

Sem vložte zadanie Vašej práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Diplomová práce

Návrh a tvorba nástroje pro optimalizaci procesů na základě analýzy BPM modelů

Bc. Richard Mach

Vedúci práce: Ing. Josef Pavlíček, Ph.D.

3. mája 2015

Pod'akovanie

Rád by som podakoval vedúcemu mojej diplomovej práce Ing. Josefovi Pavlíčkovi, Ph.D. a Ing. Pavlovi Náplavovi za metodické vedenie práce, časté konzultácie a cenné rady.

Prehlásenie

Prehlasujem, že som predloženú prácu vypracoval(a) samostatne a že som uviedol(uviedla) všetky informačné zdroje v súlade s Metodickým pokynom o etickej príprave vysokoškolských záverečných prác.

Beriem na vedomie, že sa na moju prácu vzťahujú práva a povinnosti vyplývajúce zo zákona č. 121/2000 Sb., autorského zákona, v znení neskorších predpisov, a skutočnosť, že České vysoké učení technické v Praze má právo na uzavrenie licenčnej zmluvy o použití tejto práce ako školského diela podľa § 60 odst. 1 autorského zákona.

V Prahe 3. mája 2015

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2015 Richard Mach. Všetky práva vyhradené.

Táto práca vznikla ako školské dielo na FIT ČVUT v Prahe. Práca je chránená medzinárodnými predpismi a zmluvami o autorskom práve a právach súvisiacich s autorským právom. K jej využitiu, s výnimkou bezplatných zákonných licencií, je nutný súhlas autora.

Odkaz na túto prácu

Mach, Richard. *Návrh a tvorba nástroje pro optimalizaci procesů na základě analýzy BPM modelů*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2015.

Abstrakt

Táto práca sa zaoberá procesnými mierami kvantifikujúcimi rôzne aspekty biznis procesov, akými sú ich kvalita a zložitosť. Teoretická časť práce identifikuje a kategorizuje procesné miery, rozoberá dostupné grafické notácie a formáty modelov biznis procesov. Praktická časť práce implementuje nástroj analýzy biznis procesov s názvom BPMN Measures. Tento nástroj umožňuje vypočítať vybrané procesné miery a validovať vstupný formát modelu procesu.

Kľúčová slova biznis proces, zložitosť procesu, procesné miery, BPMN, XPD, BaseX, XQuery, Bizagi Modeler

Abstract

This work deals with process metrics quantifying various aspects of business processes, such as their quality and complexity. The theoretical part of the work identifies and categorizes the process metrics, discusses the available graphical notations and formats of business process models. The practical part implements business analysis tool called BPMN Measures. This tool allows the calculation of selected process metrics and validation of input process model format.

Keywords business process, process complexity, process metrics, BPMN, XPDL, BaseX, XQuery, Bizagi Modeler

Obsah

Úvod	1
1 Cieľ práce	3
2 Metodika práce	5
3 Business process modeling	7
3.1 Business process	7
3.2 Process complexity	9
3.3 Notácie	10
3.4 Smerovanie ku kvalite procesov	12
4 Miery procesov	13
4.1 Potreba procesných mier	14
4.2 Kritéria procesných mier	14
4.3 Kategorizácia procesných mier	17
4.4 Metóda výskumu mier	18
4.5 Rozbor procesných mier	20
4.6 Size	21
4.7 Modularity	22
4.8 Complexity	24
4.9 Coupling	31
4.10 Cohesion	33
4.11 Výber procesných mier	33
5 Modely biznis procesov	37
5.1 Formáty exportu modelu biznis procesu	37
5.2 Možnosti exportu modelovacích nástrojov	40
6 Praktická časť práce	45

6.1	Aplikácia procesných mier	45
6.2	Porovnanie implementácií procesných mier	47
6.3	BPMN Measures	47
6.4	Navrhované zlepšenia špecifikácie XPDL	51
6.5	Hodnotenie	52
	Záver	55
	Literatúra	57
	A Zoznam použitých skratiek	65
	B Výsledky rešeršu	67
	C Proces „Zařazení studentů do oboru“ - staršia verzia	71
	D Proces „Zařazení studentů do oboru“ - novšia verzia	73
	E Obsah priloženého CD	77

Zoznam obrázkov

3.1	Vnorenie konštrukcií vetvenia[3]	9
3.2	Stavebné bloky jazyka EPC[13]	10
3.3	Stavebné bloky Activity Diagram[14]	11
4.1	Histogram procesných mier	20
4.2	Ukážka kontrolného grafu programu[42]	26
4.3	Paralelné aktivity[4]	29
4.4	Závislosti aktivít[4]	29
4.5	Caption for LOF	30
5.1	Porovnanie formátov výmeny modelov podľa formalizmu Workflow Patterns[53]	39
5.2	Grafická podoba procesu v notácii BPMN	42
6.1	Schéma programu BPMN Measures	48
C.1	„Zařazení studentů do oboru“	72
D.1	„Zařazení studentů do oboru“	74
D.2	„Stanovení kapacit oborů“	75
D.3	„Zjištění zájmu a rozřazení studentů“	76

Zoznam tabuliek

4.1	Mapovanie medzi OO softvérom a BPMN notáciou[20]	13
4.2	Kognitívne váhy BPM elementov[34]	25
6.1	Hodnoty procesných mier pre proces „Zařazení studentů do oboru“	53

Úvod

V dnešnom rýchlo meniacom sa prostredí sa spoločnosti snažia získať akúkoľvek výhodu aby ostali konkurencieschopné. Jednou z ciest poskytujúcou takúto výhodu je zlepšovanie samotných firemných biznis procesov.

Jorge Cardoso dokonca tvrdí nasledujúce: „Designing and improving processes is a key aspect for businesses to stay competitive in today’s marketplace.“[1] Teda, že dizajnovanie a vylepšovanie procesov je pre spoločnosti kľúčovým aspektom ich konkurencieschopnosti.

Je veľmi žiaduce aby boli procesy firmy špecifikované, priebežne kontrolované a najmä zdokonaľované. Aby bol tento manažment procesov zvládnuteľný v reálnom čase, je potreba softvérových nástrojov, ktoré umožňujú biznis proces zachytiť a ďalej s ním pracovať.

Tieto potreby dali za vznik oblasti nazývanej „Business process modeling“, teda modelovaniu biznis procesov. Zachytenie biznis procesu do modelu, ktorý je vďaka softvérovým nástrojom prístupný celej rade analytikov, doménových expertov a vývojárov umožnilo zefektívniť vývoj biznis procesov.

Výsledkom externých požiadaviek na biznis procesy akými sú technologický pokrok alebo kratšie cykly vývoja a faktom, že sa na dizajne procesov podieľa celá skupina odborníkov je postupné zvyšovanie zložitosti biznis procesov. Aby boli procesy naďalej zlepšované a tým zvyšovaná ich užitočnosť pre zainteresované strany musí byť vzrastajúca zložitosť procesu znižovaná.

Cestou k redukcii zložitosti procesov je rozpoznať jej existenciu a dokázať ju vyjadriť a odmerať. Aby sme boli schopní popísať zložitosť procesov potrebujeme inštrumenty schopné kvantifikovať vlastnosti biznis procesov. Ukazuje sa, že takéto inštrumenty sú už dlho používané na určovanie zložitosti programov a nazývajú sa softvérové miery.

Mnohé zo softvérových mier sa dajú adaptovať a použiť pre potreby ohodnotenia zložitosti biznis procesov. Popri týchto adaptovaných mierach vznikajú nové miery určené výhradne pre popis procesnej zložitosti. Problémom procesných mier je však ich relatívna mladosť. Zatiaľ čo softvérové miery popisuje množstvo prác, ktoré im dávajú teoretické základy, vierohodnosť a výpovednú

hodnotu, v prípade procesných mier je situácia odlišná. Oblasť procesných mier je málo preskúmaná a v mnohých prípadoch procesné miery stále čakajú na svoje overenie a nasadenie v praxi.

V mojej práci sa snažím rešeršou nájsť dostupné procesné miery, kategorizovať ich a popísať na ne kladené kritéria. Zaoberám sa modelovacími jazykmi procesov a formátmi výmeny modelov. Tieto teoretické základy vedú k vývoju knižnice implementujúcej množinu mier zložitosti, ktorou sa snažím prispieť k praktickému využitiu procesných mier.

Ciel' práce

Cielom teoretickej časti práce je previesť systematický prehľad a preštudovanie dostupnej literatúry a nájsť odpovede na nasledujúce otázky:

1. Je možné merať kvalitu biznis procesov pomocou nejakých mier, ukazovateľov alebo iných metód?
2. Existujú miery, ukazovatele či metódy merania kvality procesov?
3. Existuje obecný formát uloženia procesných modelov a je z neho možné počítať miery kvality návrhu?
4. Používajú sa procesné miery v praxi?

Odpoveď na prvú otázku poskytuje tretia kapitola práce. Táto kapitola sa zaoberá modelovaním biznis procesov, popisuje kritériá kladené na modelovanie a samotné biznis procesy. Ukazuje sa, že zlepšovanie procesov je jedným z kľúčových aspektov konkurencieschopnosti a kvalita procesov musí byť pre potreby ich hodnotenia merateľná. Kvalita biznis procesov je úzko spätá s ich zložitou. Hodnotu kvality a zložitosti procesov vyjadrujú procesné miery, ktoré dokážu kvantifikovať vlastnosti procesov a sú často adaptovanými mierami z oblasti programovej zložitosti.

Štvrtá kapitola práce sa zaoberá kritériami kladenými na relevantné procesné miery a predstavuje jeden z používaných formalizmov validácie mier, Weyukerovej vlastnosti. Následný rešerš vedeckých publikácií za použitia erudovaných digitálnych knižníc identifikuje dvadsaťdva rozličných mier zaoberajúcich sa aspektami zložitosti biznis procesov. Nájsené miery sú v rámci kapitoly kategorizované a bližšie popísané. Tým táto kapitola odpovedá na druhú otázku.

Možnosti exportu modelu biznis procesov popisuje piata kapitola teoretickej časti práce. Kapitola konštatuje fakt, že neexistuje obecný formát modelu vhodný pre jeho analýzu a výmenu medzi modelovacími nástrojmi. Preto kapitola sumarizuje najpoužívanejšie formáty výmeny modelov a hodnotí ich

1. CIEĽ PRÁCE

pomocou formalizmu Workflow Patterns. Výstupom tejto kapitoly je voľba formátu XPDL ako vhodného formátu pre výmenu modelu biznis procesov a výpočtu procesných mier.

Prvá sekcia praktickej časti práce obsahuje popis troch nástrojov, implementujúcich procesné miery, ktoré sa mi podarilo identifikovať v rešeršnej časti práce. Každý z týchto nástrojov sa zameriava na iné aspekty zložitosti procesov a ich existencia implikuje použitie procesných mier v praxi.

Cielom praktickej časti práce je vytvoriť knižnicu v programovacom jazyku Java, ktorá bude implementovať sadu procesných mier nad XPDL reprezentáciou modelu biznis procesu vytvoreného v notácií BPMN. Implementované procesné miery sú vybrané na základe výstupov teoretickej časti práce.

Ďalším z výstupov praktickej časti práce sú doporučené zlepšenia formátu XPDL. Tieto zlepšenia vyplynuli zo štúdia špecifikácie formátu a problémov, ktoré vyvstali pri implementácii knižnice procesných mier. Posledným výstupom je overenie funkcionality knižnice na sade procesov, ktoré vznikli v rámci mapovania procesov na ČVUT FEL.

Metodika práce

Metodika práce je založená na podrobnej analýze existujúcich procesných mier, ktorá je prevedená na základe rešerše a štúdia odbornej literatúry. Znalosti nadobudnuté štúdiom budú použité na zhodnotenie súčasného stavu procesných mier, ich vlastností a na nich kladených kritérií. Z identifikovaných procesných mier budú pomocou kvantitatívnej analýzy vybrané miery k ďalšiemu popisu a kategorizácií. Z popísaných mier budú vybraté konkrétne procesné miery k ich implementácií v praktickej časti práce.

Druhým predpokladom pre implementáciu procesných mier bude výber reprezentácie modelu biznis procesu. Tento výber prebehne na základe porovnania exportov modelu biznis procesu z dostupných modelovacích nástrojov, podľa ich možností a podpory štandardov pre výmenu modelov.

Implementácia riešenia bude prevedená v programovacom jazyku Java a bude pozostávať z implementácie knižnice, umožňujúcej výpočet hodnôt procesných mier. Jednotlivé procesné miery budú vybraté na základe výstupov rešeršnej časti práce. Okrem implementácie procesných mier ponúkne praktická časť práce možnosť overenia reprezentácie biznis modelu oproti zvolenému štandardu.

Výsledná implementácia bude zhodnotená na vzorke modelov biznis procesov vypracovaných v rámci mapovania biznis procesov prebiehajúcich na ČVUT FEL.

V závere práce bude doporučený ďalší postup a možné rozšírenia rozobranej problematiky.

Business process modeling

Business process modeling je akt vytvárania modelov biznis procesov v modelovacích nástrojoch pre ich lepšiu analýzu a prípadné zlepšenie ich vlastností.

Jedným z hlavných dôvodov vývoja biznis proces modelov je podpora komunikácie medzi expertmi v procese vytvárania softvérového produktu. Medzi týchto expertov patria doménoví experti, analytici biznis procesov alebo softvéroví vývojári.[2]

Aby tento účel mohol biznis proces model spĺňať, mal by byť jednoducho čitateľný, porozumiteľný a umožňovať efektívne meniť jeho obsah podľa nových požiadaviek a obmedzení. [3]

Ak chceme vytvárať modely, ktoré sú ľahko pochopiteľné, musíme najskôr definovať, čo fráza „ľahko pochopiteľný model“ znamená. Prirodzene nás teda zaujímajú miery popisujúce vlastnosti modelu. Medzi vlastnosti modelu, ktoré chceme merať prednostne, patrí zmieňovaná ľahkosť porozumenia, no existujú ďalšie aspekty modelu, ktoré je prínosné vedieť kvalitatívne vyjadriť.

Z mier vychádzajúcich z biznis proces modelu chceme určiť, či je model korektne konštruovaný, neobsahuje syntaktické chyby, alebo ho nie je potreba dekomponovať na menšie, ľahšie uchopiteľné modely.

Dizajn a zlepšovanie biznis procesov je jedným z kľúčových aspektov pre organizáciu, aby sa dokázala udržať konkurencieschopná na dnešnom trhu. Táto potreba zlepšovania dospelosti procesov prichádza aj od vonkajších vplyvov, napríklad v podobe zákazníkov žiadajúcich lepšie produkty, či služby.[1]

V nasledujúcej sekcii definujem biznis proces, popíšem kritéria, ktoré musí proces spĺňať, aby dosiahol „well-structuredness“ a definujem samotný objekt skúmania procesných mier, komplexitu procesu.

3.1 Business process

Jednotlivé pohľady na definíciu biznis procesu sa navzájom odlišujú. Tieto odlišnosti závisia na tom, či sa na biznis proces pozeráme z vyššieho, biznis

pohľadu, alebo podrobnejšieho, technického pohľadu. Hannus[4] definuje biznis proces nasledovne:

„Business process is an entity which is composed of associated activities and operations and which begins from the need of a customer and finishes with satisfying the need.“

Túto definíciu chápem tak, že biznis proces je subjekt skladajúci sa z prídružených aktivít a operácií, ktorý začína z potreby zákazníka a končí uspokojením tejto potreby.

Definícia Koistinena[4] je podľa môjho názoru technickejšia a znie nasledujúco:

„Business process is a collection of activities, which takes input and generates output that is of value to the customer.“

Táto definícia definuje biznis proces ako kolekciu aktivít, ktoré na základe vstupu generujú pre zákazníka hodnotný výstup.

Obe definície popisujú biznis proces z odlišných hľadísk, no rovnako vyjadrujú jeho štruktúru ako kolekciu aktivít a operácií, ktorá sa zachytáva pomocou modelu biznis procesu.

S modelom biznis procesu úzko súvisí pojem „well-structuredness“. Pre potreby BPM definoval van der Aals[5] termín „well-structuredness“ nasledujúco:

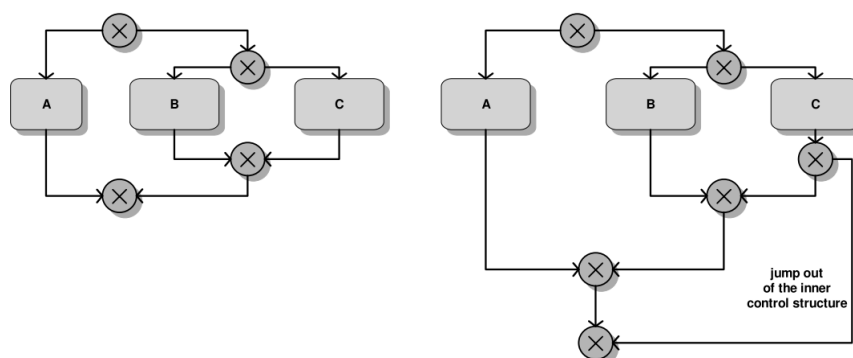
„A model is well-structured if the split/join constructions are properly nested.“

Táto na prvý pohľad jednoduchá definícia určuje kritérium dobrej štruktúry procesu ako správne vnorenie konštrukcií vetvenia procesu. Pre popis konštrukcií vetvenia, budem používať termíny split a join, namiesto slovenských prekladov - rozdelenia a spojenia. Anglické termíny sú podľa môjho názoru výstižnejšie ako ich slovenské ekvivalenty.

Formálne je termín „well-structuredness“ definovaný v oblasti Petriho sietí, ktoré modelujú proces ako špeciálny typ siete.[3]

Pre procesy vyjadruje „well-structuredness“ fakt, či sú konštrukcie vetvenia správne párované, teda či má každý split odpovedajúci join, obalujúci blok aktivít procesu. Obrázok 3.1 ukazuje dva modely, z ktorých ľavý obsahuje správne vnorenie konštrukcií vetvenia. Pravý model obsahuje „skok“ z bloku aktivít *B* a *C*, toto vetvenie nie je správne vnorené.[6]

Vo všeobecnosti môže byť vytváranie modelov procesov, ktoré nespĺňajú kritérium „well-structuredness“ považované za zlý štýl modelovania, ktorý zvyšuje zložitosť pochopenia modelu. Niektoré modelovacie jazyky, ako napríklad BPEL4WS[7] obsahujú sémantické obmedzenia, ktoré vôbec neumožňujú vytváranie modelov nespĺňajúcich toto kritérium.[3]



Obr. 3.1: Vnorenie konštrukcií vetvenia[3]

3.2 Process complexity

Podobne ako u biznis procesu sa aj v prípade komplexity ponúka viacero definícií. Fenton[8] ponúka definíciu komplexity ako množstvo zdrojov potrebných na riešenie problému. Card a Agresti[9] definujú systémovú komplexitu konkrétnejšie, ako súčet štrukturálnej a dátovej komplexity delený počtom zmien modulov.

Pre potreby procesov je však potreba vhodnejšia definícia komplexity. Takúto definíciu poskytuje IEEE[10], ktorý definuje komplexitu procesu nasledujúco:

„The degree to which a process is difficult to analyze, understand or explain. It may be characterized by the number and intricacy of activity interfaces, transitions, conditional and parallel branches, the existence of loops, data-flow, control-flow, roles, activity categories, the types of data structures, and other process characteristics.“

Podľa IEEE teda zložitosť procesu definuje, do akej miery je ťažké proces analyzovať, pochopiť a vysvetliť. To môže byť charakterizované počtom a zložitou rozhraní aktivít, prechodov, paralelných a podmienených vetvení, existenciou cyklov, toku dát, riadením toku, kategóriami aktivít, typom dátových štruktúr a ďalších charakteristík procesu.

Táto vyčerpávajúca definícia sumarizuje všetky potencionálne zdroje zložitosti procesu. Stále však ostáva otázka ako zložitosť procesu merať. Pre meranie zložitosti procesu potrebujeme notáciu schopnú zachytiť tento model. Edmonds[11] tvrdí, že komplexita biznis procesu musí byť meraná z modelu procesu. Z toho vyplýva, že musíme vybrať jazyk reprezentácie procesného modelu.

Preto v ďalšej sekcii popíšem notácie, v ktorých je možné biznis proces modelovať.

3.3 Notácie

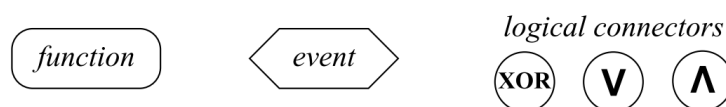
Existuje mnoho grafických notácií, v ktorých je možné vytvárať modely biznis procesov. V tejto sekcii stručne popíšem najpoužívanejšie notácie a odôvodním výber konkrétnej notácie pre praktickú časť diplomovej práce.

3.3.1 Event-driven Process Chains

Notácia Event-driven Process Chains (EPC) bola vyvinutá v roku 1992 v spolupráci so SAP a tvorí kľúčovú komponentu modelovacích konceptov SAPu v oblasti biznis inžinierstva. EPC je intuitívny grafický jazyk pre popis biznis procesov, ktorý je navrhnutý tak, aby popisoval procesy na úrovni ich biznis logiky.[12] Jazyk EPC pozostáva z nasledujúcich elementov:

- Funkcie - základné stavebné bloky jazyka, reprezentujúce aktivity, ktoré majú byť vykonané
- Udalosti - popisujú situácie pred a po vykonaní funkcie. Jednotlivé funkcie sú prepojené udalosťami
- Logické spojky - môžu byť použité na spájanie aktivít a udalostí. Pomocou nich sa určuje tok riadenia procesu. K dispozícii sú tri typy logických spojok: AND, OR, XOR.[12]

Grafickú podobu stavebných blokov EPC ukazuje obrázok 3.2.



Obr. 3.2: Stavebné bloky jazyka EPC[13]

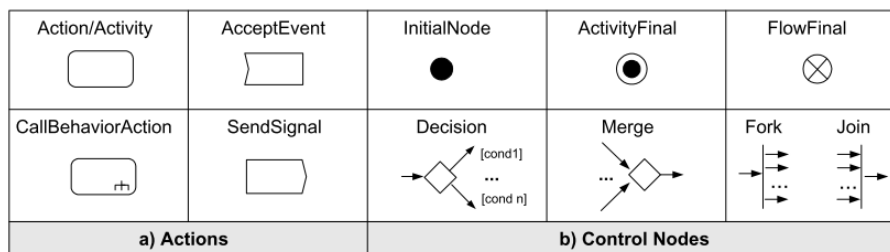
EPC sa stalo rozšíreným jazykom pre modelovanie biznis procesov vďaka úspešnosti produktov ako SAP alebo ARIS. Nanešťastie ako aj syntax, tak i sémantika EPC nie je jasne definovaná. Výsledkom toho je, že procesy modelované v EPC môžu byť nejednoznačné a nedajú sa overovať na kritériá konzistencie a úplnosti.[13]

Tieto nedostatky jazyku EPC sa riešia mapovaním na Petriho siete. Stavebné bloky EPC sú podobné ako v prípade Petriho sietí, v skutočnosti EPC korešponduje s podmnožinou Petriho sietí. Cez formalizmus Petriho sietí je možné overovať korektnosť procesov modelovaných v jazyku EPC za pomoci analytických nástrojov vyvinutých práve pre Petriho siete.[13]

3.3.2 Unified Modeling Language

Unified Modeling Language (UML) je považovaný za najširšie zameraný prostriedok pre modelovanie systémov. UML vo verzii 2.0 poskytuje niekoľko modelovacích formalizmov, schopných zachytiť statické i dynamické aspekty softvérových systémov. Pre modelovanie procesov sa zvlášť hodí podmnožina UML nazývaná Activity Diagrams (ADs).[14]

Základným stavebným prvkom Activity Diagrams, ktorý špecifikuje správanie, je Akcia. Akcia prijíma množinu vstupov a mení ju na množinu výstupov, pričom vstupná i výstupná množina môže byť prázdna. Jazyk ADs poskytuje viac ako 40 rôznych typov akcií, no popis všetkých akcií je nad rámec tejto práce.[15] Najrelevantnejšie typy akcií používaných pre modelovanie procesov ukazuje obrázok 3.3.



Obr. 3.3: Stavebné bloky Activity Diagram[14]

Jazyk UML vo verzii 2.0 je vhodný na modelovanie biznis procesov, no ako ukazuje [14], nie je rovnako vhodný pre všetky aspekty tohto typu modelovania. Na jednej strane ponúka notácia UML podporu pre riadenie toku procesov a dátového pohľadu na proces, avšak jej možnosti pre modelovanie aspektov procesov týkajúcich sa zdrojov alebo organizácie sú limitované.

V ďalšej sekcii sa preto popíšem jazyk BPMN, ktorý je považovaný za nepísaný štandard pre modelovanie biznis procesov.

3.3.3 Business Process Model and Notation

Business Process Model and Notation (BPMN) je štandardnou notáciou pre zachytenie biznis procesov. Esenciu BPMN modelu procesu vyjadrujú dve základné typy konštruktov:

- Aktivity - predstavujú akcie alebo časti práce procesu, vykonávané ľuďmi, či softvérovými aplikáciami.
- Kontrolné uzly - zachytávajú riadenie toku procesu medzi jednotlivými aktivitami. [16]

Jazyk BPMN ale tvorí oveľa početnejšia množina konštruktov, ktorá obsahuje objekty, sekvencie, správy, vetvenia a mnoho ďalších elementov, prehľad ktorých uvádza dokument BPMN 2.0 poster.¹

Podobne ako v prípade EPC existuje mapovanie BPMN do iných jazykov za účelom overenia v ňom vytvorených modelov. Pre BPMN existuje mapovanie do jazyka BPEL, pre ktorý existujú overovacie formalizmy. Avšak toto mapovanie úplne nepokrýva konštrukty BPMN a jeho overovacia sila je tak obmedzená. [17]

Druhou možnosťou je mapovanie BPMN do formalizmu Petriho sietí. Toto mapovanie pokrýva všetky možnosti BPMN až na dve situácie, v ktorých mapovanie naráža na limitácie Petriho sietí. [16]

Od roku 2009 prebieha na ČVUT FEL mapovanie biznis procesov, v rámci ktorého bolo doteraz popísaných viac ako 400 biznis procesov v notácii BPMN. Vďaka tomuto množstvu zachytených procesov, ktoré môžem využiť pre overenie funkčnosti praktickej časti mojej práce, som sa rozhodol práve pre použitie notácie BPMN.

3.4 Smerovanie ku kvalite procesov

Výsledkom externých požiadaviek na biznis procesy akými sú technologický pokrok, kratšie cykly vývoja alebo vzrastajúca konkurencia je postupné zvyšovanie komplexity procesov. Na zvládnutie tejto zložitosti existujú dva odlišné prístupy: zníženie zbytočnej zložitosti procesu a spravovanie zvyšku zložitosti. Ultimátnym cieľom znižovania a spravovania zložitosti je zlepšiť biznis proces a tým zvýšiť jeho užitočnosť pre zainteresované strany.[4]

Cestou k redukcii zložitosti procesu je rozpoznať jej existenciu a dokázať ju vyjadriť, odmerať. Ak máme mieru zložitosti pre biznis procesy, môžeme z jej výsledkov odvodiť hranice pre povolené hodnoty zložitosti. Hodnoty miery zložitosti sa dajú použiť na vyhľadanie tých procesov, ktorí sú kandidátmi na vylepšenie alebo zjednodušenie.[1]

Nasledujúca kapitola pojednáva o mierach procesov, od ich definície, kladených kritérií, prínosov a kategorizácie až po popis konkrétnych procesných mier.

¹http://www.bpmb.de/images/BPMN2_0_Poster_EN.pdf

Miery procesov

Za posledných 30 rokov boli navrhnuté miery charakterizujúce vlastnosti kódu programu, ako sú jeho zložitosť alebo zrozumiteľnosť. Medzi týmito mierami boli vyvinuté jednoduché miery pre programovacie jazyky ako C++, Java alebo Fortran. Mnohé z týchto mier poskytovali relevantný popis kvality dizajnu programu. Keďže je medzi softvérovými programami a biznis procesmi silná analógia[18], môžu byť tieto softvérové miery adaptované do oblasti analýzy biznis procesov.

Na biznis procesy modelované jazykmi ako sú BPEL[7] alebo BPMN[19] môže byť pozerané ako na tradičný softvérový program rozdelený do modulov (aktivít), ktoré prijímajú skupinu vstupov a produkujú výstupy. Interakcia týchto modulov je definovaná ich prepojeniami v sekvencii alebo pomocou elementov vetvenia (XOR, OR, AND). Medzi elementami programov a biznis procesov existuje mapovanie, ktoré približuje tabuľka 4.

Objektovo orientovaný softvér	BPMN notácia
Trieda/Balík	Subproces, proces
Metóda	Aktivita
Premenná/Konštanta	Dátový objekt
Komentár	Anotácia
Rozhranie triedy	Rozhranie procesu/subprocesu, množina aktivít v procese, ktoré posielajú alebo prijímajú správy
Lokálne dáta v triede	Dátové objekty priradené aktivitám procesu asociáciou
Dáta používané triedou	Dátové objekty priradené správam smerujúcim do aktivít procesu
Vyvolanie metódy	Prijatie sekvencie alebo správy aktivitou procesu

Tabuľka 4.1: Mapovanie medzi OO softvérom a BPMN notáciou[20]

Ako som spomenul vyššie, miery procesov sú často adaptované softvérové miery využívajúce analógiu medzi programom a biznis procesom. Okrem týchto adaptovaných mier však vznikajú aj miery určené výhradne pre analýzu procesov.

Obecnú definíciu procesných mier ponúka Cardoso[21], ktorý definuje procesnú mieru nasledujúco:

„Process metric is any type of measurement related to a process. Process metrics allows attributes of processes to be quantified.“

Podľa Cardosa je teda procesná miera akýkoľvek typ merania súvisiaci s procesom. Procesná miera umožňuje kvantifikovať atribúty procesov.

Napriek faktom, že sú softvérové miery dlhodobo etablované a že sú mnohé procesné miery odvodené zo svojich softvérových náprotivkov, je o nich prekvapujúco málo výskumov a literatúry obecné.[1] Podľa mňa je teda vhodné sumarizovať informácie o procesných mierach.

V nasledujúcich sekciách práce preto popíšem kritériá, ktoré by mala relevantná procesná miera spĺňať, kategorizujem procesné miery podľa aspektov zložitosti biznis procesu a rešeršou nájdem konkrétne, používané procesné miery.

4.1 Potreba procesných mier

Meranie komplexity modelov biznis procesov získava čoraz vyššiu váhu pri ich dizajne a úprave. Analýza referenčného modelu SAPu ukázala, že komplexita je úzko spätá s pravdepodobnosťou výskytu chýb v procese.[22] S ohľadom na toto zistenie je perspektívne používať miery zložitosti vo fáze návrhu procesných modelov s cieľom zlepšiť pochopiteľnosť modelu a znížiť jeho náchylnosť k chybám.[23]

Hodnoty mier zložitosti môžu byť použité pri efektívnom manažmente procesov, ako indikátory potreby vylepšenia produktivity a samotnej kvality procesu. So zvyšovaním sa komplexity procesu totiž klesá jeho kvalita.[21]

V neposlednom rade môžu procesné miery vo fáze dizajnu modelu procesu slúžiť ako smernice. Miera Cognitive weights 4.8.1 môže byť priebežne používaná ako odhad zložitosti porozumenia procesu a miera CFC 4.8.2 ako počet potrebných testovacích scenárov procesu. Ponúka sa aj použitie dodatočných mier z rodiny Number Of Activities 4.6.1 alebo BPM (Anti)Patterns[3] pre lepšiu kontrolu nad kvalitou procesu.

4.2 Kritéria procesných mier

Miery zložitosti by mali spĺňať sadu kritérií určujúcich kvalitu a použiteľnosť miery. Vyhovenie týmto kritériám zvyšuje kredit a dôveru v danú procesnú

mieru. Latva-Koivisto[4] navrhuje nasledujúce kritéria kladené na procesné miery:

- Validita - zaisťuje, že miera zložitosti meria to, čo sa od nej očakáva.
- Spôľahlivosť - merania získané rôznymi pozorovateľmi rovnakého procesu sú konzistentné.
- Vyčísliteľnosť - počítačový program dokáže vypočítať hodnotu miery v konečnom čase.
- Jednoduchosť implementácie - zložitosť implementácie, ktorá počíta hodnotu procesnej miery je v rozumných medziach.
- Intuitívnosť - definícia miery nie je náročná na pochopenie, je zjavná jej súvislosť so zložitou procesom.
- Nezávislosť od ďalších mier - v ideálnom prípade nie je hodnota miery závislá na ďalších vlastnostiach spojených s komplexitou, ako sú veľkosť a grafická reprezentácia procesu.

K vyššie spomenutým kritériám sa pridávajú ďalšie, stanovené v[24], ktoré rozširujú požadované vlastnosti mier:

- Automatizácia - výpočet miery musí byť možné automatizovať.
- Aditívnosť - ak sú dve nezávislé merané štruktúry vložené do sekvencie, ich kombinovaná komplexita musí byť rovná alebo väčšia ako suma komplexít jednotlivých štruktúr.
- Nezávislosť na jazyku - miera by mala byť použiteľná nezávisle na použitom modelovacom jazyku procesu. Vypočítaná hodnota procesnej miery by mala byť rovnaká či je proces modelovaný v jazyku BPEL, WSFL, BPML, YAWL alebo inom.[21]

Spomenuté kritériá nie sú jedinými podmienkami kladenými na miery komplexity. V oblasti softvérových mier sa pre validáciu hojne používa formalizmus nazývaný Weyukerovej vlastnosti. Tento formalizmus sa dá, vďaka podobnosti softvérových programov a modelov biznis procesov, využiť aj na validáciu procesných mier.

4.2.1 Weyuker Properties

Nie všetky metódy a teórie vyvinuté v oblasti softvérových mier boli akceptované v praxi. Jedným z dôvodov je nedostatočná validácia mier, a tým nedostatok dôvery vo výpovednú hodnotu ich výsledkov.[21]

Na prekonanie tejto prekážky Weyuker navrhla vlastnosti, ktoré by mala spĺňať každá miera zložitosti. Weyukerovej vlastnosti sú známe pre svoj formalizmus a analytický prístup k validácii mier zložitosti. S použitím tohto formalizmu je možné odfiltrovať miery zložitosti s nežiadúcimi vlastnosťami.[25]

Weyukerová navrhla deväť axiémov pre hodnotenie mier zložitosti. Tieto axiémy sú zamerané na validáciu mier komplexity a preto jednoduché miery založené na meraní veľkosti alebo dĺžky tieto axiémy nespĺňajú. To bolo objektom kritiky Weyukerovej vlastností, no aj napriek kritike ostáva tento formalizmus pre validáciu mier jeden z najpoužívanejších.

Vďaka výraznej podobnosti štruktúry programov a modelov biznis procesov sa Weyukerovej vlastnosti dajú použiť a používajú na validáciu procesných mier.[26]

Podľa prác [25][21][27] je deväť Weyukerovej axiémov sumarizovaných nasledujúco:

- Axióm 1: Dobrá miera by mala rozlišovať medzi dvoma rozdielnymi procesmi tak, že nevráti pre oba rovnaké výsledky.
- Axióm 2: Zmena procesu musí zmeniť jeho komplexitu. Dobrá miery by mala túto zmenu zistiť.
- Axióm 3: Existujú dva rôzne procesy, ktoré majú rovnaké dátové typy a hodnoty ale rozdielne názvy premenných. Dobrá miera by pre oba procesy mala vrátiť rovnakú hodnotu komplexity.
- Axióm 4: Dva procesy môžu vyzeráť navonok rovnako, no byť rozdielne v ich vnútornej štruktúre. Dobrá miera by mala rozlíšiť tieto procesy podľa ich vnútornej štruktúry.
- Axióm 5: Pre dva procesy je komplexita procesu vzniknutého ich spojením rovnaká, alebo väčšia ako komplexita jednotlivých procesov. Dobrá miera by mala túto zmenu komplexity detekovať.
- Axióm 6: Je možné mať dva rovnaké procesy, ktorých komplexita po spojení s tretím procesom nie je rovnaká. Akt spájania procesov má potenciál pridať doplnkovú komplexitu nad rámec originálnych procesov. Dobra miera by mala byť schopná rozlíšiť medzi týmito procesmi.
- Axióm 7: Zmena poradia elementov ovplyvňuje komplexitu, dva rovnaké procesy môžu mať rôznu hodnotu komplexity po zmene poradia ich elementov. Dobrá miera by mala túto zmenu komplexity detekovať.
- Axióm 8: Ak sa dva procesy líšia len v názvoch ich elementov, potom sú rovnaké. Dobrá miery by mala vrátiť rovnakú hodnotu komplexity pre oba procesy.

- Axióm 9: Pre dva procesy je komplexita procesu vzniknutého ich spojením rovnaká, alebo väčšia ako suma jednotlivých komplexít procesov. Dobrá miera by mala tento fakt postihnúť.

4.3 Kategorizácia procesných mier

Ku komplexite procesu prispieva mnoho faktorov. Tieto faktory sú tak rozdielne, že sa nedajú merať jednou univerzálnou mierou, ktorá by pokryla všetky ich vlastnosti. Tento fakt je dlho známy z oblasti mier softvérovej komplexity.[3]

Riešením tohto problému je vyvinúť špecifické miery pre konkrétne faktory zložitosti. Rozdelenie zložitosti na jednotlivé faktory potom implikuje samotnú kategorizáciu procesných mier.

Existuje niekoľko kategorizácií procesných mier. Tieto kategorizácie vychádzajú z analýzy zložitosti biznis procesov alebo z kategorizácie softvérových mier. Cardoso[21] identifikoval štyri hlavné perspektívy komplexity:

- Activity complexity - pohľad na komplexitu aktivít vyjadruje koľko aktivít obsahuje biznis proces. Miery zamerané na túto oblasť sú inšpirované softvérovými mierami typu „lines-of-code“ počítajúcimi riadky kódu programu.
- Control-flow complexity - komplexita riadenia toku procesu je ovplyvňovaná elementami vetvenia procesu, ktoré definujú možné prechody procesom. Miery ako CFC(viz 4.8.2) na základe počtu týchto elementov vyjadrujú zložitosť procesu.
- Data-flow complexity - stupeň dátovej zložitosti procesu rastie so zvyšujúcou sa zložitou dátových štruktúr a parametrov aktivít.[18] Miery v tejto oblasti sa zaoberajú súdržnosťou procesu, komplexitou dát alebo závislosťami medzi aktivitami procesu.
- Resource complexity - aktivity procesu potrebujú k svojej funkcionalite prístup ku rôznym zdrojom, komplexitu ktorých merajú miery z tejto oblasti.

Na základe podobnosti softvérových programov a biznis procesov sa kategorizácia procesných mier môže adaptovať práve z oblasti softvérovej zložitosti. Podľa[28] sa na kvalitu dizajnu vzťahuje nasledujúcich päť kategórií:

- Size - veľkosť modelu procesu priamo ovplyvňuje jeho zložitosť. Hoci sú miery zamerané na tento aspekt procesu jednoduché, často sa využívajú v kombinácii s ostatnými typmi procesných mier.

- Modularity - stupeň modularity procesu ovplyvňuje kvalitu jeho dizajnu, zložitosť jeho vykonávania a pochopenia. Miery v tejto oblasti merajú delenie procesu do subprocesov alebo vnorenie jednotlivých elementov vetvenia.
- Complexity - komplexita procesu rastie s počtom jeho kontrolných elementov, vetvení a samotnou veľkosťou procesu.
- Coupling - spojitosť sa zameriava na mieru prepojenosti procesu. Miery v tejto oblasti určujú počet informácií posielaných medzi aktivitami procesu alebo medzi procesmi navzájom. Stupeň spojitosti procesu závisí od komplikovanosti a typu prepojenia aktivít procesu.
- Cohesion - súdržnosť procesu sa zameriava na prepojenie procesu v rámci aktivít a berie v úvahu prenášané dáta.[29]

Pre kategorizáciu mier som sa v mojej práci rozhodol použiť zaužívané rozdelenie do piatich kategórií adaptované zo softvérovej oblasti.

V ďalšej sekcii popíšem metódu výskumu procesných mier, ktorej výsledkom sú nájdené miery, kategorizované do vyššie spomenutých skupín.

4.4 Metóda výskumu mier

V nasledujúcich sekciiach popíšem postup vyhľadávania materiálov týkajúcich sa procesných mier, digitálne databázy použité pri ich vyhľadávaní a kritériá kladené na vyhľadávané zdroje.

4.4.1 Použité digitálne knižnice

Priebeh vyhľadávania dostupnej literatúry o procesných mierach som prevádzal na základe pokynov pre vykonávanie systematických recenzií literatúry.[30] Pre zaistenie kvality literárnych zdrojov som vyhľadávanie obmedzil na erudované digitálne knižnice. Medzi použité knižnice patrili:

- Web of Science
- ACM Digital Library
- EBSCO
- IEEE Xplore
- Scopus
- SpringerLink

Oblasť, v ktorej som štúdiu prevádzal, sa zaoberá procesnými mierami. Pri vyhľadávaní relevantných zdrojov týkajúcich sa tejto problematiky som preto začínal použitím obecných kľúčových slov ako „Process metrics“ a „Bpmn measures“. Podľa nájdených zdrojov a ich príslušnosti do vyhľadávanej oblasti som granularitu vyhľadávania postupne zlepšoval spresňovaním použitých kľúčových slov.

Konečná podoba kľúčových slov sa ustálila na výrazoch „Process quality metrics“ a „Process complexity metrics“. Použitím týchto kľúčových slov som sa dopracoval k množine relevantných literárnych zdrojov.

Na základe nájdených zdrojov som bol schopný v poslednom kroku rešerše vyhľadávať konkrétne skupiny procesných mier pomocou výrazov:

- „Process coupling complexity“
- „Process cohesion complexity“
- „Control flow complexity“

Výsledky vyhľadávania literárnych zdrojov boli vytvorené v období 8. decembra 2014 až 24. februára 2015 a obsahujú zdroje publikované do tohoto dátumu.

4.4.2 Vyhľadávacie kritériá

Výsledky môjho výskumu som ďalej ovplyvňoval kritériami kladenými na vyhladané publikácie, ktorými som sa snažil ďalej zvýšiť kvalitu a relevantnosť výsledkov. Medzi použité kritériá vyhľadávania patrili:

- Celý text publikácie je dostupný online.
- Jazyk publikácie je obmedzený na anglický jazyk.
- Publikácia je citovaná v iných vedeckých prácach.
- Vyhľadaný zdroj má formu vedeckej práce, knihy alebo konferenčnej publikácie.

Pomocou vyššie spomenutých kľúčových výrazov a kritérií sa mi podarilo vyhľadať 32 relevantných literárnych prameňov zaoberajúcimi sa mierami procesov.

4.4.3 Výsledky rešerše

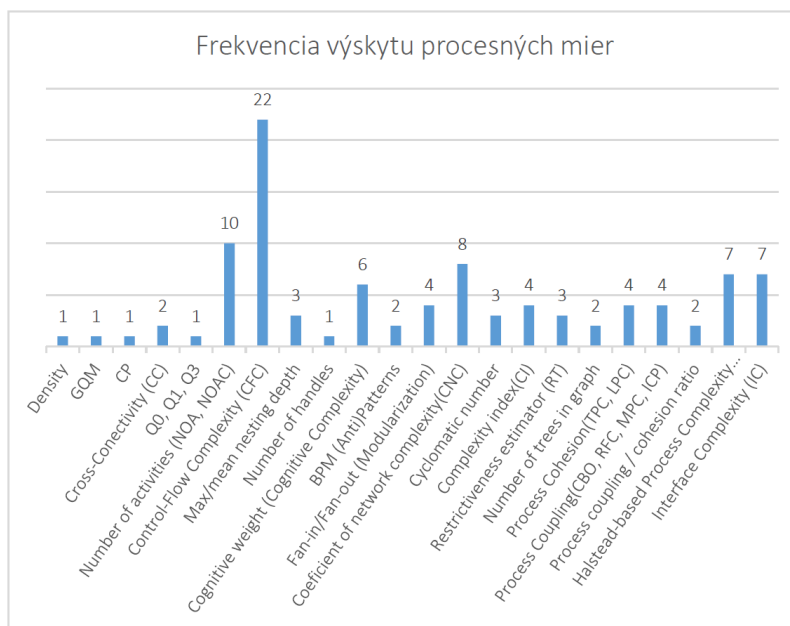
Každý z nájdených zdrojov som prečítal a vyťažil z neho informácie týkajúce sa mier zložitosti procesov. Takto zozbierané údaje o mierach som uložil vo forme tabuľky (viz príloha B). Formát tabuľky som zvolil z dôvodu ďalšieho spracovania metrik a ich kategorizácie.

Ako ukazuje príloha B v nájdených zdrojoch sa mi podarilo identifikovať 22 rozličných mier zaoberajúcich sa aspektami zložitosti biznis procesov. Z výsledkov rešerše vyplýva, že nie všetky nájdené miery sú zastúpené rovnako. Počty publikácií, ktoré sa zmieňujú o danej procesnej miere, naznačujú jej použiteľnosť a rozšírenosť.

V nasledujúcej sekcii opíšem, akým spôsobom som vybral a kategorizoval miery biznis procesov a vybrané miery bližšie popíšem.

4.5 Rozbor procesných mier

Tabuľka výsledkov rešerše ukazuje, že jednotlivé miery zložitosti procesov sú v nájdených prameňoch zastúpené v rozličnom počte. Po konzultácii s vedúcim mojej práce som sa rozhodol previesť kvantitatívnu analýzu procesných mier, na základe ktorej vyberiem signifikantné miery k ďalšej kategorizácii a popisu. Výsledné počty výskytov jednotlivých mier zobrazuje nasledujúci histogram.4.1



Obr. 4.1: Histogram procesných mier

Z nájdených mier sa budem ďalej venovať tým, ktoré majú početnosť vyššiu ako dva. Rozhodol som sa tak preto, lebo oblasť procesnej komplexity nie je zmapovaná tak dobre, ako softvérová komplexita a preto chcem pracovať len s najrozšírenejšími mierami. Týmto kritériám vyhovujú nasledujúce miery:

- Number of Activities

- Control-Flow Complexity
- Max/mean nesting depth
- Cognitive weight
- Fan-in/Fan-out metric
- Coefficient of network complexity
- Cyclomatic number
- Complexity Index
- Restrictiveness estimator
- Process Cohesion
- Process Coupling
- Halstead-based Process Complexity
- Interface Complexity

V ďalších sekciách práce bližšie popíšem konkrétne procesné miery zaradené do piatich kategórií podľa aspektov zložitosti procesu: Size, Modularity, Complexity, Coupling, Cohesion.

4.6 Size

Meranie veľkosti modelu biznis procesu patrí k najjednoduchším mieram zložitosti. V tejto sekcii popíšem miery NOA, NOAC a NOAJS, ktoré sú adaptáciami softvérovej miery LOC.

4.6.1 Number of Activities

Najjednoduchšou formou merania komplexity v prípade softvéru je počítanie riadkov kódu.[31] Táto miera sa nazýva Lines of Code (LOC). Pre programy písane v moderných programovacích jazykoch zvykne LOC vyjadrovať počet spustiteľných príkazov v programe.[3]

Vďaka podobnosti softvérovej komplexity s procesnou komplexitou sa môže koncept LOC uplatniť v sfére biznis procesov v podobe mier Number of Activities (NOA).

Ak chápeme aktivitu procesu ako analógiu príkazu v programe, môžeme odvodiť jednoduchú mieru NOA, merajúcu počet aktivít v procese.[3]

$$NOA = \text{počet aktivít v procese}$$

Táto miera však pokrýva len jeden aspekt procesu, jeho dĺžku, a neberie v úvahu ďalšie vlastnosti procesu. Pokročilejšou adaptáciou LOC miery je Number of Activities and Control-flow elements (NOAC)[32]. Táto miera sa dá aplikovať na procesy, ktoré sú „well-structured“ a nezapočítava len aktivity procesu, ale aj riadiace štruktúry ovplyvňujúce ich priebeh.

$$NOAC = \text{počet aktivít a riadiacich elementov v procese}$$

Niektoré modelovacie jazyky umožňujú tvorbu procesov, ktoré nespĺňajú kritérium „well-structuredness“. V tom prípade nie sú joins a splits procesu párované, čo vyúsťuje k menej čitateľnému procesu náchylnejšiemu na chyby. Ak chceme aj v tomto prípade merať počet aktivít a riadiacich elementov, musíme použiť tretiu mieru, Number of Activities, Joins, and Splits.[3]

$$NOAJ S = \text{počet aktivít, joinov a splitov v procese}$$

Procesné miery Number of Activities neberú do úvahy štruktúru modelu procesu[3], no aj napriek tejto slabine sú využívané ako doplnkové miery pre analýzu procesov. Neznalosť veľkosti modelu môže totiž viesť ku zavádzajúcim výsledkom pri použití ostatných mier, preto sú tieto miery potrebné aj napriek svojej jednoduchosti.[29]

4.7 Modularity

Miera modularity procesu vyjadruje hodnotu vnorenia subprocesov alebo elementov vetvenia procesu. V tejto sekcii popíšem mieru Fan-in/Fan-out zameranú na zložitosť subprocesov a jej podobnú mieru Interface Complexity. Poslednou popísanou mierou je Maximum/Mean nesting depth, ktorá vyjadruje zanorenie štruktúr vetvenia.

4.7.1 Fan-in/Fan-out (Modularization) metric

S rastúcou veľkosťou procesu rastie náročnosť na pochopenie jeho modelu, udržiavanie a prípadné zlepšovanie. Rozdelenie procesu na menšie časti, subprocesy, pomôže porozumeniu modelu. Rozdelenie taktiež často vedie k menším, znovu použiteľným modelom, ak je modularizácia použitá správne.

V oblasti softvérových mier navrhli Henry a Kafura[33] mieru založenú na toku informácií v programovej štruktúre. Táto miera identifikuje počet lokálnych informácií vstupujúcich do modulu programu (*fan-in*) a počet lokálnych informácií opúšťajúcich modul programu (*fan-out*).[34]

Pre analýzu modularizácie BPM môžeme adaptovať myšlienky predchádzajúcej softvérovej miery. Vzorec pre výpočet tejto miery modularizácie procesu je nasledujúci:

$$Modularization = (fan-in * fan-out)^2$$

kde *fan-in* je počet všetkých subprocessov, ktoré volajú skúmaný subprocess a *fan-out* je počet všetkých subprocessov, ktoré sú volané zo skúmaného subprocessu.[35]

Zvyčajne model subprocessu s vysokou hodnotou parametru *fan-in* vykonáva jednoduchú úlohu, ktorá je vyžadovaná v ostatných častiach procesu. Naopak vysoká hodnota parametru *fan-out* znamená, že je daný subprocess veľkou časťou modelovaného procesu a k svojej správnej funkcionalite potrebuje celú radu jednoduchších subprocessov.

Prítomnosť subprocessu s vysokou hodnotou oboch parametrov v modeli procesu značí, že zmena jeho návrhu môže vylepšiť proces.[3] Ak má model subprocessu vysokú komplexitu podľa Fan-in/fan-out miery, bude náročné tento proces používať a pravdepodobne je nesprávne navrhnutý.[6]

4.7.2 Maximum/Mean nesting depth

Jeden z aspektov zložitosti procesov, ktorý nepokrývali doteraz spomenuté miery, je hĺbka vnorenia.

Výskum v oblasti softvérovej komplexity ponúka miery „Maximum nesting depth“ a „Mean nesting depth“, ktoré sú vhodné k meraniu vnorenia a jeho vplyvu na celkovú zložitost programu. Vyššia hodnota vnorenia implikuje vyššiu komplexitu programu. Tieto miery sa dajú, vďaka podobnosti kontrolného grafu programu s procesným grafom, ľahko adaptovať do prostredia BPM.[3]

Hĺbka vnorenia je pre element procesu definovaná nasledujúco: „The nesting depth of an action is the number of decisions in the control flow that are necessary to perform this action.“[36] Teda hĺbka vnorenia je definovaná ako počet rozhodnutí riadenia toku potrebných k dosiahnutiu daného elementu.

Miery *Maximum nesting depth (MaxND)* a *Mean nesting depth (MeanND)* pre procesy sú definované nasledujúco:

$$MaxND = \text{maximálna hodnota hĺbky vnorenia procesu}$$

$$MeanND = \text{priemerná hodnota hĺbky vnorenia procesu}$$

Niektoré modely procesov môžu obsahovať vysoký počet elementov riadenia toku, cez ktoré sa proces dostane k svojmu koncu. Model s vnorenými XOR-split a XOR-join elementami môže byť ťažší na porozumenie než takmer lineárny model procesu, no hodnota miery CFC4.8.2 môže byť rovnaká pre oba modely.[6] Preto sa miery MaxND a MeanND dajú použiť spolu s mierou CFC za účelom presnejšieho merania zložitosti BPM.[37]

4.7.3 Interface Complexity

Práca Henry a Kafura[33] v oblasti toku informácií v programovej štruktúre bola inšpiráciou aj pre nasledujúcu mieru, Interface Complexity. Táto miera

bola adaptovaná podobne ako vyššie spomenutá Fan-in/Fan-out miera, no oproti nej navyiac pridáva dĺžku aktivít. Interface Complexity (IC) je vypočítaná pre každú aktivitu procesu nasledujúco:

$$IC = \textit{dĺžka} * (\textit{počet vstupov} * \textit{počet výstupov})^2$$

V tomto vzorci je *dĺžka* je definovaná na základe toho, či je aktivita reprezentovaná ako „black box“ alebo „white box“. V prípade „black box“ podáva aktivita informácie len o svojom rozhraní, nie o vnútornej štruktúre, *dĺžka* je definovaná ako 1. Ak je aktivita procesu popísaná ako „white box“, tak je *dĺžka* rovná počtu úloh tejto aktivity.[34]

Výhodou miery IC je fakt, že berie do úvahy „data-driven“ procesy a môže byť vypočítaná vo fáze návrhu modelu, bez nutnosti funkčnej implementácie. Nevýhodou miery je však jej správanie sa v hraničných situáciách. Napríklad ak meraná aktivita nemá výstupné interakcie s ostatnými aktivitami, bude výsledok jej komplexity podľa tejto miery 0. Táto situácia nastáva typicky s koncovými aktivitami procesu a skresľuje celkovú komplexitu procesu.[38]

4.8 Complexity

Táto sekcia práce predstavuje najpočetnejšiu kategóriu mier zložitosti. Z popisovaných mier sú nasledujúce štyri odvodené z teórie grafov: Cyclomatic number, Coeficient of Network Complexity, Complexity Index a Restrictiveness estimator.

Odlíšny prístup zvolila miera CFC zameriavajúca sa na počet možných ciest procesom alebo miera HPC určujúca počty jedinečných elementov procesu. Poslednou priblíženou mierou je Cognitive weight metric snažiaca sa priradiť procesu zložitost' na základe empirických meraní náročnosti jeho štruktúr.

4.8.1 Cognitive weight metric

Cognitive weights metric je ďalšou z adaptácií softvérových mier. Vychádza z práce Shao a Wang[39], ktorí definovali túto mieru ako množstvo úsilia potrebného na porozumenie časti softvéru. Na základe empirických štúdií priradili jednotlivým riadiacim štruktúram programu váhu podľa náročnosti danej štruktúry na pochopenie.[2]

Miera Cognitive weights je pre biznis proces model definovaná nasledujúco: „The cognitive weight of a BPM is the sum of the cognitive weights of its elements.“[2]

Podľa Gruhna je teda miera Cognitive weights definovaná ako suma kognitívnych náročností elementov modelu.

Nasledujúca tabuľka 4.8.1 zobrazuje kontrolné štruktúry BPM, k nim analogické softvérové riadiace štruktúry a priradené váhy jednotlivým BPM elementom.

BPM štruktúra	Softvérová štruktúra	Kognitívna váha
Po sebe idúce kroky v procese	Sekvencia	1
XOR-split nasledovaný odpovedajúcim XOR-join (2 možnosti)	if - then vetvenie	2
XOR-split nasledovaný odpovedajúcim XOR-join (3 a viac možností)	case vetvenie	3
AND-split nasledovaný odpovedajúcim AND-join	paralelné vykonávanie	4
OR-split nasledovaný odpovedajúcim OR-join	case vetvenie nasledované paralelnou exekúciou	7
Subproces	volanie funkcie	2
Beh viacerých inštancií aktivity	vetvenie nasledované paralelnou exekúciou	6
Ukončenie (aktivácia aktivity ukončí inú aktivitu)	–	1
Ukončenie (aktivácia aktivity ukončí všetky elementy časti modelu)	porovnateľné s volaním funkcie	2 alebo 3

Tabuľka 4.2: Kognitívne váhy BPM elementov[34]

Na základe týchto váh elementov vyjadrujúcich úroveň náročnosti ich pochopenia môžeme ohodnotiť celý model procesu a určiť jeho kognitívnu váhu.

Výskum Misra a Misra[40] sa zaoberal aplikovaním formalizmu Weyukerovej vlastností 4.2.1 na mieru kognitívnych váh. Výsledkom bolo splnenie väčšiny z Weyukerovej kritérií kladených na túto mieru, čo dokazuje, že Cognitive weight je dobre štruktúrovanou mierou.

Existujú aspekty komplexity biznis proces modelov, ktoré nemôžu byť merané pomocou Cognitive weight miery. Medzi tieto aspekty patrí najmä komplexita schémy modelu a textuálna komplexita.

Grafické zobrazenie schémy modelu môže značne prispieť k zložitosti jeho porozumenia. Pri návrhu schémy by sa malo dbať na prehľadnosť a efektivitu jej zobrazenia.[2]

Kognitívne váhy elementov predpokladajú, že každá aktivita rovnakého typu vyžaduje rovnaké množstvo úsilia na jej pochopenie. No aj v rámci rovnakého typu aktivít sa zložitost' ich porozumenia líši, k čomu prispieva aj ich popis. Popis aktivít v BPM je zvyčajne v prirodzenom jazyku a je doplnkom k porozumeniu aktivity, komplexita textového popisu elementov BPM tak ďalej zvyšuje komplexitu modelu.[2]

Najväčším nedostatkom tejto miery je však fakt, že sú jednotlivým elementom empiricky priradené ordinálne hodnoty, trpí tým výpovedná hodnota

miery. Pri porovnaní dvoch modelov procesov pomocou miery Cognitive weights nevieme presne posúdiť, o koľko je reálne jeden proces zložitejší ako ten druhý.

4.8.2 Control-flow Complexity

Miera Control-flow Complexity (CFC) je založená na myšlienkach Thomasa J. McCabe.

McCabe navrhol mieru pre meranie počtu lineárne nezávislých ciest v programe, ktorá je nezávislá na programovacom jazyku. Táto miera sa nazýva McCabe's cyclomatic complexity (MCC) a je odvodená od kontrolného grafu programu.[41] MCC je indikátorom zložitosti riadenia toku informácií v programovom module a stavia na predpoklade, že komplexita programu má vzťah k počtu možných ciest programom.[42]

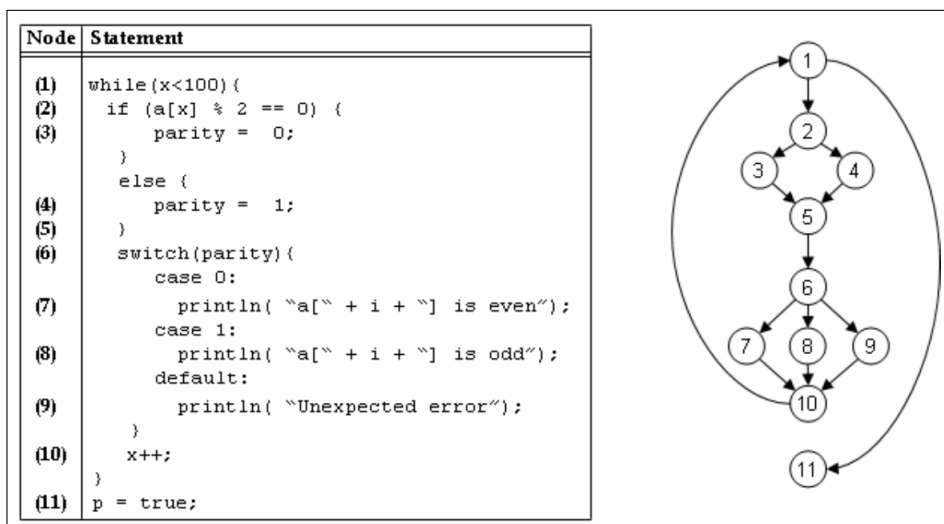
McCabe [41] definuje vzorec pre výpočet MCC nasledovne:

$$MCC = e - n + 2p$$

Vo vzorci premenná e označuje počet hrán, n je počet uzlov kontrolného grafu programu a p je počet komponent grafu.

Kontrolný graf programu predstavuje jeho logickú štruktúru, uzly reprezentujú stavy výpočtu a hrany prechody medzi stavmi. Všetky možné cesty programom majú svoju obdobu v kontrolnom grafe.[42] Medzi kontrolným grafom programu a modelom procesu je značné podobenstvo, tento fakt využíva miera Control-flow Complexity.

Obrázok nižšie 4.2 ukazuje kontrolný graf pre časť kódu určujúcu paritu čísla.



Obr. 4.2: Ukážka kontrolného grafu programu[42]

Control-flow Complexity meria komplexitu troch štruktúr vetvenia v modeli procesu - XOR-split, OR-split a AND-split. Použitie niektorej zo štruktúr vetvenia v procese vedie k zvýšeniu možných prechodov procesom a tým k väčšej komplexite procesu. Každý typ vetvenia prispieva k zvýšeniu komplexity procesu inou mierou.

- XOR-split predstavuje bod v priebehu procesu, kde sa na základe rozhodnutia vyberie práve jedna cesta z viacerých možných prechodov procesom. Zložitosť tohto vetvenia vyjadruje CFC ako funkciu $fan-out(a)$, ktorá vracia počet prechodov smerujúcich z aktivity a . [21]

Vzorec pre výpočet CFC miery pre XOR-split je definovaný nasledovne:

$$CFC_{XOR-split}(a) = fan-out(a)$$

- OR-split predstavuje bod v priebehu procesu, kde sa na základe rozhodnutia vyberie jedna, alebo viacero ciest z možných prechodov procesom. [21]
Vzorec pre výpočet OR-split vetvenia je definovaný nasledovne:

$$CFC_{OR-split}(a) = 2^{fan-out(a)} - 1$$

Pri návrhu alebo analýze procesu sa teda pridaním OR-split vetvenia musí zväžiť $2^n - 1$ stavov, ktoré môžu byť dosiahnuté prechodom týmto vetvením. Proces, ktorý začal, musí svoj priebeh ukončiť. Preto je výpočet všetkých ciest OR-splitu znížený o jednu, vždy sa musí vybrať aspoň jedna cesta, ktorou bude proces pokračovať.

- AND-split predstavuje potrebu vykonať dve alebo viacero aktivít paralelne. [21]

Vzorec pre výpočet AND-split štruktúry je definovaný nasledovne:

$$CFC_{AND-split}(a) = 1$$

Môže sa zdať, že sa pridaním paralelného radenia aktivít proces výrazne komplikuje, no z pohľadu počtu možných prechodov procesom tomu tak nie je. Pri prechode procesom sa musia vždy vykonať všetky paralelné aktivity za AND-splitom, počet možných ciest procesom sa teda zvýši o jedna. [21]

Matematicky je CFC miera aditívna, výpočet komplexity procesu teda pozostáva zo súčtu CFC mier štruktúr vetvenia. Vzorec pre výpočet miery, kde p je proces a a aktivita je nasledujúci:

$$\begin{aligned} CFC(p) = & \sum_{a \in p, a \equiv XOR-split} CFC_{XOR-split}(a) + \\ & \sum_{a \in p, a \equiv OR-split} CFC_{OR-split}(a) + \\ & \sum_{a \in p, a \equiv AND-split} CFC_{AND-split}(a) [38] \end{aligned}$$

Čím je vyššia hodnota CFC, tým je vyššia zložitosť procesu, pretože rastie počet stavov medzi štruktúrami vetvenia.[42] Validácia CFC miery ukázala jej vysokú koreláciu so zložitosťou biznis procesu a tým s jeho zrozumiteľnosťou a modifikovateľnosťou.[43]

V prípade Weyukerovej vlastností CFC plne uspokojuje sedem z deviatich vlastností a čiastočne uspokojuje jednu vlastnosť. Vďaka tomu sa dá CFC kategorizovať ako dobre štruktúrovaná a pochopiteľná miera.[21]

4.8.3 Halstead-based Process Complexity

Miery založené na práci Halsteada[44] sú jedny z najznámejších a najštudovanejších mier softvérovej komplexity. Tieto miery sa snažia merať komplexitu a porozumiteľnosť programu ako funkciu jeho operandov a operátorov. Halsteadove miery používajú štyri premenné (n_1 , n_2 , N_1 , N_2), ktoré sú zo zdrojového kódu vypočítané nasledovne:

- n_1 = počet jedinečných operátorov (if, case, while, ...)
- n_2 = počet jedinečných operandov (premenných a konštánt)
- N_1 = celkový počet operátorov
- N_2 = celkový počet operandov

V prípade procesnej miery Halstead-based Process Complexity (HPC) sa na tieto štyri merania mapujú elementy biznis procesu. Premenná n_1 bude reprezentovať počet jedinečných aktivít, splitov, joinov a elementov riadenia toku procesu. Obdobne bude premenná n_2 obsahovať počet jedinečných dátových položiek, ktoré prechádzajú aktivitami procesu. Premenné N_1 a N_2 obsahujú podobne ako v prípade softvérovej miery celkové počty elementov procesu a prechádzajúcich dát.[38]

S pomocou vyššie spomenutých premenných sú HPC miery pre odhad dĺžky, rozsahu a zložitosti procesu definované nasledovne:

- Dĺžka procesu: $N = n_1 * \log_2 n_1 + n_2 * \log_2 n_2$
- Rozsah procesu: $V = (N_1 + N_2) * \log_2(n_1 + n_2)$

- Zložitosť procesu: $D = (n_1/2) * (N_2/n_2)$ [34]

Miera CFC spomínaná v predchádzajúcej sekcii sa zameriava na počet stavov v biznis proces modeloch, zatiaľ čo HPC miera na počet komponent a dát manipulovaných v BPM.[45] HPC miera nepotrebuje pre svoje použitie poznať do detailov štruktúru modelu procesu, dokáže predpovedať chybovosť procesu a náročnosť jeho udržateľnosti.[6]

4.8.4 Coefficient of Network Complexity

Teória grafov ponúka širokú škálu mier, ktoré môžu byť adaptované vďaka podobnosti s modelom procesu do procesných mier. Jednou z jednoduchých mier adaptovaných týmto spôsobom je Coefficient of Network Complexity (CNC).

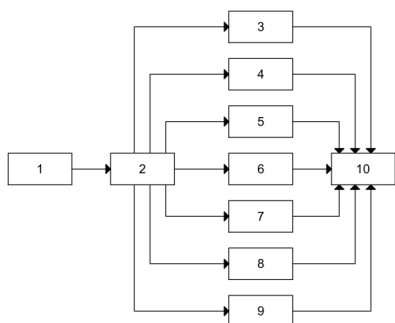
CNC miera môže byť jednoducho vypočítaná ako počet hrán grafu podelený počtom uzlov. V kontexte BPM to znamená, že počet prechodov medzi aktivitami procesu je delený počtom aktivít, joinov a splitov procesu.[38]

Vzorec pre výpočet miery je nasledovný:

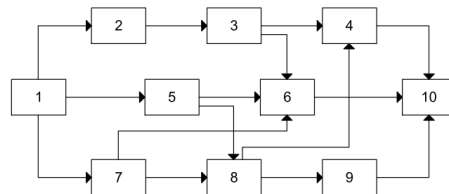
$$CNC = \text{počet prechodov} / (\text{počet aktivít, joinov a splitov}) [36]$$

Jednoduchosť CNC miery a jej výpočtu jej však neumožňuje v niektorých prípadoch správne určiť náročnosť porozumenia modelu procesu. Nasledujúce dva obrázky ukazujú abstrakciu procesného modelu, kde každý model obsahuje 10 uzlov a 15 prechodov.

Intuitívne sa ľahší na pochopenie zdá byť model na obrázku 4.3, ktorý zobrazuje paralelné aktivity. Druhý model obsahuje zložitejšie závislosti jednotlivých aktivít. Komplexita druhého modelu na obrázku 4.4 je vyššia, no CNC miera vypočíta pre oba modely rovnakú hodnotu.



Obr. 4.3: Paralelné aktivity[4]



Obr. 4.4: Závislosti aktivít[4]

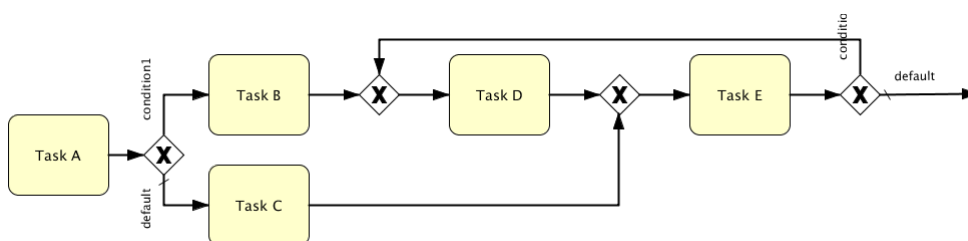
Zdá sa, že nevýhody CNC miery prevyšujú jej užitočnosť a má obmedzené použitie pri vyjadrovaní komplexnosti zložitejších procesov.[4]

4.8.5 Complexity Index

Ďalšou z mier adaptovanou z teórie grafov je Complexity Index (CI). Táto miera je definovaná nasledovne: „CI is defined as the minimal number of node reductions that reduces the graph to a single node.“ [38]

Teda miera Complexity Index je definovaná ako minimálny počet redukcií uzlov grafu, ktoré zredukujú graf na jediný uzol.

Complexity Index musí byť vypočítaný algoritmicky a nedá sa použiť na procesné modely obsahujúce cykly, ktoré majú viac než jeden vstupný alebo výstupný bod.[38] Príklad procesu s týmto typom cyklov ukazuje nasledujúci obrázok 4.5.



Obr. 4.5: Procesný model obsahujúci cyklus²

4.8.6 Restrictiveness estimator

Poslednou z adaptácií grafových mier do prostredia biznis procesov pokrytých v tejto práci je Restrictiveness estimator (RT).

Thesen[46] popisuje svojou mierou počet možných sekvencií v grafe. Tento údaj je však nemožné vypočítať v praxi pre dostatočne veľké grafy, preto boli navrhnuté odhady tejto miery, medzi inými aj RT.[38]

Restrictiveness estimator je definovaný nasledujúco:

$$RT = \frac{2 \sum r_{ij} - 6(N - 1)}{(N - 2)(N - 3)}$$

Vo vzorci označuje premenná N počet uzlov grafu a r_{ij} je element matice dosiahnuteľnosti.[4]

Hodnota miery RT je štandardizovaná na interval $[0,1]$, meria komplexitu procesného grafu vzhľadom k jeho veľkosti. Táto vlastnosť je vo všeobecnosti požadovaná, pretože komplexita by mala byť vlastnosť nezávislá na veľkosti.

V praxi to však znamená, že jednoduchý malý graf procesu môže mať vysokú hodnotu komplexity, pretože ju obsahuje v najvyššej možnej miere. Pri porovnávaní zložitosti procesov mierou RT tak treba vždy brať do úvahy veľkosť grafu.

²<http://en.bpmn-community.org/process/show/133/>

4.8.7 Cyclomatic Number

Miera Cyclomatic Number(CN) je priamou adaptáciou prác McCabe (viz 4.8.2) do oblasti mier procesných modelov. Podobne ako miera CNC, CN meria prepojenosť procesného grafu. Miera CN je nezávislá na veľkosti procesného grafu, zaoberá sa iba jeho štruktúrou. Vzorec pre výpočet tejto miery je nasledujúci:

$$CN = A - N + 1$$

Vo vzorci označuje premenná A počet hrán a N je počet uzlov procesného grafu.[4]

Kritika miery CN je rovnaká ako v prípade CNC, má rovnakú hodnotu pre dva rôzne procesné grafy s rovnakým počtom hrán a uzlov, ale intuitívne rozdielnym stupňom zložitosti.[4]

4.9 Coupling

V predchádzajúcich mierach som popísal rôzne aspekty zložitosti procesu, akými sú jeho veľkosť, modularita, komplexita riadenia toku alebo vnorenie. Táto sekcia sa zameriava na zložitosť prechodov medzi jednotlivými aktivitami procesu, na stupeň ich spojitosti a závislosti.

Definícia spojitosti pre procesy je prevzaná z definície formulovanej pre oblasť softvérového inžinierstva:

„Coupling measures the number of interconnections between the activities in a process model. The degree of coupling depends on how complicated the connections are and also on the type of connections between the activities.“[47]

Túto definíciu chápem tak, že spojitosť popisuje počet prepojení medzi aktivitami procesného modelu, pričom jej stupeň závisí na komplikovanosti a type prepojenia aktivít.

Miera spojitosti aktivity reflektuje ako dôležitá je táto aktivita v rámci procesu. Aktivity s vysokou hodnotou miery spojitosti funkčne určujú veľký počet ostatných aktivít procesu. BPM s vysokou hodnotou spojitosti indikuje vysokú úroveň informačnej závislosti medzi jeho aktivitami.[32]

Predpokladá sa, že procesy s vysokou spojitosťou budú obsahovať viac chýb, než procesy s nízkou spojitosťou.[29]

Stupeň spojitosti procesu meria niekoľko mier, v tejto práci popíšem jednu mieru vytvorenú priamo pre potreby merania zložitosti procesov a spomeniem ďalšie tri adaptované miery z oblasti softvérovej komplexity.

Pôvodná miera pre meranie spojitosti procesu je inšpirovaná softvérovými mierami a zahŕňa váhu jednotlivých typov spojení medzi aktivitami (AND, OR, XOR). Táto miera má názov CP a jej vzorec je nasledovný:

$$CP = \frac{\sum_{a_1, a_2 \in T} \text{connected}(a_1, a_2)}{|T| * (|T| - 1)}$$

$$\text{connected}(a_1, a_2) =$$

$$\begin{cases} 1 & , \text{ if } (a_1 \rightarrow a_2) \wedge (a_1 \neq a_2) \\ 1 & , \text{ if } (a_1 \rightarrow \text{AND} \rightarrow a_2) \wedge (a_1 \neq a_2) \\ \frac{1}{(2^m-1)*(2^n-1)} + \frac{(2^m-1)*(2^n-1)-1}{(2^m-1)*(2^n-1)} * \frac{1}{m*n} & , \text{ if } (a_1 \rightarrow \text{OR} \rightarrow a_2) \wedge (a_1 \neq a_2) \\ \frac{1}{m*n} & , \text{ if } (a_1 \rightarrow \text{XOR} \rightarrow a_2) \wedge (a_1 \neq a_2) \\ 0 & , \text{ if } (a_1 = a_2) \end{cases}$$

Vo vzorci miery CP premenné a_1 a a_2 predstavujú aktivity procesu, m je počet prechodov smerujúcich do konektora (AND, OR, XOR), n je počet prechodov smerujúcich z konektora a T je množina aktivít procesu.[47]

Každý prechod medzi dvoma aktivitami procesu má inú váhu podľa typu spojenia:

- *AND* zväzuje aktivity najsilnejšie, pretože proces musí prejsť každou jeho vetvou. Preto je pravdepodobnosť použitia konkrétnej vetvy 1.
- *XOR* zväzuje aktivity najslabšie, pretože je nasledovaná iba jedna z jeho vetiev. Preto je pravdepodobnosť použitia konkrétnej vetvy $\frac{1}{m*n}$.
- *OR* má váhu medzi *XOR* a *AND*, pretože proces musí prejsť aspoň jednou vetvou. Preto je pravdepodobnosť použitia konkrétnej vetvy nasledujúca:

$$\frac{1}{(2^m-1)*(2^n-1)} + \frac{(2^m-1)*(2^n-1)-1}{(2^m-1)*(2^n-1)} * \frac{1}{m*n}$$

Adaptované miery vychádzajú zo softvérových mier MPC, RFC a ICP, ktoré sa zameriavajú na komunikáciu v rámci tried programu.

Miera MSPC adaptovaná z miery MPC sa zameriava na komunikáciu v rámci procesov. Táto miera počíta správy posielané z aktivít procesu aktivitám iného procesu.[20]

Druhou z adaptovaných mier je RFP, zameriavajúca sa na spojitost v podobe volania jednotlivých aktivít. RFP vychádza zo softvérovej miery RFC a meria počet aktivít, ktorých realizácia je vyvolaná aktivitou procesu.[20]

Poslednou adaptovanou mierou je IFPC, vychádzajúca z miery ICP. Podobne ako miera MSPC sa aj IFPC zameriava na komunikáciu procesov. Na rozdiel od vyššie spomenutej miery však IFPC kladie dôraz ja na veľkosť posielanej správy.[20]

Popis zložitosti procesu mierami zameriavajúcimi sa na spojitost naráža na nedostatky pri vyjadrení znovupoužiteľnosti modelu. Druhým nedostatkom je zameranie sa na posielané informácie a nie na závislosť aktivít na ich použití.[32]

Tieto deficity sa snažia pokryť miery zamerané na kohéziu, na ktoré sa zameriam v nasledujúcej sekcii práce.

4.10 Cohesion

Kohézia vyjadruje súdržnosť častí modelu procesu. Táto oblasť zložitosti biznis procesov je jednou z najmenej preskúmaných, na túto tému existuje len niekoľko vedeckých prác.

Reijers a Vanderfeesten[18] vyvinuli kohéznu mieru pre procesy, ktorá určuje súdržnosť medzi aktivitami procesu.

Podobne ako miera spojitosti sa aj táto miera zameriava na komunikáciu a prenos informácií medzi aktivitami procesu s ohľadom na prenášané dáta. Pre každú aktivitu procesu je jeho celková súdržnosť vypočítaná ako násobok informačnej a relačnej súdržnosti aktivity.

Hodnota miery pre celý proces je priemer jej hodnôt pre jednotlivé aktivity. Hodnota tejto miery kohézie vždy prináleží intervalu $[0,1]$. [29]

Vzorec pre výpočet tejto miery je nasledujúci:

$$c = \frac{\sum_{t \in T} c(t)}{|T|}$$

Vo vzorci premenná t predstavuje aktivitu procesu, T je množina všetkých aktivít procesu a $c(t) = \lambda(t) * \mu(t)$ pričom $\lambda(t)$ a $\mu(t)$ sú relačná, respektíve informačná súdržnosť aktivity t ktorých presné znenie je možno nájsť v[18].

Údaje zo štúdie[48] ukazujú, že pri procesoch je žiadúca nízka hodnota spojitosti a vysoká hodnota súdržnosti. Existujú taktiež predpoklady, že procesy s nízkou súdržnosťou budú obsahovať väčšie množstvo chýb, než procesy s vyššou súdržnosťou.[49]

4.11 Výber procesných mier

V predchádzajúcich kapitolách mojej práce som sa zaoberal otázkou, či je potrebné merať kvalitu biznis procesov a či existujú prostriedky na meranie kvality biznis procesov. V dnešnej dobe sú tendencie smerovania ku kvalite biznis procesov. So vzrastajúcimi požiadavkami na biznis procesy rastie aj tlak na zvyšovanie ich kvality a to najmä redukovaním ich zložitosti.

Ukázalo sa, že existuje celá rada procesných mier, zaoberajúcich sa meraním kvality a zložitosti rôznych aspektov biznis procesov. Tieto miery sa od seba odlišujú v dvoch fundamentálnych rovinách.

Prvá rovina rozlišuje medzi jednotlivými procesnými mierami podľa vlastností biznis procesov, na ktoré sa miery zameriavajú. Nájdené miery vyjadrujú zložitosť riadenia toku procesu, počtu jeho elementov či prepojenosti a závislosti jeho jednotlivých štruktúr. Ako sa ukázalo, z tohto zamerania procesných mier vychádza aj ich kategorizácia.

Druhá rovina predstavuje použiteľnosť mier. Na použiteľnosť vplýva niekoľko faktorov:

- Teoretický základ miery - niektoré z predstavených mier sú založené na silných teoretických základoch, napríklad na teórií grafov, ktoré zvyšujú dôveryhodnosť miery. Naopak niekoľko mier, ktoré sú spomenuté v prílohe B, sú len experimentálnymi mierami, bez overených teoretických základov.
- Overenie miery - použiteľnosť miery zvyšuje jej overenie pomocou zaužívaných formalizmov, akými sú napríklad Weyuker Properties 4.2.1. Z popísaných mier toto overenie splňujú miery CFC a Cognitive weights.
- Štandardizácia miery - ukotvenie miery v štandardoch, či sa jedná o štandard jazyka, alebo metamodelu procesov konkretizuje špecifikáciu miery a tým prispieva k jej použiteľnosti, napríklad niektoré z mier typu NOA sú charakterizované v štandardoch BPDM.[50]
- Potrebná notácia - procesné miery potrebujú k vyjadreniu zložitosti procesu určité elementy modelu biznis procesu. Vo väčšine sa jedná o základné konštrukty, akými sú aktivity alebo prechody procesu. V prípade miery Cognitive weights, však musia elementy, ktorým je empiricky priradená váha existovať, a to nemusí byť pravidlom vo všetkých notáciách. Podobne miery, ktoré dokážu určiť správne zanorenie konštruktov vetvenia požadujú, aby sa notácia dala mapovať do Petriho sietí, formalizmu, ktorý toto overenie umožňuje. Preto podpora notácie mierou taktiež vplýva na jej použiteľnosť.

Na základe vyššie uvedených kritérií som sa rozhodol v praktickej časti mojej práce implementovať nasledujúce procesné miery:

- Control-flow Complexity
- Coefficient of Network Complexity
- Cyclomatic Number
- Fan-in/Fan-out metric

- Interface Complexity
- Maximum Nesting Depth
- Mean Nesting Depth
- Number of Activities
- Number of Activities and Control-flow elements
- CP

Modely biznis procesov

V predošlých kapitolách práce som sa zaoberal notáciami biznis procesov a procesnými mierami zaoberajúcimi sa vyjadrením kvality a zložitosti procesov. Táto kapitola sa zameria na možnosti reprezentácie modelov biznis procesov z pohľadu dostupných formátov a ich implementácie modelovacími nástrojmi.

Prvá časť kapitoly priblíži dostupné formáty výmeny modelov a uvedie formalizmus, pomocou ktorého sa dajú tieto formáty hodnotiť. Druhá časť kapitoly popíše realizáciu možnosti exportu vo vybranom formáte v konkrétnych modelovacích nástrojoch.

5.1 Formáty exportu modelu biznis procesu

Aby sa dali modely biznis procesov ďalej skúmať, musí ich byť možné exportovať z modelovacích nástrojov vo formáte vhodnom pre ďalšiu analýzu modelu. Avšak oblasť výmeny modelov procesov je heterogénna, existuje niekoľko formátov na výmenu modelov biznis procesov. Napriek tomu, že sa diskutuje o štandardizácii, absencia všeobecne uznávaného formátu je stále hlavnou prekážkou. Všeobecne akceptovaný formát je potrebný pre presun BPM medzi nástrojmi od rôznych výrobcov.[51]

Absencia všeobecného formátu výmeny modelov procesov vyústila do definície rozličných formátov rôznymi konzorciami, akými sú Object Management Group (OMG), World Wide Web Consortium (W3C), Business Process Management Initiative (BPMI), Workflow Management Coalition (WfMC) a ďalšími vývojármi softvéru či akademickými skupinami.

Z týchto definícií vzišlo množstvo navzájom nekompatibilných formátov. V nasledujúcom zhrnutí formátov popíšem tie najvýznamnejšie z nich.

- **BPDM** - Business Process Definition Metamodel je vyvíjaný konzorciom OMG, ktorý definuje koncepty, vzťahy a sémantiku pre výmenu modelov medzi rôznymi modelovacími nástrojmi. Samotný formát je definovaný pomocou XSD a XMI.

- BPEL4WS - Business Process Execution Language for Web Services je špecifikovaný ako formát výmeny modelov za pomoci XSD. BPEL modeluje aktivity procesu ako volania webových služieb, ktorých vstup a výstup je špecifikovaný správami a adresa je identifikovaná za pomoci URI. Používaným komunikačným protokolom je SOAP.
- BPML - Business Process Modeling Language navrhnutý skupinou BPMI je veľmi podobný formátu BPEL, hlavným rozdielom je možnosť špecifikovať viacero procesov v rámci jedného XML dokumentu spolu s ich komunikáciou.
- UML - podmnožina UML s názvom Activity Diagrams môže byť prenášaná medzi modelovacími nástrojmi využitím jej metamodelu a formátu XML.
- EPML - Event-Driven Process Chain Markup Language je akademickým návrhom, ktorý zachytáva elementy notácie EPC. EPML sa snaží uľahčiť výmenu grafickej informácie o modeli, preto obsahuje grafické informácie o polohe každého elementu EPC.
- WSCI - Web Service Choreography Interface navrhnutý konzorciom W3C poskytuje množinu rozšírení WSDL na popis procesu a výmeny správ.
- WSFL - Web Service Flow Language je predchodcom formátu BPEL. Riadenie toku je vo formáte WSFL modelované pomocou orientovaného grafu.
- XLANG - formát od firmy Microsoft je druhým z predchodcov formátu BPEL. Podobne ako formát WSCI používa rozšírenia WSDL pre popis procesov pomocou webových služieb.
- XPDL - XML Process Definition Language je štandardizovaný formát výmeny modelov biznis procesov navrhnutý konzorciom WfMC.[51]

Každý z týchto formátov poskytuje iné možnosti zachytenia modelu biznis procesu a je postavený na rozdielnych technológiách, i keď prevažuje využitie technológií XML. Na vyjadrenie možností zachytenia modelu každého z formátov môže byť použitý zaužívaný formalizmus s názvom Workflow Patterns.

5.1.1 Workflow Patterns

Workflow Patterns[52] sú množinou vzorov, ktoré sa týkajú rozdielnych aspektov modelovania biznis procesov, akými sú riadenie toku, dáta, zdroje a spracovanie výnimiek. Tieto vzory popisujú, nezávisle na implementácií, opakujúce sa interakcie, problémy a osvedčené riešenia v objektovo-orientovaných systémoch. Množina vzorov predstavuje funkčnosť, ktorá by mala byť podporovaná v modelovacom jazyku biznis procesov.

Formáty výmeny modelov sa dajú použitím Workflow Patterns hodnotiť podľa toho, či dokážu zachytiť konkrétny vzor. Výsledky tohto hodnotenia môžu byť použité pre posúdenie vhodnosti jednotlivých formátov. Na hodnotenie formátov je použitá množina dvadsiatich vzorov zameriavajúca sa na zachytenie rôznych perspektív procesu.

5.1.2 Výber exportného formátu

Porovnaním formátov výmeny modelov pomocou Workflow Patterns sa zaoberajú mnohé práce. V [53] Van der Aalst porovnáva vyššie spomenuté formáty XPDL, BPEL4WS, BPML, WSFL, UML, XLANG a WSCI.

Tabuľka 5.1 ukazuje vyhodnotenie týchto formátov pomocou formalizmu Workflow Patterns. Ak je vzor formátom plne podporovaný jedným z jeho konštruktov, je v tabuľke označený symbolom +. V prípade, že vzor nie je priamo podporovaný, je označený +/- . Každý pokus o riešenie vzoru konštruktami formátu, ktoré vedú k „špagetovému efektu“ je označený symbolom -.

pattern	standard						
	XPDL	UML	BPEL4WS	BPML	XLANG	WSFL	WSCI
1 (seq)	+	+	+	+	+	+	+
2 (par-spl)	+	+	+	+	+	+	+
3 (synch)	+	+	+	+	+	+	+
4 (ex-ch)	+	+	+	+	+	+	+
5 (simple-m)	+	+	+	+	+	+	+
6 (m-choice)	+	-	+	-	-	+	-
7 (sync-m)	+ ²	-	+	-	-	+	-
8 (multi-m)	-	-	-	+/-	-	-	+/-
9 (disc)	-	-	-	-	-	-	-
10 (arb-c)	+ ³	-	-	-	-	-	-
11 (impl-t)	+ ⁴	-	+	+	-	+	+
12 (mi-no-s)	+	-	+	+	+	+	+
13 (mi-dt)	+	+	+	+	+	+	+
14 (mi-rt)	-	+	-	-	-	-	-
15 (mi-no)	-	-	-	-	-	-	-
16 (def-c)	-	+	+	+	+	-	+
17 (int-par)	-	-	+/-	-	-	-	-
18 (milest)	-	-	-	-	-	-	-
19 (can-a)	-	+	+	+	+	+	+
20 (can-c)	-	+	+	+	+	+	+

Obr. 5.1: Porovnanie formátov výmeny modelov podľa formalizmu Workflow Patterns[53]

Z predchádzajúceho hodnotenia nevychádza žiaden formát ako jasný víťaz. Vzory podporované jednotlivými formátmi sa líšia, no ani jeden z formátov nepodporuje výrazne viac vzorov ako ostatné. Avšak z týchto formátov sa formát BPEL4WS, jeho predchodcovia XLANG a WSFL, a formát BPML zameriavajú na webové služby, ktoré nie sú objektom mapovania na ČVUT FEL.

V predchádzajúcich častiach práce som z dostupných grafických notácií vybral BPMN a tento výber favorizuje formát XPDL nad formátmi UML, respektíve EPML zameranými na iné notácie.

V rámci mapovania biznis procesov bol pri vytváraní ich modelov v notácii BPMN používaný modelovací nástroj od spoločnosti QPR s názvom QPR ProcessDesigner, ktorý bližšie popíšem v nasledujúcej sekcii práce. Tento nástroj podporuje export vytvoreného modelu biznis procesu v niekoľkých formátoch, medzi ktorými figuruje z vyššie spomenutých formát XPDL.

Vďaka týmto skutočnostiam som sa rozhodol v praktickej časti mojej práce využiť formát XPDL ako vstup pre analýzu modelov biznis procesov a ich ohodnotenie pomocou vybraných procesných mier.

V nasledujúcej sekcii práce bližšie popíšem formát XPDL a vybrané modelovacie nástroje s podporou exportu modelu v tomto formáte.

5.2 Možnosti exportu modelovacích nástrojov

Existuje veľké množstvo modelovacích nástrojov ponúkajúcich export vytvorených modelov v rôznych formátoch pre výmenu modelov. Ja sa zameriam na nástroje podporujúce formát XPDL. Bližšie priblížim samotný formát XPDL a spôsob akým reprezentuje model biznis procesu. Nakoniec popíšem a porovnam reálny stav exportu vo vybraných modelovacích nástrojoch.

5.2.1 Formát XPDL

Špecifikácie XPDL a BPMN sa zameriavajú na rovnaký problém z iných perspektív. XPDL ponúka XML formát, ktorý môže byť použitý pri prenášaní procesných modelov medzi modelovacími nástrojmi. BPMN ponúka grafickú notáciu na uľahčenie komunikácie medzi užívateľmi zložitých biznis procesov.[54] Ďalej sa zameriam na popis vlastností najnovšej špecifikácie XPDL - verzie s označením 2.2.

Jednou z kľúčových vlastností formátu XPDL je jeho rozšíriteľnosť, ktorá umožňuje uložiť informácie používané rôznymi modelovacími nástrojmi. Túto rozšíriteľnosť umožňujú generické konštrukty formátu, ktoré podporujú špecifické atribúty modelovacích nástrojov v rámci všeobecnej reprezentácie formátu.[54]

XPDL používa syntax XML, špecifikovanú pomocou XML schémy. Hlavnými elementami tejto syntaxe sú: *Package*, *Application*, *WorkflowProcess*, *Activity*, *Transition*, *Participant*, *DataField* a *DataType*.

Element *Package* slúži ako kontajner pre všetky ostatné elementy.

Element *Application* špecifikuje zoznam všetkých aplikácií a nástrojov, ktoré procesy definované v rámci *Package* používajú a potrebujú pre svoju funkcionálnosť.

Element *WorkflowProcess* sa používa na definovanie procesov alebo ich častí. Hlavnú časť tohto elementu tvoria elementy typu *Activity* a *Transition*.

Element *Activity* je základným blokom procesu a popisuje tri typy aktivít - *Route*, *Implementation* a *BlockActivity*, ktoré definujú vetvenie, jednotlivé kroky procesu a množiny aktivít.

Elementy typu *Transition* predstavujú prepojenia medzi jednotlivými aktivitami procesov. Element *Participant* sa používa na špecifikovanie účastníkov procesu, XPDL podporuje šesť rôznych typov účastníkov.

Posledné dva elementy *DataField* a *DataType* popisujú relevantné dáta procesu, ktoré sú používané na rozhodovanie, alebo ukazujú na údaje mimo procesu a sú predávané medzi aktivitami a subprocesmi.[53]

Nasledujúci úryvok ukazuje základnú štruktúru procesu reprezentovaného vo formáte XPDL. Úryvok sa zameriava na celkový pohľad na štruktúru, neobsahuje niektoré atribúty elementov a popis ich obsahu.

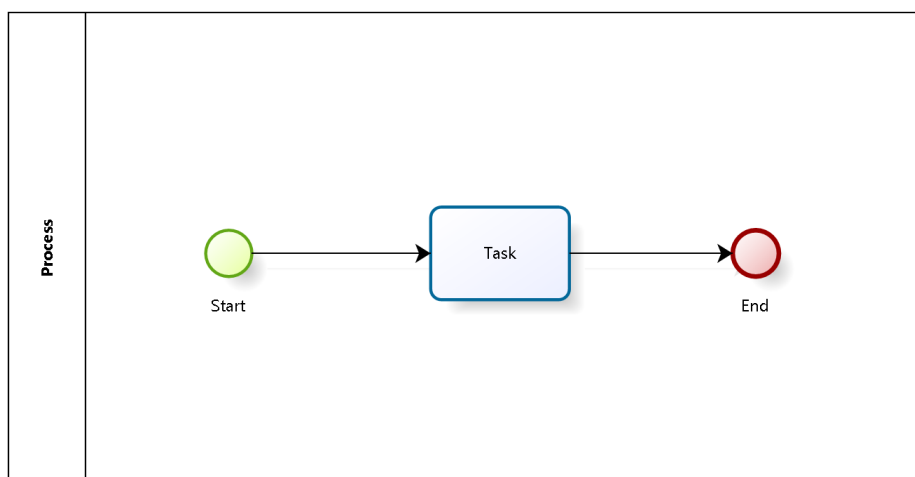
```

1 <?xml version="1.0" encoding="utf-8"?>
2 <Package xmlns:xsd="http://www.w3.org/2001/XMLSchema"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   Id="b1419bb1-773f-4a30" Name="Diagram 1"
5   xmlns="http://www.wfmc.org/2009/XPDL2.2">
6   <PackageHeader/>
7   <DataStores/>
8   <Pools>
9     <Pool Id="f8b16ad8-76dd" Name="Process"
10      Process="de8a12e0-0430">
11       <Lanes/>
12     </Pool>
13   </Pools>
14   <Associations/>
15   <Artifacts/>
16   <WorkflowProcesses>
17     <WorkflowProcess Id="de8a12e0-0430" Name="Process">
18       <ActivitySets/>
19       <Activities>
20         <Activity Id="639d6a54-9825" Name="Start"/>
21         <Activity Id="b8dc2ee3-5f33" Name="Task"/>
22         <Activity Id="793dcb24-0dca" Name="End"/>
23       </Activities>
24       <Transitions>
25         <Transition Id="34714a00-02c1" From="639d6a54-9825"
26           To="b8dc2ee3-5f33"/>
27         <Transition Id="a9147a17-2f02" From="b8dc2ee3-5f33"
28           To="793dcb24-0dca"/>
29       </Transitions>
30     </WorkflowProcess>
31   </WorkflowProcesses>
32   <ExtendedAttributes/>
33 </Package>

```

Grafickú podobu rovnakého procesu reprezentovaného v notácii BPMN ukazuje nasledujúci obrázok 5.2. Tento jednoduchý proces obsahuje jedinú aktivitu *Task* a dve udalosti *Start* a *End*.

Mnohé elementy a atribúty formátu XPDL sú definované ako nepovinné, čo



Obr. 5.2: Grafická podoba procesu v notácii BPMN

vychádza v ústrety potrebám rôznych modelovacích nástrojov a rozširiteľnosti, no praktická časť práce poukáže na to, že táto benevolencia prináša určité prekážky pri analýze modelov v tomto formáte.

5.2.2 QPR

Spoločnosť QPR ponúka niekoľko produktov zameriavajúcich sa na vytváranie a zlepšovanie kvality biznis procesov. Z týchto produktov je určený na modelovanie procesov QPR ProcessDesigner³, ktorý bol použitý vo verzii 8.1 v rámci mapovania procesov na ČVUT FEL.

ProcessDesigner ponúka podporu pre modelovanie procesov v niekoľkých notáciách, medzi ktorými figuruje aj notácia BPMN. Podporovaná je tiež kolaborácia pri vytváraní modelov procesov a možnosť zdieľania informácií medzi zainteresovanými osobami.

ProcessDesigner však neobsahuje analýzu biznis procesov pomocou procesných mier, preto sú dôležité možnosti jeho exportu modelov procesov pre ich ďalšiu analýzu. Medzi podporované formáty exportu patria XML, XPDŁ a export modelu ako množinu webových stránok formátu HTML.

Najdôležitejším aspektom QPR ProcessDesigner pre moju prácu sú možnosti exportu tohto nástroja do formátu XPDŁ. Bohužiaľ sa v rámci skúmania vhodnosti tohto nástroja ukázalo, že jeho export do formátu XPDŁ je stratový.

V staršej verzii sa QPR ProcessDesigner nazýval QPR ProcessGuide 8.1 a bol použitý pri mapovaní procesov na ČVUT FEL. Najnovšia verzia modelovacieho nástroja QPR ProcessDesigner má názov 2014.1. V oboch týchto verziách exportovaný model vo formáte XPDŁ neobsahuje všetky údaje potrebné

³<http://www.qpr.com/products/qpr-processdesigner>

k znovuzostaveniu modelu. Medzi tieto chýbajúce údaje patria napríklad informácie o príslušnosti subprocessov do jednotlivých procesov, čím sa narúša celá štruktúra modelu biznis procesu.

Túto skutočnosť som v oboch verziách overil exportovaním modelu z QPR ProcessDesigner a následným importom modelu späť do tohto nástroja. Hoci bol nástroj QPR ProcessDesigner s výhodou použitý pri mapovaní procesov, pre potreby mojej práce je nevhodný.

Nekorektnosť XPDL exportu v prípade QPR ProcessDesigner ma donútila využiť iný modelovací nástroj.

5.2.3 Bizagi Modeler

Bizagi⁴ je softvérová firma založená v roku 1989, ktorá sa zameriava na vývoj produktov pre manažment biznis procesov. V súčasnosti Bizagi ponúka tri produkty venujúce sa tejto oblasti:

- Bizagi Modeler - je aplikácia na modelovanie a simuláciu procesov v notácií BPMN. Modely procesov vytvorené v Bizagi Modeler môžu byť publikované vo formátoch Word, PDF, Wiki alebo exportované vo formáte XPDL.
- Bizagi Studio - ponúka možnosť automatizovať vytvorené procesy a testovať ich v prednastavenom prostredí. Studio umožňuje vytvárať rozhrania, formuláre a pravidlá pre používanie procesov v rámci prípravy na ich realizáciu.
- Bizagi Engine - do posledného z trojice nástrojov vstupujú modelované a automatizované procesy a sú realizované v rámci organizácie.

Bizagi Modeler sa špecializuje na notáciu BPMN a poskytuje export modelu biznis procesov vo formáte XPDL, čím sa stáva veľmi žiadúcim pre našu problematiku procesných mier.

Export modelov podľa špecifikácií podporuje formát XPDL vo verzii 2.2 a je bezstratový, export a následný import do Bizagi Modeler vo verzii 2.9.0.4 vyprodukuje funkčný model biznis procesu. Tieto skutočnosti robia z nástroja Bizagi Modeler a jeho exportu vhodný vstup pre praktickú časť práce a preto som sa rozhodol použiť práve tento modelovací nástroj.

Nasledujúca, praktická časť práce priblíži existujúce riešenia implementácie procesných mier a ich zamerania. Nasleduje opis môjho vlastného nástroja implementujúceho vybrané procesné miery z kapitoly štyri. Nakoniec sa praktická časť práce zameria na nedostatky a možné vylepšenia formátu XPDL, ktoré vyplynuli v rámci vývoja mojej implementácie procesných mier.

⁴<http://www.bizagi.com/en>

Praktická časť práce

Praktická časť práce obsahuje popis nástrojov umožňujúcich analýzu modelu procesu pomocou procesných mier. K existujúcim nástrojom pridávam vlastnú implementáciu, používajúcu formát XPDL a realizujúcu vybranú množinu procesných mier. Praktickú časť práce uzatvárajú návrhy na vylepšenie formátu XPDL týkajúce sa jeho podpory pre procesné miery.

6.1 Aplikácia procesných mier

V rámci rešeršnej časti mojej práce o procesných mierach sa mi podarilo identifikovať tri nástroje zaoberajúce sa aplikovaním procesných mier na biznis procesy. Každý z týchto nástrojov sa zameriava na iné aspekty zložitosti biznis procesov a implementuje iné procesné miery.

V tejto sekcii popíšem vlastnosti každého z identifikovaných nástrojov. V nasledujúcej sekcii 6.2 ukážem prečo má zmysel vyvíjať vlastný nástroj, ktorý podľa môjho názoru vyplní medzery nájdených nástrojov.

6.1.1 BPMN Quality Tool

BPMN Quality Tool je vyvíjaný v programovacom jazyku Java a pozostáva zo štyroch modulov. Prvý modul prijíma BPM transformovaný do formátu XMI. Táto transformácia zaručuje, že procesný model je v štandardizovanom formáte a BPMN Quality Tool je použiteľný pre každý modelovací nástroj podporujúci export modelu do tohto formátu.[35]

Výstup z prvého modelu je posúvaný do ďalšieho modelu, ktorý z neho vytvorí strom obsahujúci všetky relevantné elementy. Tento strom elementov je použitý pre výpočet procesných mier na základe výberu užívateľa cez užívateľské rozhranie. BPMN Quality Tool implementuje miery CFC, CNC, NOA, Density a CP.[35]

6.1.2 CoCoFlow

CoCoFlow (COhesion-COupling metrics for workFLOW models) sa, ako naznačuje názov tohto nástroja, zaoberá implementáciou procesných mier z kategórií kohézie a spojitosti a poskytuje isté možnosti validácie modelu procesu. Nástroj prijíma model procesu vo formáte XML a dokáže vyjadriť hodnotu kohézie informácií, relácií a aktivít. Medzi implementované miery spojitosti patria Process Coupling a Coupling/Cohesion ratio.[49]

Naviac CoCoFlow ponúka jednoduché možnosti validácie dizajnu procesu, konkrétne sa jedná o kontrolu úplnosti informačných elementov procesu, operácií a zdrojov.[49]

6.1.3 ProM

ProM⁵ je nástroj na analýzu procesov, ktorý podporuje rôzne typy analýz vzťahujúce sa k biznis procesom. Veľká časť funkcionality ProM je zameraná na „process mining“, teda na analýzu procesov založenú na získavaní informácií o procese z logovacích nástrojov. Okrem toho však ProM ponúka prostriedky pre výpočet rôznych procesných mier.[29]

Funkcionalita nástroja ProM je rozdelená do zásuvných modulov. Každý zásuvný modul (plug-in) poskytuje časť funkčnosti a ProM podporuje tvorbu nových zásuvných modulov, čím sa jeho možnosti ďalej rozširujú. ProM ponúka širokú paletu zásuvných modulov rozdelených do nasledujúcich kategórií:

- Zásuvné moduly pre import - ponúkajú možnosti načítania dát z modelov procesov v rôznych formátoch. Medzi podporované formáty patria PNML, EPML (štandardný formát pre notáciu EPC), BPEL alebo YAWL.
- Zásuvné moduly pre process mining - ProM bol vyvinutý ako nástroj pre „process mining“ preto existuje niekoľko zásuvných modulov umožňujúcich túto analýzu a prevod analyzovaných procesov do formátu EPC alebo Petriho sietí.
- Zásuvné moduly pre analýzu procesu - tu patria moduly poskytujúce overenie korektnosti modelu procesu a verifikáciu jeho modelovacieho jazyka. Tieto moduly tiež umožňujú výpočet rôznych procesných mier.
- Zásuvné moduly pre konverziu - ponúkajú rôzne konverzie formátov procesných modelov. Napríklad z Petriho sietí do formátu BPEL alebo YAWL.
- Zásuvné moduly pre export - umožňujú export modelov vytvorených v ProM do iných nástrojov.[55]

⁵<http://www.promtools.org/>

Medzi procesné miery podporované v nástroji ProM patria miery NOA, CFC a Density. Existujú tiež doplnkové zásuvné moduly implementujúce miery z kategórií kohézie a spojitosti.

6.2 Porovnanie implementácií procesných mier

Popísané implementácie procesných mier sa od seba odlišujú v mnohých aspektoch. Jedným z týchto aspektov je formát vstupu. V prípade BPMN Quality Tool je vstupom XML súbor, CoCoFlow prijíma model vo formáte XMI a ProM podporuje pomocou modulov niekoľko vstupných formátov.

Rozdielna je aj sada implementovaných procesných mier. CoCoFlow sa zameriava na procesné miery z kategórie kohézie a spojitosti. BPMN Quality Tool implementuje jednu mieru spojitosti a po dvoch mierach z kategórií veľkosti a zložitosti. ProM implementuje jednu procesnú mieru z kategórie veľkosti a dve z kategórie zložitosti s možnosťou pridať ďalšie miery pomocou zásuvných modulov.

Žiadny z identifikovaných nástrojov nepodporuje ako svoj vstup model procesu vo formáte XPDL. Rovako sa žiadny z nástrojov nezameriava na procesné miery z kategórie modularity. Preto som vyvinul vlastnú knižnicu, ktorá tieto nedostatky ostatných riešení pokrýva.

Vstupom pre moju knižnicu nazvanú BPMN Measures je model procesu reprezentovaný vo formáte XPDL. Tento formát som sa rozhodol využiť na základe výstupov teoretickej časti práce.

BPMN Measures implementuje sadu desiatich procesných mier. Knižnica ponúka dve procesné miery z kategórie veľkosti procesu, štyri miery z kategórie modularity procesu, tri procesné miery zamerané na zložitosť a jednu mieru z oblasti súvislosti procesu. Navyše BPMN Measures ponúka validáciu vstupu oproti štandardom formátu XPDL a oproti vlastnej špecifikácii XPDL, ktorá je zameraná na podporu analýzy procesu a výpočtu procesných mier.

Moja knižnica ponúka iný vstupný formát ako dostupné nástroje a sústredí sa hlavne na implementáciu procesných mier z kategórií zložitosti a modularity, ktoré v ostatných nástrojoch nie sú primárne pokryté. Počet implementovaných procesných mier v rámci môjho nástroja je tiež výrazne vyšší ako v ostatných identifikovaných nástrojoch.

Preto podľa môjho názoru BPMN Measures dopĺňa ostatné podobné nástroje a zvyšuje možné použitie procesných mier v praxi.

V nasledujúcej sekcii priblížim knižnicu BPMN Measures, popíšem jej štruktúru a funkcionality.

6.3 BPMN Measures

Z predchádzajúcej teoretickej časti práce vyplynuli kritériá kladené na vyvíjaný program. Týmito kritériami sú podporovaná grafická notácia, formát

6. PRAKTICKÁ ČASŤ PRÁCE

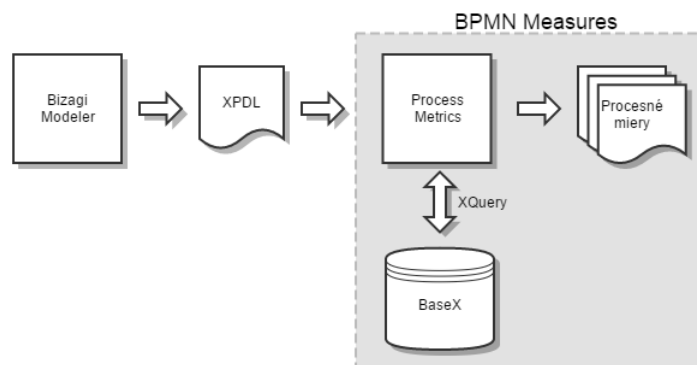
vstupu programu a množina procesných mier, ktoré budú programom implementované.

Program BPMN Measures je vyvíjaný v programovacom jazyku Java a pre svoj beh vyžaduje JRE vo verzii 7 alebo novej.

Vstupom programu je model biznis procesu vo formáte XPDL. V prípade testovaných biznis procesov sa jedná o export ich modelov vytvorených v nástroji Bizagi Modeler.

Procesný model je následne uložený do XML databázy BaseX⁶ vo verzii 8.1, ktorá spolupracuje s programom vo vstavanom režime. Databáza BaseX slúži na uloženie procesného modelu a taktiež ako procesor pre spracovanie XPath a XQuery príkazov.

Príkazy v jazyku XQuery sú používané na získavanie informácií o modeli biznis procesu z databázy BaseX. Na základe týchto informácií sú programom vypočítané hodnoty príslušných procesných mier. Schému programu zobrazuje obrázok 6.1.



Obr. 6.1: Schéma programu BPMN Measures

Funkčnosť programu je rozdelená do troch tried zameriavajúcich sa na výpočet procesných mier, validáciu vstupných súborov a integráciu do webových služieb.

Túto funkčnosť predstavujú nasledujúce triedy: *ProcessMetrics.java*, *XpdlValidator.java* a *WebConnector.java*.

⁶<http://basex.org/home/>

Trieda *ProcessMetrics.java* obsahuje nasledujúce verejné metódy, vypočítavajúce hodnoty procesných mier:

- `numberOfActivities()` - metóda vyjadruje hodnotu procesnej miery Number of Activities pomocou verejných metód `numberOfTasks()` a `numberOfSubprocesses()`.
- `numberOfActivitiesAndControlflowElements()` - metóda vyjadruje hodnotu procesnej miery Number of Activities and Control-flow elements, použitím vyššie spomenutých metód a metódy `numberOfGateways()`.
- `controlFlowComplexity()` - metóda vyjadruje hodnotu miery Control-flow Complexity.
- `interfaceComplexity()` - metóda počíta hodnotu procesnej miery Interface Complexity za pomoci metód `elementFanIn()` a `elementFanOut()`, ktoré určujú počty vstupov a výstupov elementov procesu. V tejto implementácii sa za vstupy a výstupy elementu počítajú len prechody procesu.
- `coefficientOfNetworkComplexity()` - metóda počíta hodnotu zložitosti biznis procesu podľa miery Coefficient of Network Complexity.
- `cyclomaticNumber()` - metóda počíta hodnotu rovnomennej procesnej miery Cyclomatic Number.
- `fanInFanOutMetric()` - metóda vyjadruje hodnotu procesnej miery z kategórie modularity - Fan-in/Fan-out Metric. Táto miera meria používanie subprocesov v rámci procesu za pomoci metód `processFanIn()` a `processFanOut()`.
- `cp()` - metóda počíta hodnotu zástupcu kategórie spojitosti - procesnú mieru CP.
- `maxProcessNestingDepth()` - metóda implementuje mieru Maximum Process Nesting, ktorá meria maximálne zanorenie subprocesov v rámci biznis procesu.
- `meanProcessNestingDepth()` - metóda implementuje mieru Mean Process Nesting, zameriavajúca sa na priemerné zanorenie subprocesov v rámci biznis procesu.

Všetky vymenované verejné metódy triedy vypočítavajú hodnoty daných procesných mier. Pre priblíženie implementácie v nasledujúcom úryvku popíšem metódu `controlFlowComplexity()`, ktorá vypočítava procesnú mieru CFC.

```
1 public int controlFlowComplexity() throws QueryException {
2     int elementFanOut = 0, cfc = 0;
3     String query = "declare default element namespace "
4         + "\"http://www.wfmc.org/2009/XPDL2.2\";\n"
5         + "let $wfp := /Package/WorkflowProcesses/WorkflowProcess\n"
6         + "for $act in $wfp/Activities/Activity union "
7         + "$wfp/ActivitySets/ActivitySet/Activities/Activity\n"
8         + "where $act/Route\n"
9         + "return (data($act/@Id), if (exists($act/Route/@GatewayType))"
10        + " then data($act/Route/@GatewayType) else \"Exclusive\")";
11    try (QueryProcessor proc = new QueryProcessor(query, context)) {
12        Iter it = proc.iter();
13        for (Item item; (item = it.next()) != null;) {
14            switch ((String) item.toJava()) {
15                case "Exclusive":
16                    cfc += elementFanOut; break;
17                case "Inclusive":
18                    cfc += (int) (Math.pow(2, elementFanOut) - 1); break;
19                case "Parallel":
20                    cfc += 1; break;
21                default:
22                    elementFanOut = elementFanOut((String) item.toJava());
23                    break;
24            }
25        }
26    }
27    return cfc;
28 }
```

V úryvku riadky 3 až 10 ukazujú XQuery výraz, ktorý z procesu uloženého v databáze BaseX vráti identifikátory a typy všetkých konštruktov vetvenia procesu. Riadok 11 inicializuje XQuery procesor a vykoná výraz uložený v reťazci *query*. Nasledujúci riadok získa iterátor, ktorý umožňuje prechádzať výsledok výrazu. V zvyšku úryvku prechádza cyklus nájdené konštrukty vetvenia a na základe ich typu a metódy *elementFanOut()*, ktorá zistí počty ich výstupov, vypočíta hodnotu procesnej miery CFC.

Trieda *XpdlValidator.java* poskytuje funkcionality týkajúcu sa overovania súborov formátu XPDL voči štandardom XPDL vo verziách 2.0 a 2.2 a vlastnej špecifikácie, ktorá upravuje štandard verzie 2.2. Trieda ponúka metódy:

- `validateAgainstVersion20()`
- `validateAgainstVersion22()`
- `validateAgainstCustomSchema()`

Tieto metódy implementujú validáciu vstupného súboru oproti XML schéme XPDL štandardu. Všetky metódy používajú k samotnej validácii privátnu metódu `validate()`, do ktorej vstupuje príslušná XML schéma a kontrolovaný

XPDL súbor. Posledná zmienaná metóda validuje XML súbor voči mnou upravenej špecifikácií XPDL vo verzii 2.2. Úpravy vylepšujú podmienky pre analýzu procesov a výpočet procesných mier a sú bližšie popísané v nasledujúcej sekcii práce.

Posledná trieda *WebConnector.java* obaluje funkčnosť knižnice a poskytuje verejnú metódu *getProcessMetrics()*, ktorá vracia výsledky všetkých desiatich implementovaných procesných mier pre daný biznis proces vo forme reťazca. Táto metóda je používaná pre integráciu knižnice do webovej služby.

6.4 Navrhované zlepšenia špecifikácie XPDL

Pri implementácii knižnice BPMN Measures som narazil na problémy týkajúce sa podpory výpočtu procesných mier v štandarde XPDL. Na niektoré z týchto problémov som narazil už pri štúdiu štandardu XPDL v rámci výberu formátu výmeny procesných modelov.

Preto navrhujem zlepšenie špecifikácie XPDL zameriavajúce sa na podporu analýzy biznis procesu a možnosti výpočtu procesných mier z atribútov a elementov špecifikácie.

Navrhované zlepšenia predstavuje vlastná XML schéma špecifikácie XPDL, ktorá upravuje niektoré vlastnosti pôvodnej verzie 2.2. Ako bolo spomínané v predchádzajúcej sekcii opisujúcej formát XPDL, mnohé jeho atribúty a elementy definuje špecifikácia ako nepovinné. Dôvodom je rozšíriteľnosť a najmä všestrannosť. V určitých situáciách nie je prítomnosť niektorých elementov a atribútov žiadúca, a preto sú radšej definované ako nepovinné.

V našom prípade je formát XPDL použitý ako formát exportu modelu procesu z modelovacieho nástroja podporujúceho grafickú notáciu BPMN. Tieto špecifiká a fakt, že sa export ďalej používa pre analýzu a výpočet procesných mier, sú obsiahnuté v prevedených zmenách. Moja špecifikácia XPDL obsahuje nasledujúce úpravy:

- Activity - element reprezentuje aktivitu procesu. Element má povinný atribút názov a povinne obsahuje elementy *NodeGraphicsInfo* predstavujúci grafickú podobu aktivity a *Performers* - množinu vykonávateľov aktivity. Povinnosť grafickej reprezentácie je daná tým, že sa jedná o výstup modelovacieho nástroja. Element *Performers* má využitie pri analýze rolí v procese.
- ActivitySet - element predstavuje jeden z dvoch typov subprocessov a oproti štandardu má povinný názov.
- Artifact - element predstavuje artefakty procesu, akými sú posielané správy alebo používané úložiska. Element má povinný atribút názov.

- *Coordinates* - element popisuje koordináty grafického umiestnenia objektu v procese. Keďže XPDL reprezentuje model procesu v grafickej notácii, obsahuje tento element povinné atribúty *XCoordinate* a *YCoordinate*.
- *Lane* - element zapuzdruje aktivity vykonávané určitým aktérom procesu a má povinný atribút názov a element *NodeGraphicsInfo*.
- *NodeGraphicsInfo* - element predstavuje grafickú reprezentáciu objektu. Povinne obsahuje element *Coordinates* a atribút *LaneId*, z ktorého sa dá ľahšie určiť prslušnosť objektu k aktérovi procesu.
- *Pool* - element obsahuje povinný atribút názov a element *NodeGraphicsInfo*.
- *Route* - element predstavuje riadiacu štruktúru procesu. Jeho atribút *GatewayType* popisuje typ konštruktu (AND, XOR, OR). Tento atribút je pôvodne definovaný nepovinne, obsahujú ho len konštrukty typu AND a OR. Tým sa sťažuje práca s modelom procesu, preto je v mojej špecifikácii tento atribút povinný.
- *Subflow* - element predstavuje subproces procesu, oproti štandardu má povinný názov.

6.5 Hodnotenie

Ukážku funkčnosti môjho riešenia som sa rozhodol previesť na procese „Zařazení studentů do oborů“, ktorý popisuje proces výberu a zaradenia študentov do rôznych odborov na ČVUT FEL. Tento proces prešiel vylepšením a preto existuje jeho pôvodná podoba a nová podoba po prevedených zmenách.

Pre potreby overenia funkčnosti tak vznikli dva modely v nástroji Bizagi Modeler zachytávajúce starú a novú podobu procesu. Oba tieto modely sú súčasťou obsahu príbaleného CD.

Grafickú podobu staršej verzie procesu ukazuje príloha C. Aktérmi tohto procesu sú *Študent*, *Štúdijné oddelenie* a *Dekanát*. Proces začína v rámci štúdijného oddelenia prejednaním kapacít jednotlivých odborov a zaslaním žiadosti o schválenie týchto kapacít Dekanátu.

Po schválení kapacít odborov štúdijné oddelenie sprístupní aplikáciu, v ktorej môžu študenti prejavíť záujem o odbor a informuje študentov o nutnosti výberu odboru. Študenti si pomocou aplikácie vyberú odbor. Následne štúdijné oddelenie spracuje požiadavky študentov a hromadne priradí študentom odbory a s nimi spojené študijné plány. Taktiež zverejní voľné kapacity jednotlivých odborov, čo umožní študentom zažiadať o zmenu priradeného odboru.

Ak študent požiadala o zmenu odboru, študijné oddelenie tejto žiadosti v rámci voľných kapacít vyhovie a priradí študentovi odbor.

Táto verzia procesu vznikala priebežne. Tvorili ju neskúsení analytici, ktorí v tom čase nepoznali zásady tvorby procesov. Tým dochádzalo k tvorbe veľkých, neintuitívnych a najmä neprehľadných procesných diagramov.

To nakoniec viedlo k stavu, v ktorom zamestnanci vnímali procesy ako niečo zložité a neúčinné. Prestali sa podľa nich riadiť a zoznamovať sa s nimi v procesnom portáli.

Preto sa muselo riešiť optimalizovanie procesov a zjednodušenie procesných diagramov, aby boli pre zamestnancov prehľadné a jednoducho pochopiteľné.

Výsledkom tohto snaženia je novšia verzia procesu „Zařazení studentů do oboru“, ktorú ukazuje príloha D. Aktérmi tohto procesu sú *Dekanát* a *Študent*.

Proces obsahuje dva subprocesy. Prvým z nich samotný proces začína a tento subproces opisuje stanovenie kapacít jednotlivých odborov.

Ďalší subproces sa zaoberá zistením záujmu študentov o jednotlivé odbory. Podobne ako v predchádzajúcej verzii tento subproces zahŕňa sprístupnenie aplikácie, informovanie študentov o nutnosti výberu odboru a následné hromadné priradenie odborov.

Po týchto subprocesoch sa hlavný proces zaoberá zverejnením voľných kapacít jednotlivých odborov a žiadosťami študentov o preradenie do iného odboru.

Cieľom novej verzie procesu „Zařazení studentů do oboru“ je optimalizovanie procesu a zjednodušenie jeho pochopenia pre užívateľa. Tieto vylepšenia procesu sa pokúsim vyhodnotiť pomocou môjho nástroja.

Výsledky nástroja BPMN Measures ukazujú hodnoty procesných mier pre obe verzie modelu procesu. Tieto výsledky zobrazuje tabuľka 6.5.

Procesná miera	Starý model procesu	Nový model procesu
Number of Activities	12	15
Number of Activities and Control-flow elements	14	17
Cyclomatic Number	14	14
Coefficient of Network Complexity	2.0833	1.8666
Control-flow Complexity	4	4
Fan-in/Fan-out metric	0	0
Interface Complexity	29	30
CP	0.0303	0.0286
Maximum nesting depth	0	1
Mean nesting depth	0	0.2

Tabuľka 6.1: Hodnoty procesných mier pre proces „Zařazení studentů do oboru“

Z výsledkov je zrejmé zvýšenie hodnôt procesných mier *NOA* a *NOAC* čo ukazuje, že nová verzia procesu obsahuje viac aktivít, no rovnaký počet riadiacich elementov ako staršia verzia.

Fakt, že obe verzie procesu obsahujú rovnaký počet riadiacich elementov dokladá aj hodnota miery *CFC*, ktorá uvádza štyri rôzne prechody procesom pre obe jeho verzie.

Zvýšenie hodnôt mier *Maximum nesting depth* a *Mean nesting depth* ukazuje na prítomnosť subprocesov v novej verzii procesu a tým na zvýšenie jeho modularity.

Hodnota miery *Interface Complexity*, ktorá sa aj napriek pridaniu troch aktivít v novej verzii procesu zvýšila len o jedna, ukazuje na linearizáciu procesu. Túto domnienku ďalej potvrdzuje zníženie hodnoty miery *CNC*.

Hodnoty procesných mier naznačujú, že nová verzia procesu je síce obsiahlejšia, ale modulárnejšia vďaka prítomnosti subprocesov a lineárnejšia ako stará verzia.

Ako som spomenul vyššie, cieľom tvorby novej verzie procesu bolo zníženie jeho zložitosti a zjednodušenie jeho pochopenia pre užívateľov.

Výsledky procesných mier s týmto cieľom korelujú. Nová verzia procesu je modulárnejšia, teda obsahuje subprocesy, ktoré delia proces na logické časti, zvyšujú jeho prehľadnosť a znovupoužitie. Taktiež sa zvýšila linearita procesu, nová verzia procesu je tak jednoduchšia a čitateľnejšia pre užívateľa.

Záver

Táto práca vznikla na základe potrieb skupiny CZM, ktorá sa zaoberá tvorbou a podporou procesného portálu. Ako podklady pre spracovanie boli použité procesné mapy z tohto portálu a výstupy projektu EFIN[56], na ktorom skupina CZM spolupracovala.

Cieľom mojej diplomovej práce bolo zistiť, či je možné kvantifikovať vlastnosti procesných modelov, akými sú ich kvalita a zložitosť. V prípade nájdenia takýchto metód alebo mier bolo cieľom ich klasifikovať a zistiť, či sú použiteľné v praxi a existujú ich implementácie. Nakoniec som mal implementovať vlastný nástroj umožňujúci analýzu biznis procesov.

Vytýčené ciele sa mi podarilo splniť. Teoretickú časť práce začínam definíciou „Business process modeling“ ako aktu vytvárania modelov biznis procesov. Ďalej sa zameriavam na dostupné grafické notácie pre modelovanie procesov. Popisujem nároky kladené na biznis procesy, ktoré vedú k zvyšovaniu ich zložitosti a definujem pojem zložitosť procesu.

Veľká sekcia teoretickej časti práce sa zaoberá rešerom literatúry, ktorého výsledkom je identifikovanie 22 procesných mier, vyjadrujúcich zložitosť rôznych aspektov biznis procesov. Nájdené miery kategorizujem do piatich kategórií zložitosti adaptovaných z oblasti softvérovej zložitosti a bližšie popisujem každú z mier. Nakoniec z nájdených mier vyberám podľa určených kritérií podmnožinu, ktorá je implementovaná v praktickej časti práce.

Táto sekcia práce bola taktiež podkladom pre dva vedecké články zaslané na konferencie Conference on Business Informatics a Acta Informatica Pragensia.

Ďalším krokom k implementácii nástroja pre analýzu zložitosti procesov je výber formátu procesného modelu. V práci popisujem najpoužívanejšie formáty a porovnávam ich pomocou formalizmu Workflow Patterns. Z tohto porovnania vychádza víťazne formát XPDŁ, ktorý používam ako vstup pre vyvíjaný nástroj.

Praktická časť práce obsahuje popis nástrojov umožňujúcich analýzu modelu procesu pomocou procesných mier. K existujúcim nástrojom pridávam

vlastnú implementáciu s názvom BPMN Measures, používajúcu formát XPDL a realizujúcu množinu desiatich procesných mier.

Táto knižnica ponúka okrem implementovaných procesných mier aj možnosť validácie vstupného modelu voči štandardom XPDL vrátane mnou upravenej verzie štandardu zameriavajúcej sa na podporu výpočtu procesných mier.

Možnosti budúceho pokračovania práce vidím v rozšírení podpory mojej knižnice o nové procesné miery a jej možnej integrácie do grafického nástroja vyvíjaného v rámci FEL ČVUT.

Literatúra

- [1] Cardoso, J.: Process control-flow complexity metric: An empirical validation. *Proceedings - 2006 IEEE International Conference on Services Computing, SCC 2006*, 2006: s. 167–173, ISSN 15457362, doi:10.1109/SCC.2006.82.
- [2] Gruhn, V.; Laue, R.: Adopting the Cognitive Complexity Measure for Business Process Models. *2006 5th IEEE International Conference on Cognitive Informatics*, ročník 1, 2006: s. 236–241, doi:10.1109/COGINF.2006.365702. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4216417>
- [3] Gruhn, V.; Laue, R.: Complexity metrics for business process models. *9Th International Conference on Business ...*, 2006: s. 1–12, ISSN 16175468, doi:10.1109/COGINF.2006.365702. Dostupné z: http://www.researchgate.net/publication/221281564_Complexity_Metrics_for_business_Process_Models/file/d912f5085635ace7b0.pdf
- [4] Latva-Koivisto, A. M.: Finding a complexity measure for business process models. 2001, doi:10.1.1.25.2991. Dostupné z: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.2991&rep=rep1&type=pdf>
- [5] Van Der Aalst, W. M. P.: the Application of Petri Nets To Workflow Management. *Journal of Circuits, Systems and Computers*, ročník 08, 1998: s. 21–66, ISSN 0218-1266, doi:10.1142/S0218126698000043.
- [6] Azim, A.; Ghani, A.; Tieng, K.; aj.: Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric (GQM). *Journal of Computer Science*, ročník 8, č. 5, 2008: s. 219–225.

- [7] Andrews, T.: Business Process Execution Language for Web Services. *Business*, ročník 1, č. May, 2003: s. 1–136. Dostupné z: <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>
- [8] Fenton, N. E.: *Software Metrics: A Rigorous Approach*. London, UK, UK: Chapman & Hall, Ltd., 1991, ISBN 0442313551.
- [9] Card, D. N.; Aggresti, W. W.: Measuring Software Design Complexity. *J. Syst. Softw.*, ročník 8, č. 3, Červen 1988: s. 185–197, ISSN 0164-1212, doi:10.1016/0164-1212(88)90021-0. Dostupné z: [http://dx.doi.org/10.1016/0164-1212\(88\)90021-0](http://dx.doi.org/10.1016/0164-1212(88)90021-0)
- [10] Ieee: IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990). Los Alamitos. CA: *IEEE Computer Society*, ročník 121990, 1990. Dostupné z: [http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:IEEE+Standard+Glossary+of+Software+Engineering+Terminology+\(IEEE+Std+610.12-1990\)#0](http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:IEEE+Standard+Glossary+of+Software+Engineering+Terminology+(IEEE+Std+610.12-1990)#0)
- [11] Edmonds, B.: Complexity and Scientific Modelling. *Foundations of Science*, 2000: s. 379–390, ISSN 1233-1821, doi:10.1023/A:1011383422394. Dostupné z: <http://cogprints.org/1773/>
- [12] Scheer, A. W.; Oliver, T.; Otmar, A.: Frontmatter. In *Process-Aware Information Systems*, kapitola 6, John Wiley & Sons, Inc., 2005, ISBN 9780471741442, s. i—xvi, doi:10.1002/0471741442.fmatter. Dostupné z: http://books.google.com/books?hl=en&lr=&id=ZENNdQq8p74C&oi=fnd&pg=PA119&dq=Process+Modeling+Using+Event-+Driven+Process+Chains&ots=ZfXNZRJ_8I&sig=G1ajjHw05Dh1_I1wvJQtp60iFoo#v=onepage&q=EEPC&f=falsehttp://dx.doi.org/10.1002/0471741442.fmatter
- [13] Van Der Aalst, W. M. P.: Formalization and verification of event-driven process chains. *Information and Software Technology*, ročník 41, 1999: s. 639–650, ISSN 09505849, doi:10.1016/S0950-5849(99)00016-6.
- [14] Russell, N.; Van Der Aalst, W. M. P.; Ter Hofstede, A. H. M.; aj.: On the Suitability of UML 2 . 0 Activity Diagrams for Business Process Modelling. *Reproduction*, ročník 53, č. January, 2006: s. 95–104, ISSN 14451336. Dostupné z: <http://portal.acm.org/citation.cfm?id=1151866>
- [15] Omg, T.; Adopted, D.; Final, O. M. G.; aj.: UML 2.0 Superstructure Specification. *October*, ročník 02, 2004: s. 1–786.
- [16] Dijkman, R. M.; Dumas, M.; Ouyang, C.: Semantics and analysis of business process models in BPMN. *Information and Software Technology*, ročník 50, 2008: s. 1281–1294, ISSN 09505849, doi:10.1016/j.infsof.2008.02.006.

-
- [17] Wohed, P.; van der Aalst, W. M. P.; Dumas, M.; aj.: On the suitability of BPMN for business process modelling. *Proceedings of the 4th International Conference, BPM 2006*, 2006: s. 161–176, ISSN 03029743, doi: 10.1007/11841760_12.
- [18] Reijers, H.; Vanderfeesten, I.: Cohesion and coupling metrics for workflow process design. 2004. Dostupné z: http://link.springer.com/chapter/10.1007/978-3-540-25970-1_19
- [19] Omg, O. M. G.; Parida, R.; Mahapatra, S.: Business Process Model and Notation (BPMN) Version 2.0. *Business*, ročník 50, č. January, 2011: str. 170, ISSN 13507540, doi:10.1007/s11576-008-0096-z. Dostupné z: <http://books.google.com/books?id=GjmLqXNYFS4C&pgis=1>
- [20] Khelif, W.; Zaaboub, N.; Ben-Abdallah, H.: Coupling metrics for business process modeling. *WSEAS Transactions on Computers*, ročník 9, 2010: s. 31–41, ISSN 11092750.
- [21] Cardoso, J.; Cardoso, J.: Business process control-flow complexity: Metric, evaluation, and validation. *International Journal of Web Services Research*, ročník 5, č. June, 2008: s. 49–76. Dostupné z: <https://jorge-cardoso.github.io/publications/Papers/JA-2008-017-JWSR-Business-Process-Control-Flow-Complexity-proof.pdf><http://www.scopus.com/libezproxy.open.ac.uk/inward/record.url?eid=2-s2.0-55949111041&partnerID=40>
- [22] Mendling, J.; Moser, M.; Neumann, G.; aj.: Faulty EPCs in the SAP Reference Model. *Business Process Management : 4th International Conference, BPM 2006, Vienna, Austria, September 5-7, 2006. Proceedings*, ročník 4102, 2006: s. 451–457, ISSN 0302-9743, doi:10.1007/11841760_38. Dostupné z: http://dx.doi.org/10.1007/11841760_38
- [23] Mendling, J.: Testing Density as a Complexity Metric for EPCs. *Analysis*, 2006.
- [24] Zuse, H.: *A Framework of Software Measurement*. Hawthorne, NJ, USA: Walter de Gruyter & Co., 1997, ISBN 3110155877.
- [25] Weyuker, E.: Evaluating software complexity measures. *IEEE Transactions on Software Engineering*, ročník 14, č. 9, 1988, ISSN 0098-5589, doi:10.1109/32.6178.
- [26] Muketha, G. M.; Ghani, a. a. a.; Selamat, M. H.; aj.: A survey of business process complexity metrics. 2010, doi: 10.3923/itj.2010.1336.1344. Dostupné z: <http://scialert.net/qredirect.php?doi=itj.2010.1336.1344&linkid=pdf>

- [27] Fu, X.; Zou, P.; Ma, Y.; aj.: A control-flow complexity measure of web service composition process. *Proceedings - 2010 IEEE Asia-Pacific Services Computing Conference, APSCC 2010*, 2010: s. 712–716, doi: 10.1109/APSCC.2010.27.
- [28] Conte, S. D.; Dunsmore, H. E.; Shen, V. Y.: *Software Engineering Metrics and Models*. Redwood City, CA, USA: Benjamin-Cummings Publishing Co., Inc., 1986, ISBN 0-8053-2162-4.
- [29] Vanderfeesten, I.; Cardoso, J.; V, I.; aj.: Quality Metrics for Business Process Models. *BPM and Workflow ...*, 2007: s. 1–12. Dostupné z: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.74.6133%5Cdelimiter%26E30F%5Cnhttp://eden.dei.uc.pt/~jcardoso/Research/Papers/BC-2007-013-WorkflowHandbook-Quality-Metrics-BPM-Vanderfeesten.pdf>
- [30] Kitchenham, B.; Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering. *Engineering*, ročník 2, 2007: str. 1051, ISSN 00010782, doi:10.1145/1134285.1134500.
- [31] Azuma, M.; Mole, D.: Software Management Practice and Metrics in the European Community and Japan: Some Results of a Survey. *J. Syst. Softw.*, ročník 26, č. 1, Červenec 1994: s. 5–18, ISSN 0164-1212, doi:10.1016/0164-1212(94)90091-4. Dostupné z: [http://dx.doi.org/10.1016/0164-1212\(94\)90091-4](http://dx.doi.org/10.1016/0164-1212(94)90091-4)
- [32] Khelif, W.; Makni, L.: Quality metrics for business process modeling. ... *Conference on APPLIED ...*, ročník 9, č. 1, 2009: s. 195–200. Dostupné z: <http://www.wseas.us/e-library/conferences/2009/genova/ACS/ACS-32.pdf>
- [33] Henry, S.; Kafura, D.: Software Structure Metrics Based on Information Flow. *IEEE Transactions on Software Engineering*, ročník SE-7, č. 5, 1981: s. 510–518, ISSN 0098-5589, doi: 10.1109/TSE.1981.231113. Dostupné z: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1702877%5Cdelimiter%26E30F%5Cnhttp://dl.acm.org/citation.cfm?id=1313341.1313821
- [34] Thammarak, K.: Survey Complexity Metrics for Reusable Business Process. *1st National Conference on Applied Computer Technology and Information System (ACTIS)*, 2010: s. 18–22. Dostupné z: <http://www.bsc.ac.th/v4/ACTIS2010-05.pdf>
- [35] Makni, L.; Khelif, W.; Haddar, N. Z.; aj.: A Tool for Evaluating the Quality of Business Process Models Overview on current metrics for BPM: s. 230–242.

-
- [36] Kluza, K.; Nalepa, G. J.: Proposal of Square Metrics for Measuring Business Process Model Complexity. *Federated Conference on Computer Science and Information Systems 2012*, 2012: s. 919–922.
- [37] Gruhn, V.; Laue, R.: Approaches for Business Process Model Complexity Metrics. *Technologies for Business Information Systems*, 2007: s. 13–24, doi:10.1007/1-4020-5634-6. Dostupné z: <http://www.springerlink.com/content/wt7344r681731685/>
- [38] Cardoso, J.; Mendling, J.; Neumann, G.; aj.: A Discourse on Complexity of Process Models (Survey Paper). *Lecture Notes in Computer Science*, 2006: s. 117–128. Dostupné z: <http://www.mendling.com/publications/06-BPI.pdf>
- [39] Shao, J.; Wang, Y.: A new measure of software complexity based on cognitive weights Une nouvelle métrique de complexité logique basée sur les poids cognitifs. *October*, ročník 28, č. 2, 2003: s. 69–74, ISSN 08408688, doi:10.1109/CJECE.2003.1532511. Dostupné z: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.7800&rep=rep1&type=pdf>
- [40] Misra, S.; Misra, a. K.: Evaluating cognitive complexity measure with weyuker properties. *Proceedings of the Third IEEE International Conference on Cognitive Informatics, ICCI 2004*, , č. 3, 2004: s. 103–108, doi:10.1109/COGINF.2004.1327464.
- [41] McCabe, T. J.; McCabe, T. J.: A complexity measure. *IEEE Transactions on Software Engineering*, ročník SE-2, č. 4, 1976: s. 308–320, ISSN 0098-5589, doi:10.1109/TSE.1976.233837.
- [42] Cardoso, J.: How to Measure the Control-flow Complexity of Web Process and Workflows. In *Workflow Handbook 2005*, 2005, ISBN 0970350988, s. 199–212.
- [43] Rolón, E.; Cardoso, J.; García, F.; aj.: Analysis and validation of control-flow complexity measures with BPMN process models. *Lecture Notes in Business Information Processing*, ročník 29 LNBIP, 2009: s. 58–70, ISSN 18651348, doi:10.1007/978-3-642-01862-6_6.
- [44] Halstead, M. H. M. H.: *Elements of software science*. Operating and programming systems series, New York: Elsevier, 1977, ISBN 0-444-00205-7, elsevier computer science library. Dostupné z: <http://opac.inria.fr/record=b1084731>
- [45] Solichah, I.; Hamilton, M.; Mursanto, P.; aj.: Exploration on software complexity metrics for business process model and notation. *2013 International Conference on Advanced Computer Science*

- and Information Systems (ICAC SIS)*, 2013: s. 31–37, doi:10.1109/ICAC SIS.2013.6761549. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6761549>
- [46] Thesen, A.: Measures of the restrictiveness of project networks. *Networks*, ročník 7, č. 3, 1977: s. 193–208, ISSN 1097-0037, doi:10.1002/net.3230070302. Dostupné z: <http://dx.doi.org/10.1002/net.3230070302>
- [47] Vanderfeesten, I.; Cardoso, J.; Reijers, H. a.: A weighted coupling metric for business process models. *CEUR Workshop Proceedings*, ročník 247, 2007: s. 41–44, ISSN 16130073.
- [48] Selby, R.; Basili, V.: Analyzing Error-Prone System Structure. *IEEE Transactions on Software Engineering*, ročník 17, č. 2, 1991: s. 141–152.
- [49] Vanderfeesten, I.; Reijers, H. a.; van der Aalst, W. M. P.: Evaluating workflow process designs using cohesion and coupling metrics. *Computers in Industry*, ročník 59, 2008: s. 420–437, ISSN 01663615, doi:10.1016/j.compind.2007.12.007.
- [50] Debnath, N.; Salgado, C.; Peralta, M.; aj.: Optimization of the business process metrics definition according to the BPD standard and its formal definition in OCL. *2010 ACS/IEEE International Conference on Computer Systems and Applications, AICCSA 2010*, 2010, doi:10.1109/AICCSA.2010.5586986.
- [51] Mendling, J.; Neumann, G.; Nüttgens, M.: A Comparison of XML Interchange Formats for Business Process Modelling. *Proceedings of EMISA 2004 - Information Systems in E-Business and E-Government*, 2004: s. 129–140, doi:10.1.1.1.7503.
- [52] Van der Aalst, W. M. P.; Ter Hofstede, a. H. M.; Kiepuszewski, B.; aj.: Workflow patterns. *Distributed and Parallel Databases*, ročník 14, 2003: s. 5–51, ISSN 09268782, doi:10.1023/A:1022883727209.
- [53] Van Der Aalst, W. M. P.: Patterns and XPD: A Critical Evaluation of the XML Process Definition Language. *Language*, 2003: s. 1–30. Dostupné z: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.5.739&rep=rep1&type=pdf>
- [54] Coalition, W. M.: Workflow Standard Process Definition Interface—XML Process Definition Language. *Xpdl.Org*, 2012. Dostupné z: <http://www.xpdl.org/nugen/p/gseonklyf/a/2009-4-7xpdl2.doc>

- [55] Lassen, K. B.; van der Aalst, W. M. P.: Complexity metrics for Workflow nets. *Information and Software Technology*, ročník 51, č. 3, 2009: s. 610–626, ISSN 09505849, doi:10.1016/j.infsof.2008.08.005. Dostupné z: <http://dx.doi.org/10.1016/j.infsof.2008.08.005>
- [56] *Metodika EFIN*. Praha: Ministerstvo školství, mládeže a tělovýchovy, první vydání, 2012, ISBN 978-80-87601-09-9.

Zoznam použitých skratiek

- BPDM** Business Processes Definition Metamodel
- BPEL** Business Process Execution Language
- BPM** Business Process Model
- BPML** Business Process Modeling Language
- BPMN** Business Process Model and Notation
- EPC** Event-driven Process Chains
- EPML** EPC Markup Language
- HTML** HyperText Markup Language
- IEEE** Institute of Electrical and Electronics Engineers
- JRE** Java Runtime Environment
- LOC** Lines of Code
- OO** Object Oriented
- PNML** Petri Net Markup Language
- SOAP** Simple Object Access Protocol
- UML** Unified Modeling Language
- URI** Uniform Resource Identifier
- WSDL** Web Services Description Language
- WSFL** Web Services Flow Language
- XMI** XML Metadata Interchange

A. ZOZNAM POUŽITÝCH SKRATIEK

XPath XML Path Language

XPDL XML Process Definition Language

XSD XML Schema

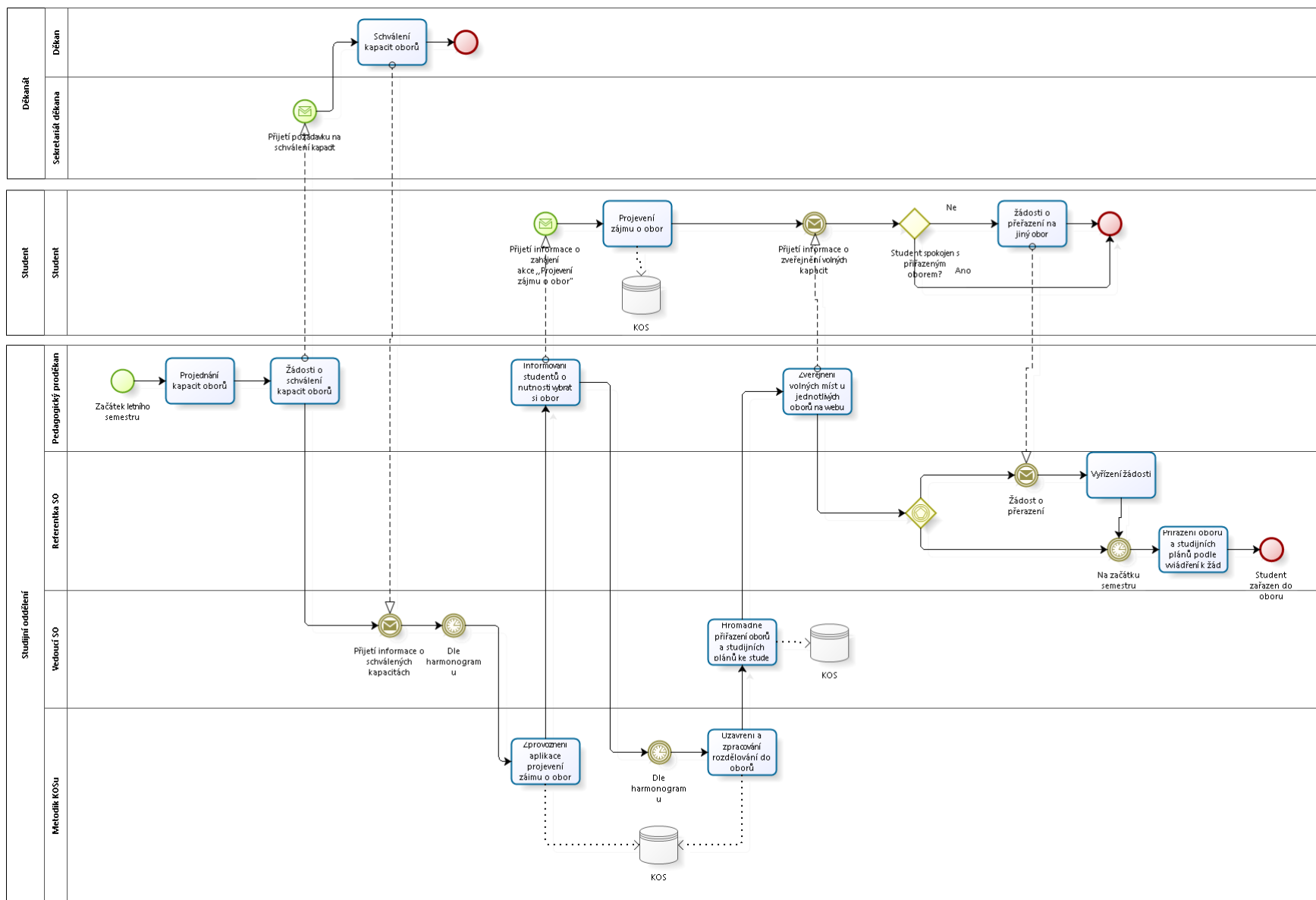
YAWL Yet Another Workflow Language

Výsledky rešeršu

Publication title	Number of activities (NOA, NOT)	Control-Flow Complexity (CFC)	Max/Mean nesting depth	Number of handles	Cognitive weight (Cognitive Complexity)	BPM (Anti)Patterns	Fan-in/Fan-out (Modularization)
Complexity Metrics for Business Process Models	1	1	1	1	1	1	1
Quality Metrics for Business Process Models	1	1					
Finding a complexity measure for business process models							
Cohesion and Coupling Metrics for Workflow Process Design							
A Cohesion Metric for the Definition of Activities in a Workflow Process							
A Discourse on Complexity of Process Models	1	1					
Control-flow Complexity Measurement of Processes and Weyuker's Properties		1					
How to Measure the Control-flowComplexity of Web Processes and Workflows		1					
Testing Density as a Complexity Metric for EPCs							
Towards Thresholds of Control Flow Complexity Measures for BPMN models		1					
An Empirical Study of Error Patterns in Industrial Business Process Models !		1					
Software Structure Metrics Based on Information Flow							
Evaluating workflow process designs using cohesion and coupling metrics							
Process control-flow complexity metric: An empirical validation		1					
On a Quest for Good Process Models: The Cross-Connectivity Metric		1					
A weighted coupling metric for business process models							
Adopting the Cognitive Complexity Measure for Business Process Models					1		
An Ensemble of Complexity Metrics for BPEL Web Processes		1					
Complexity Metrics for Workflow Nets		1					
A new measure of software complexity based on cognitive weights					1		
Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric	1	1	1		1	1	1
A Survey of Business Process Complexity Metrics	1	1					
Survey Complexity Metrics for Reusable Business Process	1	1			1		1
Business Process Control-Flow Complexity: Metric, Evaluation, and Validation		1					
New Quality Metrics for Evaluating Process Models							
Proposal of Square Metrics for Measuring Business Process Model Complexity	1	1	1		1		
Coupling metrics for business process modeling							
Quality metrics for business process modeling	1	1					
Exploration on software complexity metrics for business process model and notation		1					
Analysis and Validation of Control-Flow Complexity Measures with BPMN Process Models		1					
A Control-Flow Complexity Measure of Web Service Composition Process		1					
On the Correlation between Process Model Metrics and Errors	1	1					
A Tool for Evaluating the Quality of Business Process Models Overview on current metrics for BPM	1	1					1
	10	22	3	1	6	2	4

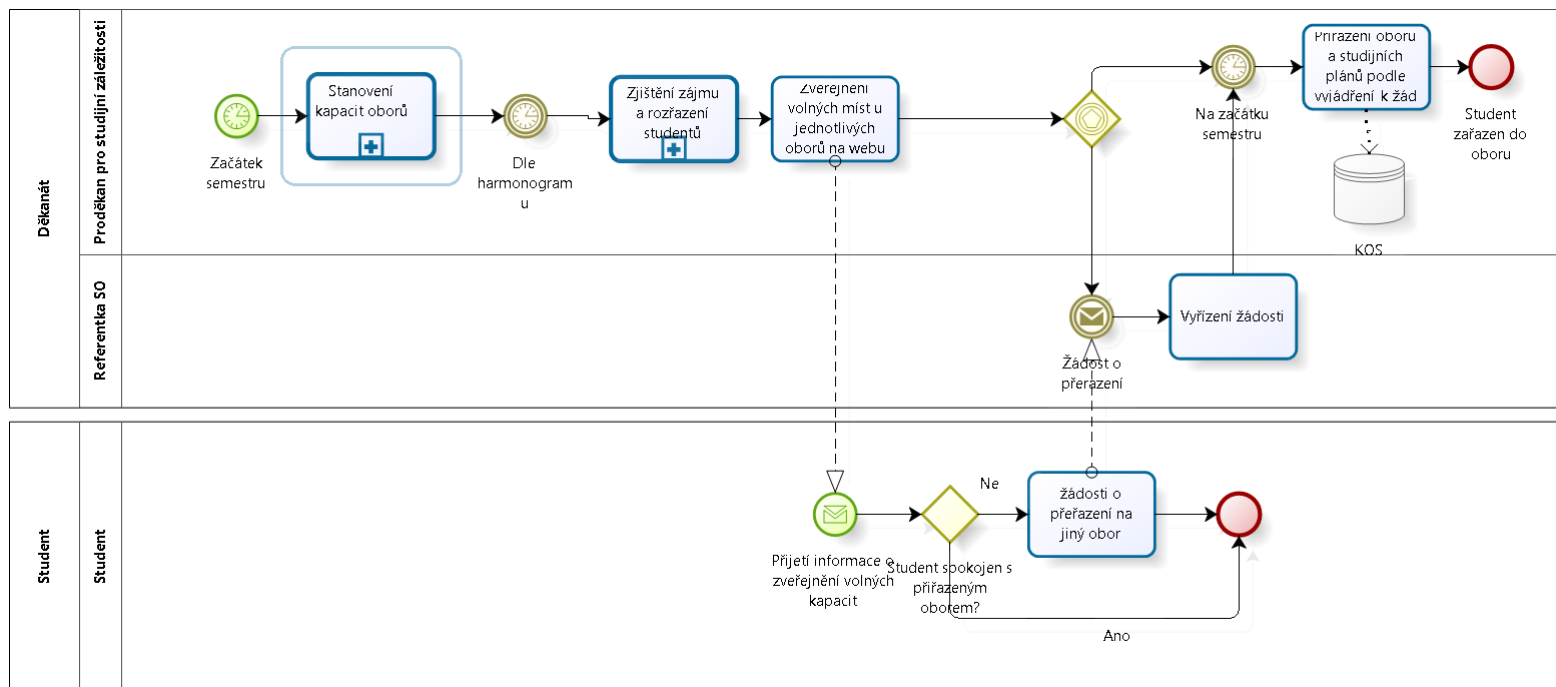
Coefficient of network complexity(CN C)	Cyclomatic number	Complexity index(CI)	Restrictiveness estimator (RT)	Number of trees in graph	Process Cohesion (TPC, LPC)	Process Coupling(CBO, RFC, MPC, ICP)	Process coupling / cohesion ratio	Halstead-based Process Complexity (HPC)	Interface Complexity (IC)	Density	Cross-Connectivity (CC)	CP	GQM	Q0, Q1, Q2
1		1												
1	1	1	1	1										
					1	1	1							
					1									
1		1	1					1	1					
											1			
1	1	1	1	1										
					1	1	1							
												1		
													1	
	1													
														1
1								1	1					
								1	1					
1														
1														
8	3	4	3	2	4	4	2	7	7	1	2	1	1	1

**Proces „Zařazení studentů do
oboru“ - staršia verzia**

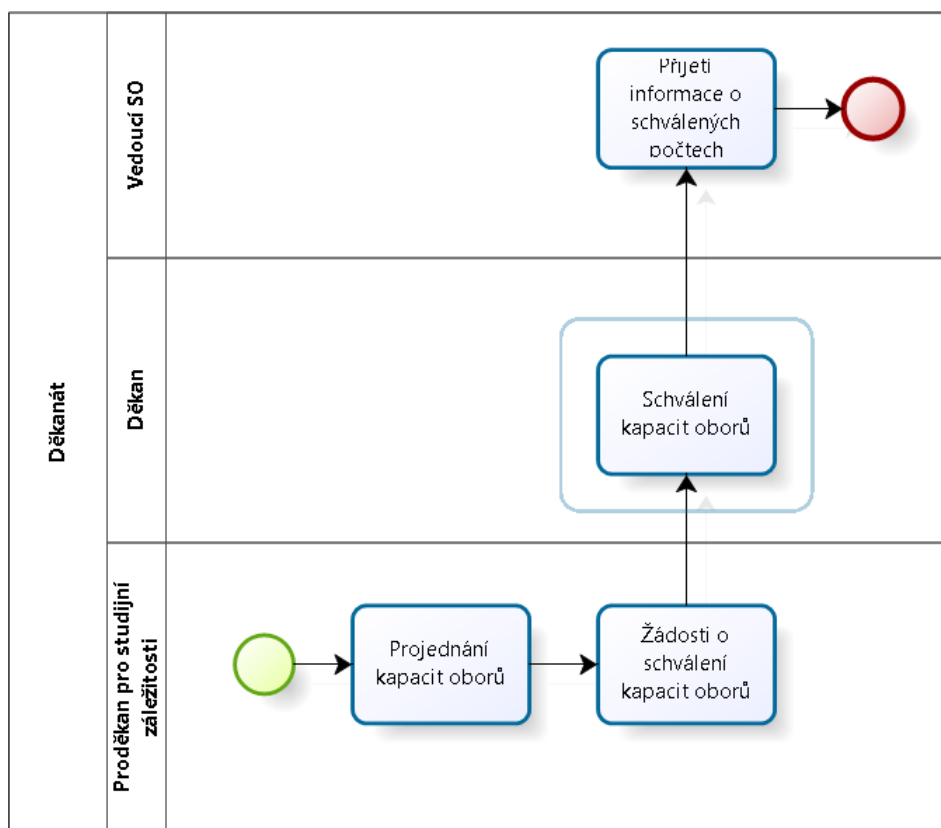


Obr. C.1: „Zařazení studentů do oborů“

**Proces „Zařazení studentů do
oboru“ - novšia verzia**

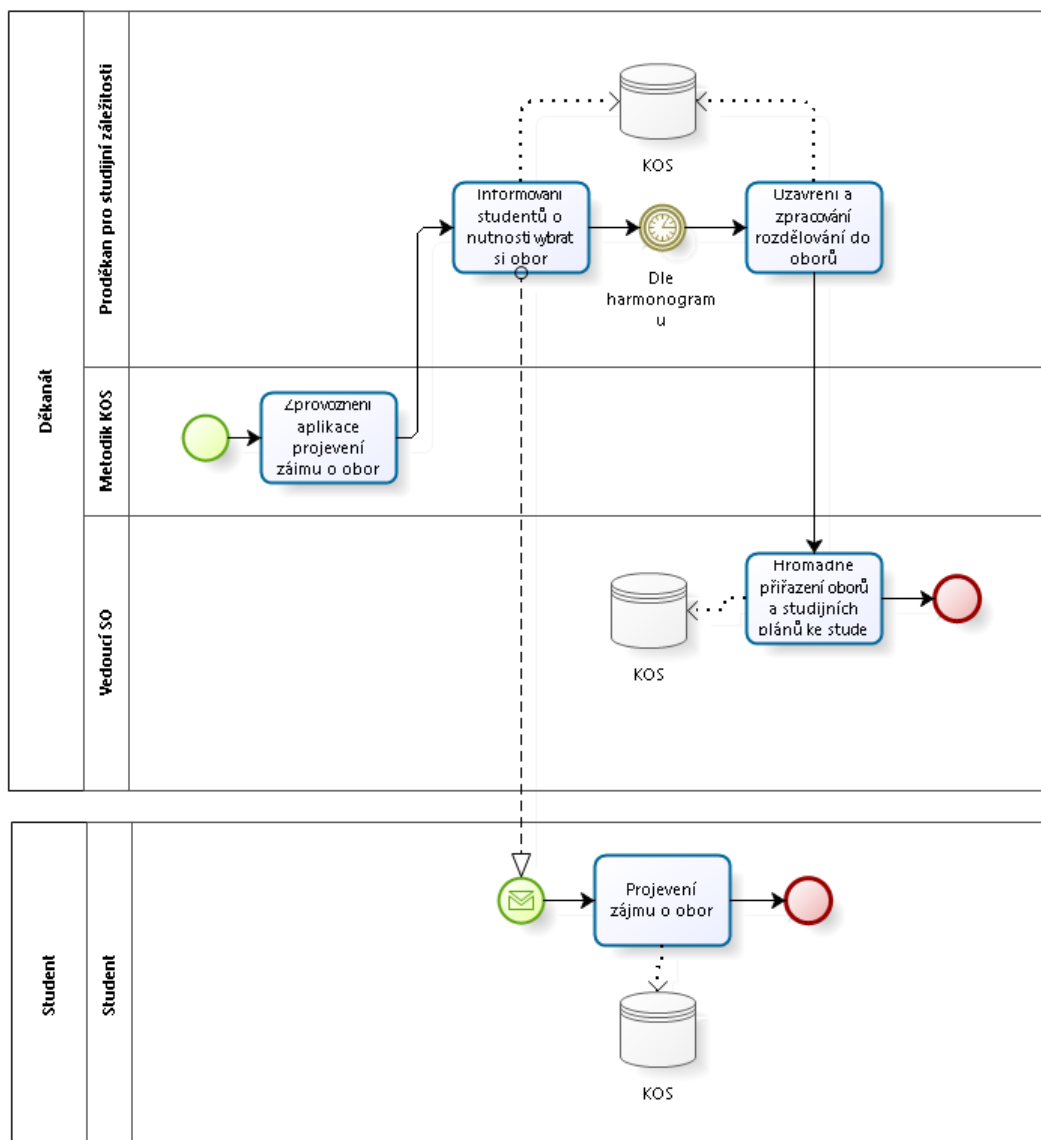


Obr. D.1: „Zařazení studentů do oboru“



Obr. D.2: „Stanovení kapacit oborů“

D. PROCES „ZAŘAZENÍ STUDENTŮ DO OBORŮ“ - NOVŠIA VERZIA



Obr. D.3: „Zjištění zájmu a rozřazení studentů“

Obsah priloženého CD

readme.txt.....	stručný popis obsahu CD
impl	zdrojové kódy implementácie
├─ BpmnMeasures	implementácia knižnice Bpmn Measures
├─ models.....	modely procesu
thesis.....	zdrojová forma práce vo formáte L ^A T _E X
├─ img.....	obrázky používané v práci
text	text práce
├─ DP_Mach_Richard_2015.pdf	text práce vo formáte pdf