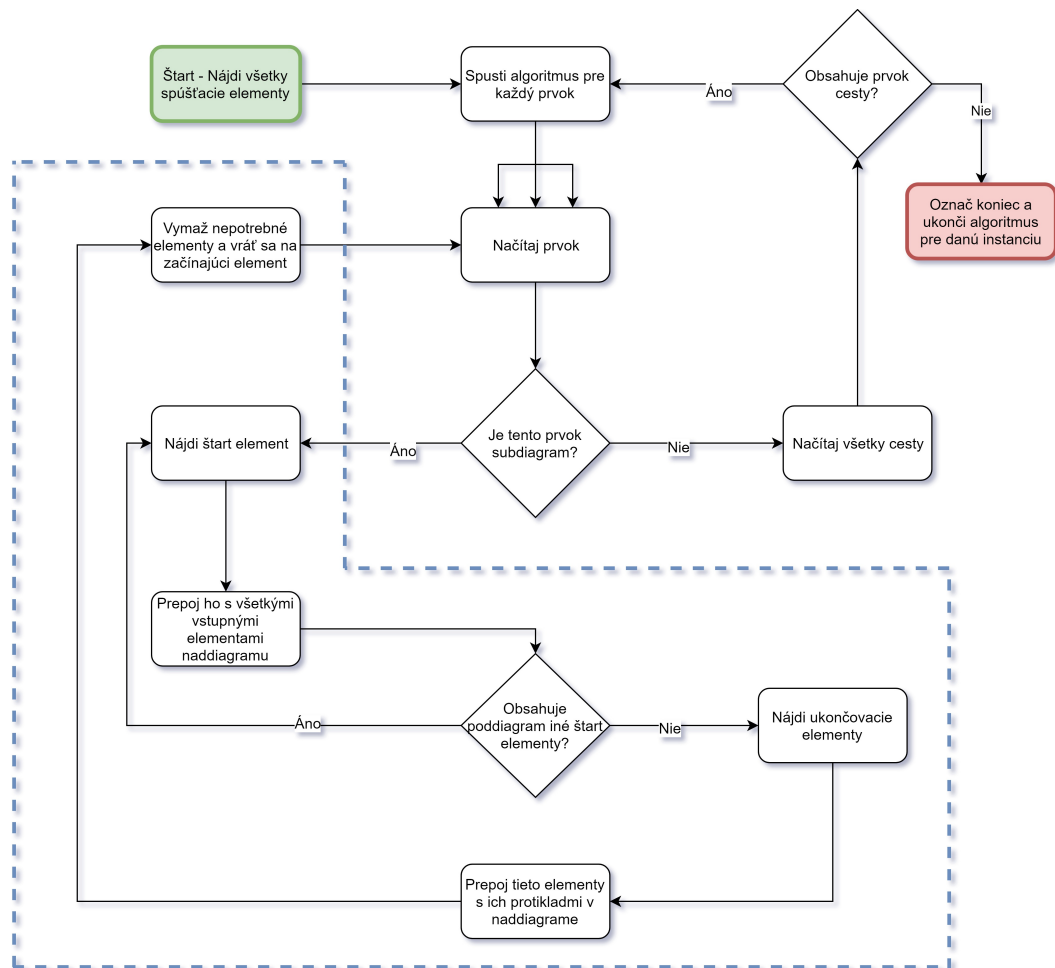
4. Prevod generického modelu na WF model

Vstup: Generický model a Užívateľom nadeľované podmienky prevodu

Výstup: Diagram v notácii WF

Podľa nadeľovaných podmienok sa začne prevádzať generický model na WF model

- Najskôr algoritmus spracuje prvý začiatkový stav v generickom modeli
- Následne začne iterovať všetky prechody a rekurzívne spúšťať instance prevodu
- Akonáhle algoritmus narazí na poddiagram (užívateľom nadeľovaný typ) začne rekurzívne spúšťať rovnaký algoritmus so špeciálnou podmienkou, ktorý sa používa na konverziu workflow samotného. Výstupom instance algoritmu je Aktivita typu State Machine, Flow Chart alebo SequenceFlow
- Prevod je zakončený vo chvíli, keď sa všetky instance algoritmu ukončia
- Následne sa workflow uloží do XAML notácie a prevedie sa korekcia tried a namespace



Obr. 24: Algoritmus prevodu workflow

Po skončení algoritmu sa užívateľovi sprístupní tlačítko na spustenie workflow.

5. Sledovanie behu workflow

- (a) Počas behu workflow sa jednotlivé aktivity (resp. External ID týchto aktivít) zapisujú do súboru
- (b) Tento súbor paralelne číta aplikácia a zisťuje External ID aktivít. Pri takomto postupe bolo nutné nadefinovať delegát reagujúci na OnChange metódu
- (c) Následne sa použije Externá funkcia GetImage a prevedie začiatočný BPMN model na obrázok
- (d) Pomocou External ID algoritmus vyhľadá prvok a zistí jeho polohu
- (e) Do obrázku sa podľa súradníc zakreslí obdĺžnik a vykreslí sa užívateľovi spolu s celým workflow

8 Implementácia a Testovanie

Naimplementoval sa nástroj podľa návrhu vyplývajúceho z analýzy. Nástroj v základe dokáže previesť workflow v notácii BPMN do notácie WF s zachovaním pôvodnej ideí diagramu. Následne bolo v implementácii zahrnutý aj algoritmus, ktorý dovoľuje voľbu poddiagramov ako aj ich typov. Import/export je naimplementovaný ako stávajúci modul v aplikácii s využitím dostupných knižníc z .NET 4.5.2. Monitorovanie chodu aplikácie je naimplementované ako samostatná Windows Form.

Použité technológie

1. .NET 4.5.2

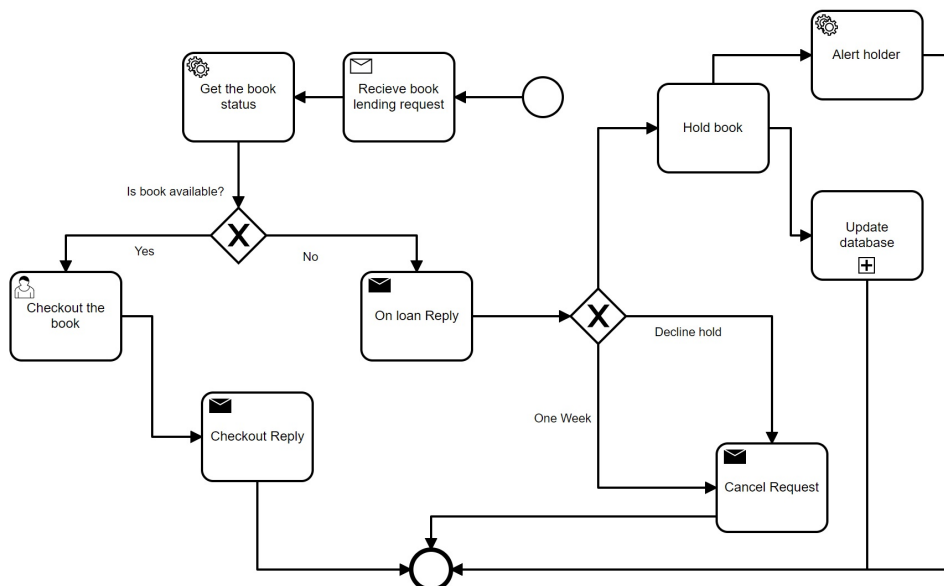
Použité programy

1. Pre vývoj nástroja sa použil Microsoft Visual Studio 2015.
2. Pre tvorbu vstupných dat sa použil Camunda 7.8.0.

Pre testovanie sa vybrali zložitejšie príklady nájdené na internete. Vizualna stránka týchto prevodov sa z dôvodov lepšej prehľadnosti upravovala ručne.

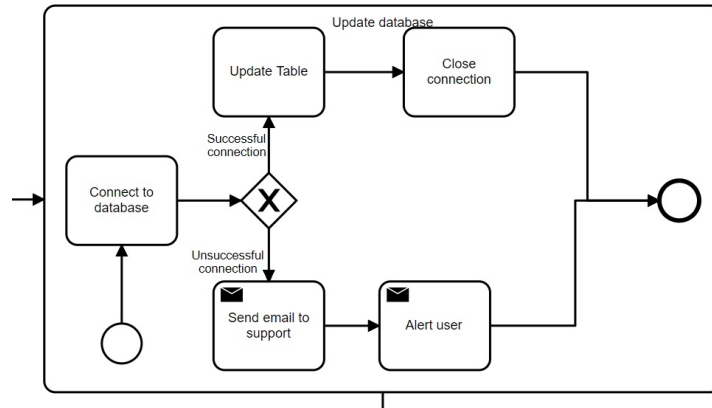
8.1 Book lending process

Prvým príkladom, ktorý je zobrazený nižšie je “Book lending process”. Tento workflow slúži na spracovanie požiadavky na požičanie knihy. V workflow sa nachádza paralelné spracovanie, podmienky ako aj poddiagram.



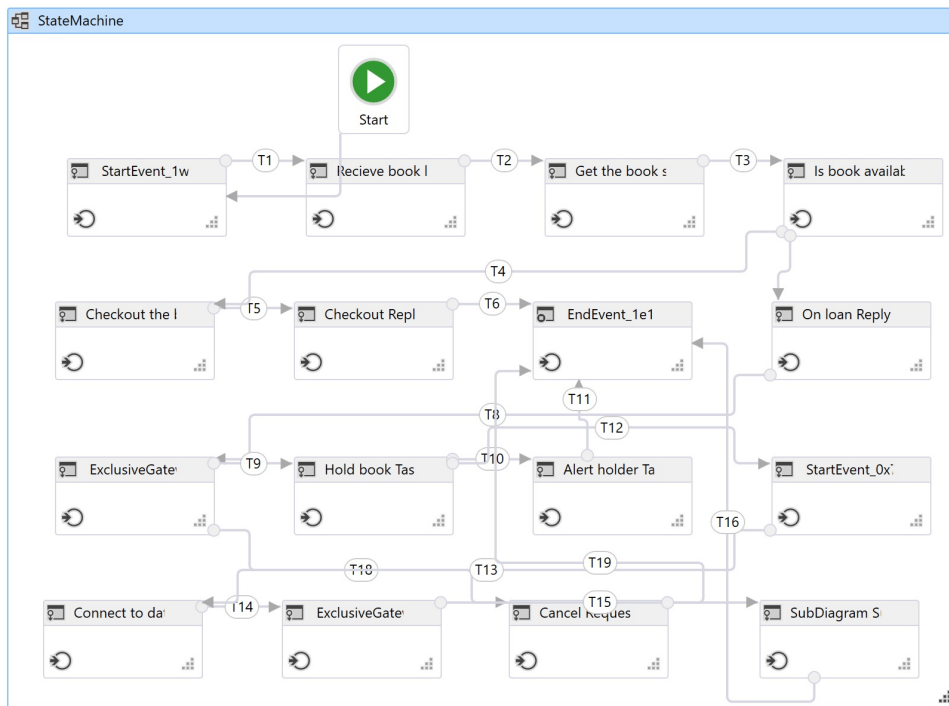
Obr. 25: Book lending process v notácii BPMN

V diagrame je vidieť element “Update database”, vnútorná štruktúra tohto elementu sa nachádza na nasledujúcom obrázku.

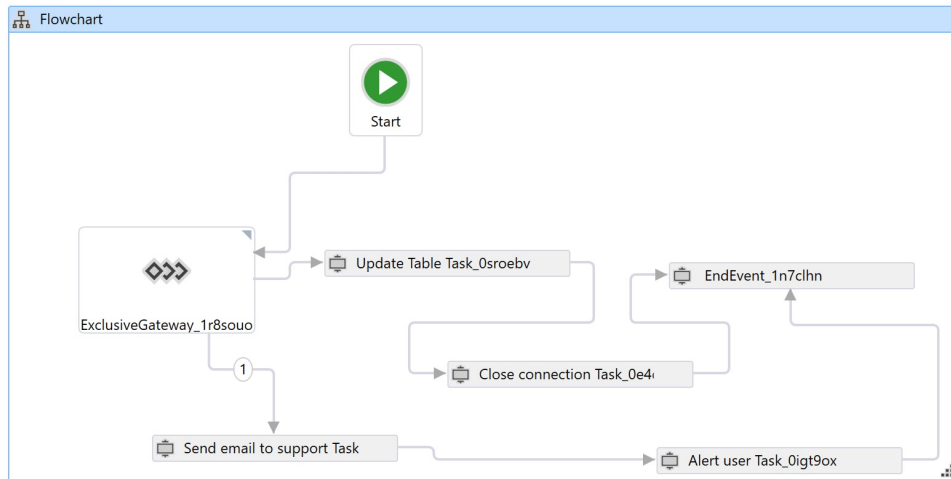


Obr. 26: Vnútorná špecifikácia Update database elementu

V nástroji sa nadefinoval poddiagram typu State Machine a celkový typ workflow ako Flow Chart. Výsledok migrácie je zobrazený na nasledujúcom obrázku.



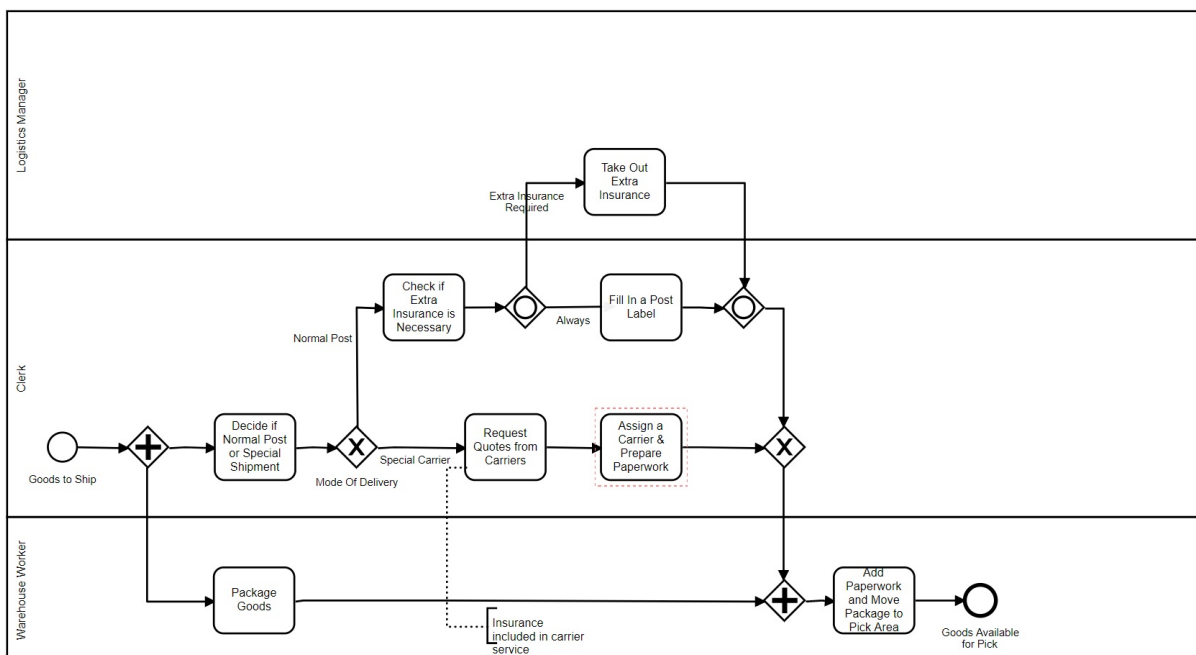
Obr. 27: Výsledok prevodu nástroja do notácie WF



Obr. 28: Vnútorná štruktúra poddiagramu

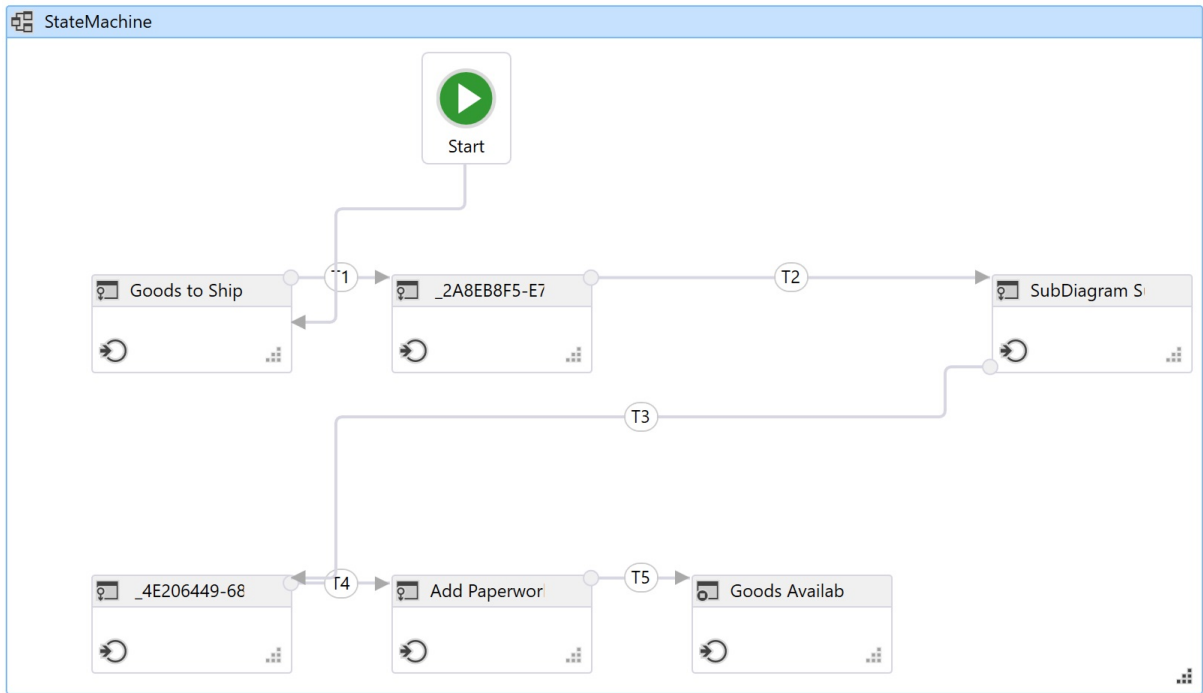
8.2 Shipment Process of a Hardware Retailer

Ako druhý príklad sa vybral “Shipment Process of a Hardware Retailer”. V tomto workflow sa nachádza veľa rozcestí s spoločným koncom, takže možnosti poddiagramov je podstatne viac ako v predchádzajúcom príklade.



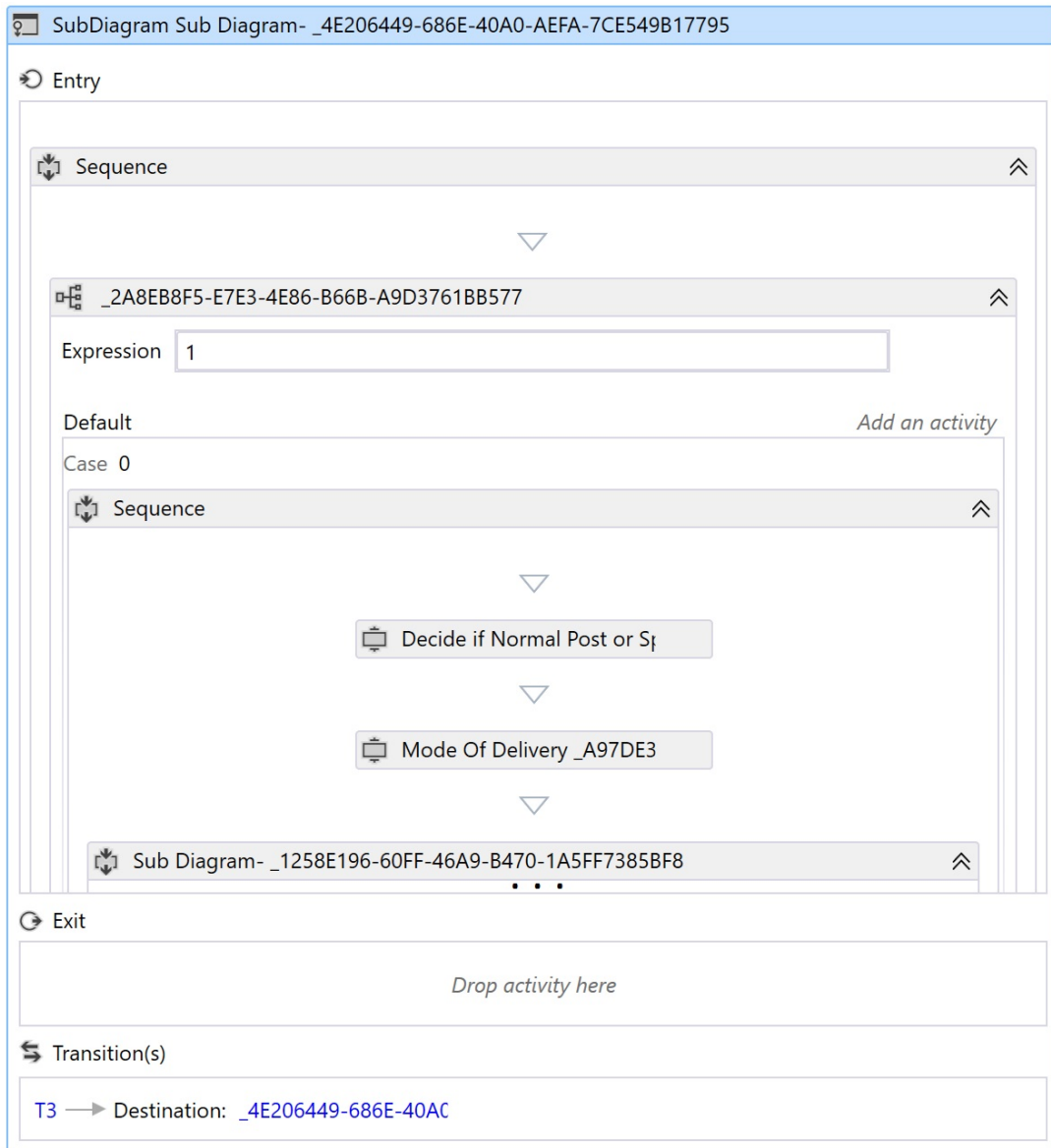
Obr. 29: Shipment Process of a Hardware Retailer v notácii BPMN

V nástroji sa nastavil základný typ workflow ako State Machine a všetky ostatné navrhované poddiagramy ako ControlFlow. Výsledný State Machine vyzerá nasledovne.



Obr. 30: Výsledný workflow po migrácii

Pre lepšiu orientáciu v ControlFlow poddiagrame sa výsledok orezal.



Obr. 31: Výsledný poddiagram po migrácii

9 Možnosti prevodu a obmedzenia

Možnosti nástroja zahrňujú širokú škálu prevodu medzi ktoré patrí napríklad spracovanie podmienok v workflow na úrovni schémy či spracovanie prepojení poddiagramov. Medzi najdôležitejšie možnosti patrí:

- Nástroj obsahuje detekciu vnútorných paralelizmov/rozcestí a ponúkne užívateľovi ich prevod do ControlFlow, aby bolo možné previesť celý workflow do ostatných typov diagramu
- Základným prvkom každej úlohy v BPMN notácií je v notácií WF vytvorená prázdna trieda CustomActivity, ktorá slúži ako šablóna pre budúcu integráciu externých kódov
- Nástroj dokáže nájsť cyklus/paralelizmus a upozorniť užívateľa o jeho vplyvu na možnosti migrácie
- Migrácia nástrojom zachováva asociačné texty v podobe poznámok pri jednotlivých aktivitách v notácií WF
- Po samotnej migrácii sa užívateľovi sprístupní možnosť daný workflow spustiť v notácií WF a sledovať jeho beh na vizuálnej notácií BPMN
- Pri zvolení Flow Chart diagramu sa nezobrazí priebeh aplikácie počas rozhodovacích prevodov
- Je odporúčané nástroj spúšťať v programe Visual Studio
- Výsledný workflow v notácií WF je možné upravovať v nástroji Visual studio
- Po zmigrovaní workflow sa odporúča vypnúť a zapnúť nástroj, čo spustí automatický build workflow

Všetky štandardné workflow spĺňajúce nasledujúce podmienky sú prevediteľné ak spĺňajú nasledujúce podmienky:

- V nástroji nie je možné previesť konverzačný a choreografický typ BPMN diagramu
- Nástroj v stávajúcej verzii nedokáže previesť paralelizmus s viac ako jedným spoločným koncom
- Cykly sú možné previesť len do State Machine alebo Flow Chart, ak majú jednoduchú štruktúru a neročleňujú sa do inej vetvy programu
- Odkazy na externé kódy v jednotlivých úlohách sa neprevádzajú
- Nástroj podporuje aktuálnu verziu BPMN 2.0 s jeho stávajúcimi prvkami, ktoré sa dali previesť, WF má obmedzený počet prvkov a tým aj funkcionality, nebolo teda pri každom prvku BPMN možné potvrdiť plnú prevediteľnosť

- Pre korektnú identifikáciu začiatku je možné previesť workflow s práve jedným nepodmie-
neným začiatkom - to isté platí aj pri poddiagramoch
- Z BPMN notácie sa neprevádzajú informácie o Pooloch či SwimLanes ako ani farebné
prevedenia jednotlivých prvkov
- Poddiagram v notácii BPMN typu CallActivity musí mať validnú štruktúru v súbore aby
bol začlenený do úplneho workflow prevedeného do WF notácie

Po prevedení workflow splňujúceho predošlé požiadavky sa výsledný WF štruktúrou vo väč-
šine prípadov podobal/bol identický voči vstupnému BPMN workflow. Bol teda dosiahnutý
prevod z BPMN notácie do WF notácie na úrovni štruktúry schémy s zachovaním pôvodnej
idei.

10 Existujúce nástroje a ich porovnanie s naimplementovaným riešením

Pri investigácii tohto bodu došlo k problému s veľkým nedostatok stávajúcich riešení. Riešenia slúžili väčšinou na spustenie diagramu, nie na jeho migráciu. V podstate sa našlo iba jedno riešenie, ktoré slúžilo na migráciu BPMN do WF. Nástroj sa volá **HeadSight.BPMN2WF**. Nástroj je zastaralý a vyžadoval potrebnú údržbu pre spustenie.

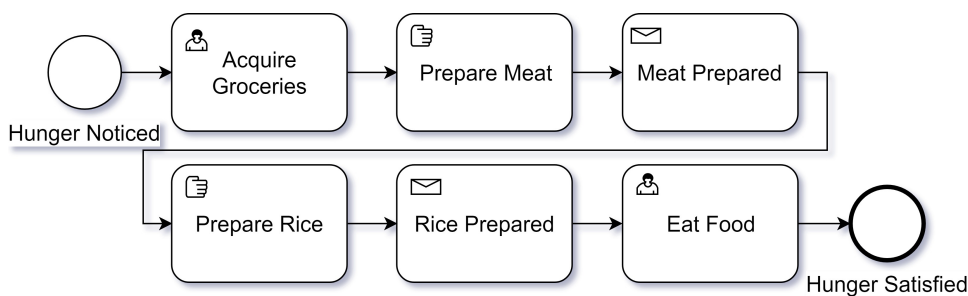
Po investigácii nástroja som zistil, že nástroj obsahuje tri základné úrovne prevodu:

1. Simple Converter
2. Advanced Converter
3. Conversion via Configuration file

Simple Converter väčšinou vykonal prevod, avšak len v minime prípadov úspešne. Výsledkom bola prevažne iba časť workflow. Program zastavil tvorbu workflow hneď na začiatku a tým výsledok obsahoval maximálne 4 - 5 stavov (z 20 stavového workflow). Naproti tomu **Advanced Converter** vypísal error hneď pri koverzii takmer každého workflow. **Conversion via Configuration file** sa vynechal z dôvodu nedostatku vstupných dát. Tento nástroj bol jediný, ktorý sa našiel a pri podrobovaní testovania nesplnil ani minimálne kritéria. Nástroj neumožňoval voľbu poddiagramov a vždy vytvoril typ Flow Chart. Akonáhle bola konverzia dokončená, nástroj dovoľoval uloženie diagramu v WF notácií. Aj napriek tomu sa porovnávala konverzia tohto nástroja s vlastným migračným nástrojom.

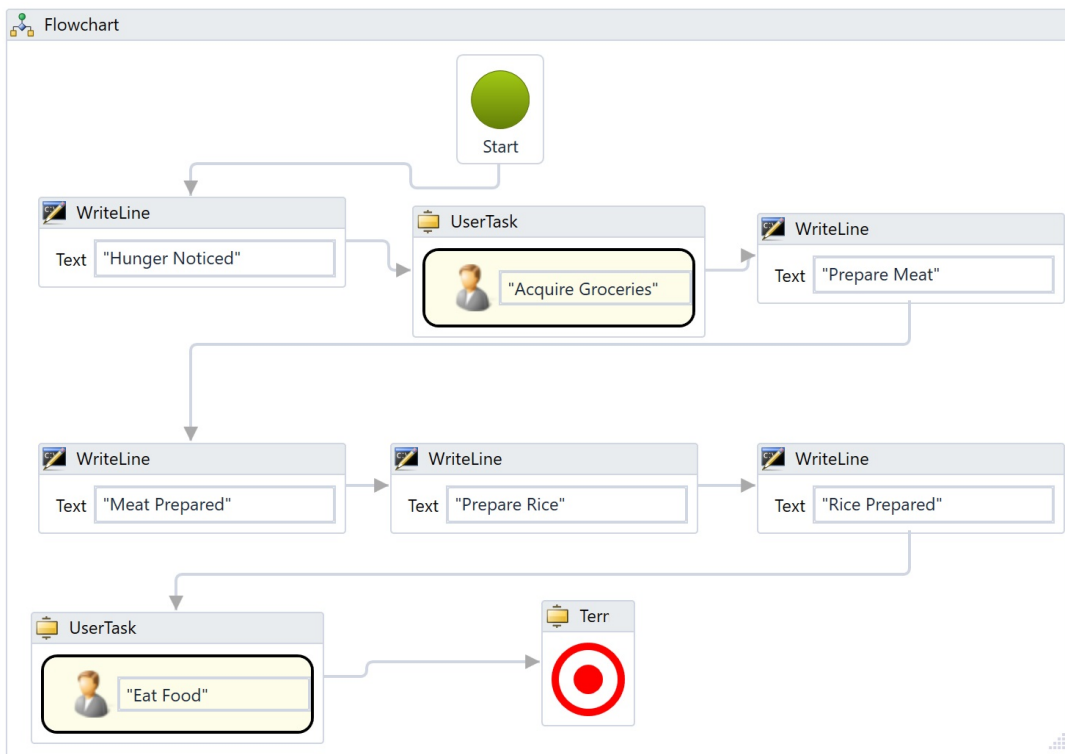
10.1 Príklad 1

Prvý príklad demonštroval základnú schopnosť prevodu. Vstupom bol BPMN súbor s sekvenciou šiestich úloh nasledujúcich za sebou. Príklad sa podarilo previesť obom nástrojom. Výsledné workflow sa pre lepšiu orientáciu graficky upravili.



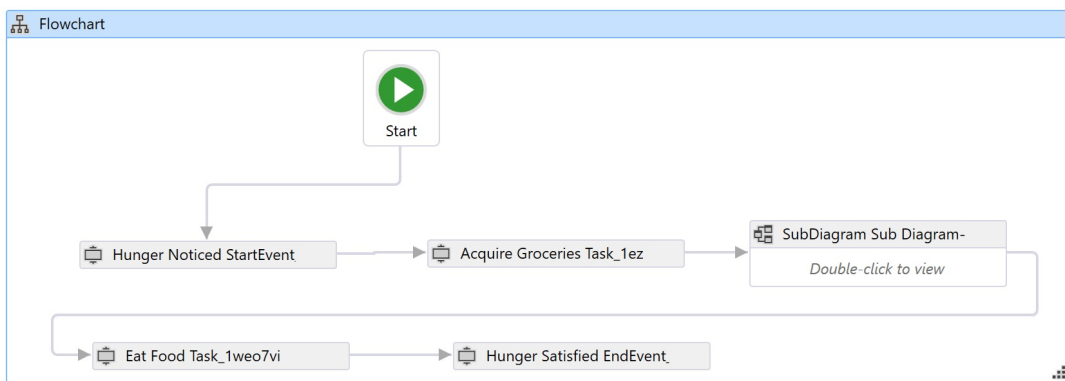
Obr. 32: Základný workflow v notácii BPMN

Nástroj HeadSight pri spustení v Simple Converter režíme nedokázal previesť daný workflow, použil sa pre to Advanced Converter režím s Beautify created XAML. Výsledok sa nachádza na nasledujúcom obrázku.



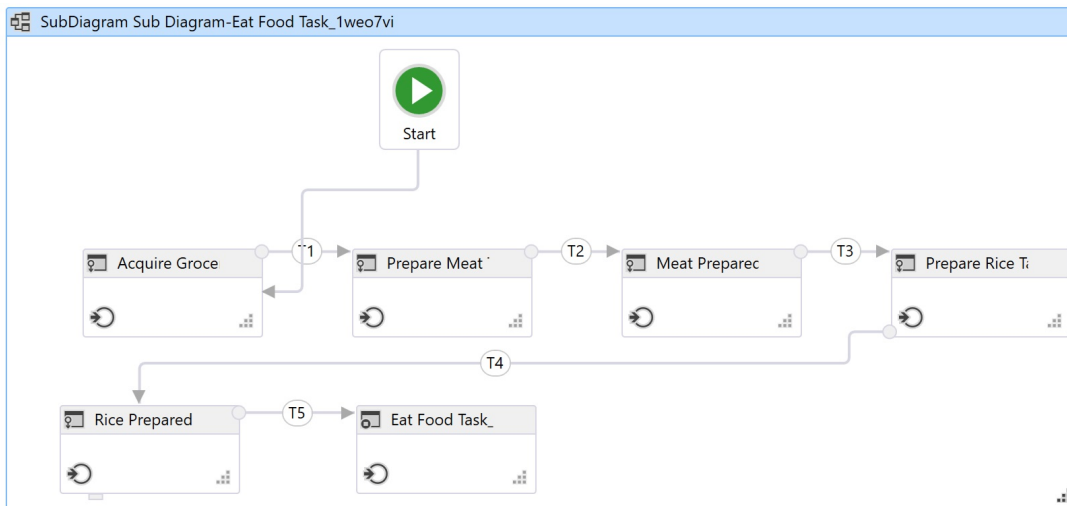
Obr. 33: Výsledný základný workflow po migrácii v nástroji HeadSight

V mojom nástroji sa nastavil poddiagram začínajúci v stave “Acquire Groceries“ a končiaci v stave “Eat Food“, ako typ poddiagramu bol zvolený State Machine pre jeho prehľadnosť a ako typ celkového diagramu sa zvolil Flow Chart. Výsledok hlavného diagramu je na nasledujúcom obrázku.



Obr. 34: Výsledný základný workflow po migrácii mojím nástrojom

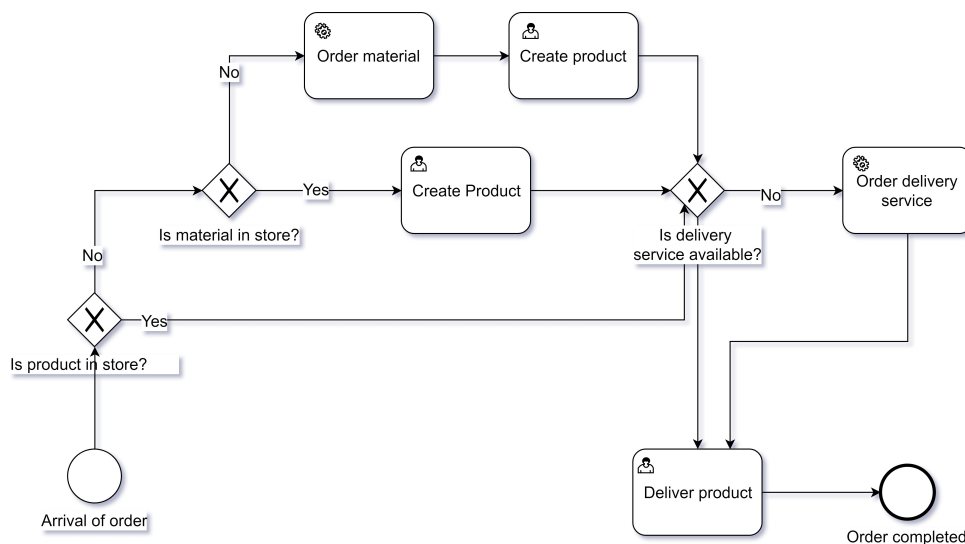
Na výslednom workflow môžeme vidieť, že pribudli dva stavy identifikujúce začiatok a koniec poddiagramu. Tieto dva stavy boli vytvorené aplikáciou umelo pre jednoduchšie nadviazanie vonkajších prvkov workflow s vnútornými. Obsah poddiagramu je zachytený na nasledujúcom obrázku.



Obr. 35: Výsledný základný poddiagram po migrácii mojím nástrojom

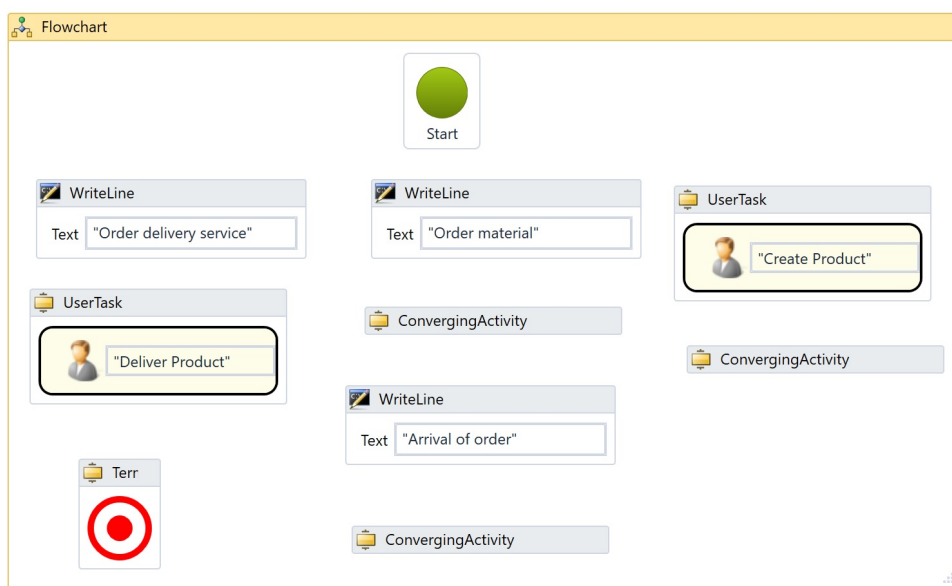
10.2 Príklad 2

Popri testovaní došlo k nespočtu prípadov, kedy prevod pomocou nástroja HeadSight neprebehol vôbec pri akejkoľvek možnej konfigurácii. Musel som preto voliť jednoduchšie modely, aby bolo vôbec možné porovnať nástroje. Nasledujúci príklad obsahoval tri jednoduché podmienky a päť stavov.



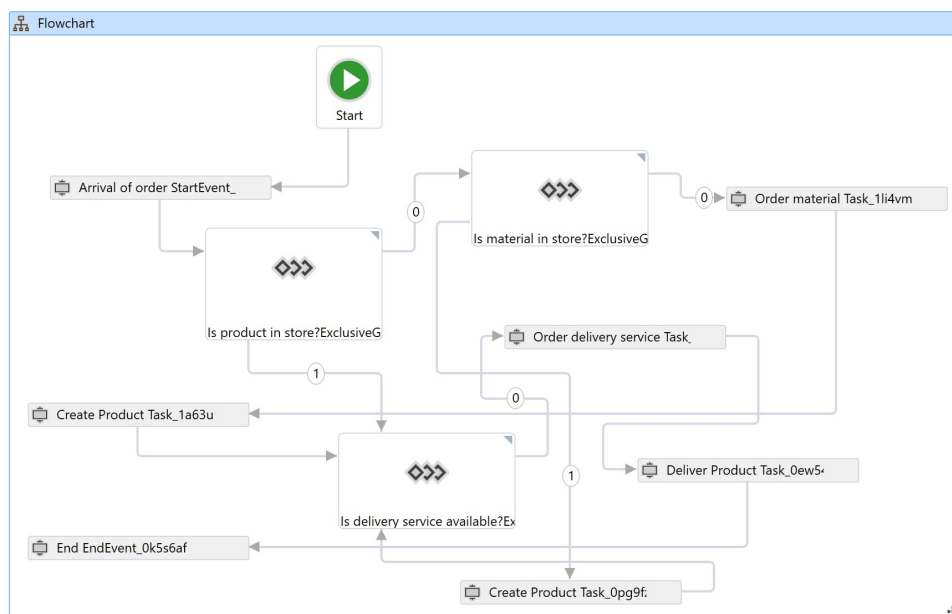
Obr. 36: Zložitejší workflow v notácii BPMN

Avšak aj po použití relatívne jednoduchého diagramu sa nástroj HeadSight zasekával pri oboch možných konfiguráciách. Výsledok nakoniec zahŕňal zhluk stavov, ktoré neboli prepojené.



Obr. 37: Výsledný zložitejší workflow po migrácii HeadSight

Naproti tomu môj migračný nástroj dokázal daný workflow relatívne správne previesť na WF notáciu. V aplikácii som nenastavoval žiadne poddiagramy, aby výsledok bol porovnateľný s programom HeadSight. Výsledok je na nasledujúcom obrázku.



Obr. 38: Výsledný zložitejší workflow po migrácii mojím nástrojom

Ďalšie testovanie dopadlo podobne ako posledný pokus. Nástroj HeadSight nedokázal vo väčšine prípadov workflow previesť do notácie WF. Avšak stále to bol jediný nástroj na ktorý sa počas investigácie narazilo.

11 Záver

Pri tvorbe práce som sa oboznámil s notáciou BPMN, ako aj s technológiou Windows Workflow Foundation. Podrobne som rozpísal každú technológiu ako aj ich možnosti. Popísal som slabé a silné stránky každej technológie, rovnako ako aj ich porovnanie.

Z existujúcich workflow systémov som našiel HeadSight a porovnal ho s mojim nástrojom. Výsledky som rozpísal do dvoch príkladov, ktoré demonštrovali väčšinu prípadov použitia. Nástroj HeadSight zhodnocujem ako nedostačujúci na riešenie problematiky a požiadaviek vypísaných v tejto práci. Žiadny ďalší nástroj splňujúci základnú požiadavku prevodu diagramov z notácie BPMN a WF nebol nájdený.

Vytvoril som nástroj slúžiaci na prevod BPMN do WF notácie. Vypísal som problémy, s ktorými som sa stretol, ako aj ich riešenia. Podrobne som popísal algoritmus prevodu ako aj možné rozšírenia nástroja.

Nástroj podporuje poskytovanie informácií o prebiehajúcich procesoch a o perzistentnom uložení diagramu v notácií WF. Poskytovanie informácií som sa rozhodol zobrazovať na BPMN notácií. Správu verzií som sa rozhodol vynechať z dôvodu nevyužitelnosti tejto požiadavky.

Analyzoval som podporné funkcie a naimplementoval som import-export vytvoreného modelu. Ďalej som navrhol a naimplementoval automatické vyhľadávanie subdiagramov. Prevod modelu z WF do BPMN som sa rozhodol vynechať pre nedostatok času, tento prevod vyžaduje rovnako veľa práce ako prevod z BPMN do WF. Na prevod z WF do BPMN som pri implementácii nástroja zanechal potrebné informácie v objektoch, aby bol tento prevod možný. Integrácia vytvorených komponentov do nástroja Visual Studio som sa rozhodol vynechať z časovej náročnosti.

11.1 Možnosti rozšírenia

Aplikáciu by bolo možné rozšíriť v nasledujúcich smeroch:

1. Načítanie BPMN workflow aktuálne nepodporujú choreografické a konverzačné diagramy. Rozšírenie vďaka modularite bude značne zjednodušené.
2. Ďalším rozšírením by mohlo byť automatická konverzia externých užívateľských kódov. Rozšírenie bude vyžadovať investigáciu konverzie kódov prevažne z platformy java na platformu C#. Možnosťou by bolo vytvoriť spustiteľný jar kód, ktorý by následne spúšťala implementácia C# aktivity.
3. Možnosť pre rozšírenie je aj v prezentácii aktuálneho behu aplikácie užívateľovi v notácií WF. Toto rozšírenie bude vyžadovať širokú investigáciu, prípadne implementovanie rozsiahleho modulu.

4. Údržba nástroja je podmienená novými verziami špecifikácie BPMN. Akonáhle bude vydaná nová verzia, spolu s ňou sa môžu vyskytnúť aj nové prvky. Nástroj bude nutné rozšíriť o tieto prvky.

Literatura

- [1] Workflow Management dostupné z <http://wwwis.win.tue.nl/~wvdaalst/publications/p120.pdf>.
- [2] Business Process Model and Notation (BPMN) Version 2.0 Normative Documents dostupné z <http://www.omg.org/spec/BPMN/2.0/PDF> p. 27 – 42.
- [3] BPMN PDF Quick Reference Guide dostupné z https://www.nomagic.com/images/guides/no_magic_quick_reference_guide_bpmn.pdf.
- [4] BPMN 2.0 handbook – Dr. Bruce Silver dostupné z <http://www.conradbock.org/white-bpmn2-process-bookmark-web.pdf> p. 10 – 12.
- [5] Business Process Mapping using BPMN 2.0 Modelling Notation Tutorial dostupné z http://www.cragssystem.co.uk/bpmn_tutorial/bpmn_2_0_tutorial.pdf.
- [6] BPMN 2.0 by Example dostupné z http://www.omgwiki.org/bpmn2.0-ftf/lib/exe/fetch.php?media=public:sub-teams:bpmn_2.0_by_example_version_alpha_8.pdf.
- [7] Modeling Business Processes with BPMN dostupné z http://www.dis.uniroma1.it/~marrella/slides/Sem_PM_11-12_BPMN.pdf.
- [8] An Introduction to BPMN 2.0 dostupné z <http://www.temida.si/~bojan/UNG-PIS-2013/Introduction%20to%20BPMN.pdf> p. 2 – 7.
- [9] An introduction to business process modeling notation dostupné z <http://ccsl.ime.usp.br/baile/files/IntroductionToBPMN.pdf>.
- [10] BPMN Modeling and Reference Guide Digital Edition, Stephen A. White, PHD and Derek Miers
- [11] The Workflow Way: Understanding Windows Workflow Foundation dostupné z <http://www.davidchappell.com/TheWorkflowWay--Chappell.pdf>
- [12] Microsoft press Windows Workflow Foundation step by step dostupné z <https://epdf.tips/microsoft-press-windows-workflow-foundation-step-by-step.html> .
- [13] Introduction To Windows Workflow Foundation dostupné z <http://www.guysmithferrier.com/downloads/workflow.pdf> p. 41 – 44.
- [14] White, Bayer: Pro WF 4.5. Apress, 2013, ISBN 978-1-4302-4383-0, DOI 10.1007/978-1-4302-4384-7.p. 30 – 35
- [15] Windows Workflow Foundation Tutorial dostupné z <http://billatnapier.com/wwf.pdf>.

- [16] Activiti in Action: Executable business processes in BPMN 2.0.
- [17] Windows Workflow Foundation using C# dostupné z <http://www.itcourseware.com/docs/1-08-00241-000-07-10-13/evals/1-08-00241-000-07-10-13-sample.pdf>.
- [18] Požiadavky na software, Karl E.Wiegens 2008.
- [19] Mistrovství ve Windows Presentation Foundation, Charles Petzold 2008 p. 84 – 93.
- [20] Zpracování obrazu a algoritmy v C#, Michal Dobeš, 2008 p. 104 – 114
- [21] The Railway Engineer, Lawrence Saunders, S. R. Blundstone,1921
- [22] Informácie na webovej stránke: <https://www.signavio.com/post/what-workflow-management-is/>
- [23] Introduction to Workflow,Aalst and Kees van Hee, 2004 dostupné z <http://pages.di.unipi.it/ferrari/CORSI/SISD/Lezioni/WFM1.pdf>
- [24] Overview of the Workflow Engine dostupné z https://docs.oracle.com/cd/B13789_01/workflow.101/b10286/wfapi.htm
- [25] Informácie na webovej stránke: <https://www.ibm.com/support/knowledgecenter>
- [26] The Workflow Engine Model, Informácie na stránke: [https://docs.microsoft.com/en-us/previous-versions/office/developer/office2000/aa188337\(v=office.10\)](https://docs.microsoft.com/en-us/previous-versions/office/developer/office2000/aa188337(v=office.10))
- [27] Implementation of workflow engine technology to deliver basic clinical decision support functionality. Huser, V. Rasmussen, L. V. Oberg, R. Starren, J. B. (2011)
- [28] Presenting Windows Workflow Foundation, 2005. Paul Andrew, James Conard, Scott Woodgate, Jon Flanders, George Hatoun, Israel Hilerio, Pravin Indurkar, Dennis Pilarinos, Jurgen Willis