



EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání



Název projektu	Rozvoj vzdělávání na Slezské univerzitě v Opavě
Registrační číslo projektu	CZ.02.2.69/0.0./0.0/16_015/0002400

Techniky a nástroje v oblasti řízení podnikových procesů

Distanční studijní text

Roman Šperka, Michal Halaška, Dalibor Šimek

Karviná 2018



**SLEZSKÁ
UNIVERZITA**
OBCHODNĚ PODNIKATELSKÁ
FAKULTA V KARVINĚ

- Obor:** Podniková informatika, aplikovaná informatika, podniková ekonomika, management.
- Klíčová slova:** Podnikový proces, management, BPM, řízení, process mining, zlepšování, simulace, Bizagi, Disco, ProM.
- Anotace:** Hlavním cílem výukového textu je představit využití základních technik, metod a nástrojů z oblasti řízení procesů se zaměřením na procesy probíhající v podnicích (nebo alternativně ve firmách, společnostech a organizacích). Tento text je vhodný jako distanční opora ve výuce předmětů na vysokých školách s důrazem na řízení podnikových procesů (BPM – Business Process Management), zejména v oblasti průběhu podnikových procesů, jejich simulací a analýzy reálných podnikových dat. Technickou náročností je určena zejména aplikovaným informatickým oborům studia. Předmětem zájmu výukového textu je analýza podnikových procesů, která je základem pro pochopení konkrétní podnikové domény s cílem nalézt slabé místa v procesech a na-vrhnout jejich zlepšení. V prvních kapitolách autoři představují teoretické východiska pro pojmy, jako jsou podnikový proces, řízení a životní cyklus podnikových procesů. Následně bude prezentována notace BPMN, jednotlivé fáze životního cyklu BPM a detailní popis process miningu. Závěr opory tvoří popis softwarových nástrojů (Bizagi, Disco a ProM) a případové studie z oblasti BPM a process miningu. Celý text je proložen množstvím praktických příkladů.

Autoři: **Doc. RNDr. Ing. Roman Šperka, Ph.D.**
Ing. et Ing. Michal Halaška
Ing. Dalibor Šimek

Recenzenti: **Prof. Ing. Miroslav Hučka, CSc.**
Doc. Mgr. Petr Suchánek, Ph.D.

ISBN **978-80-7510-312-3**



Toto dílo podléhá licenci:

Creative Commons Uveďte původ-Zachovejte licenci 4.0

Znění licence dostupné na:

<http://creativecommons.org/licenses/by-sa/4.0>

Obsah

ÚVODEM.....	6
RYCHLÝ NÁHLED STUDIJNÍ OPORY.....	7
1 ZÁKLADNÍ POJMY Z OBLASTI PROCESNÍHO ŘÍZENÍ.....	8
1.1 Podnikový proces.....	11
1.2 Typy podnikových procesů.....	18
1.3 Workflow.....	21
1.3.1 Automatické provedení podnikového procesu.....	21
1.3.2 Typy Workflow systémů.....	23
1.4 Metody pro modelování podnikových procesů.....	25
2 MODELOVÁNÍ A SIMULACE.....	30
2.1 Systém.....	30
2.2 Model.....	32
2.3 Modelování.....	34
2.4 Klasifikace modelů.....	35
2.5 Simulace.....	36
3 BUSINESS PROCESS MODELING NOTATION (BPMN).....	41
3.1 Business Process Diagram (BPD).....	42
3.1.1 Symboly používané v BPMN.....	43
3.2 Využití BPMN.....	48
4 ŘÍZENÍ PODNIKOVÝCH PROCESŮ (BPM).....	51
4.1 Příklady podnikových procesů.....	51
4.2 Životní cyklus BPM.....	52
4.3 Identifikace procesů.....	57
4.3.1 Označovací etapa.....	57
4.3.2 Vyhodnocovací etapa.....	58
4.3.3 Procesní architektura.....	59
4.4 Objevování procesů.....	61
4.4.1 Objevování založené na důkazech.....	62
4.4.2 Objevování založené na rozhovorech.....	62
4.4.3 Objevování založené na workshopech.....	63

4.5	Analýza procesů	63
4.5.1	Kvalitativní analýza procesů	63
4.5.2	Kvantitativní analýza procesů	65
4.6	Redesign procesů	66
4.7	Implementace a monitorování procesů	67
5	PŘÍPADOVÁ STUDIE BIZAGI	71
5.1	Představení Bizagi Suite	72
5.1.1	<i>Bizagi</i> a jeho zastoupení v životním cyklu BPM	74
5.2	Identifikace procesů	75
5.2.1	Situační analýza ukázkové firmy z pohledu procesní vyspělosti	76
5.2.2	Tvorba procesní architektury v ukázkové firmě	77
5.2.3	Prioritizace procesů	79
5.3	Objevování procesů v ukázkové firmě	81
5.3.1	Analýza dokumentů	82
5.3.2	Pozorování procesních účastníků	83
5.3.3	Rozhovory s doménovými experty	84
5.3.4	Popis a model procesu zpracování závazků	84
5.4	Analýza procesu zpracování závazků	85
5.4.1	Kvalitativní analýza ukázkového procesu	86
5.4.2	Kvantitativní analýza ukázkového procesu	89
5.5	Implementace procesu zpracování závazků	94
5.5.1	Data model	95
5.5.2	Tvorba formulářů	96
5.5.3	Obchodní pravidla	96
5.5.4	Uživatelé	97
5.5.5	Integrace	98
5.5.6	Exekuce procesu a pracovní portál	99
5.6	Monitorování a kontrola ukázkového procesu	99
6	PROCESS MINING	105
6.1	Podstata a základní pojmy process miningu	106
6.2	Data potřebná pro process mining	110
6.2.1	Záznamy	112
6.3	Procesní modely	117

6.3.1	Přechodový systém	117
6.3.2	Petriho síť	118
6.4	Objevování podnikových procesů	121
6.4.1	Alfa-algoritmus	124
6.4.2	Pokročilé metody objevování	131
6.5	Zjišťování shodnosti modelů procesů	132
6.6	Další perspektivy process miningu	141
6.6.1	Organizační perspektiva.....	141
6.6.2	Časová a pravděpodobnostní perspektiva	142
7	PŘÍPADOVÁ STUDIE DISCO	144
7.1	XES	144
7.2	Instalace softwarového produktu Disco	146
7.3	Analýza procesů v Disco.....	148
8	PŘÍPADOVÁ STUDIE PROM	162
8.1	Instalace softwaru ProM.....	163
8.2	Analýza procesů v ProM	165
	LITERATURA	183
	SHRNUTÍ STUDIJNÍ OPORY	186
	PŘEHLED DOSTUPNÝCH IKON.....	187

ÚVODEM

Studijní opora „Techniky a nástroje v oblasti řízení podnikových procesů“ je určena studentům, kteří se zajímají o problematiku řízení podnikových procesů (angl. BPM, Business Process Management). Technickou náročností je zaměřena na aplikované inženýrské obory studia na vysokých školách. Student nemusí mít žádné speciální znalosti z předchozího studia, ale orientace v základech modelování, simulací, logice a technologii informačních systémů je vítaná. Studentem studiem opory získá následující dovednosti – bude umět přemýšlet procesně ve smyslu podnikové struktury; identifikovat, objevovat, analyzovat a navrhnout zlepšení podnikových procesů; provádět detailní analýzy podnikových dat s důrazem na identifikaci slabých míst podnikových procesů.

Text je strukturován do 8 kapitol. Každá kapitola začíná stručným seznámením s jejím obsahem v rychlém náhledu kapitoly, dále obsahuje stručné cíle a klíčová slova. V samotném textu se vyskytují distanční prvky, které Vás upozorní na důležité definice, na texty pro zájemce, k zapamatování a případové studie. Distanční prvky samostatný úkol a úkol k zamyšlení budou řešeny v rámci prezenční výuky, takže si jejich řešení připravte dopředu. Každá kapitola končí kontrolními otázkami, krátkým shrnutím a odpověďmi na otázky.

RYCHLÝ NÁHLED STUDIJNÍ OPORY

Studijní opora „Techniky a nástroje v oblasti řízení podnikových procesů“ obsahuje komplexní a hloubkové představení problematiky identifikace, objevování, analýzy, redesignu a implementace podnikových procesů v podnicích, ve kterých je kladen důraz na procesní řízení, nebo které o procesním řízení uvažují.

Opora začíná představením základních teoretických pojmů z oblasti procesního řízení, modelování a simulací podnikových procesů, objasněním použití notace BPMN (Business Process Modeling Notation) a životním cyklem řízení podnikových procesů. V druhé části textu je kladen důraz na představení softwarových nástrojů, které se v této oblasti využívají a na detailní a praktické případové studie, které nabízejí studentům si proces řízení podnikových procesů vyzkoušet.

1 ZÁKLADNÍ POJMY Z OBLASTI PROCESNÍHO ŘÍZENÍ



RYCHLÝ NÁHLED KAPITOLY

V této kapitole se seznámíte se základní definicí podnikového procesu, představíme si jeho strukturu a charakterizujeme pohled na funkční a procesní řízení podniku. Na meta-modelu podnikového procesu jsou znázorněny základní prvky a účastníci podnikového procesu. Na objasnění praktického náhledu na provedení podnikového procesu použijeme model procesu objednávky pomocí notace BPMN. Následně si vysvětlíme rozdíl mezi modelem a jeho instancí procesu a definujeme pojem workflow. Neopomeneme ani základní členění podnikových procesů a objasníme podstatu hodnotového řetězce podle Portera. Kapitola v závěru představuje základní metody popisu podnikových procesů a grafických notací, které se používají při modelování podnikových procesů.



CÍLE KAPITOLY

Po prostudování této kapitoly budete umět:

- Vysvětlit rozdíl mezi funkčním a procesním řízením podniku.
 - Definovat podnikový proces, vyjmenovat a charakterizovat základní členění podnikových procesů v souladu s podnikovými aktivitami.
 - Objasnit rozdíl mezi modelem procesu a instancí procesu.
 - Definovat pojem workflow podnikového procesu a jeho základní druhy.
 - Popsat metody popisu a grafické notace procesních modelů.
-



KLÍČOVÁ SLOVA KAPITOLY

Procesní a funkční řízení, podnikový proces, rozhodovací procesy, podpůrné procesy, instance, model, modelování, workflow, notace podnikových procesů.

Současné koncepce managementu zdůrazňují několik významných charakteristik, které mají podstatný vliv na fungování dnešních podniků a organizací – (1) využívání informačních technologií (IT) a podnikových informačních systémů, (2) významná úloha lidského faktoru, vedení lidí (*leadership*), učící se organizace a týmová práce, (3) zdůraznění nutnosti procesní orientace podniků. Procesní přístup se v posledním období stal standardem

v podnikovém managementu. Můžeme ho rozčlenit do více etap, v kterých byly identifikovány především výrobní procesy jako nezbytný předpoklad finální kvality výrobků a proto byla hlavní pozornost podniků věnována právě jim. Na druhé straně administrativní procesy byly charakteristické funkčním přístupem, který se datuje do prvních desetiletí minulého století. V tom čase Henry Ford aplikoval **principy dělby práce**, které definoval dříve Adam Smith na výrobní proces automobilů a rozdělil pracovní činnosti na menší části, které umožňovaly pracovníkům se specializovat na určité činnosti a tím dosáhnout lepší produktivity. Tím na druhé straně narůstaly požadavky na vedení lidí a jejich koordinaci. Tento problém vyřešil A. Sloan, který aplikoval princip dělby práce na management podobným způsobem jako na výrobní procesy. Tím se manažeři středního řízení firmy začali specializovat na určité **funkce podniku** (např. výroba, prodej, logistika, apod.) a top manažeři řídili firmu na základě finančních ukazatelů hospodaření podniku. Takovýto funkční přístup byl typický pro období hromadné výroby hlavně v poválečném období, ale posléze došlo k výrazným ekonomickým změnám, které začaly funkční přístup odsouvat na vedlejší kolej. Těmito důvody byli především boje výrobců o zákazníka na globalizovaném trhu a role zákazníků v ekonomice, kteří se stali určujícími činiteli, co mají firmy vyrábět na základě jejich preferencí.

Za prvního autora **procesního přístupu** k řízení organizací se dá označit Michael Porter, který ve své práci o analýze konkurenceschopnosti podniků navrhnul hodnotový řetězec (více kapitola 1.2). Základním principem jeho přístupu byl fakt, že nezobrazoval podnik jako pyramidovou strukturu řízení, ale jako řetězec, ve kterém firma generuje marži a přispívá ke kladnému finančnímu výsledku celé firmy. V 80. letech minulého století zaznamenali vícere americké firmy jako např. IBM, Xerox nebo Ford prudkého zlepšení produktivity a to na základě radikálních změn svých procesů především s **využitím IT** a stali se příkladem pro zbytek světa. Postupně se začala uplatňovat myšlenka, že je lepší, když se procesy mění postupně, ne skokově a to především s ohledem na setrvačnost myšlení a chování lidí.

Významným faktorem, který ovlivňuje konkurenceschopnost podniků v současném turbulentním prostředí, které je charakteristické konkurencí na globálních trzích a využíváním informací v rozhodovacím procesu, je hlavně schopnost řízení změn napříč celým podnikem (firmou). Tyto změny, které jsou způsobeny dynamikou a proměnami podnikatelského prostředí, jsou iniciovány požadavky na inovace a zaváděním nových služeb a produktů na globální trh. Významnou úlohu hraje taky radikální reengineering nebo automatizace stávajících procesů, aktivit a pracovních postupů v podniku, firmě nebo organizaci. Pro zajištění kontinuální správy těchto změn a samotné řízení podniků je nutné radikálně změnit úhel pohledu na vnímání podniku jako celku. Výsledkem je **procesní pohled**, který každou firmu, nebo podnik vnímá jako soubor obchodních, výrobních, administrativních nebo сумárně podnikových procesů. Podnikové procesy překračují jednotlivá oddělení ve firmě a dodávají své výstupy zákazníkovi – jak externímu (koncovému), tak internímu.



K ZAPAMATOVÁNÍ

Reengineering znamená zásadní přehodnocení a radikální redesign (rekonstrukci) podnikových procesů tak, aby mohlo být dosaženo dramatického zdokonalení z hlediska kritických měřítek výkonnosti, jako jsou náklady, kvalita, služby a rychlost. Jedná se tedy o zavádění radikálních, především organizačních změn v podniku.

Přechod firem z klasického **funkčního** (úkolového) řízení a pohledu na základě organizační struktury k procesnímu pohledu, kde jsou jednotlivé podnikové procesy spjaté se strategickými cíli organizací a plně vyhodnocované a sledované na základě stanovených metrik, vyžaduje odborníky s procesním pohledem a kvalifikací, které umožňují tento přechod provést.¹



DEFINICE

Podle Evropské komise² se **podnikem** rozumí každý subjekt vykonávající hospodářskou činnost, bez ohledu na jeho právní formu. K těmto subjektům patří zejména osoby samostatně výdělečně činné a rodinné podniky vykonávající řemeslné či jiné činnosti a obchodní společnosti nebo sdružení, která běžně vykonávají hospodářskou činnost.

Obchodní zákoník³ (§5) ho definuje jako soubor hmotných, jakož i osobních a nehmotných složek podnikání. K podniku náleží věci, práva a jiné majetkové hodnoty, které patří podnikateli a slouží k provozování podniku nebo vzhledem ke své povaze mají tomuto účelu sloužit.

Podnik je podle Kislingerové (1999) tedy funkčním celkem – entitou, která je nadána schopností přinášet určitý užitek, generovat určitý výnos v současnosti i budoucnosti.

¹ PEKÁRKOVÁ, Lucie. *Techniky modelování a optimalizace podnikových procesů*. Brno, 2007. Diplomová práce. Masarykova univerzita, Fakulta informatiky.

² NAŘÍZENÍ KOMISE (ES) č. 800/2008 ze dne 6. srpna 2008. Lex.europa.eu [online]. Evropská komise, 2008 [cit. 2017-09-30]. Dostupné z: <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:214:0003:0047:cs:PDF>

³ Zákon č. 513/1991 Sb., obchodní zákoník [online]. Businesscenter.cz, 2017 [cit. 2017-09-30]. Dostupné z: <https://business.center.cz/business/pravo/zakony/obchzak/cast1.aspx>

1.1 Podnikový proces

Pro uskutečnění aktivit, které mají za cíl pochopit procesní náhled na podnik, firmu nebo organizaci je nutné vytvořit abstrakci procesu tak, aby umožňovala znázornit všechny aktivity, které v podnikovém procesu probíhají. Pozornost je věnována také vztahům mezi různými aktivitami, lidem v rozličných rolích a zařízením, které do podnikového procesu nějakým způsobem zasahují. Níže si zmíněné pojmy vysvětlíme podrobněji.

Podnikový
proces

Klíčový pojem v moderní teorii managementu (řízení) podniku představuje **podnikový proces** (*business proces*). V literatuře nalezneme různé definice. Každá z nich zdůrazňuje jiné aspekty podnikového procesu, přesto je však možné vysledovat některé společné vlastnosti.

DEFINICE



Podle Řepy (2006) je **podnikový proces** chápán jako souhrn činností přeměňující za pomoci lidí a dalších nástrojů vstupy na určité výstupy, které mají hodnotu pro zákazníky nebo jiné procesy.

Vlastnosti
procesu

Každý proces má tedy několik **základních atributů**. Z výše uvedené definice vyplývá, že proces využívá zdroje, transformuje vstupy na výstupy a skládá se z uspořádaných činností (kroků). Taktéž je možné jej dekomponovat na aktivity a subprocessy. Dále můžeme proces specifikovat jeho jednoznačným začátkem a koncem, včetně návazností na jiné procesy a opakovatelností. Navíc bychom mohli odvodit, že proces je spouštěn určitým signálem. Každý proces má své parametry, které můžeme měřit (např. kvalita, včasnost, průběžná doba, časová délka, náklady, apod.). Funkčnost procesů závisí na posloupnosti jednotlivých kroků (procedurách) a zdrojích. Každý proces má interní nebo externí vstupy či dodavatele a zároveň zákazníky a také má svého vlastníka. Vlastníkem procesu je pro každý proces právě jedna osoba. Táto osoba je odpovědná za způsob provádění procesu (jeho nastavení), tzn. za jednotlivé jeho aktivity a za dodržování stanovených postupů.

Vstupem procesu se rozumí objekt, resp. jeho stav před působením předmětného procesu. Ten se stává předmětem působení procesu. Může to být např. příkaz či plán, poptávka, přijatá faktura nebo polotovar, atd. **Výstupem procesu** je objekt, resp. jeho stav po působení tohoto procesu. Může to být např. hotový výrobek, uhrazená faktura, vrácená faktura, vyškolený pracovník nebo vyskladněný materiál, apod.).

Definice procesu je jeho reprezentace ve formě, která umožňuje jeho automatizované zpracování. Automatizovaným zpracováním je například jeho spuštění pomocí Workflow Management Systému (WfMS), nebo vytvoření modelu a jeho simulace (více o workflow naleznete v kapitole 1.3). Definice procesu obsahuje hlavně informace o kritériích zahájení procesu, přerušování a ukončení činnosti, síti činností a jejich vztahů, aplikacích a datech

nebo o údajích o účastnících. Definice procesu pro účely WfMS obvykle tvoří souhrnný popis různých objektů (entit):

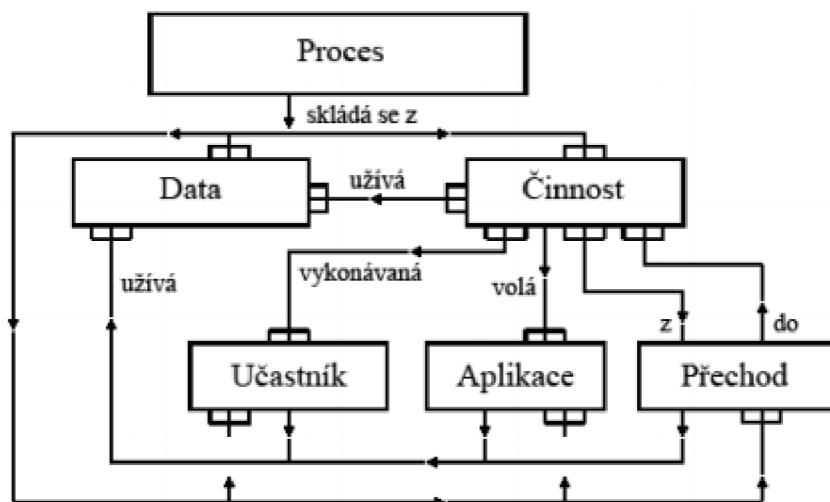
- Proces (popis celého procesu)
- Činnost (definice činností, z nichž se proces skládá)
- Účastník (seznam/deklarace účastníků procesu)
- Přejchod (definice přechodů mezi činnostmi)
- Data (deklarace dat procesu)
- Aplikace (deklarace aplikací používaných procesem)

Vzájemné vztahy entit znázorňuje tzv. metamodel procesu (obr. 1-1).



K ZAPAMATOVÁNÍ

Workflow Management System je softwarový systém pro nastavení, spouštění a monitorování definované sekvence úkolů, které označujeme jako **workflow**.



Obrázek 1-1: Metamodel procesu.

Zdroj: http://is.muni.cz/th/60555/fi_m/DP-pekarkova.pdf

Podnikový proces se provádí za účelem splnění určitého **podnikového cíle** (*business goal*). Jeho provedení je časově ohraničeno, přičemž pro jeho zahájení platí vstupní podmínky. Ukončení procesu je charakterizováno dosažením určitých koncových podmínek. Jednotlivé činnosti (aktivity), které probíhají v rámci procesu, jsou vzájemně provázány.

Aalst (2004, s. 33) ve své práci uvádí, že proces popisuje úlohy, které musí být uskutečněny a také specifikuje pořadí, v jakém mají být provedeny. Aktivita, které realizují příslušné úlohy, nemohou být řazeny libovolně. Jsou vždy uspořádány do určité posloupnosti tak, aby bylo dosaženo požadovaného podnikového cíle.

K ZAPAMATOVÁNÍ



Aktivita tvoří jeden logický atomický krok v procesu, kterého část popisuje. Aktivita může být manuální, tedy taková, která nevyžaduje počítačovou podporu, nebo automatizovaná (**workflow**). Workflow aktivita vyžaduje lidské a/nebo technické zdroje k pokračování vykonávání procesu (nejčastěji IT). Automatizovaná aktivita v průběhu podnikového procesu, kterého je součástí, je schopna být automaticky uskutečňována pomocí WfMS. Manuální aktivitu naopak není možno automatizovat, proto pod WfMS nespadá.

Každá aktivita se začne vykonávat pouze tehdy, pokud jsou splněna všechna omezení pro její provádění (*execution constraints*). Libovolná operace, která se v rámci podniku provádí, může být popsána jako proces. Podnikový proces představuje např. vyřízení objednávky od zákazníka, zajištění aktualizace aplikace u klientů, ale také například vývoj samotného produktu. Podnikové procesy interagují s jinými procesy. Jejich vzájemný kontakt přitom není omezen pouze na procesy v rámci firmy, která je ustanovuje, ale velmi často dochází také k interakci mezi procesy různých podniků (*business-to-business processes*).

K ZAPAMATOVÁNÍ



Soubor vzájemně se doplňujících dovedností je **role**. Role jsou přiřazovány k jednotlivým aktivitám. Cílem rolí je umožnit jejich plnění v rámci vykonávání procesu. Role reprezentuje skupinu zdrojů. Zdroj je prostředek nebo jejich skupina, které jsou nutné k vykonání aktivity. Zdroje mohou být lidské (pracovníci) anebo technické (zařízení, stroje, systémy, apod.). Zdrojem mohou být například výrobní prostředek, informace, vykonavatel aktivity, další vykonavatel nebo jiný proces.

Vykonavatel procesu je ten, kdo provádí aktivity spojené s procesem. Vykonavatel musí být při analýze a popisu procesu jednoznačně identifikován na různých hladinách konkrétnosti a pro různé účely. Role tedy definují zodpovědnost, kompetenci a chování jednotlivých osob nebo skupiny osob, které spolupracují. Jednotlivé lidské zdroje jsou mapovány na požadované role podle toho, jak jsou požadované kompetence slučitelné se schopnostmi těchto osob.

Nyní si představíme jednoduchý příklad procesu, který zachycuje proces přijetí objednávky ve firmě. Podnikový proces je složený z následujících činností:

- Přijetí objednávky od zákazníka,
- analýza objednávky,
- posláni zprávy o zamítnutí objednávky,
- odesláni zprávy o přijmutí objednávky,
- odesláni objednávky do oddělení objednávek,
- uzavření procesu přijetí objednávky.

Výše uvedený textový popis podnikového procesu pomocí aktivit, které je nutné provést, není úplný. Podle definice podnikového procesu výše musí být stanoveno také pořadí, v jakém budou jednotlivé aktivity uskutečněny. K zachycení vazeb mezi aktivitami se v modelování používá grafické notace⁴.

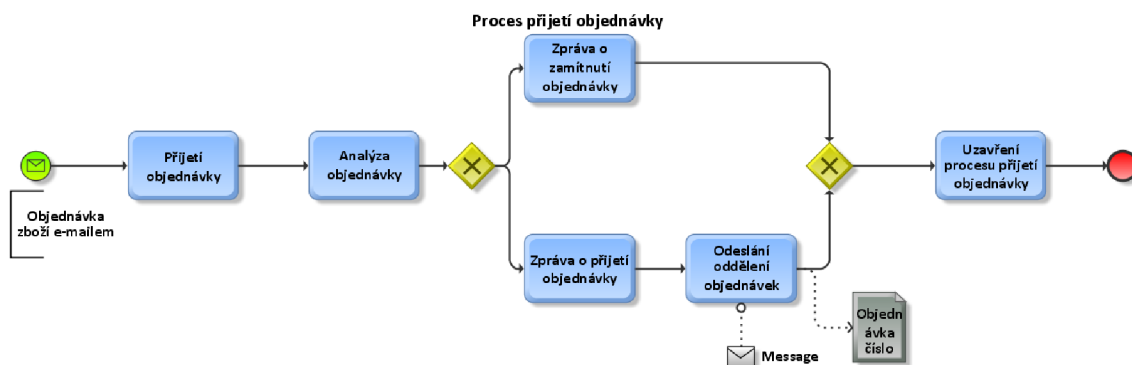
Podnikový proces přijetí objednávky je znázorněn na obr. 1-2. Proces začíná obdržetím e-mailové zprávy s konkrétní objednávkou zboží. Není nezbytně nutné přesně specifikovat formu zprávy, resp. samotné objednávky. Forma může být slovní, nebo písemná ve formě firemního formuláře, který používá firma pro objednávky svých zákazníků. V ideálním případě to může být datový objekt, například datového typu XML, který je výstupem účetního systému zákazníka a podnikový informační systém (např. MS Dynamics) je schopen ho přijmout a poslat ke zpracování. V dalším kroku musí proběhnout ověření objednávky a pracovník firmy musí posoudit, jestli došla objednávka splňuje všechny požadavky ke zpracování, nebo zdali je zboží k dispozici, apod. Dále mohou nastat dvě situace – (1) objednávka z nějakého konkrétního důvodu není pracovníkem firmy přijata a zákazníkovi je poslána zpráva, která popisující zdůvodnění odmítnutí jeho objednávky, nebo (2) objednávka splňuje podmínky pro vyřízení a firma zašle zákazníkovi potvrzení přijetí objednávky a taky odešle tuto objednávku do oddělení objednávek k dalšímu zpracování. Objedávka se odešle formou zprávy „Message“ a příslušného datového objektu „Objedávka číslo“. Objednávkový proces je nakonec uzavřen přijetím objednávky.

Příklad přijetí objednávky

Grafická notace, která byla použita pro návrh modelu objednávkového procesu, se nazývá Business Process Modeling Notation (zkratka BPMN). Obdélníky v procesních modelech této notace znázorňují jednotlivé aktivity. Orientované šipky pak označují vzájemné vztahy aktivit. Aktivita, u které šipka začíná, musí předcházet aktivitě, do které šipka směřuje. Speciální symbol ve tvaru diamantu se využívá pro znázornění větvení procesu a následného spojení jednotlivých větví. Diamant obsahuje písemný znak, který indikuje, jakým způsobem se proces rozdělí. Ve vzorovém procesu jsme potřebovali vyjádřit alternativu – má se provést pouze jedna vybraná větev znamenající (1) přijetí, nebo (2) nepřijetí

⁴ Pojem notace označuje formální prostředky pro zápis skutečnosti, nejčastěji ve formě grafické.

objednávky. Pro tento účel se používá diamant se znakem X. Bez ohledu na výsledek je nutné nakonec provést aktivitu uzavření objednávkového procesu. Proto se v grafickém zápisu procesu obě větve opět spojí pomocí symbolu diamantu se znakem X. Ten vyjadřuje ukončení větvení. Podrobněji se notaci BPMN budeme věnovat v kapitole 4.



Obrázek 1-2: Objednávkový proces.

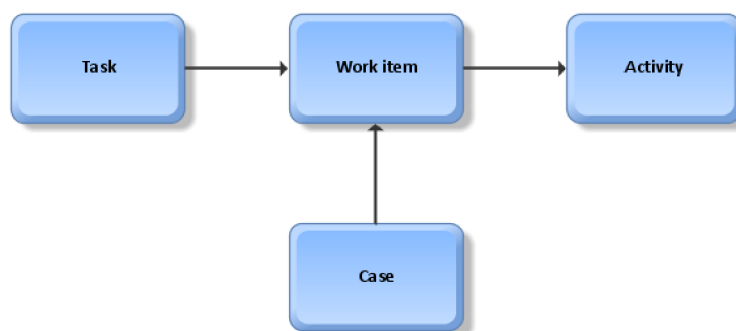
Zdroj: vlastní zpracování

Procesní
model a in-
stance

Další typický podnikový proces, který se realizuje v každé firmě, která prodává zboží je reklamační proces. Metoda, znázorněná na obr. 1-2 a popsaná v textu výše se dá úspěšně použít pro popis toho, jakým způsobem se má provádět libovolná reklamacie, kterou musí nějaká firma uspokojit. Tato metoda představuje tzv. šablonu, která popisuje průběh práce v procesu vyřizování reklamace, nebo objednávek, atd. Tato šablona se nazývá **model procesu** nebo **procesní model**. Konkrétní případ reklamace uskutečněný podle procesního modelu potom nazýváme **instancí procesu**. Podobným způsobem můžeme uvažovat o jednotlivé aktivitě, která je prováděná v rámci libovolného procesu. Na úrovni procesního modelu je aktivita popsána **modelem aktivity** a její konkrétní provedení potom představuje **instanci této aktivity**.

Podnikové procesy představují důležitý koncept, který pomáhá vedení podniku pochopit, jakým způsobem jejich podnik funguje. Definice procesů navíc nepředepisuje, jaké implementační strategie nebo platformy budou použity (Weske, 2007, s. 10). Aktivity jsou popisovány abstraktně. Díky tomuto přístupu je možné docílit lepší integrace informačních systémů (IS), které podnik používá k provedení činností v rámci podnikového procesu.

Zaměřme se nyní na základní stavební prvky každého procesu, na aktivity. Aktivita představuje logickou jednotku práce, kterou můžeme popsat jako atomický proces. Aktivita je tedy buď provedena celá, anebo není provedena vůbec. Výrok o nedělitelnosti aktivity je však relativní. Pokud budeme vytvářet např. model výroby určitého produktu, můžeme ho definovat pomocí jediné aktivity „výroba produktu“. Tato aktivita ale bude ve skutečnosti tvořena poměrně komplexním subprocessem, který obsahuje řadu dalších aktivit. Záleží tedy na zvolené míře abstrakce procesního modelu s ohledem na jeho velikost a srozumitelnost. Weske (2007) uvádí ještě další pojmy, které jsou spojené s aktivitou. Jedná se o **úkol** (*task*) a **pracovní položku** (*work item*). Jejich vzájemný znázorňuje obr. 1-3.

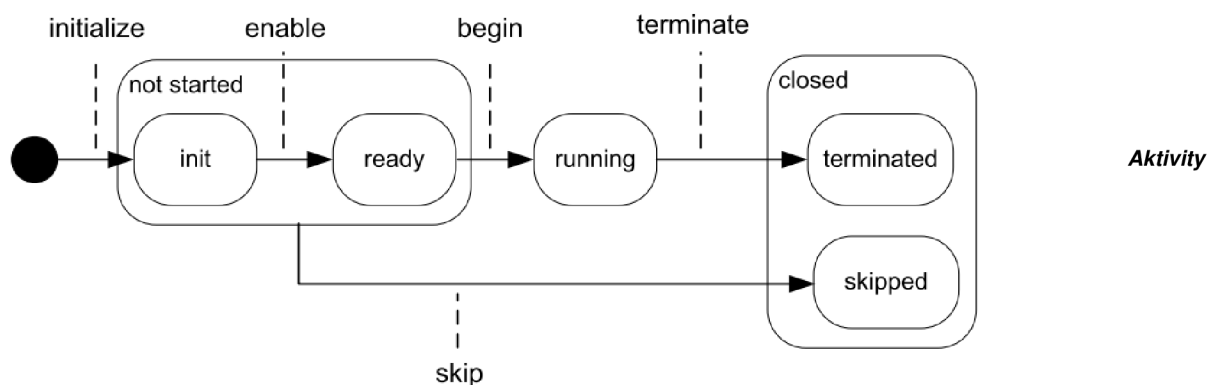


Obrázek 1-3: Úkol, pracovní položka, aktivita a případ.

Zdroj: vlastní zpracování

Úkol odpovídá dříve definovanému pojmu model aktivity. Pracovní položka představuje úkol, který je specifikovaný pro určitý případ, tzn., jedná se o instanci aktivity. Samotná aktivita je podle Weskeho (2007) realizace pracovní položky vybraným zdrojem. V tomto textu se však budeme zabývat jenom modelováním procesů, a proto budeme používat pouze pojmy model aktivity a instance aktivity tak, jak byly vysvětleny výše. Aktivity v rámci procesu můžeme podle klasifikovat do tří skupin:

1. manuální aktivity,
2. systémové aktivity,
3. aktivity uživatelských interakcí.



Obrázek 1-4: Stavový diagram instance aktivity.

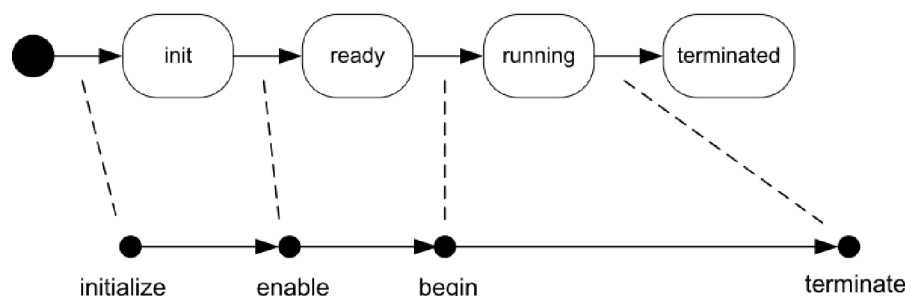
Zdroj: Weske (2007, s. 86)

Manuální aktivity mohou být vykonány pouze lidmi. S použitím IS se nepočítá. Zástupcem této skupiny aktivit je např. fyzické zaslání nové objednávky v rámci objednávkového procesu. **Aktivity uživatelských interakcí** realizují lidské zdroje. Pro jejich úspěšné dokončení je ovšem potřebná podpora IS. Jako příklad lze uvést uzavření objednávkového procesu, kdy pracovník firmy vloží do registru objednávek data, které popisují právě řešený

případ. **Systémové aktivity** obsahují aktivity, jejichž provedení je zajištěno pouze prostřednictvím IS. Příkladem takové aktivity by mohlo být automatické odeslání zprávy, která oznamuje přijetí nebo nepřijetí objednávky firmou.

Nyní se budeme podrobněji věnovat modelování aktivity. Podle Weskeho (2007, s. 86) lze instanci aktivity popsat pomocí stavů, ve kterých se její uskutečnění nachází a příslušných přechodů mezi těmito stavy. Jednoduchý stavový diagram instance aktivity prezentuje obrázek 1-4.

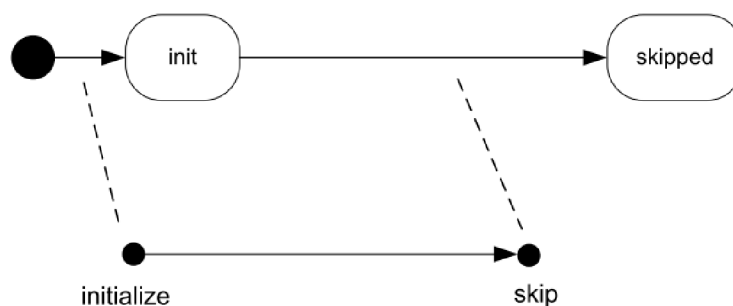
V momentě vyvolání aktivity (přechod „initialize“) se její instance nachází ve stavu vyvolání („init“). Aby mohla být aktivita připravena k realizaci (stav „ready“), musí být nejdříve povolena přechodem „enable“. V tomto stavu je aktivita stále považována za nezahájenou („not started“). Pokud v kontextu provádění procesu nepotřebujeme aktivitu uskutečnit, dojde k jejímu vynechání přechodem „skip“ do stavu „vynechána“ („skipped“). V opačném případě je aktivita započata přechodem „begin“ a po dobu provádění setrvává ve stavu běhu („running“). Ukončení aktivity zajistí přechod „terminate“ a aktivita přejde do ukončeného stavu („terminated“). Tento stav společně se stavem „přeskočeno“ vyjadřuje, že aktivita je uzavřena („closed“).



Obrázek 1-5: Událostní diagram provedené aktivity.

Zdroj: Weske (2007, s. 86)

Průchod aktivity jednotlivými stavy je možné modelovat pomocí **událostí**. Událost vyjadřuje skutečnost, že došlo k přechodu z jednoho stavu instance aktivity do jiného. Uspořádáním událostí v čase potom můžeme zachytit průběh libovolné aktivity. Událost reprezentuje pojmenovaný okamžik. Tento okamžik je důležitý z pohledu vykonávání aktivity. Délka jeho trvání je nulová. Doba, během níž se aktivita nachází v určitém stavu, je definovaná časovým intervalem mezi uskutečněním dvou po sobě jdoucích událostí. Pořadí událostí vyjadřuje diagram událostí. Obr. 1-5 znázorňuje vztah diagramu událostí ke stavovému diagramu aktivity popsanému výše. Pro větší přehlednost jsou odděleny dva možné průběhy aktivity – úplně provedená aktivita (obr. 1-5) a vynechaná aktivita (obr. 1-6).



Obrázek 1-6: Událostní diagram vynechané aktivity.

Zdroj: Weske (2007, s. 86)

V prvním případě můžeme průběh aktivity popsat pomocí čtyř událostí – vyvolání („initialize“), povolení („enable“), zahájení („begin“) a ukončení („terminate“). Jakmile nastane událost vyvolání, aktivita přejde do stavu „init“. V tomto stavu setrvává, dokud nedojde k události „povolení“. Od tohoto momentu je aktivita připravena a k jejímu spuštění dochází s příchodem události „zahájení“. Ukončení aktivity je potom vyjádřeno výskytem události „ukončení“. Druhý případ je ještě jednodušší. Vyvolání je opět spojeno s událostí „vyvolání“. Potom ovšem dojde k události „vynechání“ („skip“) a aktivita není provedena, což je reprezentováno příslušným stavem.

1.2 Typy podnikových procesů

Procesní řízení se zaměřuje na opakované činnosti, které je možno separovat a následně popsat, tz. vytvořit a popsat pomocí procesního modelu. Pokud tedy nalezneme v podnikovém procesu opakované řetězce činností (aktivit), mohli bychom vzbudit dojem, že se jedná o primitivní řetězce aktivit, které jsou vykonávány rutinně, příp. naprogramovanými vykonavateli. To je ale falešný obraz, neboť hledání významných procesů v organizacích je vždy velmi obtížný úkol. Přestože máme řadu metodik a postupů, jedná se vždy o sofistikované řešení.

Jak již bylo popsáno, proces je sled kroků navržených za účelem vytváření výrobků nebo služby. Některé procesy mohou být plně vykonávány v jediné organizační jednotce firmy, ale většina procesů prochází napříč celou strukturou firmy. Podnikové procesy obvykle procházejí napříč funkčními jednotkami podniku, což ale nezajišťuje jejich optimální řízení. Optimální řízení procesu znamená, že celý proces je řízen, monitorován a vykazován **vlastníkem procesu**. Ten řídí proces s cílem naplnění požadavků zákazníka. Podnikové procesy mají ze své definice vždy **zákazníky**. Zákazníky můžeme rozdělit do dvou skupin:

- **Externí zákazník** – využívá výstupy podnikového procesu vně podniku,
- **Interní zákazník** – využívá výstupy podnikového procesu uvnitř podniku a je interním subjektem firmy.

Podobně můžeme rozlišovat podnikové procesy podle jejich vztahu k subjektům, které do nich vstupují, nebo jimi jsou ovlivněny. Z tohoto hlediska je možné procesy dělit na (Gála a Pour, 2009, s. 27):

- **Interní procesy** – probíhající v rámci jednoho podniku, případně pouze v jeho organizačních jednotkách (divizích, pobočkách, závodech, apod.). Pro tyto procesy je charakteristické, že činnosti firmy jsou vztažené pouze k dané firmě nebo útvaru. Tedy zejména k vlastním pracovníkům firmy. Příkladem interního procesu je výrobní zakázka a její řízení.
- **Mezipodnikové, externí procesy** – zahrnující vztahy podniku k externím subjektům (obchodním partnerům, státní správě a samosprávě, apod.), překračující hranice firmy. Jsou realizovány částečně u dodavatelů, u spolupracujících firem nebo u konečného zákazníka. Hlavní črtou těchto podnikových procesů je, že jejich jednotlivé činnosti se dělí mezi několik subjektů, které si v rámci průběhu procesu vzájemně předávají vstupní a výstupní informace, případně materiál, polotovary, produkty, atd. Příkladem takových procesů je například řízení obchodních zakázek.

Typy procesů

Na základě kategorie zákazníků a stanovených cílů podnikových procesů lze procesy v podniku dělit na (Gála a Pour, 2009, s. 27):

- **Řídící procesy** – zahrnují v sobě činnosti spojené s definováním strategických cílů podniku a zajištěním realizace těchto cílů v rámci celého podniku. Mezi tyto procesy tedy řadíme krátkodobé (operativní) plánování, stanovení cílů, odměňování a alokaci zdrojů, zpětnou kontrolu, apod.
- **Základní procesy** – vytvářejí produkt (výrobky nebo služby), který má hodnotu pro externího zákazníka. Jejich výsledkem je produkování výstupů, které požaduje externí zákazník. Základní procesy podporují hlavní podnikatelskou činnost firmy, která představuje naplnění poslání firmy, strategických cílů, atd. Na základě konkrétních vizí a poslání firmy lze podle jejich významu hlavní procesy dále členit na **klíčové procesy** (*core processes*).
- **Vedlejší procesy** – jsou podobné hlavním podnikovým procesům, ale nejsou z hlediska vize a poslání firmy důležité natolik, aby se výrazným způsobem podílely na hlavní podnikatelské činnosti firmy. Vedlejší procesy mohou být uskutečňovány paralelně s hlavními procesy nebo sdílenými procesy. Jejich výstupy jsou určeny převážně pro externího zákazníka. Jako takové jsou hlavními kandidáty na vyloučení z vlastní činnosti podniku formou outsourcingu⁵. Jako příklad můžeme uvést proces provozu školy pečení v rámci pekárny.

⁵ Outsourcing (angl. „out“, vně, a „source“, zdroj) představuje, že podnik vyčlení různé podpůrné a vedlejší činnosti a svěří je smluvně jiné firmě jako sub-kontraktorovi, který je specializovaný na příslušnou činnost. Cílem je snížit náklady na vlastní provoz.

- **Podpůrné procesy** – jejich výstupem je vytvoření vhodných podmínek, které podporují funkcionality hlavních procesů. Jejich charakteristickým znakem je vytvoření přidané hodnoty pro externího zákazníka ve formě produktu, který externí zákazník sice nevidí, ale který je nezbytný pro efektivní řízení podniku. Příkladem je kontrola kvality.
- **Sdílené procesy** (nebo služby) – vytvářejí podmínky, které umožňují funkci všech podnikových procesů. Vytvářejí hodnotu pro interního zákazníka. Příkladem je proces přijetí objednávek.

Hodnotový řetězec

Na proces můžeme nahlížet také jako na **hodnototvorný nebo hodnotový řetězec**. Každý krok procesu (aktivita, činnost) by měl přidat jistou hodnotu výrobku nebo poskytované službě oproti kroku předchozímu. Podle úrovně řízení můžeme podnikové procesy dále členit na procesy operativního, taktického a strategického řízení. Podle oblasti řízení např. na procesy obchodního, finančního, marketingového nebo logistického řízení.



PRO ZÁJEMCE

Hodnotový řetězec (*value chain*) rozčleňuje firmu na její strategicky významné aktivity. Důvodem je porozumění vývoji nákladů a rozeznat existující potenciální zdroje odlišností. Firma získá konkurenční výhodu tím, že bude tyto strategicky důležité aktivity dělat levněji a lépe než její konkurenti.

Michael Porter vytvořil hodnotový řetězec jako primární nástroj, který slouží k nalezení možností, jak vytvořit **větší hodnotu pro zákazníka**. Firmy obhospodařují soubor činností, jejichž hlavním cílem je designovat, produkovat, distribuovat a podporovat své výrobky nebo služby. Hodnotový řetězec podniku a způsob jakým podnik provádí jednotlivé aktivity, jsou odrazem jeho historického vývoje, strategie, postoje k realizaci této strategie a vnitřní ekonomiky těchto aktivit samotných. Rozdíly mezi jednotlivými hodnotovými řetězci u konkurence jsou hlavním zdrojem **konkurenční výhody**. Hodnotový řetězec znázorňuje jeho celkovou hodnotu a skládá se z hodnototvorných aktivit a marže. Hodnototvorné aktivity jsou technologicky a fyzicky odlišné činnosti, které podnik provozuje. Jsou to stavební kameny, jimiž firma vytváří produkt, který má pro jeho koncového zákazníka určitou hodnotu. **Marže** je rozdíl mezi koncovou cenou a souhrnnými náklady na uskutečnění potřebných hodnototvorných aktivit.

Robert S. Kaplan a **David P. Norton** kladou značný důraz na hodnotový řetězec ve svém modelu **Balanced Scorecard** (používaná zkratka BSC). Hlavním faktorem, který ovlivňuje hodnotu produktu pro zákazníka, je zákaznická perspektiva a pro vlastníky to je finanční perspektiva.

Hodnotový řetězec popisuje každý podnik a jeho procesy jako probíhající řadu dílčích aktivit, které společně ovlivňují pozici podniku ve vztahu ke konkurenci a k zákazníkům. Tyto aktivity se podílejí na vytváření hodnoty, proto se také využívají k hodnocení vlivu

jednotlivých dílčích aktivit na celkovou hodnotu podniku a mohou být následně zdokonalovány.

BSC je metoda v managementu a představuje systém vyvážených ukazatelů výkonnosti podniku. Vytváří vazbu mezi operativními činnostmi a strategií se zaměřením na měření výkonu. Metoda BSC vznikla v roce 1992 jako reakce na fakt, že řada strategických záměrů podniků nebyla přetavena do reality. Mnoho podniků má problém s reálným propojením operativních činností se strategií tak, aby byla strategie implementovaná do všech podnikových oblastí a tím bylo možné ověřit **dosazení strategických cílů**. Jedním z důvodů je, že východiskem operativních plánů jsou většinou jenom finanční ukazatele, které nejsou schopné v dostatečné míře charakterizovat celý podnik. Je nutné proto sledovat a vzájemně skombinovat **všechny ukazatele (metriky)** – kromě finančních také ukazatele zaměřené na zaměstnance, firemní procesy a v neposlední radě taky na zákazníky.

Hodnotovými výhodami zákazníka jsou ty vlastnosti produktů, které přinášejí podniku spokojenost a loajalitu zákazníků v cílových tržních segmentech. V jednotlivých segmentech se tyto výhody liší, ale je možné je rozčlenit do tří skupin, které platí pravděpodobně v každém ze segmentů. Patří mezi ně:

- Vztahy se zákazníky,
- vlastnosti výrobku nebo služby,
- pověst a image podniku.

1.3 Workflow

1.3.1 AUTOMATICKÉ PROVEDENÍ PODNIKOVÉHO PROCESU

Významným přínosem zavedení procesního řízení v podniku je mimo jiné také možnost **automatizace řízení** a **koordinace probíhajících aktivit**. Podnikový proces, jehož provádění je částečně, nebo úplně počítačově automatizováno se nazývá **workflow**. Tohoto progressu v řízení je možné dosáhnout pomocí explicitní znalosti definice podnikového procesu. Zatímco tradičně řízení podniku spočívá na vedoucích pracovnících, kteří kontrolují korektní provádění aktivit manuálně na základě stanovených podnikových postupů a svých zkušeností, při existenci exaktního popisu podnikového procesu může být dohled nad správou pořadí vykonávaných činností přenechán informačnímu systému – **workflow systému**. Workflow systém na základě existujícího procesního modelu zajišťuje realizaci předepsaných činností. Tyto činnosti jsou přidělovány automaticky přiděleným pracovníkům. Řešitelem může být nejenom člověk, ale také další IS. Proto obecně diskutujeme o tzv. **přidělování zdrojů**.

Workflow systémy kromě samotné koordinace a řízení procesu zajišťují také další prospěšné funkce, mezi které patří podpora modelování procesu, jeho administrace, simulace nebo monitoring jeho průběhu. Pro kompletní informaci o názvosloví workflow systémy bývají podle jiné terminologie také nazývány systémy pro řízení podnikových procesů (angl. „Business Process Management Systems“ – BPMS).



DEFINICE

Workflow je automatizace části nebo celého podnikového procesu. V průběhu procesu jsou předávány informace, dokumenty a/nebo úkoly od jednoho účastníka procesu k dalšímu podle stanovených procedurálních pravidel.

Workflow umožňuje existující podnikové procesy zprůhlednit a pomáhá zajistit zvýšení efektivnosti, zjednodušení a zkrácení průběhu celého podnikového procesu. Na druhé straně ale klade vysoké nároky na jednoznačnost specifikace a přesnost procesu kvůli automatizaci. Základní **rozdíl mezi workflow a podnikovým procesem** je v tom, že workflow je řízeno k tomu určeným softwarem (WfMS nebo ERP⁶). WfMS definuje, tvoří a řídí průběh podnikového procesu. Workflow umožňuje interpretovat definici procesu, komunikovat s jeho účastníky a v případě nutnosti spouštět jiné aplikace. Z technického hlediska je samotný WfMS také zajímavý, neboť propojuje technologie, metodiky⁷ a principy různých oblastí řízení a ICT, jako např. elektronická pošta, řízení úkolů, znalostí, dokumentů technologie klient/server, databázové zpracování, modelování a monitoring procesů a jiné.

Počítačové modely podnikových procesů upřesňují všechny nutné atributy a parametry pro vykonání složitých procesů. Parametry jsou jak jednotlivé kroky (například dotaz v databázi, vložení záznamu zákazníka, atd.), tak určení podmínek a pořadí, za nichž mohou být tyto kroky vykonány. Podmínkou může být např. určení odpovědnosti za krok procesu nebo datový tok mezi jednotlivými kroky procesu. Jednou z nejnáročnějších částí implementace a realizace podpůrného počítačového systému jsou složité soustavy transakcí nebo

⁶ ERP (Enterprise Resource Planning) je informační systém, který integruje a automatizuje velké množství podnikových procesů, které souvisejí s odpovídajícími činnostmi firmy nebo podniku. Typicky se jedná o prodej, distribuci, logistiku, výrobu, fakturaci a účetnictví nebo správu majetku, apod.

⁷ Metodika je teoreticko-praktické schéma, které specifikuje postup provádění odborné činnosti. Vychází z vědeckého poznání a empirie. Také přesně vymezuje konkrétní postupy pro výkon zvolené činnosti.

operací, které je nutné koordinovat a organizovat na cílených výsledcích či případných sehláních a také v závislosti na jejich pořadí.

Workflow zatím nemá česky ekvivalent, proto je i v dalším textu opory používána anglická varianta. Do češtiny by tento výraz mohl být přeložen jako „tok prací“. Infrastruktura podniku je tvořena kombinací všech jeho procesů. Tedy i těch, které zdokumentované nejsou, neboť jsou vytvořené a uložené v hlavách pracovníků nebo v různých směrnících, případně mohou být vyžadovány v rámci zvyků, neformálních pravidel, atd. Některé postupy jsou tedy předávány mezi pracovníky pouze ústně. Jak vidíme, toto je ten nejméně vhodný způsob, jak s infrastrukturou nakládat. Jakýkoli pokus rozšíření nebo o zlepšení této infrastruktury vyžaduje dokumentaci. Jedině tak můžeme o ni mluvit, poznat ji, aktualizovat a revidovat ji. Důležitá je také rychlost provádění podnikových procesů. Současné řídicí postupy kladou na první místo rychlost jednotlivých procesů, kterými jsou především vývoj a výroba produktu, případně logistika. Jde o rozhodující podnikové procesy v konkurenčním boji, které ovlivňují rozvoj a udržitelnost podnikání. Vždy bychom měli myslet na to, že konkurence již může mít řadu workflow procesů implementovanu. Tím před námi získává výhodu nejen v rychlosti implementace, ale také v kvalitě.

Od implementace WfMS je možné očekávat **snížení nákladů** a **zvýšení efektivity práce**, které vyplývají ze zavedení standardních postupů, zlepšení kvality a organizace práce, zjednodušení podnikových procesů, dokumentace dosud jen v hlavách uložených postupů (což při odchodu pracovníka výrazně zjednoduší výškolení jeho následovníka). Na základě vyhodnocení dokumentace postupů je také jednodušší navrhnout změny. V každé chvíli je možné jednoduše zkontrolovat stav jednotlivých případů, jejichž vybavení se pomocí WfMS značně zrychluje. V oblasti dokumentů má implementace WfMS také své výhody – veškeré změny v datech nebo dokumentech jsou autorizovány a průběh každého případu je zaznamenán v historii, kterou není možné dodatečně měnit. WfMS přirozeně podporuje i řízení kvality, které je důležitou složkou na konkurenčním trhu.

1.3.2 TYPY WORKFLOW SYSTÉMŮ

Členění systémů do skupin se společnou charakteristikou je běžné, pokud potřebujeme získat lepší pohled na široké spektrum systémů. WfMS jsou taktéž členěny podle různých hledisek do skupin a typů. **Základní členění WfMS** je z hlediska charakteru podporovaných podnikových procesů. Z tohoto pohledu můžeme WfMS rozčlenit do čtyř základních skupin – administrativní, kolaborativní, produkční a ad hoc systémy.

- **Administrativní workflow** slouží k vyřizování běžné agendy. Tento typ WfMS zajišťuje rutinní administrativní činnosti. Příkladem jsou registrace vozidla, vyřízení objednávky, vystavení faktury, ve školství např. žádost o mimořádný termín zkoušky apod. V každém podniku bychom našli množství podnikových procesů tohoto typu. Bývají dobře strukturovatelné, nejsou složité, často se opakují, nemají moc alternativních možností a běžně jsou spojené se standardizovanými formuláři. Pro administrativní workflow je typické, že téměř každý pracovník

v podniku je jeho potencionálním účastníkem, z čehož plyne důležitost dostupnosti systému pro každého. Také by se mělo myslet na to, že účastníci administrativního workflow jsou jen příležitostní, že toto není jejich hlavní pracovní náplní. Proto je nutné administrativní workflow co nejvíce simplifikovat. Z charakteru administrativního workflow vyplývá, že se značně liší v jednotlivých organizacích a také, že podléhá časovým změnám. Příkladem administrativního workflow je například zpracování žádosti o dovolenku. Ačkoli jde o podnikový proces, který je snadno strukturovatelný, jeho implementace je v různých firmách odlišná.

- **Kolaborativní workflow** podporuje spolupráci v týmu. Charakteristická je v tomto případě existence nějakého dokumentu, pomocí něhož si pracovníci vyměňují své poznatky a který je vlastně výsledkem jejich společné práce. Kolaborativní procesy většinou obsahují nějaký opakovaný cyklus několika iterací téhož kroku, dokud se všichni nesjednotí na výsledku. Taktéž může dojít i k návratu do předchozí iterace. Příkladů kolaborativních workflow procesů najdeme v každé organizaci několik – může jít o tvorbu propagačních materiálů, dokumentace, návrh nové služby, změnu designu produktu apod. Jedno mají tyto podnikové procesy společné – vždy je jako výstup dodán dokument, na kterém spolupracuje několik pracovníků a který prochází různými schvalovacími kolečkami. Další typickou vlastností kolaborativních workflow procesů je jejich výrazná dynamičnost. Některé kroky jsou definovány až na základě průběhu předchozích aktivit. Pro kolaborativní workflow je charakteristické, že účastníci podnikového procesu pracují společně, podnikové procesy jsou tak méně rigidní a je pro ně typická dynamická změna jejich definice. Pro dobré řešení je vhodné, aby nebylo vtíravé a umožňovalo kreativitu účastníků podnikového procesu. Také musí být flexibilní, protože kreativní pracovníci často využívají předem nedefinované postupy. Průchodnost těchto systémů nebývá obvykle důležitá. Příkladem kolaborativního workflow by mohlo být představení marketingového konceptu pro uvedení nového výrobku na trh. Základní črty takového marketingového konceptu dle postupů firmy poskytuje pro tento případ určená šablona workflow. Marketingový manažer na základě dostupných zdrojů, svých znalostí a omezení zpracuje plán do detailů. Poté mají manažeři z poboček s lepšími informacemi o daném trhu možnost připomínkovat plán a navrhopvat úpravy dle vlastních představ. V daném časovém horizontu se pak implementuje globální plán, jehož stav musí být průběžně k dispozici všem participujícím pracovníkům.
- **Produkční workflow** podporuje hlavní (klíčové) podnikové procesy, které vytvářejí přidanou hodnotu k finálnímu produktu a na kterých je závislá spokojenost zákazníků. Tyto podnikové procesy jsou také dobře strukturovatelné, i když jejich struktura může být někdy složitá. Výskyt takových podnikových procesů je častý, pracovníci jim věnují většinu své pracovní doby. Alternativní průběhy hlavních procesů jsou předem definovány a jejich výskyt je relativně omezený. Tyto procesy jsou vlastně jistou variací výroby v továrně. Dělníci vykonávají

řadu činností, ale jenom jediná je hlavní. Tato činnost určuje jejich profesi, zařazení a identifikuje jejich rutinní úkoly. Na základě toho vznikl i název pro tento typ workflow. Produkční workflow je typické tím, že flexibilita změn definice podnikového procesu není důležitá, neboť jejich výskyt není častý. Navíc změna definice podnikového procesu není starostí koncových uživatelů, ale specialistů. Z toho vyplývá, že změna definice podnikového procesu většinou souvisí s podstatnějšími změnami v celém podniku. Produkční workflow vyžaduje integraci s jinými podnikovými systémy. Je celkem zřejmé, že čím je kratší prodleva mezi jednotlivými kroky podnikového procesu, tím je celý systém více produktivní. Typickým produkčního workflow může být zpracování žádosti o úvěr v bankách. V tomto případě se jedná o opakované zpracování poskytnutí standardního produktu.

- **Ad hoc workflow** je poslední skupinou WfMS z pohledu charakteru procesů. Tato skupina WfMS je založena na náhodnosti vzniku workflow procesu. Jedná se o podnikové procesy, kterých průběh není popsán předem. Nejsou proto standardizované a jsou většinou jedinečné. K jejich definici musí dojít v okamžiku jejich vzniku. Jistým způsobem jsou podobné administrativním workflow. Rozdílem je, že postup definice podnikového procesu má trend ke zpracování výjimek, odchylek a ojedinělých situací. Z toho vyplývá podstatný rys této skupiny procesů – zatímco celý podnikový proces je jedinečný, jeho účastník se běžně podílí na mnoha opakovatelných a podobných subprocesů. Ad hoc WfMS vyžadují od pracovníků vysokou míru samostatnosti. Proto je nutná široká dostupnost workflow produktu a jednoduchá definice workflow procesu. Příkladem ad hoc workflow je požadavek zákazníka na dodání velkého množství produktu s vlastním návrhem balení, které je specifické a vymyká se běžným postupům a možnostem podniku. Dalším možným příkladem je odpověď na zadání odběratele, zpracování závěrečného reportu projektu nebo selhání logistické přepravy, která byla smluvně garantovaná.

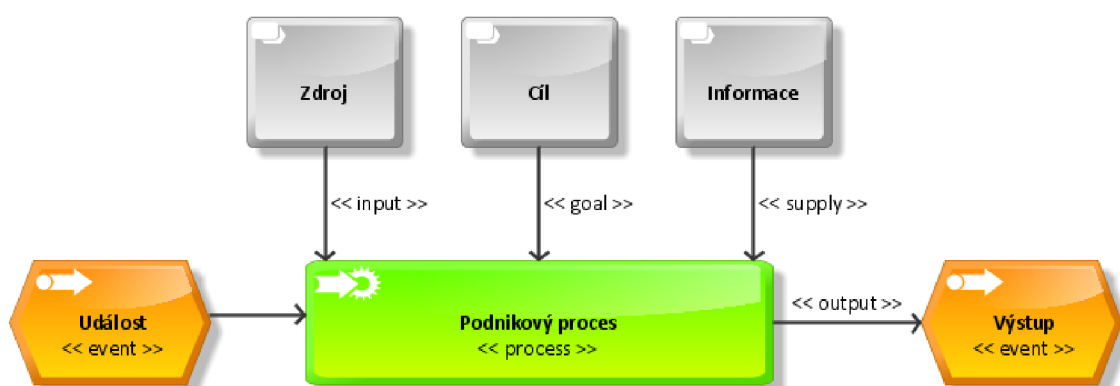
1.4 Metody pro modelování podnikových procesů

Pro modelování procesů se používají různé metody. Mezi základní patří **BP**EL (*Business Process Execution Language*) (Basl, 2012, s. 115). Z obecného hlediska je možné pro modelování podnikových procesů použít např. univerzální modelovací jazyk **UML** (*Unified Modelling Language*), případně v upravené podobě podle H. Ericssona se čtyřmi základními pohledy na podnik (Řepa, 2006):

- **Procesní pohled** – zahrnuje podnikové procesy, aktivity a hodnoty v podniku, které tyto činnosti vytvářejí. Charakterizuje vzájemnou spolupráci podnikových

procesů a využívání zdrojů za účelem dosažení strategických cílů, které jsou definované ve vizi⁸ podniku.

- **Chování organizace** – zahrnuje vnitřní „chování“ a interakci jednotlivých prvků podniku (procesy a zdroje). Cílem analýzy těchto interakcí je přidělení odpovědnosti za jednotlivé zdroje.
- **Strategický pohled** – zahrnuje klíčové pojmy jako např. strategické cíle a hodnoty podniku. Soustředí se na úmysly a hlavní problémy, které mají být procesní změnou vyřešeny.
- **Strukturní pohled** – zahrnuje zdroje organizace, jako jsou znalosti, dokumenty, organizační jednotky, informace, produkty apod.



Obrázek 1-7: Metamodel podnikového procesu.

Zdroj: upraveno podle Řepy (2006)

K analýze podnikových procesů slouží soustava technik, mezi které patří také **diagram procesů**. Jak ilustruje obr. 1-7, jsou znázorněny základní objekty, které s procesem souvisejí:

Diagram procesů

- **Řídící objekty** – objekty, které řídí průběh podnikového procesu.
- **Cíle** – jichž má být prostřednictvím podnikového procesu dosaženo. Takovým cílem může být například spokojený zákazník nebo kvalitní produkt.
- **Vstupy** – objekty, které jsou podnikovým procesem přetvářeny nebo spotřebovávány. Jsou jimi suroviny, informace nebo lidská práce.
- **Výstupy** – objekty, které jsou produktem nebo výsledkem podnikového procesu.

⁸ Víze sumarizuje to, čím chce podnik, společnost nebo organizace být, popisuje budoucnost, která je významně odlišná od současnosti tím, že vytyčuje směr dlouhodobých změn v společnosti a je zdrojem inspirace. Poskytuje důležitá rozhodovací kritéria pro tvorbu základních strategických cílů a směrů, v kterých je společnost chce dosáhnout.

- **Podpůrné objekty** – suroviny nebo informace, které jsou podnikovým procesem využívány, ale nejsou přetvářeny ani spotřebovávány.

Pro modelování podnikových procesů jsou využívány různé **modelovací standardy** a **normy**. Modelovací normy určují prvky modelu podnikového procesu a jejich význam. Příklad standardů (Basl, 2012, s. 116), které jsou běžně podporovány různými nástroji pro modelování podnikových procesů, jsou následující:

Standardy

- **BPEL** – Business Process Execution Language (BPEL) je technická norma, která se využívá při popisu spustitelných procesních modelů, které jsou určeny k provádění, automatizaci a integraci. Provádění je zabezpečeno webovými službami (*web services*⁹).
- **BPMN** – Business Process Modeling Notation (BPMN ve verzi 2.0) je technicky specifikovaná norma pro vytváření procesních modelů. Více se o BPMN dočtete v další kapitole.
- **UML** – Unified Modeling Language (UML) je univerzální modelovací jazyk pro analýzu a návrh funkcionalit softwarových produktů, ale používá se i v doméně modelování podnikových procesů. Primárně pomáhá zmazat mezeru mezi návrhem softwaru, který je pochopitelný pro lidi mimo ICT a detailním návrhem softwarových systémů.
- **EPC** – Event-driven Process Chain (EPC) je významný standard pro modelování podnikových procesů. Notace, která je zaměřena netechnicky, dovoluje rychle a jednoduše uživatelům mimo ICT optimalizovat a dokumentovat jejich workflow.

Při modelování podnikových procesů se používají tři základní metody popisu:

- **Neformální metody popisu** – popis podnikových procesů prostřednictvím přirozeného jazyka. Nemá danou syntaxi ani sémantiku. Vyskytuje se v podobě nestrukturovaného volného textu jako například „Michal přijme cenovou nabídku od dodavatele, zašle ji Mirkovi a ten zjistí, jestli nabízená cena není mimo rozsah běžné ceny u tohoto typu služby.“. Výhody této metody jsou například v tom, že pro kompletaci popisu není potřebné specifické vzdělání, ani zvláštní nástroje. Porozumění popisu je poměrně snadné. Nevýhodou je zavádění změn a složitá správa, nevhodnost pro prezentaci, nepřehlednost, nemožnost automatického zpracování a analýzy. Dále pak skutečnost, že každý má jiný styl psaní a rozumí jiným detailům. Problémy často vznikají při popisu souběžnosti a alternativy.

Základní metody popisu

⁹ Webová služba je softwarový systém umožňující interakci dvou strojů na síti. Je popsána ve strojově zpracovatelném formátu, konkrétně WSDL. S webovou službou ostatní stroje komunikují způsobem, který je předepsaný v popisu služby, pomocí protokolu SOAP přepravené pomocí jiných, již zavedených protokolů, tzv. tunelování firewallu.

- **Semiformální metody popisu** – popis podnikových procesů pomocí grafických notací. Pro tyto metody je typická volná sémantika a přesná syntaxe. Existuje poměrně velké množství dostupných notací. Příkladem je BPMN (další kapitola), UML, EPC, IDEF, RAD (*Rapid Application Development*), NPC a další. Grafický zápis je mnohem přehlednější než volný text. Výhody jsou v jednodušší správě a zavádění změn, možnost automatického zpracování a analýzy. Při dodržení několika omezení je možné semiformální popis převést do formálního tvaru. Nevýhodami jsou potřeba zvláštních nástrojů, potřeba základní znalosti notací i při čtení, složitější tvorba, často dochází k rozdílným výkladům obsahu. Každá z použitých notací má v podstatě své výhody a nevýhody, ale použití více druhů vede k chaosu. Semiformální notace mají přesně danou syntaxi, ale nedikují použití precizní a úplné sémantiky.
- **Formální metody popisu** – umožňují jednoznačně definovat podnikový proces pomocí precizně dané syntaxi a sémantice použité metody. Jinak řečeno, jde nám o takové procesní modely, které je možné navrhovat v počítači, ukládat je tam, analyzovat a popřípadě dále využívat ve WfMS nebo ERP systémech. V momentě, kdy taková notace získává rysy precizní syntaxe i sémantiky, stává se z ní formalizmus, nebo formální specifikace. Formální specifikace je jinými slovy technika, která umožňuje jednoznačně popsat modelovaný systém.



OTÁZKY

1. Vysvětlíte procesní pohled na podnik a rozdíly oproti funkčnímu pohledu.
2. Popište obecně co je podnikový proces.
3. Jaké členění je typické pro podnikové procesy?
4. Jaký je rozdíl mezi workflow a podnikovým procesem?
5. Jaký je rozdíl mezi modelem podniku a jeho instancí?
6. Jaké typy workflow se v současnosti používají v praxi?
7. Jaké metody popisu podnikových procesů poznáte? Vysvětlíte rozdíly.



SHRNUTÍ KAPITOLY

V této kapitole jsme se seznámili se základními pojmy z oblasti modelování podnikových procesů. Vysvětlili jsme si podstatu a složení procesu, typy podnikových procesů a procesní náhled na podnik, workflow a jeho typy. Závěrem kapitoly jsme nahlédli na základní notace modelování podnikových procesů.

ODPOVĚDI



1. Kapitola 1, str. 9.
 2. Podkapitola 1.1, str. 11.
 3. Podkapitola 1.2, od str. 18.
 4. Podkapitola 1.3.1, str. 22.
 5. Podkapitola 1.3.
 6. Podkapitola 1.3.2. str. 23 a 24.
 7. Podkapitola 1.4, str. 27 a 28.
-

2 MODELOVÁNÍ A SIMULACE



RYCHLÝ NÁHLED KAPITOLY

Tato kapitola se věnuje představení základních pojmů v oblasti modelování – co je systém a jaké druhy systémů poznáme, z čeho se skládá a jiné. Dále se budeme věnovat událostem, transakcím, aktivitám a atributům systému. Část textu představuje také modelování, existující modely a klasifikaci modelů. Druhá polovina kapitoly prezentuje základní principy vědecké disciplíny – simulace. Vysvětlíme si definici simulace, simulačního experimentu a představíme základní členění simulací – spojitou a diskrétní simulaci.



CÍLE KAPITOLY

Po prostudování této kapitoly budete umět:

- Definovat co je systém, model, modelování a simulace.
- Vyjmenovat a objasnit základní typy modelů, simulací a základních simulačních konstruktů.



KLÍČOVÁ SLOVA KAPITOLY

Systém, modelování, model, simulace, program, běh, experiment, spojitá a diskrétní simulace.

Když chceme například podle Kindlera (2001) definovat modelování, je nutné před tím objasnit hned několik důležitých pojmů jako například systém, model a další. Množství termínů, které budeme v této kapitole používat, jsou cizí slova, nejčastěji adaptované z anglického jazyka. Každý počítačový specialista je nucen ve své odborné činnosti číst anglickou literaturu. Zároveň je však potřebné říct, že množství cizích slov, které se v oblasti modelování používají, mají jiný význam jako v běžném životě.

2.1 Systém

Systém

Termín **systém** se v dnešní době vyskytuje v mnoha oblastech a v různých významech. V teorii modelování se studuje nějaká věc, nebo její možné varianty. Věc je nějaký objekt, který se vyskytuje v reálném světě, který skutečně existuje (např. člověk, úřad, firma apod.), nebo o kterém předpokládáme, že by existovat mohl (počítač, škola, výrobní linka

nebo továrna, který by měla být postavena a jiné.). Věc, jak ji chápou filozofové, se uvažuje v její úplné složitosti nebo spolu se všemi nejasnostmi její existence. Stejně však chápou, že není v lidských možnostech celou věc rozumně pochopit a zvládnout. Podobně to chápou i různé technické obory, a proto se uvažují na zkoumaných věcech abstrakce oprošťující od přílišných detailů. Abstrakce usnadňují zainteresovaným osobám o věcech rozumně přemýšlet a komunikovat. Tyto oprošťující detaily nebo aspekty nejsou pro fungování řešené věci kritické. Taková abstrakce se v teorii modelování nazývá systémem.

Podle druhu profese, která systém na věci „sleduje“, „provádí“, „definuje“ bude tento systém dostávat přívlastek (např. politický systém, elektronický systém, mechanický systém a jiné.). Co se týče času, ten v abstrakci může nebo nemusí být zanedbán. Například firma Google, která pořizuje fotodokumentaci do své mapové cloudové aplikace se nemusí zajímat o to, jak vypadala krajina před nebo po datu focení, nebo kolik domů je právě rozestavěno.

DEFINICE



Statickým systémem nebere do úvahy význam času. V **dynamickém systému** se naopak čas nezanedbává. Množina okamžiků, v kterých dynamický systém existuje, se nazývá **existence** dynamického systému. Každý dynamický systém je v každém okamžiku své existence v nějakém **stavu**. **Událost** je změnou stavu dynamického systému. Systém je složen z **prvků**.

Taky se může stát, že systém je tvořen jenom jedním prvkem, ale taková situace je poměrně ojedinělá. Běžně se systém skládá z více prvků, které mají svoje typické chování. Když poznáme chování jednotlivých prvků, je možné z toho odvodit chování celého systému. Prvky systému, tedy prvky abstrakce na nějaké věci, mohou odpovídat elementům, které na věci fyzicky (vztahové složky, prostorové), nebo logicky (provázanost dané věci, nebo jejich částí, schopnosti) rozeznáváme.

Počet prvků v dynamickém systému se může během jeho existence měnit, např. biologický systém se může smršťovat a růst. V ekonomických a technických systémech jde nejčastěji o to, že prvky mohou do systému „vstupovat“ a ze systému „vystupovat“. Tyto prvky se nazývají **transakce** (*temporary elements, transactions*). Ve skutečnosti takové prvky nevznikají, ale přicházejí do systému z jeho „okolí“, a nezanikají, nýbrž systém opouštějí. Vzhledem k tomu, že samotný systém je abstrakce, která našemu myšlení nahrazuje zkoumanou věc, abstrahujeme i od okolí, které reálně pro danou věc existuje, ale pro systém nikoliv. Jinak řečeno, když je nějaká složka reality přítomna v části prostředí, od kterého abstrahujeme, tak v naší abstrakci je to stejné, jako kdyby neexistovala. Příkladem transakcí jsou např. zakázky směřující do podniku, pacienti nastupující do nemocnice, zákazníci přihlašující se do e-shopu nebo vozidla, které vstupují do dopravního systému,

který je – jakožto systém – abstrahován tak, že je vyčleněn ze svého okolí, z něhož do něj vozidla přijíždějí a kam jej vozidla opouštějí.



DEFINICE

Permanentními prvky neboli **aktivitami** (*activities, permanent elements*) se nazývají prvky, které jsou v dynamickém systému během celé jeho existence.

Vysvětlili jsme si, že když o systému přemýšlíme, zanedbáváme vše, co do něho nechceme zahrnout. Z toho vyplývá, že když transakce dynamický systém jednou opustí, je „ztracena“ z našich úvah a nemá smysl zabývat se jejím návratem. Jinak řečeno, pokud by bylo možné, že transakce systém skutečně opustila a pak se do něj vrátila, nemělo by být možno určit, že jde o tu samou transakci. Pokud se identita transakce naopak určit má, pak tato transakce nemůže systém opustit, nýbrž v něm zůstává, může být nějak skryta a vyskytuje se bez důležitosti na to, co se v systému děje, ale nemůže systém opustit a musí být po celou dobu uvažování do abstrakce začleněna.

Atributy

Prvky systému mají své vlastnosti, které se nazývají **atributy**. Příkladem může být teplota vody v chladícím systému tepláren (jedná se o tzv. **reálný** nebo **aritmetický atribut**, protože ten nabývá aritmetických hodnot, reálných čísel), dále to může být funkčnost čidla v produkčním systému (jedná se o tzv. **booleovský atribut**, protože nabývá booleovských hodnot „ano“ a „ne“, konkrétněji značeno „měří“ a „neměří“), nebo jméno dodavatele firmy (jedná se o tzv. textový atribut, který nabývá textových hodnot). Atributy tedy přiřazují prvkům určité hodnoty a ty se u prvků dynamického systému mohou v čase měnit.

2.2 Model

Jak se náš praktický život stává složitějším, tím více modelování jako vědecká metoda nabývá důležitosti v téměř každé lidské činnosti. Stává se z něj jedna ze základních metod poznání. Teorie modelování se zabývá všemi oblastmi poznání – od zkoumání mikrosvěta po objekty velkých rozměrů, např. zemětřesení, atmosférické proudy anebo tvoření hvězdných mlhovin. Modelování se prosazuje taky v oblasti zkoumání sociálních jevů, nevyjímaje jevů poznání lidské psychiky.

Termín „**model**“ se používá na označení materiální i nemateriální napodobeniny objektu, přičemž nezávisí na tom, za jakým účelem se dělá. Samotná napodobenina se může vytvořit v souvislosti s různými praktickými úlohami, např. může plnit určité funkce při vyučování (jako např. makety, schémata, tabulky a jiné, které se používají jako názorné výukové pomůcky) anebo se používá za badatelským účelem.

Model

K vytvoření nebo návrhu modelů se běžně přistupuje v těch případech, kdy je obtížné anebo vůbec nemožné zkoumat jev nebo objekt v jeho přirozené formě. V mnoha případech

je zkoumaný jev natolik složitý (důležitý), že když ho pozorujeme v jeho bezprostřední formě, dochází k vzniku rizika schematizaci anebo zjednodušení reálného obrazu procesů, které v něm probíhají. Podobné nebezpečí může vzniknout při zkoumání jevů, které se vyvíjejí velice rychle anebo velice pomalu. Mnohem složitější je situace v případech, kdy se zabýváme jevy, které patří do minulosti nebo do budoucnosti. Současná praxe a věda se neustále zabývá řešením takových typů úloh.

Jednou ze základních otázek, které se objevují při startu řešení každého nějaké úlohy praktické, teoretické, pragmaticky-praktické apod.) je, jestli máme k dispozici předběžné vědomosti o předmětných jevech a procesech.

K ZAPAMATOVÁNÍ



Apriorní informace je předběžná vědomost se o objektu a může se chápat jako počáteční. Někdy je taková informace tak komplexní a detailní, že vytýčenou úlohu je možné řešit bez dodatečného zkoumání objektu. Takové situaci poté odpovídá také nutnost vytvořit, resp. navrhnout modely.

Existují i případy, kde je apriorní informace pro řešení úlohy nedostačující. V tomto případě je potřebné výzkum prodloužit, aby bylo možné získat další informace o objektu zkoumání. Ačkoliv to vždy není možné. Snadno mohou nastat i takové situace, kdy apriorní informace vůbec není k dispozici. I v tomto případě je jedním z rozhodujících prostředků pro tvoření konkrétního systému poznatků proces modelování. Ve všech takových situacích práce výzkumníků probíhá specifickým způsobem:

Apriorní informace

- Místo zkoumání objektu přímo v jeho přirozené formě se přistupuje k tvorbě anebo využití dostupné náhrady objektu zkoumání a celý výzkumný proces se v tomto případě přenáší na objekt nový, dokud se znalosti, které jsou získané při jeho zkoumání, týkají reálného objektu. V teorii modelování se v souvislosti s výše uvedeným používají tyto termíny – (1) **originál** se používá pro přirozený objekt a pro identifikaci objektu, který je předmětem bádání, taktéž kváziobjekt, náhrada, (2) **analogie** – pro označování objektu nového, vytvořeného nebo používaného na výzkum originálu.
- Každá náhrada reálného objektu musí vyhovět několika požadavkům. Hlavně se určitý jev zkoumá jako model tehdy, pokud je zdrojem nového poznání objektu našeho zájmu, nějaké vhodné informace a ověření. Aby splnil svou úlohu, musí se model nacházet v nějakém vztahu vůči objektu, musí mu být podobný. Ve většině případů objektivní vztah originálu a modelu tkví v tom, že model odráží reálný objekt. Současně každý model reprezentuje vybraný objekt v zjednodušené formě. V případě, že model modeluje svůj objekt v celé jeho mnohostrannosti a složitosti,

není potřeba jeho tvorby. Modely se tvoří, abychom získali **nové vědomosti** o zkoumaném objektu. Obvykle mnohem jednodušeji se dá vytvořit model než originál.

Slovo „model“ se používalo v běžné řeči nejprve pro předlohu Kindler (2001). V odborném jazyku z minulosti zůstal z praxe termín „funkční model“ jako první exemplář nového výrobku. Jeho funkce imitují nový výrobek, ačkoliv jiné vlastnosti nového výrobku (např. estetické) tento exemplář ještě nemá. Z této praxe vznikla i interpretace slova model pro něco nezvyklého, zvláštního nebo nákladného. V modelování a simulaci se termín model používá pro vyjádření analogie mezi dvěma systémy. Jednoduché příklady nabízí socha (model zvířete nebo osoby apod. v neživém materiálu), dětský vláček (model skutečného vlaku ve zmenšeném měřítku) nebo zeměpisná mapa (model části země na papíře).



PRO ZÁJEMCE

Vztah dvou systémů – modelovaného a modelujícího je dán tím, že každému prvku P modelovaného systému je přiřazen prvek Q modelujícího systému. Dále pak každému atributu g prvku P je přiřazen atribut h prvku Q a pro hodnoty atributů g a h je daná určitá relace. Charakter relace není nějak obecně omezen, ale v případě, že g i h jsou aritmetické atributy, bývá takovou relací tolerance (mapa zobrazuje jen přibližně), úměrnost, kombinace tolerance a úměrnosti (např. rozměry složek a částí dětského vláčku jsou přibližně úměrné odpovídajícím rozměrům skutečného vlaku) apod. Když jsou modelovaný i modelující systém statické, říkáme, že daný model je **statický model**.

2.3 Modelování

Modelování

V oblasti výpočtové techniky, informačních a komunikačních technologií dominuje anglicky psaná literatura, které je potřebné se přizpůsobit a uvědomit si, že modelovat (*to model*) a modelování (*modelling* nebo *modeling*) může mít téměř neomezeně mnoho významů.



DEFINICE

Podstatou **modelování** ve smyslu výzkumné techniky je náhrada zkoumaného systému jeho modelem (přesněji systémem, který jej modeluje), jejímž cílem je získat pomocí pokusů (nebo experimentů) s modelem informaci o původním zkoumaném systému.

Platí tedy, že vytvoříme model, ve kterém modelovaným systémem je zkoumaný systém. Přičemž my budeme experimentovat s modelujícím systémem. Cílem bude zjistit něco

o modelovaném systému. Kdyby bylo cílem nahrazení modelovaného systému modelujícím systémem v reálném světě, šlo by o modelování ve stylu vytváření protězy. Pokud by cílem experimentování bylo dozvědět se něco o modelujícím systému bez vztahu k systému modelovanému, šlo by vlastně jen o přímě experimentování s modelujícím systémem.

Modelující systém může být abstraktní matematická struktura (např. vzorec), která je měněna lidskou myslí a interpretovaná třeba na tabuli., může to být fyzikální analogie. Je zřejmé, že v dnešní době se z mnoha důvodů stále více uplatňuje ve funkci modelujícího systému výpočet na počítači.

2.4 Klasifikace modelů

Ve vědeckém světě i v praxi se většinou vytváří myšlené konstrukce, které hrají úlohu modelů. Ačkoliv možné jsou taky takové případy, kdy se jako modely použijí reálně existující objekty. Tyto mohou být podobné, nebo se pokládají za podobné zkoumanému objektu. V této souvislosti mohou být modely přirozené, případně uměle vytvořené objekty.

- Při návrhu a realizaci nových systémů nebo při výrobě určitých výrobků se často vytvářejí **prototypy**. Prototypy reprezentují důležité vlastnosti objektu, kterým se zabýváme. Když se při zkoumání těchto prototypů získá nová informace, pak se zkoumají jako modely. Často se vytvářejí také tzv. **pokusné provizoria**, které znamenají kvantitativní model pro plánovaný technologický komplex. Když pokusné provizorium dodává negativní výsledky, tak se mění původní projekt nebo plán.
- Metoda modelování nachází specifické využití při zkoumání jevů, jako je např. chování určité sociální skupiny, veřejné mínění, spotřebitelská poptávka apod. V takovýchto případech se používá teorie reprezentativního výzkumu. Podle této teorie se vybírá určitá skupina osob, které se zkoumají z hlediska vytyčeného cíle, například se zkoumá jejich reakce na jejich chování v dané situaci, jisté události atd. Když postupujeme správně, můžeme získat zdroj hodnotné informace. Tato informace se může postupně rozšířit na celou populaci, nebo na všechny členy předmětné sociální skupiny. Mluvíme konkrétně o svérázném pokusném provizoriu, které má sice charakter modelu, ale je nutné respektovat formulovanou teorii statistiky a dodržovat určitá pravidla.
- Značný rozvoj praktické a vědecké činnosti si vynutil a umožnil tzv. **matematické modelování**, které do značné míry odstraňuje nedostatky jiných forem sociálního modelování. Matematický model je vlastně jakýsi popis struktury a chování objektů v systému pomocí matematických prostředků. Matematické modely jsou různé. Je možné, aby vyjadřovaly základní charakteristiky systému, např. rovnice, které popisují pohyb systému, resp. stavy, dále pak tabulky a grafy pro popis přechodu systému z jednoho stavu do dalšího atd. Když se matematický aparát aplikuje v oblastech, které jsou na to vhodné, je nenahraditelným a silným prostředkem na překo-

nání subjektivismu v teorii a praxi. V dnešní době se matematické modelování používá s využitím ICT, což do teorie a praxe modelování přináší zásadní nové momenty. Využití výpočtové techniky je jedním z nejjistějších prostředků realizace vědeckého postupu v řízení, přičemž se metoda modelování dostává na kvalitativně vyšší úroveň.

- V současnosti se stávají důležitějšími úlohy tzv. **funkčních a strukturních modelů**. Možnosti vytváření těchto modelů jsou ovlivněny skutečností, že každý objekt má svou funkční stránku, tzn. určité chování a současně se odlišuje vnitřním strukturním obsahem. Díky tomu je možné modelovat chování i strukturu daného systému. Strukturní a funkční modelování nachází široké uplatnění v teorii řízení, hlavně při projektování a budování systémů řízení. Jedním ze základních požadavků při budování řídicího systému je naprojektovat takovou strukturu, která by ideálně realizovala funkce řízení.

V souvislosti s úrovní organizace objektu rozeznáváme modely **sublokální, lokální, superlokální** a **superglobální**. Zajímavé jsou modely, které se rozlišují podle počtu modelovaných objektů. Modely, označované jako **singulární** se vztahují na jeden objekt určitého rozměru (např. na konkurenceschopnost jedné skupiny firem během sledovaného období), **binární modely** (zahrnují dva objekty stejného rozsahu - např. vztah mezi dvěma firmami), **multipletní modely** (zahrnují víc než dva objekty). Pozornost si zasluhuje i klasifikace modelů, která je spjatá s charakterem procesu jejich vytváření. Do čistě diskrétních modelů spadají „modelující“ (adaptivní) modely, které např. dovolují, aby se řídicí systém stále adekvátněji přizpůsoboval měnícím se podmínkám.

Modely se mohou členit podle množství metod, které se používají při jejich návrhu. V tomto smyslu existují tzv. **simplexní modely** (při využívání jedné metody), **duplexní modely** (při využití dvou metod), **komplexní modely** apod.

2.5 Simulace



DEFINICE

Podle Kindlera (2001) a Kolektivu autorů (1986) je **simulace** výzkumná technika, jejíž podstatou je náhrada zkoumaného dynamického systému jeho simulátorem, přičemž se simulátorem se experimentuje se záměrem získat informace o originálu – původním zkoumaném dynamickém systému.

Obecně (Kindler, 2001) slovo simulace znamená předstírání bezvědomí, nemoci, duševní poruchy apod. Odborně vzato, tento význam slova simulace by měl patřit do sociální psychologie. Pojem simulace, jak s ním pracuje informatika a jak ho chápou i ostatní obory,

když aplikují ICT, má však zcela jiný obsah. Jednoduše řečeno, v této oblasti je simulace chápána jako modelování ve smyslu výzkumné metody, při němž použitý model je simulační.

V tomto smyslu budeme termín simulace dále chápat i my. Platí zde vše, co bylo zmíněno výše o modelování jakožto výzkumné technice. Především cílem simulace je získat informace o simulovaném systému. Jeho pouhá náhrada simulátorem není dostatečná. Taková náhrada se někdy nazývá **emulací**. Například simulátor jednoho počítače P1, který realizovaný na počítači jiného typu je jakousi protézou, která nahrazuje P1, který např. není pro nás dostupný, ale máme pro něj programy. Dále platí, že simulátor nemusí být realizován na počítači, ale dnes je takto implementován stále častěji. Číslicový počítač má své výhody především v tom, že je možné k němu přistupovat vzdáleně a dá se použít i k jiným účelům (to se dá využít, když ho zrovna nepotřebujeme použít k simulaci). Taky je obecně známo, že nespotřebovává mnoho energie a nekazí životní prostředí, i když se s tím dá polemizovat. Nezanedbatelná výhoda využití ICT je i v tom, že výpočty na počítačích (a tedy i pokusy se simulátorem) je možné reprodukovat.

Simulátor

Zdůrazněme, že aby šlo o simulaci, musí být cílem pokusů se simulátorem snaha dozvědět se něco o simulovaném systému. Když je simulátor realizován jako výpočet na počítači, může se složkou simulace stát i experimentování se simulátorem, jehož cílem je získat informaci o něm samotném a ne o simulovaném systému. Nastává to například tehdy, když ověřujeme, zda v příslušném programu není programátorská chyba nebo zda v něm není použita nevhodná numerická metoda. Tento proces se nazývá ověření správnosti modelu nebo jinak **verifikace modelu**.

Verifikace modelu

Sloveso simulovat (*to simulate*) budeme ve shodě s anglicky psanou odbornou literaturou chápat jako provádět simulaci. Odborník, který provádí simulaci, je v angličtině nazýván *simulationist*, odpovídající český termín neexistuje.

V minulosti byl simulátor realizován na speciálních zařízeních a podle nich dostávala příslušná simulace označení – např. galvanická, hydrodynamická, elektromechanická, odporová, mechanická, analogová (pomocí analogových počítačů) a hybridní (pomocí hybridních analogově-číslíkových počítačů). Dnes byly všechny tyto druhy nahrazeny simulací, při níž je simulátor realizován na číslicovém počítači, tedy **simulace číslicová** (*digital simulation*). V dalším textu budeme přívlastek číslicová vynechávat, protože žádnou jinou simulaci řešit nebudeme. Když je však zřejmé, že jde o simulaci číslicovou, spojuje se možnost vyjádřit něco přívlastkem s charakterem simulovaného systému. Když je charakter systému spojitý, tzn., hodnoty jeho atributů se mění v čase jen spojitě, mluví se o **spojité simulaci** (*continuous simulation* – simulace spojitých systémů). Když je simulovaný systém diskrétní, tzn., nenastávají v něm spojitě změny v čase, mluví se o **diskrétní simulaci** (*discrete event simulation* – simulace diskrétních událostí). Když je simulovaný systém kombinovaný, tzn. má-li jak vlastnosti typické pro spojitý systém tak vlastnosti typické pro diskrétní systém, mluví se o kombinované diskrétně-spojité simulaci nebo častěji jenom o **kombinované simulaci** (*combined simulation*).

Typy simulací

Významy právě představených tří základních typů simulace jsou v různých situacích a různými autory více či méně modifikovány nebo i kombinovány. Připomínáme, že systém je definován na věci. Na jedné a téže věci může být definován jak spojitý, tak diskrétní (nebo i kombinovaný) systém. A tak se klidně může stát, že je někdy jedna věc zkoumána pomocí spojitě, diskrétní a případně i kombinované simulace. Například odborník v oboru mechatroniky „vidí“ na počítači spojitý systém, a může tedy např. zkoumat procesor pomocí spojitě simulace; avšak odborník v oboru počítačových věd vidí na počítači diskrétní systém a může aplikovat na studium téhož procesoru diskrétní simulaci. Dále je zřejmé, že i když simulujeme spojitý systém na počítači, který je číslicový, je zajímavé, že „uvnitř počítače“ existuje jakýsi diskrétní dynamický systém. Tento diskrétní systém vzniká aplikací numerické metody a „diskretizací modelovaného spojitého systému“. I v tomto případě však jde o spojitou simulaci, protože (viz text výše) použitý přívlástek odráží to, jak definujeme simulovaný systém my, a není v tomto případě důležité, co se děje v simulátoru.

Simulační program

Program, který řídí výpočet při simulaci, se nazývá **simulačním programem**. Neexistuje jednotné vysvětlení, zda se pod tímto označením rozumí program ve strojovém kódu, který skutečně výpočet řídí, nebo zdrojový kód programu, napsaný v programovacím jazyku. Je zřejmé, že důvod této nejednotnosti spočívá v tom, že v praxi kvůli ní nedochází k žádným fatálním nedorozuměním nebo selháním. I v tomto textu budeme používat označení simulační program jak pro zdrojový kód, který napíše autor simulačního modelu v programovacím jazyku, tak pro strojový kód, který z něho vznikne kompilací nebo interpretací, čili automatickým převodem do strojového kódu.

Simulační pokus

Pokus se simulačním modelem se nazývá **simulační pokus** (*simulation experiment*). V české literatuře se často vyskytuje termín **simulační běh**, ale ten nemá žádnou analogii v literatuře ve světových jazycích. Slovo **run** (tedy běh) je navíc vhodné jako protiklad ke slovu překlad (nebo kompilace). Například „chyba zjištěná při kompilaci“ vs. „chyba při běhu“ s poznámkou, že při běhu neběží jen simulační pokusy (experimenty). Když na počítači běží simulační pokus, je nutné evidovat při něm i čas, který by dané fázi výpočtu odpovídal v simulovaném systému. Mezinárodní autority v oboru simulace doporučily, aby se hodnoty času nazývaly **simulárním časem** (*similar time*), avšak odborná veřejnost používá ne zcela přesný, ale všeobecně rozšířený termín **simulovaný čas** (*simulated time*).

Simulační studie

Posloupnost simulačních experimentů, které mají stejný účel se nazývá **simulační studie** (*simulation study*). V současnosti je obvykle prováděna jako jeden výpočet (*task*). Před každým experimentem se simulovaný čas vrátí na výchozí hodnotu a rovněž se změní sledované hodnoty způsobem, který je vrátí k výchozímu stavu. Navázání jednoho simulačního experimentu na další lze tedy nejlépe chápat jako přirovnání k tomu, že simulovaný systém se nahradí systémem zcela novým, který z historie původního systému nic „nedědí“. Jeden „běh“ odpovídá tedy jedné simulační studii, tedy sekvenci několika simulačních experimentů.

Simulační krok

V odborné literatuře se zavádí ještě termín **simulační krok** (*simulation step*). Simulační krok je časový úsek výpočtu, během něhož se nemění hodnota simulovaného času. Simulační studie se tedy skládá ze simulačních pokusů nebo experimentů a ty se dále skládají ze

simulačních kroků. Dále platí, že na začátku každého pokusu se simulovaný čas vrátí na svou výchozí hodnotu (obvykle na nulu) a s výjimkou prvního simulačního kroku každého simulačního pokusu se hodnota simulovaného času zvětší o nějaký nezáporný přírůstek. Když je tento přírůstek pro celý simulační pokus stejně velký, mluví se o **ekvidistantním simulovaném čase**, v ostatních případech se mluví o **neekvidistantním simulovaném čase**.

OTÁZKY



1. K čemu se využívá abstrakce při vytváření modelů?
2. Jaké systémy rozlišujeme v souvislosti s časem, který je ovlivňuje?
3. Co znamená pojem apriorní informace?
4. Vysvětlete rozdíl mezi modelovaným a modelujícím systémem.
5. Jaké jsou to singulární modely?
6. Co je simulace a jaký je rozdíl mezi spojitou a diskrétní simulací?
7. Popište pojem simulární čas.

SHRNUTÍ KAPITOLY



V úvodu této kapitoly jsme prezentovali základní pojmy z oblasti systémů, modelů a vědecké disciplíny, která se modely zabývá – modelování. Seznámili jsme se se základními typy systémů a modelů. Druhá část kapitoly se zabývala základními principy další vědecké disciplíny – simulací. Obezámili jsme se se základními typy simulací – diskrétní, spojitou a kombinovanou. Představili jsme pojmy jako simulační program, pokus, běh a simulární čas.

ODPOVĚDI



1. Podkapitola 2.1, str. 30 a 31.
2. Podkapitola 2.1, str. 31.
3. Podkapitola 2.2, str. 33.
4. Podkapitola 2.2, str. 33 a 34.
5. Podkapitola 2.4, str. 36.
6. Podkapitola 2.5.
7. Podkapitola 2.5, str. 39.

3 BUSINESS PROCESS MODELING NOTATION (BPMN)

RYCHLÝ NÁHLED KAPITOLY



Tato kapitola stručně představuje jednu z nejpoužívanějších modelovacích notací v oblasti analýzy podnikových procesů – Business Process Modeling Notation (BPMN). Tato notace je standardem v předmětné oblasti a využívá se při práci s řízením podnikových procesů i při jejich realizaci v podobě workflow. Obeznamíme se se základními skupinami grafických prvků BPMN a na konkrétních příkladech si vysvětlíme použití grafických symbolů v procesních modelech. Ukážeme si také příklady diagramů podnikových procesů.

CÍLE KAPITOLY



Po prostudování této kapitoly budete umět:

- Charakterizovat notaci BPMN.
- Vyjmenovat a objasnit využití skupin symbolů BPMN.

KLÍČOVÁ SLOVA KAPITOLY



BPMN, tokové objekty, BPD, aktivity, brány, artefakty, události, plavecké dráhy.

Business Process Modeling Notation (BPMN) je grafickou notací. Konkrétněji je to soubor grafických objektů a pravidel, podle kterých mohou být objekty v podnikových procesech mezi sebou spojovány. Smyslem této notace je zachycení toku práce v rámci probíhajících podnikových procesů. Za vznikem této notace stojí iniciativa BPMI (*Business Process Management Initiative*), jejímž hlavním cílem bylo vytvořit komunikační prostředek, který bude srozumitelný pro všechny účastníky procesního řízení v organizaci, podniku nebo firmě. Konkrétně tedy pro účastníky procesu, technické vývojáře, business analytiku, analytiku, kteří monitorují procesy apod. Pomocí BPMN se zmenšila komunikační mezera mezi návrhem a implementací procesu a pro mnohé implementační nástroje se stala de facto standardem pro modelování podnikového procesu. Notace BPMN je jednoduchá na pochopení a používání a zároveň nabízí možnost modelovat i komplexní podnikové procesy.



Business Process Modeling Notation je soubor principů a pravidel, který slouží pro grafické znázorňování podnikových procesů pomocí procesních diagramů. Jinak řečeno jde o standard pro modelování podnikových procesů.

Do této domény také spadá převod mezi návrhem podnikových procesů a jejich implementací v jazyce pro spouštění procesů. Jedná se o ryze technickou problematiku, které se v tomto textu detailněji věnovat nebudeme. BPMN procesní modely určují logiku průběhu procesů, které se pak prostřednictvím jednotlivých elementů převádějí do jazyka BPEL. To umožňuje manuální transformaci procesního modelu do spustitelné podoby v podobě workflow. Díky volnosti modelování v BPMN není možné udělat tento převod automaticky. Když některé nástroje tuto funkci nabízejí, je to za cenu určitých omezení při samotném modelování procesů. V současnosti je BPMN ve verzi 2.0¹⁰ (vydaná v roce 2011).

3.1 Business Process Diagram (BPD)

BPMN standard definuje jediný typ diagramu – **Business Process Diagram (BPD)**. BPD diagram je tvořen sítí grafických objektů, které jsou tvořeny zejména aktivitami a diagram zobrazuje tok práce, resp. informací mezi nimi. Jednotlivé grafické objekty jsou od sebe odlišeny, což přispívá k přehlednosti diagramu. Tvary těchto objektů jsou jednoznačně dány a je nutné je dodržovat. Můžeme ovšem volit pro ně vlastní barvy, například z odlišovacích účelů u skupiny objektů stejného typu. V některých situacích lze použít v diagramu i vlastní grafický objekt, ten se však nesmí překrývat s žádným již existujícím a rovněž by neměl ovlivňovat samotný tok procesu. Měl by jej pouze upřesňovat, či poskytovat nějaké dodatečné informace.

BPD diagramy jsou tvořené z jednoduchých elementů, které umožňují snadnou tvorbu procesních modelů. Tyto modely by pak měly být intuitivně pochopitelné většinou podnikových analytiků. Tvary elementů byly navrženy s ohledem na již používané nástroje v procesním modelování. Například aktivity se znázorňují pomocí čtyřúhelníku a rozhodnutí jsou značené diamantem. Při vývoji BPMN byl kladen důraz na to, aby pomocí něj bylo možné zachytit i komplexní podnikové procesy, které probíhají různými odděleními napříč podniky. Pro lepší zvládnutí těchto požadavků bylo navrženo několik kategorií elementů, které napomáhají snadnější orientaci v jejich základních druzích. V rámci každé ze základ-

¹⁰ Documents Associated With Business Process Model And Notation™ (BPMN™) Version 2.0 Release Date: January 2011. *Object Management Group* [online]. 2017 [cit. 2017-10-21]. Dostupné z: <http://www.omg.org/spec/BPMN/2.0/>

ních kategorií je možné upřesňovat definované elementy prostřednictvím rozšiřujících informací. Rozšíření nesmí narušit základní charakteristiky elementů, tím by snižovala jejich srozumitelnost.

3.1.1 SYMBOLY POUŽÍVANÉ V BPMN


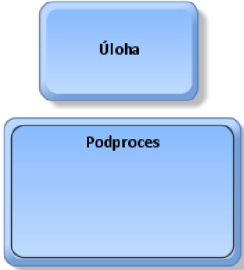

V BPMN existují čtyři základní kategorie grafických prvků:

- tokové objekty (*Flow Objects*),
- spojovací objekty (*Connecting Objects*),
- plavecké dráhy (*Swimlanes*),
- artefakty (*Artefacts*).

Tokové objekty

Do kategorie tokových objektů patří tři základní elementy. Jsou to (1) **událost** (*event*), (2) **aktivita** (*activity*) a (3) **brána** (*gateway*). Stručný popis těchto elementů a symboly jejich zápisu je zachycen v tabulce 3-1.

Tabulka 3-1: Tokové objekty.

Událost	Událost je znázorněná kruhem a reprezentuje něco, co se děje během podnikového procesu. Existují tři typy událostí – start, přechod a konec.	
Aktivita	Aktivita je reprezentovaná obdélníkem se zaoblenými rohy a reprezentuje činnost, která je vykonávána podnikem. Aktivita může být jednoduchá, nebo složená. Může být dvou typů – úloha nebo podproces. Podproces je někdy označený znaménkem plus ve středu spodní části obdélníku.	
Brána	Brána je znázorněná jako diamant a používá se na realizaci větvení podnikového procesu a spojování sekvenčních toků.	

Zdroj: vlastní zpracování

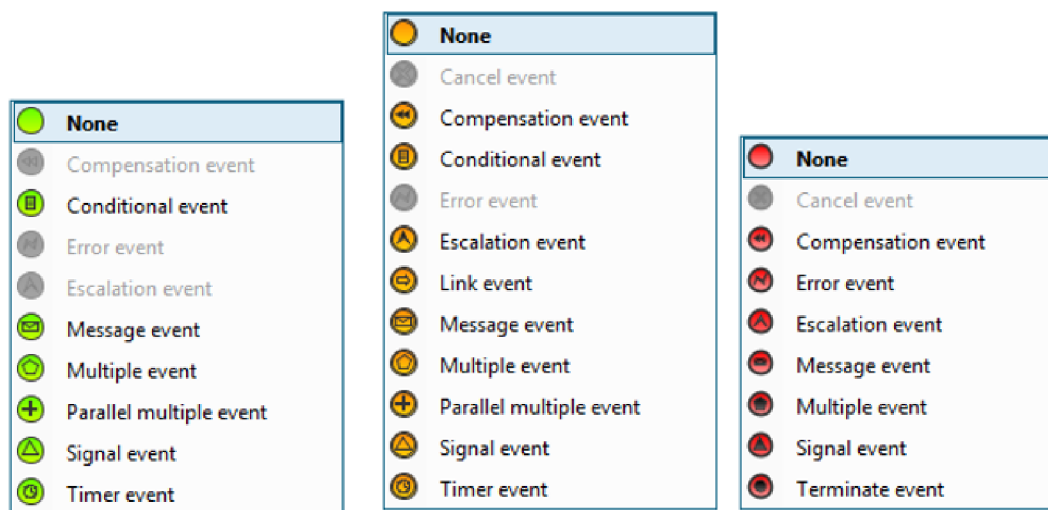
Události ovlivňují tok procesu a běžně obsahují spouštěč (příčinu) anebo důsledek (výsledek). Definované jsou tři typy **událostí**. Na začátku procesního toku se umísťuje startovací událost (*start*), v průběhu procesu používáme mezilehlé (*intermediate*) události a procesní tok ukončujeme koncovými událostmi (*end*). Pro každou událost můžeme do kruhu

umístit symbol, který upřesňuje spouštěč nebo výsledek. V BPMN je definováno více podtypů událostí. Ne všechny podtypy se mohou používat s každým typem události. Na obrázku 3-1 jsou zobrazeny všechny podtypy událostí. Mezilehlými událostmi můžeme navázat na aktivity, čímž specifikujeme jejich alternativní ukončení. – např. výskyt chyby v průběhu procesu nebo vypršení času na průběh aktivity a jiné podle uvedeného seznamu.

Události

Podtypy událostí se liší pro startovací, přechodové a koncové události (zelená, oranžová a červená barva). Jedná se např. o tyto podtypy:

- Pravidlo (*conditional event*) – splnění uvedeného pravidla (např. kladný zůstatek peněz na účtu).
- Zpráva (*message event*) – přijetí zprávy od účastníka podnikového procesu.
- Různý (*multiple event*) – ke vzniku události dochází různými způsoby.
- Paralelně různý (*parallel multiple event*) – ke vzniku události dochází různými způsoby, navíc paralelně.
- Signál (*signal event*) – ke vzniku události dochází po obdržení signálu.
- Časovač (*timer event*) – načasování spuštění procesu po uplynutí určitého časového intervalu.



Obrázek 3-1: Seznam podtypů událostí.

Zdroj: vlastní zpracování

Aktivita

Aktivita je označení pro činnost, která probíhá v podniku. Aktivita může být atomická nebo se může skládat z více úloh. Rozdělení a sloučení sekvenčního toku činností se v diagramu zabezpečuje bránou. Brána je znázorněna jako diamant, v jehož středu jsou zobrazené zpřesňující symboly. Ty specifikují, o jaký typ větvení toku se jedná. Na výběr máme

rozhodovací a paralelní dělení, které je založené na vstupních datech anebo na vyskytnuté události (obr. 3-2). K těmto dělením jsou definovány odpovídající sloučení.

BPMN standard definuje element brána jako **datový XOR, událostně řízený XOR, OR, AND** nebo jako **komplexní bránu**. Odpovídající symboly jednotlivých typů uvádí obrázek 3-2. Brána typu XOR se použije v situaci, kdy proces může pokračovat dvěma a více větvemi, z nichž ale pouze jedna bude při provádění procesu iniciována. BPMN rozlišuje dva typy XOR brány. Datová XOR brána provádí směřování toku procesu na základě jeho dat. Výběr větve je proveden na základě vyhodnocení podmínky, která je spojena s elementem brány. Standard dovoluje také označit výchozí (defaultní) cestu, která se provede, pokud žádná jiná větev nesplňuje stanovenou podmínku. Událostně řízená XOR brána má odlišnou sémantiku, protože všechny potenciálně možné následující aktivity jsou povoleny a očekávají přijetí zprávy. Také v tomto případě je vybrána pouze jedna větev tím, že nakonec dorazí odpovídající zpráva. Brána typu OR dovoluje na základě dat procesu zvolit jednu nebo více větví, které se provedou. Brána typu AND vyjadřuje paralelní provedení aktivit všech přípustných větví a nakonec komplexní brána je využita pro složitější větvení popsané v přípojeném výrazu v diagramu.

Brány



Obrázek 3-2: Typy rozhodnutí pro brány.

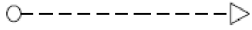

Zdroj: vlastní zpracování

Spojovací objekty

Spojovací objekty (*connecting objects*) se v BPD asociují se záměrem vytvořit základní kostru struktury podnikového procesu. Existují tři spojovací objekty, které toto zajišťují. Jejich přehled a popis nabízí tabulka 3-2. Spojovací objekty usnadňují propojení a taktéž umožňují připojení tzv. artefaktů (viz text níže).

Tabulka 3-2: Spojovací objekty.

Sekvenční tok	Sekvenční tok (<i>sequence flow</i>) je znázorněn plnou čarou, která je ukončena plnou šipkou. Zobrazuje pořadí, v jakém se v procesu vykonávají aktivity.	
---------------	--	--



Tok zpráv	Tok zpráv (<i>message flow</i>) se označuje přerušovanou čarou, která je ukončena prázdnou šipkou. Používá se na zobrazení toku zpráv mezi dvěma různými účastníky procesu, kteří posílají a přijímají tyto zprávy. Účastníky mohou být např. dva různé bloky (<i>pools</i>).	
Asociace	Asociace (<i>association</i>) je znázorněna tečkovanou čarou, která je ukončena šipkou a používá se na spojení dat, textu a jiných artefaktů s neukotvenými objekty. Asociace zobrazují vstupy a výstupy aktivit.	

Zdroj: vlastní zpracování

Plavecké dráhy

Plavecké dráhy (*swimlanes*) jsou použité i v dalších metodologiích modelování procesů (např. UML diagramy aktivit) jako pomůcka na organizování aktivit do vizuálně oddělených celků s cílem ilustrovat různou funkci těchto celků (např. různé oddělení v podniku). Notace BPMN podporuje dva typy směrodajných čar.

Tabulka 3-3: Plavecké dráhy.

Blok	Blok (<i>pool</i>) reprezentuje účastníka v podnikovém procesu. Je taktéž grafickým kontejnerem pro rozklad skupiny aktivit z jiných bloků, např. v kontextu B2B ¹¹ (<i>Business-to-Business</i>) situace.	
Dráha	Dráha (<i>lane</i>) je částí bloku, která se táhne po jeho celé délce horizontálně nebo vertikálně. Používá se na organizování a třídění aktivit.	

Zdroj: vlastní zpracování

Pool ohraničuje podnikový proces a graficky vymezuje jeho hranice. V rámci jednoho poolu se vyskytuje jenom jeden podnikový proces. Interakce mezi pooly je zajišťována pomocí zasílání zpráv. Pooly se v BPD využívají k dokumentaci dvou nebo více oddělených podnikových entit anebo účastníků procesů. Aktivity každého účastníka jsou uzavřené v jeho vlastním poolu, čím je specifikováno jeho jasné ohraničení. Předmětný pod-



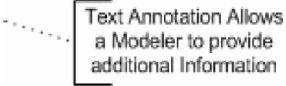
¹¹ Business-to-business (B2B) je označení pro obchodní vztahy mezi obchodními společnostmi. Významnou vlastností modelu B2B je větší důraz na logistiku a zajištění samotného obchodu, oproti důrazu na získání zákazníka, jako je tomu v případě konceptu B2C. Paradigma B2B je nejstarší složkou elektronického podnikání (e-business).

nikový proces nemůže spolupracovat s okolními procesy pomocí sekvenčních toků. K spolupráci mezi dvěma pooly se používá mechanismus zasílání zpráv. Zprávy naopak není možné použít v rámci jednoho poolu. Na obrázku 3-3 dráha rozděluje pool na menší části po celé jeho délce. Slouží na smysluplné uspořádání aktivit. Může znázorňovat oddělení, role nebo funkce podniku. Komunikace mezi jednotlivými dráhami může probíhat pomocí sekvenčních toků. Toky zpráv se však nesmí používat na komunikaci mezi tokovými objekty v dráhách jednoho poolu, jak bylo uvedeno výše.

Artefakty

Notace BPMN byla navržena tak, aby měli odborní pracovníci – modeláři a také různé modelovací nástroje volnou ruku v rozšiřování prvků základní notace a při zachycení dalších souvislostí, které mohou být potřebné ve specifické situaci. Proto může být do diagramů doplněné libovolné množství tzv. artefaktů, které jsou vhodné v kontextu modelovaných podnikových procesů. Současná verze notace BPMN má předdefinované tři typy artefaktů (tab. 3-4). Kromě toho je možné navrhnout a používat vlastní typy artefaktů, které upřesňují průběh podnikového procesu.

Tabulka 3-4: Artefakty.

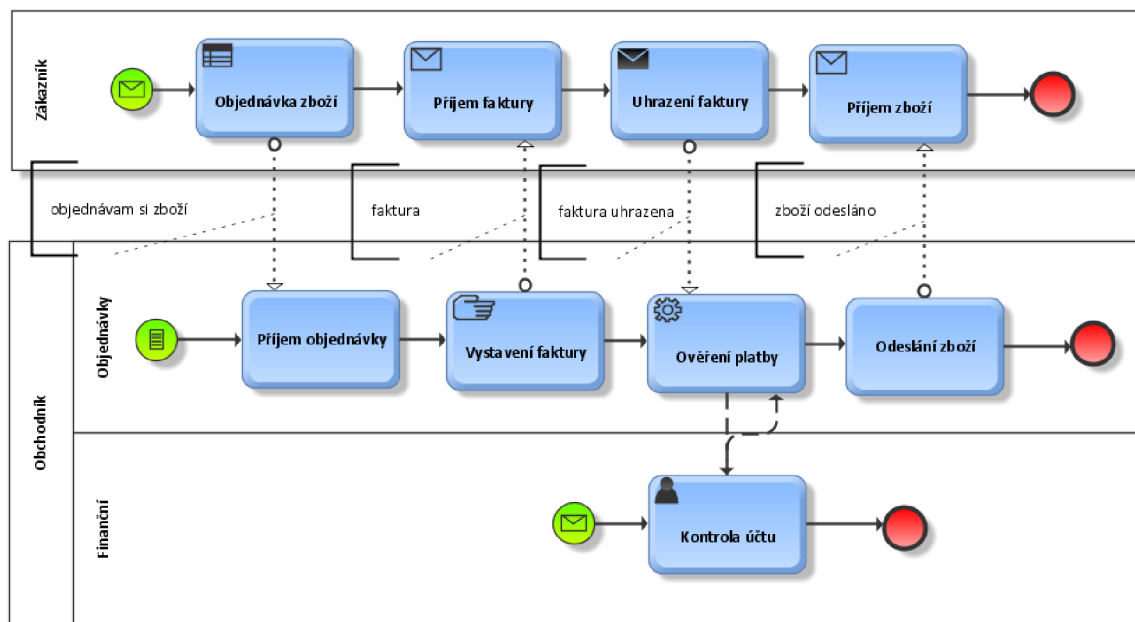
Datový objekt	Datové objekty (<i>data objects</i>) znázorňují, jaká data jsou požadovaná a produkována aktivitami. K aktivitám se připojují pomocí asociací.	
Skupina	Skupina (<i>group</i>) je znázorněna obdélníkem se zaoblenými rohy zachycena tečko-čárkovanou čarou. Používá se na analytické nebo dokumentační účely. Nemá vliv na sekvenční tok.	
Anotace	Anotace (<i>annotation</i>) poskytují účastníkům diagramu doplňující textové informace.	

Zdroj: vlastní zpracování

Datový objekt slouží na znázornění toku dat v podnikovém procesu. Pomocí něho modelujeme, jaká data jsou požadována v průběhu procesu a jaká data systém produkuje na výstupu. Datový objekt je k aktivitám připojený pomocí tzv. asociace. Graficky je reprezentovaný obdélníkem s ohnutým rohem. Skupina je znázorňovaná přerušovaným obdélníkem a používá se na analytické a dokumentační účely. Business analytik ji může použít na přehlednění diagramů, vytvořením skupin, které slučují související podnikové procesy. Vytvořené skupiny neovlivňují tok podnikového procesu.

Anotace může tvůrce diagramu využít na zachycení dodatečné textové informace. Anotace zvyšuje názornost procesního modelu a poskytuje vysvětlující textovou informaci o

elementech, které se vyskytují v diagramech. Anotace je k objektu připojená pomocí asociace.



Obrázek 3-3: Příklad využití bloku a dráhy.

Zdroj: vlastní zpracování

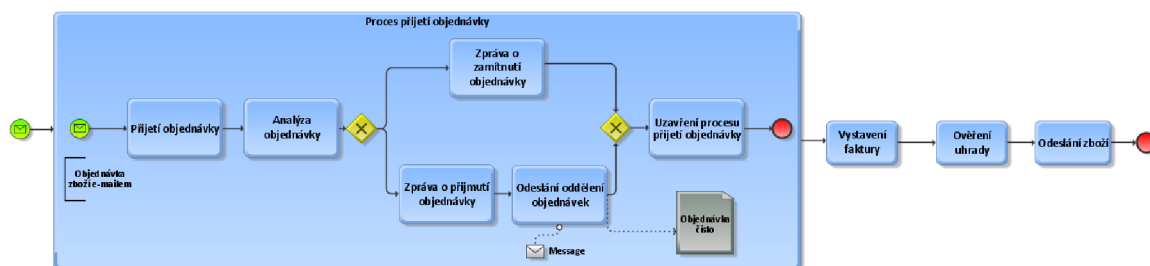
3.2 Využití BPMN

Pomocí BPMN je možné modelovat podnikové procesy na různých úrovních. Přesněji, od detailního popisu jednotlivých dílčích podnikových procesů až ke globální orchestraci procesů, které se mohou jevit jako černá skříňka¹² (*blackbox*). Notace BPMN tím může oslovit různorodé publikum, kterému podává širší spektrum informací na různých úrovních detailů. Podle míry záběru modelovaných podnikových procesů se BPD člení do dvou základních skupin:

- **Interní podnikové procesy** – jsou znázorněné v hierarchii diagramů (více kapitola 4.3). Nejvyšší úroveň obsahuje hlavní podnikový proces v podniku (někdy se označuje jako klíčový), který je prostřednictvím podprocesů podrobně specifikovaný. Nejnižší úroveň podrobně modeluje všechny aktivity, které v procesu probíhají. Příkladem procesu vysoké úrovně je obrázek 3-4. Diagram se skládá z podprocesů, které reprezentují nižší úrovně. Rozbalením některého z podprocesů je možno získat podrobný diagram jeho průběhu.
- **Kooperativní mezipodnikové procesy** – kooperativní typ diagramu znázorňuje mezipodnikové procesy (*Business to Business – B2B*). Jeho hlavním cílem je

¹² V programování jsou černé skříňky objekty, které mají pevně definované rozhraní, a jejich vnitřek je ukryt.

zachycení vztahů mezi dvěma a více podnikovými procesy. Důraz je kladen na modelování vzájemné komunikace.



Obrázek 3-4: Interní podnikový proces vysoké úrovně.

Zdroj: vlastní zpracování

Modelování podnikových procesů je disciplínou, která se používá hlavně v **BPR** (*Business Process Reengineering*). BPR je procesní přístup, kterého cílem je optimalizovat podnikové procesy tak, aby přinášely maximální efekt při minimální spotřebě podnikových zdrojů. Výsledkem BPR aktivit je řada podstatných změn v podnikových procesech, které jsou doprovázeny i změnami v organizační a kvalifikační struktuře podniku, a také ve způsobech organizace práce a řízení. Další z přístupů, se kterými se setkáváme v této oblasti je **BPI** (*Business Process Improvement*). BPI je systematický přístup, kterého snahou je pomoci podniku najít a optimalizovat její klíčové procesy tak, aby byly ve výsledku efektivnější. Notace BPMN se v obou přístupech používá jako základní technika analýzy a popisu aktivit v rámci podnikového procesu. Dále se využívá k reakci na události a tím vlastně ke komplexnímu ovlivnění chování podniku. Díky vytvořeným procesním modelům je posléze jednodušší provádět optimalizace. Informace na okraj – na popisu podnikových procesů je založena například certifikace ISO 9001.

Z pohledu více infromatického je nejdůležitějším důvodem modelování podnikových procesů situace, kdy chce podnik zavést informační systém (IS). Procesní modely v tomto případě slouží k pochopení, jaké funkcionality má IS pokrývat pro podporu dotčených podnikových procesů. Funkce nového IS přímo vyplývají z toho, jak firma vlastně funguje, jaký business provozuje, v jakém prostředí, s jakými událostmi se potýká, s jakými daty se pracuje, jaké firemní pravidla tam platí a jaké podnikové procesy realizuje. Na druhé straně se modely podnikových procesů používají např. u softwarových dodavatelů k pochopení zadání, a slouží jako podklad pro různé investiční záměry – hlavně v oblasti podpory a automatizace procesů a jsou důležitou součástí dokumentace IS. **IS je pouze nástrojem, který by měl podporovat podnikové procesy. Cílem je správně fungující a prosperující podnik.**

OTÁZKY



1. K čemu se notace BPMN využívá a jaký základní diagram používá?
2. Jaké základní kategorie prvků obsahuje notace BPMN?

3. Vyjmenujte tokové objekty v notaci BPMN.
 4. Jaké jsou důvody modelování podnikových procesů z podnikového a informačního pohledu?
-



SHRNUTÍ KAPITOLY

Tato kapitola se stručně věnovala představení notace pro modelování podnikových procesů – BPMN. Objasnili jsme základní BPD (Business Process Diagram) notace, dále pak základní kategorie grafických prvků v BPMN a symboly, které notace používá. Na příkladech jsme dokumentovali tokové a spojovací objekty, plavecké dráhy a artefakty. V závěru kapitoly jsme popsali důvody pro modelování podnikových procesů a vysvětlili jeho přínos pro návrh funkcionalit informačních systémů v podnicích.



ODPOVĚDI

1. Kapitola 3 a podkapitola 3.1, str. 39 a 40.
 2. Podkapitola 3.1.1.
 3. Podkapitola 3.1.1, str. 42.
 4. Podkapitola 3.2, str. 47 a 48.
-

4 ŘÍZENÍ PODNIKOVÝCH PROCESŮ (BPM)

RYCHLÝ NÁHLED KAPITOLY



Tato kapitola představuje v úvodu příklady podnikových procesů jako např. objednávka-hotovost, nabídka-objednávka a žádost-schválení. Jádrem kapitoly je popis životního cyklu řízení podnikových procesů (BPM) a jeho jednotlivých fází – identifikace, objevení, analýza, redesign, implementace a monitorování podnikových procesů. Při popisu typů procesů, simulace a automatizace podnikových procesů bude čtenář odkázán na předešlé kapitoly této opory. Závěrem získáte ucelený pohled na životní cyklus podnikového procesu s podporou ICT v podobě automatizovaného procesu.

CÍLE KAPITOLY



Po prostudování této kapitoly budete umět:

- Uvést příklady podnikových procesů.
- Popsat životní cyklus řízení podnikových procesů a jeho jednotlivé fáze.

KLÍČOVÁ SLOVA KAPITOLY



Řízení, podnikový proces, BPM, discovery, implementace, automatizace, BPMS.

4.1 Příklady podnikových procesů

Každý podnik bez ohledu na jeho typ nebo formu musí řídit celou řadu procesů. Mezi typické podnikové procesy, které se vyskytují ve většině podniků, bezesporu patří například:

- **Objednávka-hotovost** (*order-to-cash process*) – tento typ podnikového procesu začíná u dodavatele zboží nebo služeb. Je realizovaný tehdy, když zákazník potvrdí nákup a je ukončený tehdy, když zákazník objednané zboží nebo službu dostane a za dodávku zaplatí. Tento podnikový proces souvisí s aktivitami, které pokrývají ověření nákupní objednávky (*purchase order*), odeslání zásilky, dodání zásilky, fakturace, platba za zásilku, a nakonec ověření platby.

- **Nabídka-objednávka** (*quote-to-order process*) – tento typ podnikového procesu běžně předchází procesu order-to-cash. Podnikový proces začíná v okamžiku, když dodavatel dostane od zákazníka tzv. požadavek na nabídku (RFQ, *Request for Quote*) a končí tehdy, když zákazník potvrdí objednávku, která je založená na přijaté nabídce. Na tento proces navazuje order-to-cash a společně vytvářejí kombinovaný podnikový proces nabídka-hotovost (*quote-to-cash*).
- **Žádost-schválení** (*application-to-approval*) – tento typ procesu začíná v případě, že někdo v podniku žádá např. o udělení mimořádné odměny za práci nad rámec běžných pracovních povinností, o povolení k zahraniční pracovní cestě nebo o vyplacení benefitu apod. Je to poměrně běžný proces ve vládních a veřejných institucích. Mezi další příklady patří např. žádost o dotaci, proplacení nákladů za poplatky, nebo přijímací řízení na vysoké škole apod.

Mezi další typické podnikové procesy patří např. nákup-platba (*procure-to-pay*) nebo problém-řešení (*issue-to-resolution*). Způsob, jakým jsou podnikové procesy navrženy a následně prováděny v podnicích, ovlivňuje kvalita služeb, kterou vnímá koncový zákazník a efektivnost, s kterou jsou služby dodávány. V případě, že podnik dokáže tyto podnikové procesy zabezpečovat efektivněji než konkurence, může je realizovat i komerčně, jako službu nabízenou dalším firmám.

Přidaná hodnota

Každý podnikový proces obsahuje kromě dříve zmiňovaných událostí, aktivit, úloh a rozhodovacích bodů i **aktéry**, kterými jsou pracovníci, podniky, informační systémy, hmotné (prototypy, materiál, zboží, polotovary atd.) a nehmotné objekty (elektronické záznamy). Provedení podnikového procesu by mělo vést k jednomu nebo více výstupům a v ideálním případě by měl taktéž přinést **přidanou hodnotu** všem jeho aktérům. V takovém případě označujeme výstup podnikového procesu jako pozitivní. V případě, že podnikový proces nepřináší žádnou přidanou hodnotu aktérům, označujeme jeho výstup za negativní. Mezi všemi aktéry podnikových procesů je nejdůležitější zákazník, kterým může být zaměstnanec daného podniku (**interní zákazník**) nebo to může být **externí zákazník** mimo daný podnik.

4.2 Životní cyklus BPM

Výhody BPM

Řízení podnikových procesů (BPM, Business Process Management) je vědeckou disciplínou, která se zabývá způsoby, jakými je v podnicích vykonávána práce pro zajištění konzistentních výstupů. BPM se zaměřuje na výhody, které přinášejí příležitosti pro **zlepšení postupů**, jakými jsou práce, a hlavně podnikové procesy prováděny. V tomto kontextu může mít slovo „zlepšení“ různý význam podle toho, co je cílem podniku. Typickým příkladem zlepšování jsou například redukce chybných výrobků a výsledků, snižování nákladů na výrobu a provoz nebo optimalizace pracovního času. Tyto snahy o zlepšení mohou být jednorázového charakteru nebo mohou v podnicích probíhat kontinuálně.

BPM není jenom o způsobu, jakým jsou aktivity nebo činnosti vykonávány, ale hlavně o řízení celého komplexu činností, rozhodnutí a událostí, které se označují jako podnikové procesy (viz kapitola 1). Každá společnost, organizace, nebo podnik má podnikové procesy, které musí řídit, jak jsme uvedli na začátku této kapitoly. Porozumění těmto procesům a jejich řízení je klíčovým faktorem pro **zajištění konkurenceschopnosti a efektivnosti** podniků. Pomocí orientace na procesy (procesní řízení) podniky řídí ty aktiva, které jsou nejdůležitější, aby měly pozitivní dopad na zákazníka. BPM je kolekcí metod, principů a nástrojů pro analýzu, návrh, exekuci a monitorování těchto podnikových procesů. Jako základ pro řízení podnikových procesů se používají **procesní modely a metriky procesního výkonu**. Hlavní myšlenky a přístupy, které jsou použity v této kapitole, jsou založené na referenční publikaci Dumase et al. (2013).

První otázku, kterou by si měl každý podnik na začátku jakékoliv BPM aktivity položit je, jaké podnikové procesy chce zlepšit. Před samotným započítím o úvahách nad startem BPM aktivity pravděpodobně v podniku existovali nějaké problémy v operativě a účastníci by měli mít představu, jakých podnikových procesů se to týká. Jestli např. manažer logistiky ve firmě vidí, že se mu hromadí dodávky na skladě a přepravce je nestíhá distribuovat, je zřejmé, že se problémy týkají externího dodavatele a v první řadě by se mělo určit, kde daný podnikový proces začíná a kde končí, jestli se jedná o stejného dodavatele logistických služeb, nebo konkrétního zboží apod. Tyto otázky souvisejí se zkušenostmi podniku s procesním řízením v minulosti. Kdyby se jimi podnik již zabýval, je pravděpodobné, že by měl vytvořený seznam podnikových procesů a alespoň částečně definován jejich rozsah. V podnicích, které se BPM aktivitami doposud nezabývali, musí BPM tým v první řadě identifikovat problematické podnikové procesy, určit jejich rozsah a vztahy mezi nimi. Tato počáteční fáze BPM činnosti se nazývá **identifikace procesů** (*process identification*). Tato fáze vede k vytvoření tzv. **procesní architektury** (*process architecture*), která je obvykle reprezentovaná kolekcí podnikových procesů různého druhu a vztahy mezi nimi. Obecným pravidlem BPM činnosti je, aby analyzované podnikové procesy vedly ke konzistentním pozitivním výstupům a aby přinášeli **přidanou hodnotu** podniku nebo organizaci především ve vztahu k zákazníkům. Měření této hodnoty je klíčovým prvkem BPM aktivity. Není možné něco řídit nebo kontrolovat, když to neumíme měřit. Před detailní analýzou podnikových procesů je nutné stanovit si **metriky procesního výkonu** (*process performance metrics*). Tyto procesní metriky budou určovat stav procesu. Konkrétní typy metrik mohou souviset s náklady např. režijní náklady, mzdy, externí služby, poplatky apod. Další typy metrik mohou být vázané na čas, např. průměrný čas, který uplyne při vyřizování reklamace. Tato metrika se obecně označuje jako **cyklický čas** (*cycle time*). Třetí typ metrik se týká kvality, např. **chybovost** (*error rate*). Chybovost je procentuálně podíl neúspěšně ukončených instancí procesu z celkového počtu instancí. Může to být například celkový podíl nedoručených zásilek z důvodu nesprávně vyplněné adresy zákazníka. Je zřejmé, že podnik by se měl snažit takové chybovosti vyvarovat a minimalizovat ji. Identifikace metrik procesního výkonu a výkonnostních cílů je obecně součástí identifikační fáze BPM činnosti i když je možné ji v některých případech posunout do fází pozdějších.

Po identifikaci předmětných podnikových procesů a určení metrik procesního výkonu je další fází BPM aktivity pochopení detailů podnikových procesů. Tato fáze se označuje **objevování procesů** (*process discovery*). Hlavním výstupem této fáze je jeden nebo více procesních modelů, které zachycují současný skutečný stav procesů, jedná se o tzv. **as-is procesní modely**. As-is procesní modely znázorňují myšlení lidí v podniku o tom, jakým způsobem je vykonávána jejich rutinní pracovní náplň. Tyto procesní modely by měli být jednoduše pochopitelné a měli by se využívat při komunikaci mezi členy BPM týmu. Podnikové procesy je možné modelovat např. pomocí přirozeného jazyka ve formě psaného textu. Jenomže takovýto charakter popisu často vede k mylné interpretaci, a proto se při modelování běžně používají diagramy. Diagramy umožňují jednoduchým způsobem zachytit průběh podnikových procesů. V případě, že se při tvorbě diagramů bude využívat nějaká konkrétní notace modelování podnikových procesů (např. BPMN v kapitole 3), která je snadno pochopitelná pro všechny členy procesního týmu, tak prostor pro nedorozumění je minimální. Také v tomto případě se může použít forma textového popisu. Dokonce kombinace diagramů a textu se jeví jako nejvhodnější.

Po detailním pochopení as-is podnikových procesů je další fází BPM činnosti **identifikace a analýza problémů** v těchto procesech. Jedním z potenciálních problémů procesu vyskladnění zásilek může být příliš vysoký cyklický čas. Toto může způsobovat různá zpoždění v úlohách, které se musí splnit při dodávkách zboží zákazníkům. Pro analýzu takového problému potřebuje analytik dostat informace o časech, které konkrétní pracovník strávil přímo svou prací na jednotlivých rutinních úkolech a časech, kdy čekal na výsledky práce někoho jiného (*idle time*). Tento čas se nazývá **čekací čas** (*waiting time*). Analytik potřebuje také informace ohledně množství **opakované práce** (*rework*) v podnikovém procesu. Opakovaná práce znamená, že se jedna nebo více úloh opakuje, protože se v podnikovém procesu něco nepovedlo. Procesní analytik by v podobných případech potřeboval vědět, v kolika (procentuálně) případech k tomuto dochází. Jestliže analytik tuto informaci má, může využít různé BPM techniky pro identifikaci příčin vysokého cyklického času a při určení změn, které se musí v podnikovém procesu realizovat pro jeho snížení.

Další problémy, s kterými by měl být procesní analytik obeznámen, se týkají případů, kdy instance procesů končí negativním výstupem. To znamená, že se v podnikovém procesu něco nepovede, např. faktura se nevystaví, objednávka nebude vybavená apod. Při analýze takového typu problému musí procesní analytik identifikovat, jak často k těmto negativním výstupům dochází, ale hlavně proč k nim dochází. Častým důvodem bývá nedorozumění v komunikaci mezi účastníky podnikového procesu, chybami, které jsou způsobené daty (nesprávné podkladové materiály apod.). Identifikaci, klasifikaci a detailním pochopením příčin negativních výstupů může procesní analytik naleznout řešení problému v podnikovém procesu. Identifikace a diagnostika problémů a příležitostí na zlepšení podnikových procesů se označuje jako **analýza procesu** (*process analysis*).

Po analýze a případné kvantifikaci problémů v podnikových procesech začíná další fáze životního cyklu BPM – **identifikace a analýza potenciálních změn** problémů. Procesní analytik v této fázi přemýšlí nad možnostmi jak problémy, které byli identifikovány, vyřešit.

Musí myslet na to, že jakékoliv změny, které by vyřešili jeden problém, mohou vést ke vzniku dalších problémů. Například při řešení problému s plným skladem by mohlo dojít k záměně balíčků a nesprávnému doručení koncovému zákazníkovi. To by mohlo ve svém důsledku vést k vyšší míře negativních výstupů vyskladňování. Změna podnikového procesu není nikdy jednoduchá. Lidi jsou navyklí pracovat určitým způsobem a každou změnu vnímají negativně. Často se změny v podnikových procesech týkají taky změn v IS podniku, který se používá při vykonávání příslušného podnikového procesu. Taková změna bývá často finančně náročná a může se kromě předmětného podniku nebo organizaci týkat taky např. dodavatelů nebo odběratelů.

Analytik, který má informace o jednom nebo více problémech v podnikovém procesu může navrhnout jeho **redesign** (znovu navržení) – jeho novou verzi, která řeší identifikované problémy v as-is procesním modelu. Vzniká nová verze podnikového procesu, která se označuje jako *to-be procesní model*, tj. vylepšená verze skutečnosti. To-be podnikový proces je hlavním výstupem další fáze životního cyklu BPM. Důležitou charakteristikou této fáze je, že redesign a analýza bývají přepojené, protože analytik musí mít k dispozici dostatek analytických informací při výběru nejlepší redesignové možnosti.

**Redesign
procesů**

Po znovu navržení podnikového procesu a vytvoření to-be procesního modelu je nutné navržené změny práce a změny v IS i realizovat. Tato fáze se nazývá **implementace procesu** (*process implementation*). Obecně se implementace procesů skládá ze dvou vzájemně se doplňujících oblastí – **organizační change management** (*organizational change management*) a **automatizace procesů** (*process automation*). Organizační change management se týká všech činností, které jsou potřebné pro změnu způsobu práce všech účastníků podnikového procesu. Tyto činnosti zahrnují:

**Implementace
procesů**

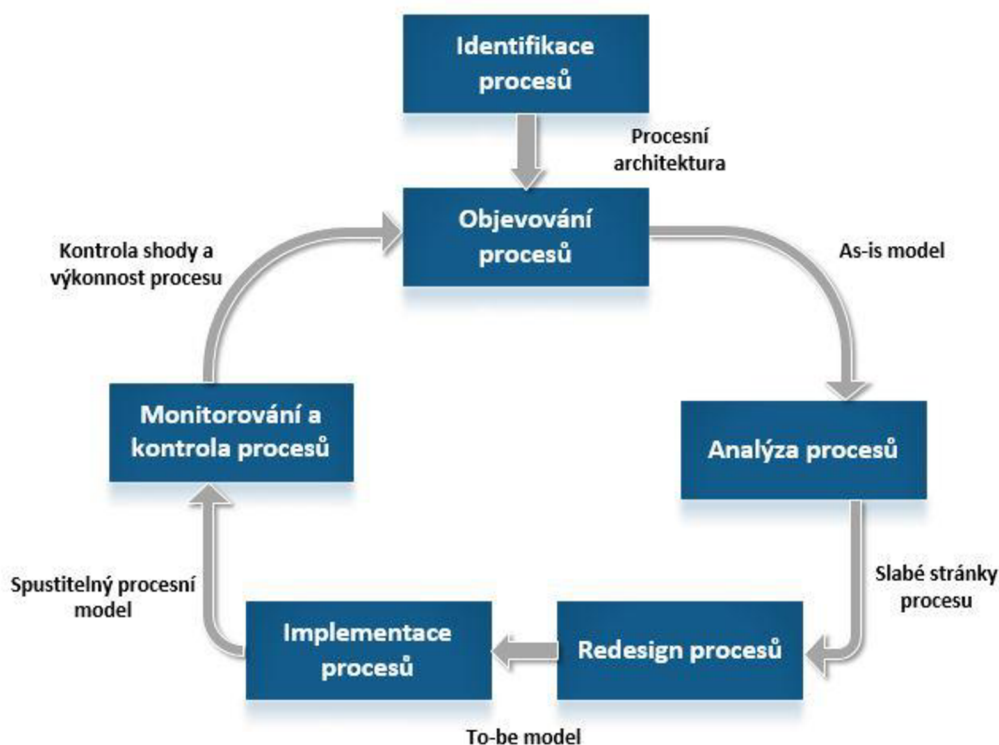
- Vytvoření plánu change managementu, který bude všechny zúčastněné strany informovat, kdy a jaké změny nastanou a jaká přechodná opatření nastanou, aby bylo možné se vyhnout problémům při přechodu na to-be procesy.
- Vysvětlení přínosů zaváděných změn pro podnik nebo organizaci.
- Školení uživatelů na nový způsob práce a monitorování změn pro zajištění hladkého přechodu na to-be procesy.

Automatizace podnikových procesů sestává z nastavení nebo implementace ITC systému pro podporu to-be procesů. Tento systém by měl sloužit pro podporu vykonávání jednotlivých kroků podnikového procesu jeho účastníky. Účastníkům procesu jsou přiřazeny úlohy; jsou jim poskytovány informace, které potřebují při plnění úloh; je jim poskytována pomoc při stanovení priorit při výkonu práce; mají k dispozici automatizovanou kontrolu práce a výstupů s cílem implementovat co nejvíce automatizovaných úloh.

Po realizaci to-be podnikových procesů nastává fáze, kdy je potřebné vyhodnotit jejich fungování a implementovat změny těch částí procesů, které nesplňují očekávání. Proto by měl být podnikový proces průběžně monitorován a monitorovací data by měli být průběžně

**Monitoring
a kontrola
procesů**

analyzována pro identifikaci nedostatků a pro lepší průběh podnikových procesů. Tyto činnosti zajišťuje **monitoring a kontrola procesu** (*process monitoring and controlling*) jako další fáze BPM životního cyklu. Řízení podnikového procesu vyžaduje neustálou kontrolu, protože jeho nedostatek způsobuje pokles jeho výkonu, případně až degradaci. Podnikový proces se musí neustále přizpůsobovat technologiím, konkurenci a především požadavkům zákazníků. To je důvod, proč jsou všechny fáze životního cyklu BPM realizovány v kruhu (obr. 4-1). Monitoring a kontrola procesů se vrací zpět na začátek životního cyklu BPM k objevování a analýze podnikového procesu.



Obrázek 4-1: Interní podnikový proces vysoké úrovně.

Zdroj: upravené podle Dumase et al. (2013).

BPM tým

Životní cyklus BPM pomáhá pochopit význam technologií v procesním řízení. Klíčovou technologií procesního řízení je ICT. Systémoví inženýři, kteří zastupují ICT v BPM činnosti podniku nebo organizace jsou součástí většího celku. Jestli chceme z BPM využít maximum musí být členy BPM týmu kromě systémových analytiků také procesní analytici, kteří chápou příslušnou business doménu a dokážou využít synergický efekt znalostí z obou světů. BPM projekt má interdisciplinární charakter a spolupracují na něm manažeři různých organizačních úrovní podniku nebo organizace, dále běžní a administrativní pracovníci, analytici (systémoví, procesní, business) a celé ICT týmy.

4.3 Identifikace procesů

DEFINICE



Identifikace procesů začíná s výskytem business problému/ů. Podnikové procesy, které se spojují s daným problémem, jsou identifikovány, je určen jejich rozsah a vztahy mezi nimi. Výstupem této fáze BPM je revidovaná nebo nová **procesní architektura**, která znázorňuje celkový pohled na všechny participující podnikové procesy a jejich vzájemné vztahy v podniku nebo v organizaci. Paralelně se někdy s těmito činnostmi identifikují také **procesní metriky výkonnosti**. Procesní architektura slouží jako framework pro definici rozsahu a priorit procesního modelování a celého BPM projektu.

V této podkapitole jsou stručně popsány metody, které se využívají při identifikaci podnikových procesů ve dvou etapách – **označovací** (*designation*) a **vyhodnocovací** (*evaluation*). Označovací etapa se zabývá definicí počátečního seznamu podnikových procesů. Vyhodnocovací etapa má za úkol najít vhodná kritéria po definici priorit těchto procesů. Výstupy obou etap jsou následně využity u tvorby procesní architektury.

4.3.1 OZNAČOVACÍ ETAPA

Jestli je podnik jenom na začátku přechodu na procesně orientovaný, první jejich aktivitou je správné označení všech probíhajících podnikových procesů. Základním problémem je hierarchicky charakter podnikových procesů a určení toho, co tvoří jeden nezávislý podnikový proces a které jeho části jsou považované za součásti jiného podnikového procesu. Rozdělení podnikových procesů jsme se věnovali v kapitole 1.2 tohoto textu.

Myšlenkou procesního řízení je aktivně řídit podnikové procesy v organizaci a tím uspokojovat potřeby zákazníků. Když **označíme procesy**, které jsou příliš rozsáhlé, tak bude velice obtížné je řídit ve smyslu rychlosti akce a jejich rozsahu. Některé procesy mohou být příliš komplexní a jejich odstavení by mohlo způsobit ochromení činnosti podniku nebo organizace. Proto by měl být počet a rozsah označených procesů pro BPM aktivity kompromisem mezi dopadem a schopností tyto podnikové procesy řídit.

V případě velkých diverzifikovaných podniků je obvyklé identifikovat stovky podnikových procesů. V případě menších podniků jsou to desítky procesů. V této fázi BPM činnosti nejde zúčastněným stranám o přesnou specifikaci procesů a jejich vzájemných vztahů, ale v první řadě o jejich pochopení a seznámení se se situací ohledně jejich rozsahu, propojení a výkonnosti. V této aktivitě jsou přínosné referenční procesní modely.

4.3.2 VYHODNOCOvací ETAPA

Pro vyjádření důležitosti podnikových procesů se používají různá kritéria. Mezi nejpo-
užívanější patří:

- **Důležitost** (*Importance*) – posuzuje se strategický přínos každého podnikového procesu. BPM aktivita by se měla soustředit na ty podnikové procesy, které mají největší vliv na dosahování strategických cílů podniku nebo organizace.
- **Disfunkce** (*Dysfunction*) – posuzuje se kondice každého podnikového procesu. Procesy s největšími problémy by měli mít největší přínos z hlediska přínosů BPM aktivity.
- **Proveditelnost** (*Feasibility*) – posuzuje se do jaké míry je každý podnikový proces citlivý na aktivity procesního řízení. Obecně platí, že jestli se předmětné podnikové procesy týkají politických nebo kulturních aspektů, tak se stávají překážkou procesního řízení. Procesní řízení by se mělo soustředit především na takové procesy, od kterých je možné očekávat přínosy.

Maturity

Při posuzování těchto kritérií se předpokládá, že máme k tomu dostatek informací. Podnik nebo organizace musí mít jasno, jaké je jejich strategické směřování a to alespoň na abstraktní úrovni. Více podniků považuje za strategické např. schopnost změnit sortiment nabízeného zboží v situaci, kdy zákazníci změní preference. Při posuzování disfunkcí podnikových procesů je důležité, aby podnik byl schopen kvantitativně vyhodnotit výkonnost jednotlivých podnikových procesů. U kvalitativního posouzení podnikových procesů je to snadnější. Další informace o disfunkci podnikových procesů můžeme získat od zákazníků např. pomocí dotazníků. V oblasti proveditelnosti je už celkem běžnou praxí, že podniky procházejí různými programy na zlepšení výkonnosti v různých dimenzích (např. telekomunikace, ICT, technologie obecně, apod.). Když se změny realizují příliš často, stává se, že management společnosti, ale i běžní zaměstnanci reagují negativně na časté programy zlepšování. U procesního řízení je důležitý pozitivní postoj vrcholného managementu podniku. Sofistikovanější přístup k vyhodnocovací fázi je založený na **vyspělosti** (*maturity*) podniku.

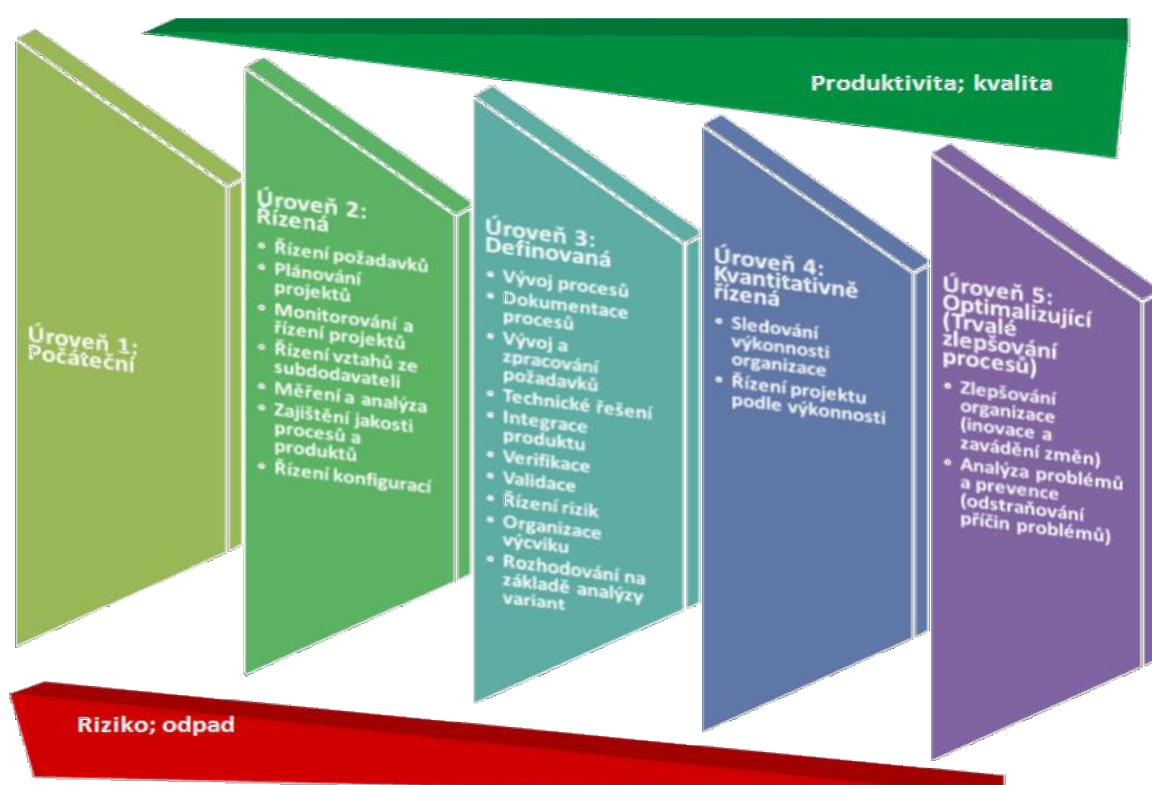


PRO ZÁJEMCE

Nejznámějším frameworkem na vyhodnocení vyspělosti procesního řízení je *Capability Maturity Model Integrated* (CMMI). Rozsah procesních oblastí a stupeň jejich podpory určuje vyspělost podniku na pěti CMMI úrovních (obr. 4-2):

- **Úroveň 1** (*Initial*) – procesy v podniku běží náhodně a bez jasné definice. Neexistuje jejich kontrola.
- **Úroveň 2** (*Managed*) – v podniku probíhá plánování, monitorování a kontrola projektů. Jsou nasazeny analýza a metriky.

- **Úroveň 3 (Defined)** – procesy jsou definovány a společnost nabízí školení účastníkům procesů, aby byli schopní se podílet na analýze a dokumentování podnikových procesů.
- **Úroveň 4 (Quantitatively managed)** – výkonnost procesů v podniku je sledována. Využívají se kvantitativní techniky.
- **Úroveň 5 (Optimizing)** – funguje management výkonnosti procesů s analýzou příčin a následků (*causal analysis and resolution*).

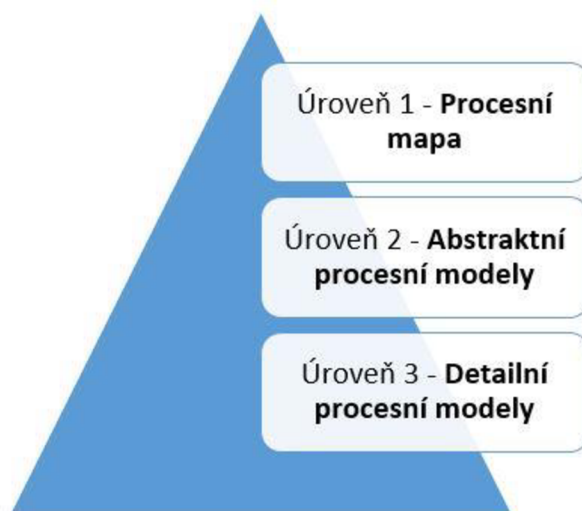


Obrázek 4-2: CMMI.

Zdroj: vlastní zpracování

4.3.3 PROCESNÍ ARCHITEKTURA

Procesní architektura je konceptuální model, který znázorňuje procesy v podniku a explicitně stanovuje vztahy mezi nimi. Tyto vztahy jsou definované v obou směrech. Procesy mohou být ve vztahu **konzument-producent** – jeden proces produkuje výstup, který je zároveň vstupem do druhého procesu. Procesní architektura definuje různé úrovně detailů (obr. 4-3).



Obrázek 4-3: Procesní architektura.

Zdroj: upravené podle Dumase et al. (2013).

Část procesní architektury, která reprezentuje úroveň 1, se nazývá **procesní mapa** (*process landscape model*). Znázorňuje hlavní podnikové procesy na velice abstraktní úrovni. Každá z částí procesní mapy odkazuje na konkrétní podnikový proces na úrovni 2 procesní architektury. Tato úroveň znázorňuje procesy na jemnější úrovni granularity, ale pořád abstraktním způsobem. Každá část **abstraktních procesních modelů** na úrovni 2 odkazuje na **detailní procesní modely** úrovně 3. Detailní procesní modely obsahují všechny detaily procesů jako je předávání aktivity (*control flow*), účastníky podnikových procesů, datové vstupy a výstupy.

Typy případů a business funkce

Pro definici procesní architektury se využívají různé přístupy, jako např. přes **typy případů** (*case types*) a **business funkce** (*business functions*). Případy představují běžné situace, se kterými podnik pracuje. Je to produkt nebo služba, kterou podnik nabízí zákazníkům a klasifikuje ho na základě mnohých parametrů, např. typ (spotřební zboží, stroj, software, apod.) nebo komunikační kanál (např. Internet, telefon, e-mail, apod.). Funkce je jednoduše řečeno to, co podnik dělá např. prodej, výroba, distribuce, apod. Pro vytvoření procesní architektury se doporučuje postupovat systematicky:

1. Identifikovat typy případů (typ produktu, typ služby, kanál a typ zákazníka).
2. Identifikovat funkce pro typy případů (např. pomocí řízených rozhovorů se zaměstnanci se zaměřením na funkce jednotlivých oddělení).
3. Navrhnout případově-funkční matici (různé funkce podniku pro různé typy zákazníků).
4. Identifikovat procesy.

4.4 Objevování procesů

DEFINICE



Objevování procesů neboli as-is procesní modelování dokumentuje současný stav všech relevantních podnikových procesů. Výstupem této fáze BPM činnosti je jeden nebo několik as-is procesních modelů. Důležitou částí této aktivity je sběr dostatečného množství informací o skutečném stavu procesů.

Vyhledávání informací o podnikových procesech je časově náročnou aktivitou a vyžaduje nastavení způsobů, jakým budeme informace efektivně sbírat. Pro vyřešení těchto problémů popíšeme nyní čtyři kroky objevování podnikových procesů:

1. **Definovat nastavení** – výběr a sestavení týmu, který bude pracovat na podnikovém procesu.
2. **Sběr informací** – pochopení podnikového procesu s využitím různých exploračních metod pro sběr informací o podnikovém procesu.
3. **Modelování** – organizace aktivit, potřebných pro vytvoření modelu s využitím mapování procesu systémovým způsobem.
4. **Zajištění kvality procesního modelu** – výsledný procesní model musí plnit nastavené kritéria kvality pro dosažení důvěry v procesní model.

Za analýzu a modelování podnikového procesu běžně odpovídá procesní analytik, který často není obeznámen se všemi detaily procesu. Pro získání dostatečných informací o podnikovém procesu se využívají doménoví experti, kteří zastřešují informace z různých perspektiv podnikových procesů. Nejčastěji se doménovým expertem stává přímo účastník procesu, ale může to být i manažer nebo vlastník procesu, případně dodavatel nebo zákazník. Vzhledem k tomu, že doménovými experty jsou různé osoby, objevování procesů musí čelit výzvám především v oblastech jako **fragmentové procesní znalosti, myšlení v případech a nezkušenost s procesními modelovacími jazyky**. První výzva souvisí s faktem, že účastníci podnikového procesu mají detailní znalosti ze svoji částí procesu, ale jejich celkový pohled na proces je obecný. Druhá výzva odráží zkušenost, že experti nemají problém nad úvahami o konkrétních případech a aktivitách v procesu, ale zobecnit průběh podnikového procesu je pro ně problém. Třetí výzva se týká problému, že doménoví experti neumí číst procesní modely, které vytvořil někdo jiný.

Při objevování podnikových procesů se používají různé techniky. Mezi nejdůležitější patří:

- **Objevování založené na důkazech** (*evidence-based discovery*)
- **Objevování založené na rozhovorech** (*interview-based discovery*)
- **Objevování založené na workshopech** (*workshop-based discovery*)

4.4.1 OBJEVOVÁNÍ ZALOŽENÉ NA DŮKAZECH

Objevování založené na důkazech používá různé druhy důkazů pro analýzu existujících podnikových procesů. Mezi tři základní metody patří **analýza dokumentů**, **pozorování** a **automatické objevování procesů**.

Analýza dokumentů vychází z předpokladu, že v podniku nebo v organizaci je běžně dostupná dokumentace existujících podnikových procesů. Tato dokumentace nezřídka vykazuje více problémů. Často se stává, že dokumentace se nevyskytuje v procesně orientované podobě, např. v podobě organizačních diagramů. Dalším problémem bývá granularita dokumentů nebo důvěryhodnost dokumentů. Při **pozorování** přímo sledujeme průběh jednotlivých případů a snažíme se pochopit, jakým způsobem proces probíhá. Procesní analytik se může přímo stát zákazníkem a podnikový proces spouštět, nebo ho může pozorovat pasívně. Problémem je, že zaměstnanci nejsou zvyklí, aby je někdo při práci pozoroval, a může se stát, že nebudou práci vykonávat běžným způsobem. **Automatické objevování procesů** využívá enormní vývoj ICT v podobě operativní podpory podnikových procesů pomocí IS. Informační systémy zaznamenávají všechny operace, které v podnicích probíhají prostřednictvím tzv. logovacích souborů o událostech. Data o událostech mohou být ukládány takovým způsobem, který umožňuje jejich rekonstrukci v podobě procesních modelů pomocí tzv. process miningových metod (dolování procesů). Více se této problematice věnujeme na konci opory. Výhodou automatického objevování procesů je, že logovací soubory zachycují realizaci podnikového procesu velmi přesně včetně času. Nevýhodou je, že struktura některých dat neumožňuje jejich zpracování nebo jsou generované procesní modely nepřehledné a komplexní. V takových případech se používají techniky filtrování a clusteringu.

4.4.2 OBJEVOVÁNÍ ZALOŽENÉ NA ROZHOVORECH

Objevování založené na rozhovorech je metoda, která je založena na rozhovorech s doménovými experty. Procesní analytik si pro rozhovory vybírá ty doménové experty, kteří mají specifické znalosti k určité části podnikového procesu. Používají se dvě techniky vedení rozhovorů. První technikou je začít rozhovor od konce podnikového procesu (*downstream*) – od produktu a postupně se dostat až na začátek procesu. Druhou technikou je postupně odhalovat tok procesu (*upstream*). Obě techniky mají své výhody a umožňují postupně odhalovat jaké vstupy nebo výstupy jsou potřebné pro realizaci dalších podnikových procesů a jaké jsou mezi nimi vztahy. Procesní analytik následně vytváří prvotní procesní model – **draft procesní model**, který se následně iterativně vylepšuje. Doporučuje se

forma volného rozhovoru, ne strukturovaného interview. Takovýto postup je časově velmi náročný vzhledem k iteracím, které v něm probíhají.

4.4.3 OBJEVOVÁNÍ ZALOŽENÉ NA WORKSHOPECH

Objevování založené na workshopech patří taky mezi příležitosti, jak získat hodnotné informace o podnikových procesech. Podněty z diskuse mohou být přímo přidány do procesního modelu. Kromě běžných účastníků workshopu se ho účastní **facilitátor**, který vystupuje jako moderátor, aby nedošlo k tomu, že diskuse bude odbíhat od tématu a **operátor**, který vkládá detaily do procesního modelu. Dále by měl být účastný procesní analytik a vlastník procesu. Zajištění účasti a hladký průběh workshopu si vyžaduje důslednou přípravu a plánování. Workshopy se opakují tři až pět krát za sebou. Diskuse probíhá nad aktivitami v podnikovém procesu, a když dojde ke shodě nad první aktivitou procesu, pokračuje se další až na konec procesu.

Třetí fází objevování podnikových procesů je modelování. Modelování podnikových procesů jsme se podrobněji věnovali v kapitolách 2 a 3 této opory. Výsledkem modelování je vytvořený procesní model, který musí mít určitou kvalitu (čtvrtá fáze objevování procesů). Rozeznáváme **syntaktickou**, **sémantickou** a **pragmatickou** kvalitu modelů. Pro zajištění syntaktické kvality se používá **verifikace**. **Validace** se používá pro zajištění sémantické kvality a **certifikace** pro pragmatickou kvalitu.

Modelování a kvalita

4.5 Analýza procesů

DEFINICE



Při **analýze procesů** jako další fázi BPM činnosti, identifikujeme problémy, které vyplývají z as-is procesních modelů. Snažíme se tyto problémy kvantifikovat pomocí metrik procesního výkonu. Výsledkem této fáze je strukturovaný seznam problémů, který je seřazený podle jejich dopadu na celkový průběh podnikových procesů nebo podle výkonu, který musíme vynaložit na jejich řešení.

4.5.1 KVALITATIVNÍ ANALÝZA PROCESŮ

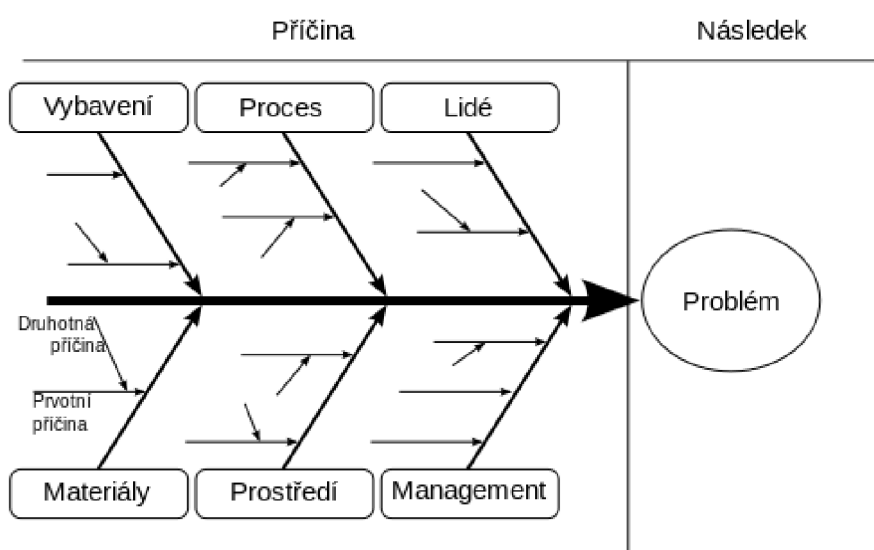
Kvalitativní analýza se soustředí v první řadě na štíhlost (*lean*) podnikových procesů identifikací nadbytečných částí procesů s cílem je odstranit. Analyzuje slabé části podnikových procesů. Tyto části způsobují problémy a negativně ovlivňují výkonnost procesů. S využitím kvalitativní analýzy analyzujeme dopady problémů v podnikovém procesu s úmyslem použít získané informace v další fázi BPM aktivity – při redesignu podnikových procesů.

Klasifikace hodnoty

Value-added analýza, neboli analýza přidané hodnoty se skládá ze dvou kroků – **klasifikace hodnoty** a **eliminace nadbytku**. Analýza přidané hodnoty se snaží nalézt nepotřebné kroky v podnikovém procesu a eliminovat je. Krok v tomto případě značí úlohu v podnikovém procesu, nebo její část, případně přecházení toku procesu mezi dvěma úlohami. Platí přitom, že všechny kroky, které přináší hodnotu zákazníkovi, jsou *value-adding* (VA). Některé kroky hodnotu nepřináší, ale jsou důležité pro business samotný. Příkladem může být kontrola údajů o zákazníkovi, která jemu samotnému nic nepřináší, ale pro podnik to znamená správné údaje při doručování zásilek a menší podíl negativních výstupů procesu doručování. Takové kroky v podnikovém procesu označujeme jako *business value-adding* (BVA). Zákazník za ně není ochoten platit, ale pro podnik, který podnikový proces realizuje, přináší užitek. Třetím typem kroku, který nepřináší hodnotu zákazníkovi, se nazývá *non-value-adding* (NVA). Po identifikaci a klasifikaci kroků v podnikovém procesu můžeme pokračovat odstraněním nadbytečných úloh (*eliminating waste*). Obecným pravidlem je, že by jsme měli eliminovat, resp. minimalizovat NVA kroky v procesu. Některé NVA kroky mohou být eliminovány automatizací podnikových procesů a souvisí taky s odstraňováním účastníků procesů.

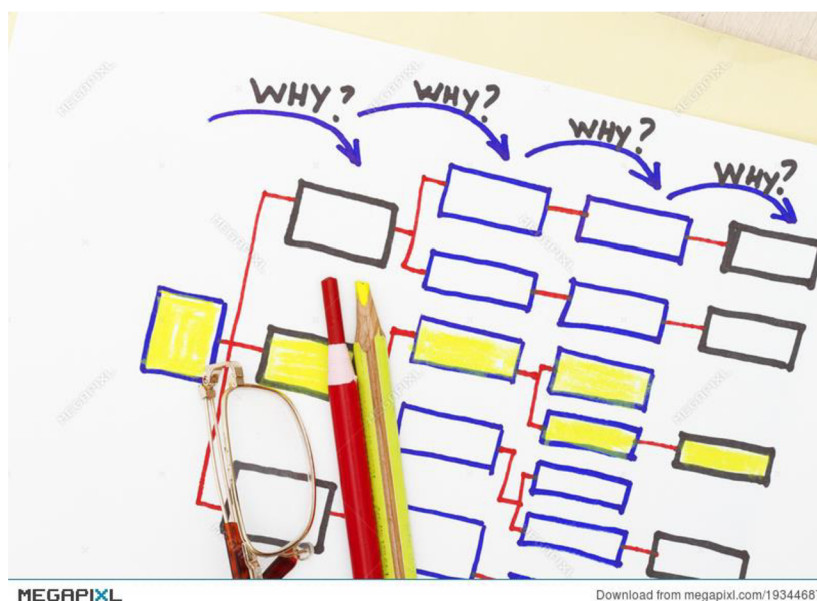
Root cause analýza

Při analýze podnikových procesů platí, že i relativně dobrý podnikový proces může být lepší. V každém procesu se mohou vyskytovat odchylky, chyby, nedorozumění, nepotřebné kroky a jiné formy nadbytku při rutinním běhu podnikového procesu. Úkolem procesního analytika je identifikovat a dokumentovat tyto problémy rozhovory a pozorováním přímo na místě a bude přicházet do interakce s mnoha účastníky procesu, přičemž každý z nich může mít jiný názor na výkonnost celého podnikového procesu. **Root cause analýza** je skupinou technik, které se soustředí na identifikaci a porozumění příčinám problémů nebo nečekaných událostí. Mezi nejvíce používané techniky root cause analýzy patří tzv. **diagramy příčin a následků** (*cause-effect diagrams*, obr. 4-4) a **proč diagramy** (*why-why diagrams*, obr. 4-5).



Obrázek 4-4: Diagram příčin a následků.

Zdroj: https://cs.wikipedia.org/wiki/Diagram_p%C5%99%C3%AD%C4%8Din_a_n%C3%A1sledk%C5%AF#/media/File:Is-hikawa_Fishbone_Diagram_cz.svg



Obrázek 4-5: Why-why diagram.

Zdroj: <https://images.megapixl.com/1934/19344687.jpg>

4.5.2 KVANTITATIVNÍ ANALÝZA PROCESŮ

Kvantitativní analýza procesů nabízí hodnotné pohledy dovnitř podnikových procesů, přičemž se zaměřuje na výkonnostní metriky procesů, jako např. cyklický a čekací čas, odchylky, apod. Mezi nejdůležitější techniky kvantitativní analýzy patří **analýza toků** (*flow analysis*), **analýza front** (*queueing analysis*) a **simulace**.

Každý podnik by chtěl mít podnikové procesy levnější, lepší a rychlejší. Z toho plynou základní **výkonnostní dimenze**, kterými jsou **čas**, **náklady** a **kvalita**. Někdy se k nim přidává i reakce na změny – **flexibilita**. Každá ze jmenovaných dimenzí může být rozdělena na procesní výkonnostní metriky, které též nazýváme *key performance indicators* (KPIs). Procesní výkonnostní metriky jsou kvantifikovatelné. Poznáme např. více druhů nákladů – výrobní náklady, režijní náklady, fixní a variabilní náklady, operativní náklady, apod. Každý z těchto typů nákladů můžeme blíže specifikovat pomocí agregační funkce, např. průměr, rozptyl, minimum, maximum, apod. Čas je nejčastěji analyzovanou dimenzí. **Cyklický čas** je čas průběhu jednoho podnikového procesu (neboli případu) od začátku do konce. Redukce cyklického času (maximálního nebo průměrného) bývá cílem další fáze životního cyklu BPM – redesignu procesů. Další typy časů, které se používají při analýze podnikových procesů, jsou: (1) **čekací čas** (*waiting time*), kdy podnikový proces, resp. jeho aktivita čeká na volný zdroj, a (2) **čas zpracování** (*processing time*), po který zdroj (např. účastník procesu) pracuje na případě. Flexibilita je schopností reagovat na změny.

Výkonnostní dimenze

Může jít o schopnost managementu podniku změnit strukturu, pravidla a zodpovědnosti v podnikovém procesu jako reakci na potřeby trhů a obchodních partnerů, nebo o schopnost samotného podnikového procesu zpracovat různé případy a změnu obsahu pracovní náplně. Další metody, které se používají při identifikaci a definici procesních výkonnostních metrik jsou referenční modely, průmyslné benchmarky, nebo např. balanced scorecard. Simulace procesů je jednou z nejrozšířenějších a nejoblíbenějších technik na kvantitativní analýzu procesů. Simulaci se věnujeme v kapitole 2.5 této opory a její praktické nasazení při aplikaci na podnikové procesy budeme demonstrovat v další kapitole. Mezi nejznámější nástroje pro modelování a simulaci podnikových procesů patří bezesporu ARIS Business Simulator, Bizagi Process Modeler, BIMP nebo Signavio.

4.6 Redesign procesů



DEFINICE

Cílem **redesignu procesů** jako další fáze životního cyklu BPM je nalézt změny v podnikových procesech, které by pomohly vyřešit problémy, identifikované v minulé fázi BPM aktivity. Jedná se o zlepšování podnikových procesů. V tomto okamžiku si může podnik nastavit výkonnostní cíle. Změny mohou být analyzované a srovnané s nastavenými výkonnostními metrikami. Tato fáze se kombinuje s různými analytickými technikami a navrhovanými změnami, co vede k novému redesignu (návrhu) podnikového procesu, resp. k jednomu nebo vícerym to-be procesním modelem, které sestávají vstupem do další fáze BPM aktivity.

Důkladná analýza podnikových procesů vede k různým nápadům a směrům, jak by se mohly předmětné podnikové procesy změnit, aby lépe fungovali s ohledem na stanovené výkonnostní metriky. Tady vzniká prostor pro redesign procesů, neboli jejich znovunávrh. Komplikací je, že redesign nebývá řízený systematickým způsobem, nýbrž spíše je založený na kreativní činnosti. Stává se proto, že vícere redesignové možnosti, které mohou být přínosem, bývají ve svém důsledku opomenuty.

Při redesignu se využívají dvě základní metody: (1) **heuristický procesní redesign** (*heuristic process redesign*) a (2) **produktově orientovaný design** (*product-based design*). Podnikový proces vytváří a dodává nějaký produkt nebo službu pro své zákazníky. Kdyby někdo chtěl zlepšit kvalitu tohoto produktu nebo služby, tak nejlepším způsobem, jak toho dosáhnout je zlepšit příslušný podnikový proces. Všechny podnikové procesy se časem vyvíjejí a výsledkem může být mnohem komplexnější proces, než jaký byl zamýšlený na začátku jeho návrhu. Jeho komplexnost může být důvodem pro jeho výkonnostní problémy, které ve svém důsledku budou mít negativní vliv na zákazníka. Při redesignu podnikového procesu musíme vzít v potaz tyto skutečnosti:

- Interní a externí zákazníci podnikového procesu.
- Operativní pohled na podnikový proces – implementační detaily procesu, počtu a druhu jeho aktivit.
- Informace, které podnikový proces vytváří a využívá.
- Technologie, kterou podnikový proces využívá.
- Organizace a účastníci podnikového procesu – rozlišujeme dvě úrovně: (1) pracovníci jako účastníci podnikových procesů, kteří jsou alokováni k jednotlivým aktivitám včetně vztahů mezi nimi a (2) organizační struktura jako role, oddělení, projektové skupiny, uživatelé, apod.
- Chování podnikového procesu – jakým způsobem je podnikový proces realizovaný, jaké je pořadí jednotlivých aktivit a jejich plánování.

Existuje velké množství informačních zdrojů, které se týkají procesního redesignu, různých metodologií, případových studií, apod. Všechny přestupy můžeme rozdělit do tří skupin – **metodologie**, **techniky** a **nástroje**. Metodologie je souhrnem metod, postupů, principů, které se používají k řešení problému. Technika je přesně specifikovaný postup, který vede k řešení daného problému. Při procesním redesignu se používá např. brainstorming, out-of-box-thinking, Paretova analýza, kognitivní mapování, apod. Z technického pohledu se v procesním redesignu používají různé techniky a notace jako např. BPMN, IDEF, simulace, apod. To samé platí pro nástroje, které podporují simulaci výše uvedených technik a notací.

Při procesním redesignu se nejčastěji začíná s existujícím podnikovým procesem, který se analyzuje a navrhuje se jeho změny (heuristický procesní redesign). Ve výjimečných případech se nebere do úvahy existující podnikový proces, ale postupuje se radikálně projektem na „zelené louce“. V tomto případě je výsledek nejistý (produktově orientovaný design).

4.7 Implementace a monitorování procesů

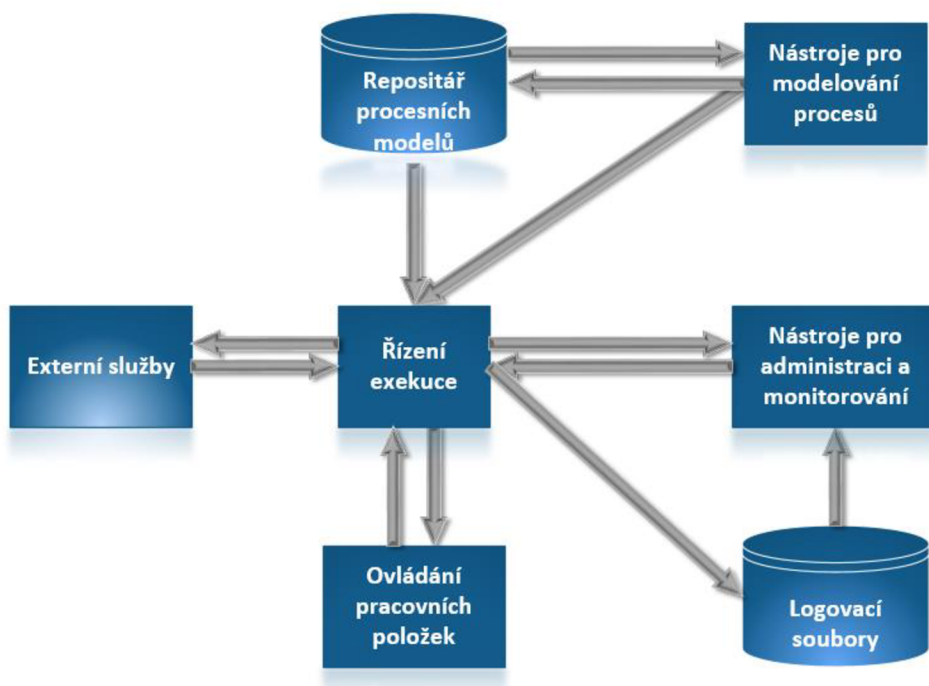
IS, které využívají znalosti o tom, jak spolu různé aktivity, které probíhají v podnikových procesech, souvisí, se nazývají **procesně orientované IS** (*process-aware*). Hlavní kategorií procesně orientovaných IS se nazývá *Workflow Management* systémy (WfMS), nebo v současnosti častěji *Business Process Management* systémy (BPMS). Speciálním znakem BPMS je, že pro spuštění podnikového procesu využívají jeho explicitní (nebo formální) popis ve formě procesního modelu. Účelem BPMS je koordinovat automatizovaný podnikový proces takovým způsobem, že aktivita v procesu je načas provedena správným účastníkem procesu.



DEFINICE

Ve fázi **implementace procesů** jsou připravené změny pro přechod od as-is k to-be procesním modelům implementované nejčastěji v podobě WfMS (podrobněji kapitola 1.3) nebo BPMS. Implementace procesů se skládá ze dvou rovin – **organizační change management** a **automatizace procesů**. Automatizace představuje vývoj a nasazení ICT technologií a IS, které podporují a spouštějí to-be procesní modely.

BPMS je standardním typem softwarového systému, který se nabízí na trhu s různými funkcionalitami, které pokrývají celý životní cyklus BPM, resp. podnikového procesu. Od jednoduchých systémů, které pomáhají při návrhu a automatizaci podnikových procesů, až po komplexní řešení, které pokrývají i monitorování a process mining pro zajištění kvality implementovaných procesů. BPMS ve své složitější podobě, např. v bankách umožňují i napojení na systémy třetích stran a na sociální sítě. Nutným předpokladem automatizace podnikových procesů je podpora formátu, ve kterém se nachází procesní model. BPMS následně tento procesní model načte a umožní jeho nastavení a exekuci včetně různých podmínek a časových omezení pro nasazení v běžné podnikové praxi. Takový procesní model, resp. jeho explicitní popis je nutné aktualizovat pro zajištění jeho aktuálního obsahu. BPMS proto existují jako komplexní produkty nebo se specializují jenom na určitou část životního cyklu BPM, např. jako analytické, modelovací nebo simulační nástroje. Typická BPMS architektura je znázorněna na obr. 4-6.



Obrázek 4-6: Architektura BPMS.

Zdroj: vlastní zpracování

DEFINICE



Když redesignované procesy běží, tj. jsou spuštěné, tak dochází ke sběru a ukládání dat o těchto procesech. Tyto data jsou analyzována a je možné rekurentně zjistit, jaká je aktuální výkonnost podnikových procesů od realizované změny. Ve fázi **monitorování a kontroly procesů** jsou identifikované a případně opravené odchylky od požadovaného stavu (*deviations*), úzké hrdla (*bottlenecks*) a opakující se chyby v procesech. Může dojít ke vzniku nových problémů a celá BPM aktivita startuje od začátku.

OTÁZKY



1. Popište podnikový proces objednávka-hotovost.
2. Jaké etapy identifikace procesů předcházejí vytvoření procesní architektury?
3. Jaké typy objevování procesů poznáme a na čem jsou založeny?
4. Jaký je rozdíl mezi kvalitativní a kvantitativní analýzou procesů?
5. K čemu slouží BPMS?

SHRNUTÍ KAPITOLY



V této kapitole jsme představili příklady podnikových procesů a vysvětlili jsme si podstatu životního cyklu řízení podnikových procesů (BPM, Business Process Management). Stručně jsme si popsali jednotlivé fáze životního cyklu – identifikace procesů, objevování procesů, analýza procesů, redesign procesů, monitorování a kontrolu procesů. Závěrem jsme přiblížili fungování softwarových systémů na podporu návrhu, implementace a exekuce podnikových procesů.

ODPOVĚDI



1. Podkapitola 4.1, str. 51 a 52.
2. Podkapitola 4.3, str. 56 a 57.
3. Podkapitola 4.4.
4. Podkapitola 4.5.
5. Podkapitola 4.7, str. 67.

5 PŘÍPADOVÁ STUDIE BIZAGI

RYCHLÝ NÁHLED KAPITOLY



Hlavním obsahem této kapitoly bude názorná ukázka využití životního cyklu BPM v podnikové praxi. Za pomoci softwarového nástroje Bizagi bude procházeno všemi fázemi včetně jejich zastoupení právě v prostředí Bizagi. Kapitola tedy bude kopírovat životní cyklus BPM teoreticky popisován v předchozí kapitole, proto se nedoporučuje postoupit k této kapitole bez prostudování předchozí.

Součástí této kapitoly je obsáhlá případová studie ve strojírenském podniku. Případová studie se zabývá adopcí procesního řízení jako nástroje na zvyšování procesní vyspělosti a zefektivňování firemní procesů. Bude se postupovat od úvodního nastavení procesní architektury a rozdělení procesů podle jejich podstaty. Následovat bude identifikace procesů vedoucí k tvorbě as-is procesního modelu. Důležitým aspektem je také analýza procesů, kde za použití simulace procesů v Bizagi Modeler se budou testovat a vybírat ty nejlepší to-be procesní modely, které následně postoupí do implementační a optimalizační fáze. Tím se životní cyklus uzavírá. Jako další součást kapitoly budou dílčí úkoly podporující praktickou demonstraci principů procesního řízení prakticky používaného v podnicích napříč odvětvími.

CÍLE KAPITOLY



Po prostudování této kapitoly budete umět:

- Využít software Bizagi jako podporu na řízení podnikových procesů,
- Testovat výkonnost různých procesních modelů v simulačním rozhraní,
- Metody sloužící pro úspěšnou implementaci BPM do podniku,
- Holistický přístup k praktickému řízení procesů.

KLÍČOVÁ SLOVA KAPITOLY



Bizagi, BPM, Simulace procesů, procesní model, redesign procesů, implementace, životní cyklus, zpracování závazků.

5.1 Představení Bizagi Suite

Bizagi je akronymem pro Business agility v překladu – podniková obratnost. Tento pojem vychází z historického vývoje disciplíny BPM, která prošla rozličnými etapami od **radikálního** přístupu k implementaci procesních technik (BPR) až po více **agilní** přístup vycházející z principů agilního způsobu projektového řízení. Proto se slovo agility – hbitost, obratnost hojně využívá nejen ve vědeckých publikacích, ale také u dodavatelů IT systémů podporující procesní řízení, jakým je také *Bizagi*.

Předsta-
vení spo-
lečnosti
Bizagi

Samotná společnost stojící za *Bizagi* vznikla v roce 1989, kdy byla BPM disciplína v počátcích. Původní záměr společnosti bylo doručovat platformu na podporu postupného zlepšování podnikových procesů. Tato platforma se postupem času stala globální a má stovky zákazníků, mezi kterými jsou firmy jako GE, Adidas, a další. Velká výhoda také spočívá v možnosti synchronizace s vysoce rozšířenými softwary jako např. IBM, SAP nebo Oracle a prostřednictvím **API** také s ostatními IT systémy v podniku.

Samotné *Bizagi* se prezentuje jako platforma sloužící pro zažehnutí digitální transformace ve firmách.

Výhody platformy *Bizagi*:

- Velice jednoduchá ovladatelnost zaměřená na uživatelskou přívětivost. Široká nabídka školicích kurzů a online materiálu pro samostudium. Podpora plyne také ze strany *Bizagi* komunity, která se shlukuje na fóru na *Bizagi* webových stránkách. To vše pomáhá při získávání důležitých informací pro firmy na jejich BPM cestě.
- Celistvé řešení *Bizagi* má pozitivní ohlasy od zákazníků i směrem k poměru nákladů a přínosů při implementaci do podniku v porovnání s ostatními dodavateli obdobných řešení.
- Společnost *Bizagi* se odlišuje také nastavením svého obchodního modelu. Dva ze tří softwarů nabízí zdarma. Na prvních dvou modulech – *Bizagi Modeler* a *Bizagi Studio* lze finálně adoptovat a pilotně otestovat fungování v podniku až do rozsahu 20 uživatelů a také na neomezeném množství procesních modelů. Až v situaci, kdy je podnik připraven na zavedení, se poptá řešení, které bývá nejčastěji individuálně naceněno.

Nevýhody platformy *Bizagi*:

- Jednoduchá implementace nabádá uživatele k rozhodnutí, že adopce ve vlastní režii bude nejvýhodnější. To může být častokrát kritická chyba, neboť mohou

chybět kompetence v oblasti procesního řízení a vůbec základních principů, které budou obsahem této kapitoly, a to může vyplynout až v nepovedený start a v nejhorším případě celý BPM projekt může zaniknout.

- Přestože *Bizagi* umožňuje řídit jak strukturované, tak nestrukturované procesy, nedostatečně se ovšem věnuje komplexnímu *case managementu* (řízení případů).
- Vzhledem k velikosti firmy a rychlosti změn v oblasti BPM společnost *Bizagi* pomalu přestává hbitě reagovat na změny a disrupce v oboru řízení a především automatizace podnikových procesů, kam vstupují firmy využívající například umělou inteligenci anebo tzv. robotickou procesní automatizaci.

Bizagi se skládá ze tří komponentů:

Balík SW
Bizagi

- **Bizagi modeler** – tato část softwarového *Bizagi* balíku slouží pro tvorbu procesních modelů respektující BPM notaci. Vše probíhá v jednoduchém rozhraní, kde se dá vytvořit téměř jakýkoli procesní model za pomoci jednoduchého a intuitivního systému „*drag and drop*“¹³. Součástí *Bizagi Modeler* je také simulační prostředí. Do toho prostředí vstupuje nastavený procesní model a zde se již jednoduchými 4-mi kroky nastaví veškeré proměnné vstupující do procesní simulace. V *Bizagi Modeler* tedy začíná BPM cesta a také zastupuje první fáze v životním cyklu BPM. Po dokončení tvorby procesních modelů lze tyto modely vyexportovat a dokumentovat v různých formátech, jakými je např. Word, PDF, a další. *Bizagi* obecně podporuje kolaboraci týmů, a proto i v *Bizagi Modeler* se nachází rozšíření pro spolupráci, kdy lze k procesním modelům přizvat další spolupracovníky, a dokonce s nimi komunikovat prostřednictvím vbudovaného chatu.
- **Bizagi Studio** – na modelech a dalších výstupech z *Bizagi Modeler* následně buduje *Bizagi Studio*, které slouží pro tvorbu procesních aplikací. *Bizagi* často zdůrazňuje, že se jedná o budování aplikací bez použití kódu – programování. To se dá brát jako výhoda, ale může to působit také kontraproduktivně v případě nutných větších zásahů do aplikací, popř. při řešení chyb v nastavení. Uživatelské rozhraní, na kterém se pracuje, se zde dělí na dvě. Jednou je prostředí, kde se nastavují a budují procesní aplikace. Celý tento proces je rozdělen do několika kroků, kterými se postupně prochází. Nastavuje se zde – datový model vázaný na procesní mapu, definování formulářů (rozhraní pro exekuci procesů), obchodní pravidla, uživatelé, integrace s ostatními softwary zapojenými do procesů a finálně exekuce procesů, kde se pilotně testuje, jestli je vše správně nastaveno. Po vybudování této procesní aplikace určené pro exekuci jednotlivých procesů se již vše odehrává v dalším uživatelském rozhraní, kam již přistupují konkrétní

¹³ V překladu táhni a pusť. Nejběžněji se využívá v prostředí Windows, kdy uchopíme virtuální objekt a přetáhneme jej na požadované místo.

uživatelé, ať již z pozice procesních účastníků nebo majitelů procesů a finální proces operativně vykonávají. *Bizagi Studio* se bude využívat ve fázi implementace procesů v rámci životního cyklu BPM.

- **Bizagi Engine** – poslední součást *Bizagi* balíku je *Bizagi Engine*. Za jeho pomoci lze distribuovat procesní aplikace napříč celým podnikem do všech zařízení. *Bizagi* je responzivním softwarem a nabízí jak rozhraní pro PC, tak i pro mobily nebo tablety. Lze také používat software přímo v prostředí webového prohlížeče nebo si stáhnout program a využívat jej přímo ze zařízení. Zde již veškerá práce probíhá v uživatelském rozhraní *Bizagi* pracovního portálu. Zaměstnancům jsou přiděleny konkrétní procesy, kterých se účastní a oni pak veškerou práci provádí právě přes toto rozhraní. Vše je tedy pod jednou střechou, z čehož je možné čerpat spoustu informací a dat skrze analytický nástroj přímo zabudovaný do programu. Data se týkají efektivity a výkonu prováděných procesů. Ty se poté používají pro další rozhodování při redesignu procesů popř. dalších strategických rozhodnutích.

Bizagi Modeler a *studio* jsou nabízeny zadarmo ke stažení včetně obsáhlé dokumentace vhodné pro implementaci. Za *Bizagi Engine* se již platí na bázi licencí za jednotlivé uživatele. Cena se rozděluje pro případ nasazení *Bizagi Engine* v cloudové formě nebo lokálně ve firmě. Zde se také účtuje ročně částka za údržbu.

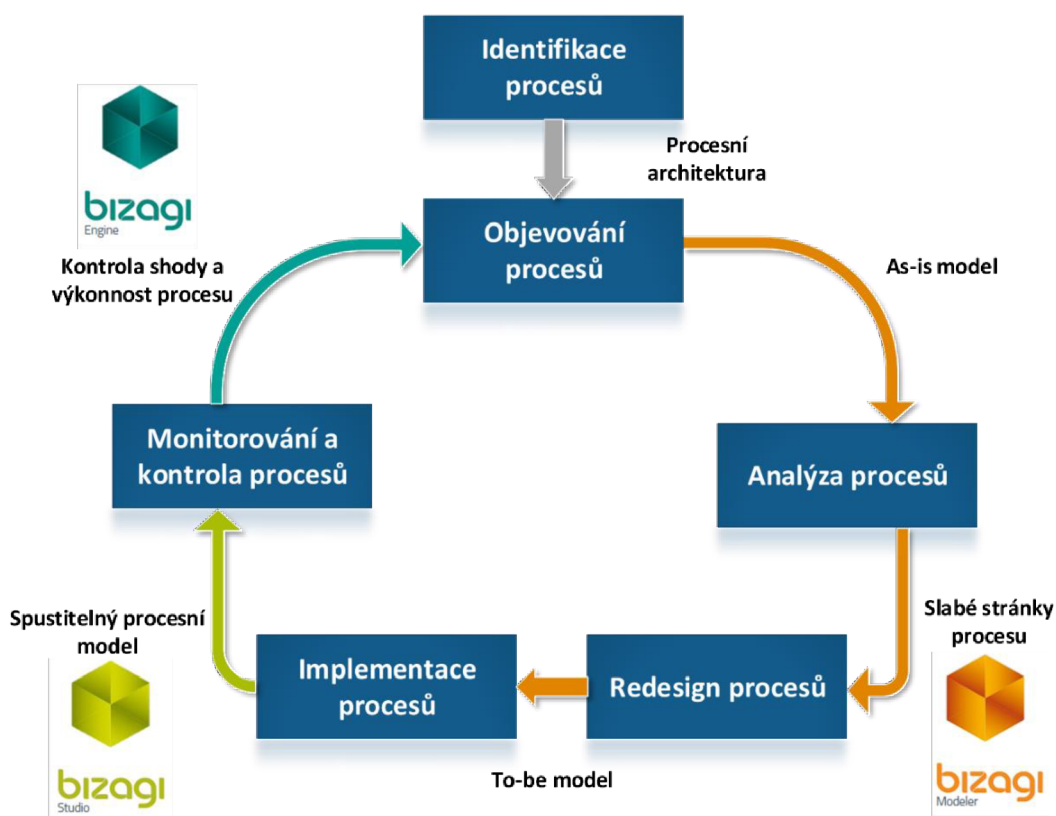


K ZAPAMATOVÁNÍ

Bizagi Modeler je program na tvorbu procesních modelů a jejich testování skrze simulace, *Bizagi Studio* slouží pro tvorbu procesních aplikací nad procesní modely, které slouží jako rozhraní pro exekuci těchto procesů a *Bizagi Engine* následně distribuuje tyto aplikace do potřebných zařízení napříč podnikem a také poskytuje analytický nástroj pro vyhodnocování a analýzu vykonávaných procesů.

5.1.1 BIZAGI A JEHO ZASTOUPENÍ V ŽIVOTNÍM CYKLU BPM

Na obrázku 5-1 je zaznačen kompletní životní cyklus BPM jako v předchozí kapitole, ale navíc je do něj doplněno zastoupení jednotlivých komponentů z balíku *Bizagi* pro zpřehlednění. Vidíme, že největší zastoupení v životním cyklu BPM má program *Bizagi Modeler*, který se využívá od objevování procesů přes analýzu až po redesign procesů. Implementace procesů je časově i technicky nejnáročnější fáze a v případové studii bude prováděna v *Bizagi Studio*. Poslední fáze monitorování a kontrola procesů je zastoupena v *Bizagi Engine*, kde se skrze analytický nástroj bude vyhodnocovat výkonnost procesů.



Obrázek 5-1: Zastoupení Bizagi v životním cyklu BPM.

Zdroj: vlastní zpracování

5.2 Identifikace procesů

Fáze identifikace procesů jako jediná nevstupuje přímo do uzavřeného životního cyklu BPM a také není podporována žádným z programů *Bizagi*. To však nemění fakt, že se jedná o klíčovou aktivitu, která následně ovlivní všechny další fáze.

Tato podkapitola se bude soustředit na procesní architekturu v podniku a na zodpovězení dvou hlavních otázek, které umožní vstoupit do další fáze:

- Jaké procesy v podniku probíhají?
- Na jaký proces by měl vstoupit do životního cyklu BPM na základě prioritizace dle zvolených kritérií?

**SAMOSTATNÝ ÚKOL**

Představte si standartní autobazar. Jaké procesy budou řídicí, klíčové a podpůrné v takové firmě? Uveďte alespoň 3 příklady ke každému z nich.

5.2.1 SITUAČNÍ ANALÝZA UKÁZKOVÉ FIRMY Z POHLEDU PROCESNÍ VYSPĚLOSTI

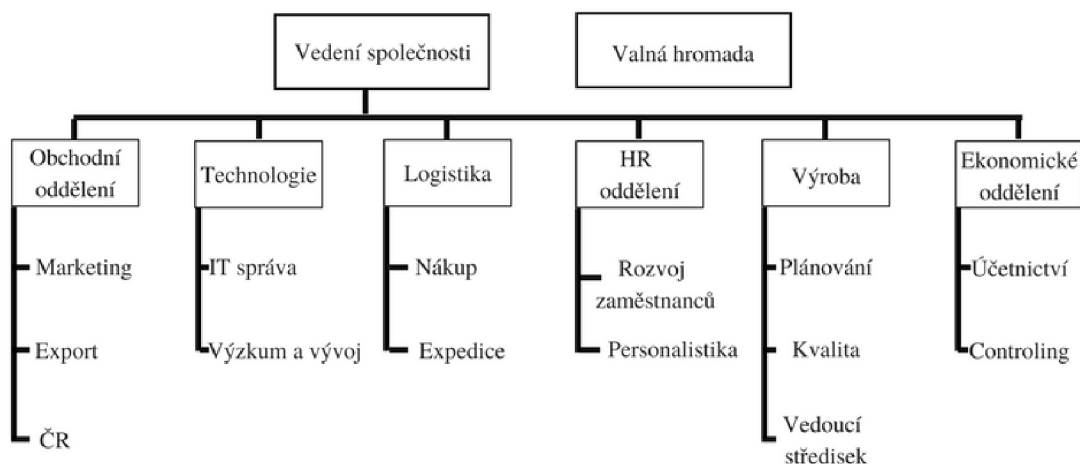
Pro správné zvolení procesu a celkového implementačního přístupu je nutné si nejdříve udělat zhodnocení **podnikové vyspělosti** (viz. podkapitola 4.3.2) z pohledu řízení podnikových procesů. To, v jaké fázi vyspělosti se firma nachází, determinuje její vztah k procesnímu řízení a také úroveň celkové procesní orientace.

*Podniková
procesní
vyspělost*

V současné době se firma nachází v situaci maximálního využití výrobních kapacit, což je dáno celkovou situací na trhu i v ekonomice obecně. Nezaměstnanost je na velmi nízké úrovni a firmy v tomto odvětví musí častokrát i odmítat zakázky. To je pro jakékoli rozvojové nebo změnové programy nepřilíš vhodné prostředí, protože jednoduše vedení tvrdí, že na tyto věci není čas a všichni se musí zabývat provozem a operativou. V minulosti již ve firmě probíhaly určité snahy o zavedení principů BPM, které ovšem skončily právě kvůli nadměrné vytiženosti. V rámci tohoto projektu se zaměřili na několik procesů v administrativě, kde vytvořili as-is procesní modely a ihned bez předchozího analyzování a optimalizace se pustili do automatizace těchto procesů. Povědomí u většiny pracovníků v ekonomickém a obchodním oddělení o procesním řízení již existuje, to platí i pro vedení firmy, které celý tento pilotní projekt iniciovalo. Tím, že zaměstnanci byli do těchto projektů zapojeni a jejich odezva byla spíše kladná, lze odhadem usoudit, že potenciál ve firemní kultuře pro iniciaci dalšího BPM projektu je značný. Z pohledu napojení na firemní manažerské nástroje (rozpočty, organizační struktury atd.) a na firemní IT infrastrukturu, je situace ve firmě v rámci BPM technik nulová, tzn., neexistuje žádný takový nástroj a předchozí projekt se touto oblastí nezabýval. Z analýzy současné situace lze tedy usoudit, že podnik se stále nachází na první úrovni BPM vyspělosti, kterou je fáze „*Initial*“ – **počáteční**. Tuto skutečnost je důležité mít na paměti v průběhu dalšího postupu, např. při zapojování zaměstnanců do BPM aktivit, výběr pilotního procesu, vzdělávání zaměstnanců apod.

**ŘEŠENÁ ÚLOHA**

Rozdělte následující **firemní procesy** v rámci procesní architektury mezi **řídicí** procesy, **klíčové** procesy a **podpůrné** procesy a popište vztah mezi procesní architekturou a organizační strukturou. Všechny procesy pochází z firmy z případové studie, která má následující organizační strukturu.



Obrázek 5-2: Organizační struktura sledovaného podniku.

Zdroj: vlastní zpracování

Vybrané firemní procesy:

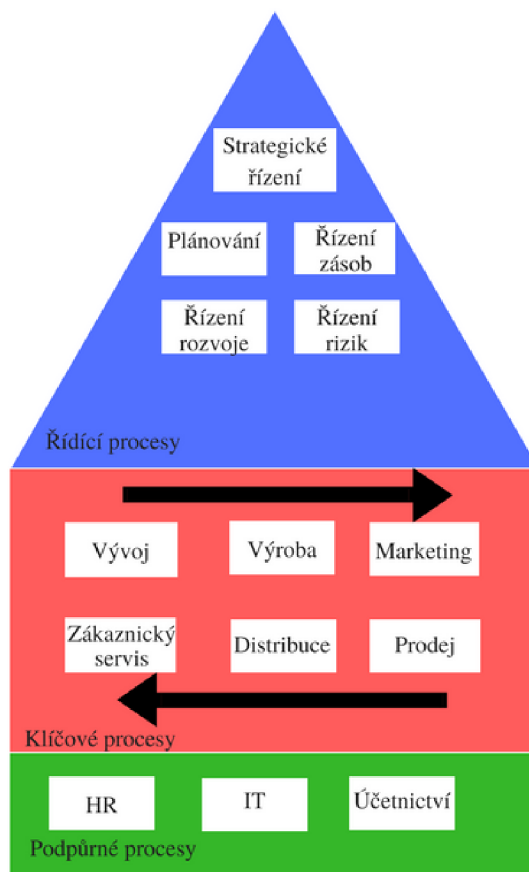
- Nábor zaměstnanců,
- zpracování dokladů,
- prototypování nového produktu,
- tvorba plánu výroby,
- vytvoření požadavku na nákup,
- hodnocení kvartální výkonnosti společnosti.

5.2.2 TVORBA PROCESNÍ ARCHITEKTURY V UKÁZKOVÉ FIRMĚ

Na obrázku 5-3 je zobrazena procesní architektura v případové firmě. Procesy jsou uvedeny na značně **abstraktní úrovni**, aby bylo možné obsáhnout všechny oblasti podnikových procesů. Jedná se tedy o první úroveň detailnosti tzv. **procesní mapa**. Tato architektura tedy znázorňuje veškeré firemní procesy na základní úrovni a dekompozicí jednotlivých prvků v ní se lze postupně dostat přes procesní modely až na velice konkrétní úroveň sub-procesů.

Procesní architektura (viz. podkapitola 4.3.3) byla vybudována na základě **rozhovorů** s důležitými osobami v rámci více zájmových skupin uvnitř podniku. Rozhovor se týkal

procesů a aktivit probíhajících jak na operativní, tak na strategické úrovni a jejich vzájemného vztahu a **provázanosti**. Také byla analyzována organizační struktura (uvedena v řešené úloze výše), která také posloužila pro vytvoření celistvého obrazu o hlavních skupinách procesů v podniku.



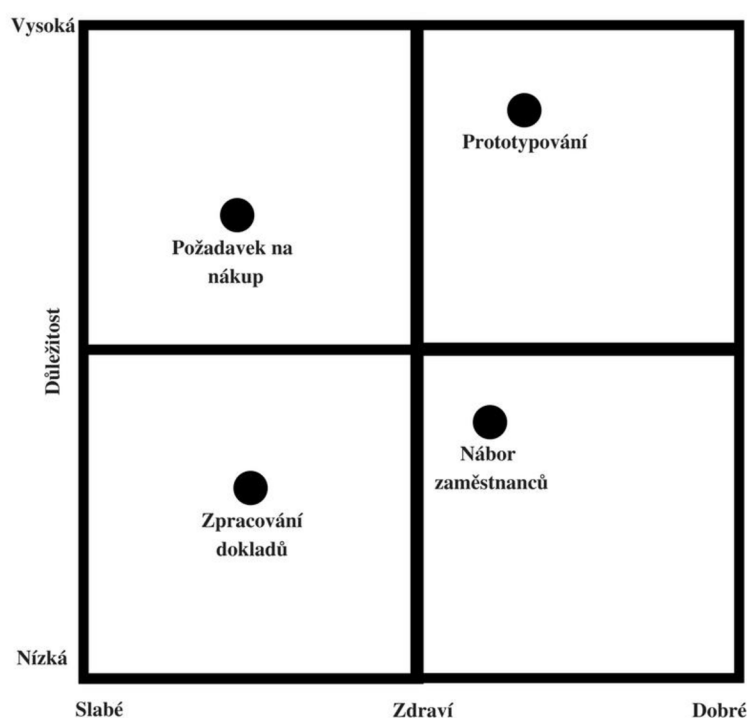
Obrázek 5-3: Procesní architektura ukázkového podniku.

Zdroj: vlastní zpracování

První částí a pomyslnou střechou celé architektury tvoří **řídicí procesy**. Nadřazeným procesem je strategické řízení, které určuje celkové směřování firmy a využívání dostupných zdrojů do oblastí dle priorit stanovených ve firmní strategii. Ostatní řídicí procesy již fungují samostatně a ze své podstaty zasahují a ovlivňují další procesy napříč podnikem. Další část firemní procesů poté tvoří **klíčové procesy**, prostřednictvím kterých se generuje hodnota pro zákazníka od počátečního vývoje přes výrobu, marketing, prodej, distribuci až po zákaznický servis. Rozložení v této části procesní architektury odpovídá **hodnotovému řetězci**. Naznačuje tedy posloupnost klíčových procesů přispívajících k budování hodnoty pro zákazníka v pořadí znázorněném šipkami. Poslední částí v procesní architektuře jsou **podpůrné procesy**, které umožňují, podporují a zefektivňují exekuci klíčových procesů. V případovém podniku existují tři základní procesní skupiny – řízení lidských zdroj (HR), správa informačních systémů a technologií a účetnictví. Tyto procesy sami o sobě nepřinášejí hodnotu zákazníkovi, což ale neubírá na jejich důležitosti v kontextu celkového firemního řízení procesů.

5.2.3 PRIORITIZACE PROCESŮ

Po procesní architektuře přichází na řadu hlubší rozklíčování hlavních abstraktních procesů na sub-procesy druhé úrovně, z kterých již lze vybudovat procesní modely a jasně stanovit účastníky a majitele procesu. Tato aktivita je v této případové studii spíše **kvalitativního charakteru**, kdy se vede **rozhovor** s vedoucími jednotlivých oddělení, a otázky jsou směřovány na základní dekompozici procesů první úrovně stanovených v procesní architektuře. Vedoucí oddělení mají v dané oblasti největší přehled o procesech a u spousty z nich jsou jejich majiteli, což z nich tvoří ideální osoby právě pro hlubší identifikaci.



Obrázek 5-4: Prioritizace procesů.

Zdroj: vlastní zpracování

Ukázky rozklíčovaných procesů jsou zaznamenány v samostatném úkolu výše v této podkapitole. Také v této fázi ještě není potřeba procesy dále mapovat nebo s nimi jiným způsobem pracovat, jde pouze o jejich nalezení a zaznamenání. Vybrané procesy byly ilustrativně zaznamenány do matice na obrázku 5-4. Pro vyšší přehlednost nebyly zaznamenány všechny. Matice zobrazuje na jedné ose **důležitost** daného procesu v rámci celkového firemního kontextu a na horizontální ose je **zdraví** procesu, které se získá prostřednictvím pozorování, stínování nebo rozhovory s procesními majiteli a účastníky. Následně se zanáší jednotlivé procesy do matice a jejich pozice se určuje podle vzájemného vztahu právě těchto dvou proměnných. Jedná se o pomůcku, která v této případové studii je hodně o subjektivním pohledu účastníků a majitelů procesu, protože ve firmě neexistuje žádný nástroj na měření efektivity a kvantifikaci prováděných procesů. Jedinou možnou kvantifikaci procesů je možné realizovat skrze logovací soubory využívaného ERP softwaru a jeho

následné zpracování prostřednictvím *Process miningu*, jak bude ukázáno v další případové studii v poslední kapitole těchto skript. I přes jistou subjektivitu se lze od takto roztržiděných procesů odrazit.

Proces **prototypování** je součástí klíčového procesu vývoj, tím pádem je, co se týče důležitosti, na vysoké pozici. Rozhovor s vedoucím oddělení technologie odhalil, že proces zde není jasně strukturovaný, protože přehnaná strukturovanost by mohla omezit kreativní myšlení, tak důležité pro tvorbu inovací. Jinak proces prototypování nevyjadřuje žádné známky špatného procesní kondice. Kombinace nevhodnosti (neproveditelnosti) zapojení technik procesního řízení s dobrou úrovní procesního zdraví, směřuje k faktu, že tento proces není vhodný pro úvodní BPM projekt.

Proces **náboru zaměstnanců** je součástí HR oddělení a zabezpečuje průchod zaměstnance od fáze zájemce až do fáze zaškolený pracovník. S představitelem tohoto oddělení byl opět veden rozhovor týkající se současného stavu a existujících problémů při vykonávání tohoto procesu. Až na menší problémy týkající se efektivní práce s náborovými dokumenty, jejich správa a archivace, nebyly v procesu identifikovány žádné další problémy, a i z pohledu výkonu proces funguje dle stanovených indikátorů. Proto je procesní zdraví spíše na dobré úrovni a výsledný efekt po zapojení do BPM programu by nebyl dostatečný. To by mohlo způsobit špatnou návratnost investice z vložených zdrojů a zastavení dalších BPM projektů do budoucna.

V této fázi vyspělosti nelze vybrat **klíčový proces** (rozdělení procesů kapitola 1.2), možná by úspěch prospěl při obhajobě u vedení, ale neúspěch by zabil veškeré další snahy do budoucna o postoupení v modelu procesní vyspělosti. Také kapacitní vytíženost firmy je vysoká. Rozum by velel vybrat si jeden z klíčových procesů (např. požadavek na nákup nebo prototypování) a snažit se s ním dosáhnout stanoveného cíle ve formě efektivity nebo zvýšení kvality výstupů. To může platit v případě, že již firma dosáhla určité fáze procesní orientace a její procesní vyspělost je na vyšší úrovni. Avšak v této počáteční fázi je nesmírně důležité začít od malých projektů nezaměřujících se na klíčové procesy, jejichž možné ohrožení v průběhu implementace by mohlo následně ohrozit chod celé firmě na nejdůležitějším hodnotovém řetězci směrem k zákazníkům.

Proto volba úvodního procesu, kterému se dále tato případová studie bude věnovat, padla na proces **zpracování dokladů** (závazků), který je součástí podpůrného procesu účetnictví. Jeho jasná strukturovanost, pravidelnost a také podpůrný charakter jsou důležitými faktory pro jeho výběr. Po rozhovoru s hlavní firemní účetní byly naznačeny určité nedostatky v tomto procesu, které spočívají především v dlouhé době zpracování a oběhu dokladu. Na základě informací tedy bylo zdraví procesu stanoven na spíše horší úrovni. Důležité je zde také zmínit vysokou míru proveditelnost, která značí potenciál úspěchu při využití BPM nástrojů a technik, což vychází z faktu, že velká část procesu probíhá v elektronické podobě a jediný externí program vstupující do exekuce procesu je účetní ERP software.

5.3 Objevování procesů v ukázkové firmě

Úlohy objevování procesů ve firmách má nejčastěji na starosti **procesní analytik**, který zodpovídá za projekty týkající se analyzování a vylepšování procesů. To platí za předpokladu, že firma má již určitou zkušenost s projekty zaměřenými na procesní řízení. To však v ukázkové firmě neplatí. Tuto úlohu tedy mohl plnit externí konzultant, nově rekrutovaný zaměstnanec s potřebnými kompetencemi nebo současný zaměstnanec povoláný pro novou pracovní pozici. Firma vyhodnotila, že nejvýhodnější bude najmout externího konzultanta, který tuto roli bude zastávat. Bude mít k dispozici tzv. **doménové experty**. Ti mají důkladnou znalost, jak proces probíhá, protože se ho buď přímo účastní, nebo ho vlastní (vlastník procesu). Aby fungovala fáze objevování procesů, musí fungovat také kooperace mezi procesním analytikem a doménovým expertem, protože výsledek v podobě platného procesního modelu, závisí na znalosti od obou zúčastněných rolí. Doménoví experti v případě vybraného procesu v ukázkové firmě mají znalost o průběhu procesu z každodenní operativy, avšak nemají znalost modelovacích jazyků. Tu má externí procesní analytik, který ale zas nemá znalost samotného procesu zpracování závazků v podniku.

Role

K ZAPAMATOVÁNÍ



Doménový expert je osoba, která má důkladnou znalost o průběhu analyzovaného procesu. Nejčastěji je to konkrétní pracovník, který proces vykonává nebo osoba zodpovědná za proces (procesní vlastník)

Procesní analytik oproti tomu má vysokou znalost o řízení a analýze procesů, včetně znalosti modelovacích jazyků a aktivně se této oblasti věnuje. Avšak jeho znalost konkrétní procesů je nižší.

Před samotnou realizací této fáze životního cyklu BPM je důležité identifikovat **výzvy**, před kterými osoby zodpovídající za úspěch této iniciativy stojí:

- Již zmiňovaná kapacitní vytíženost projevující se i v ekonomickém oddělení a z toho plynoucí neochota doménových expertů spolupracovat
- Jakákoli neznalost doménových expertů základních modelovacích principů
- Roztříštěnost znalostí o procesu mezi několika pracovních pozic (procesních účastníků)
- Neschopnost doménových expertů zobecňovat procesy. Tím pádem se drží na úrovni konkrétních aktivit



ÚKOL K ZAMYŠLENÍ

Představte si, že z pozice **procesního analytika** vstupujete do studijního oddělení naší fakulty. Máte za úkol vymodelovat administrativní proces změny typu studia z prezenční na kombinovanou. Jaké by byl postup? Do jakých dokumentů byste nahlídli? Jaké konkrétní otázky byste zvolil při rozhovoru?

Postup

Strategie organizovaného sběru informací je založena na několika krocích logicky na sebe navazujících a má vyplýnout ve vybudování procesního modelu. Prvotní snahy budou směřovány do **analýzy dokumentů**, které nenarušují pracovníky a nevyžadují jejich zapojení, což v případě plného vytížení firemních kapacit, je na místě. Další taktika méně zasahující je **pozorování** účastníků procesu při vykonávání daného procesu. Poslední taktika budou **rozhovory** s doménovými experty, kde se bude postupně pracovat na vylepšování draftu procesního modelu. V případě, že i po tomto kroku budou nejasnosti ohledně vymodelovaného procesu, musí přijít na řadu **workshop** pod vedením facilitátora se zástupci účastníků procesu, aby se přímo na tomto workshopu dosáhlo konsensu o finální podobě procesního modelu. Avšak tato aktivita je časově více náročná, proto se využije pouze v krajním případě.

5.3.1 ANALÝZA DOKUMENTŮ

V prvním kroku se budou analyzovat dostupné dokumenty tak, aby procesní analytik získal informace o procesu a mohl si utvořit alespoň základní představu předtím, než přejde ke kroku, který již počítá s účastí doménových expertů. Před získání dokumentů byl proveden krátký rozhovor s vedoucím ekonomického oddělení, který směřoval k odhalení všech pracovních pozic, jenž se procesu zpracování závazků účastní. Představení si **cesty** jednotlivých faktur skrz ekonomické oddělení není pro vedoucího obtížný úkol, proto ihned identifikoval **4 pozice**, které přímo vstupují do procesu:

- Finanční asistent
- Recepční
- Administrativní manažer
- Účetní

Je možné, že se počet zúčastněných ještě změní, ale jako základ tato informace stačí. Následuje analýza **popisu pracovních pozic**, právě u těchto 4 pozic. Tyto dokumenty jsou jedny z mála dokumentů, které má každá firma zejména i kvůli legislativě. Tím, že popis pracovní pozice musí být ze zákona přiložen k pracovní smlouvě, jedná se spíše o dokument vznikající z nutnosti nežli popis činnosti, z kterého se dá vycházet. Lze z něj vyčíst

hrubé obrysy jednotlivých aktivit, z toho některé očividně přímo souvisí s procesem zpracování závazků, jako např. zavedení faktur a dokladů do ERP softwaru, tvorba platebních příkazů nebo také kontrola přijatých dokladů. Po průzkumu dokumentů se zanáší aktivity a sub-procesy v různých úrovních abstraktnosti nebo konkrétnosti do pracovního souboru a z něj se bude v další fázi vycházet při formování těchto aktivit do posloupnosti a značení spojitostí mezi nimi.

U jedné pozice – finanční asistent byl z důvodu vysoké fluktuace na této pozici sepsán podrobný dokument usnadňující zapojení nového pracovníka do operativních činností. Tento dokument poskytuje již příliš mnoho detailů a konkrétních pracovních úkonů včetně tzv. *workflow* (viz. kapitola 1.3). Ten není vymodelován, ale spíše popsán slovně i graficky.

Dohromady tvoří dokumenty pouze základ pro další objevování a pouze s jejich znalostí nelze validně vybudovat procesní model. Procesní analytik je ale pomocí těchto dokumentů uveden do kontextu procesu zpracování závazků a v dalších fázích znalosti nabitě zde využije. Dalším plánovaným krokem je tedy pozorování procesních účastníků při výkonu ukázkového procesu.

5.3.2 POZOROVÁNÍ PROCESNÍCH ÚČASTNÍKŮ

V další úrovni objevování procesů se procesní analytik s majitelem procesu domluvil na **pasivním pozorování**, které bude vykonávané v průběhu pracovní doby, kdy všichni účastníci procesu jsou zapojeni v procesu a nejlépe proces aktivně vykonávají. Procesní analytik zvolil nejdříve taktiku **sledování cesty** jednoho konkrétního závazku (faktury) napříč všemi aktivitami od přijetí přes zpracování, nejlépe až po zaplacení. Následně bude sledovat konkrétní účastníky při jejich práci po dobu nezbytně nutnou pro sběr dostatečného množství vstupů zabezpečující kvalitu informací.

Tím, že proces zpracování závazků je dobře strukturovaný a objevuje se v něm málo výjimek a jiných problémů, nebude tato fáze náročná jako u více nahodilých a méně strukturovaných procesů, kde se může jednat o komplexní a náročnou aktivitu.

Po výstupech z pasivního pozorování si procesní analytik udělal jasno v procesní posloupnosti a propojení procesů. To mu umožnilo navrhnout první procesní model tzv. *draft*, který se bude následně iterativně vylepšovat skrz **rozhovory** s doménovými experty.



SAMOSTATNÝ ÚKOL

Jak by podle vás vypadal draft procesního modelu zpracování e-shopové objednávky u vašeho oblíbeného e-shopu. Nemusíte se nutně držet BPM notace, jde spíše o prvotní draft určený pro další validaci.

5.3.3 ROZHOVORY S DOMÉNOVÝMI EXPERTY

Při rozhovorech s doménovými experty bývá největší překážkou jejich skeptický přístup. Myšlení je nutné přetransformovat z pohledu, že vidí výjimky a odlišnost každé situace do pozice, kdy pochopí smysl modelování procesů a dokáží do určité míry **generalizovat** aktivity v procesu na úroveň pro procesní modelování potřebnou.

Při rozhovorech procesní analytik dodržuje stejnou posloupnost jako u pozorování - **dopředný postup**, kdy se začíná u přijetí faktury u pozice recepční. Rozhovor také nemá žádnou předem definovanou strukturu, jsou pouze stanoveny informace, ke kterým se procesní analytik snaží dopátrat. Požadavky na informace vychází z předcházejících fází, zejména pak z fáze pozorování. U některých pracovních pozicí jde skutečně o rychlý rozhovor, protože např. úloha recepčního v procesu je pouze předat nebo přeposlat fakturu dále. U některých dalších pozicí jako finanční asistent je situace komplikovanější, protože do procesu vstupuje několikrát a stará se o velkou část aktivit spojených s procesem zpracování závazků. Další věc u **otevřeného rozhovoru** je povzbudit otevřenost doménových expertů, kteří se formální strukturou nebudou cítit tak svázáni a otevřenost přispěje k získání transparentních informací.

5.3.4 POPIS A MODEL PROCESU ZPRACOVÁNÍ ZÁVAZKŮ

Po několika rozhovorech se ucelily výstupní informace, daly se dohromady s informacemi získanými z analýzy dokumentů a z pozorování a výsledkem je tento **finální procesní popis**:

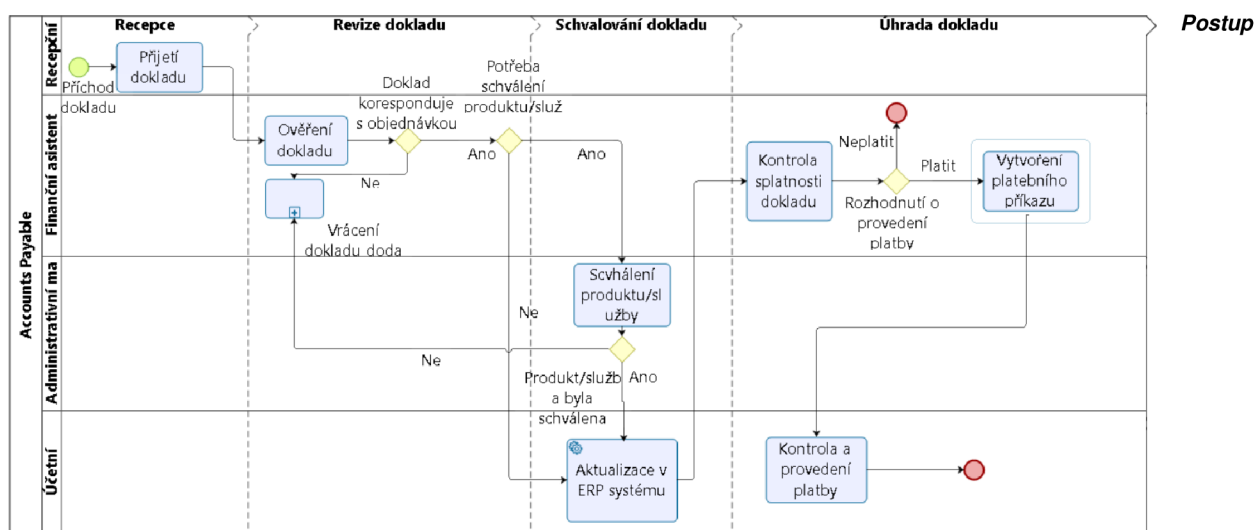
Popis procesu

Celý proces zpracování závazků začíná na pozici recepčního, kde se doklad přijímá, a to ve dvou možných formách – elektronicky (např. prostřednictvím emailové zprávy) nebo fyzicky (společně s dodávkou). Je zde také spárován s příslušnou objednávkou, která každému závazku předchází. Poté probíhá porovnání přijaté faktury s údaji uvedenými na objednávce. Když informace na faktuře (produkty, datum splatnosti, částky) korespondují s objednávkou, faktura je schválena. Když ne, faktura je odmítnuta, okomentována a vrácena dodavateli. Sub-proces navrácení dodavateli se v této případové studii neřeší. V případech, kdy závazek nemá žádnou objednávku, která předcházela, musí být do faktury zaneseny další dodatečné informace (označení produktu, údaje o dodavateli). Další brána ur-

čuje, jestli je potřeba dalšího schvalování od osoby, která je majitelem procesu tvorby objednávky (administrativní manažer). Pokud schválení není potřeba, proces se přesouvá k aktualizaci ERP systému, kterou má v režii firemní účetní. Když schválení od administrativního manažera neprojde, je faktura opět vrácena dodavateli. Po aktualizaci ERP systému probíhá kontrola splatnosti faktury opět u finančního asistenta, který po zvážení faktorů, jakými jsou blížíící se splatnost na dokladu a likvidita podniku, určí, jestli je faktura připravena na uhrazení. Jestli ne, proces zde končí. V opačném případě finanční asistent vytvoří platební příkaz, který přepoše účetnímu a ten příkaz zkontroluje a provede platbu skrze internetové bankovníctví.

5.4 Analýza procesu zpracování závazků

U analýzy procesu a následné tvorby to-be procesního modelu se jedná o nejvíce kreativní fázi ze všech, která je hodně otevřená ve smyslu používání striktních postupů a metod. Samozřejmě, že jako u každé fáze i zde je obrovské množství technik, a proto je potřeba určit, které se pro danou situaci nejvíce hodí. Z kapitoly 4.5 vyplývá, že máme dvě obecné skupiny analýzy, a to **kvalitativní** a **kvantitativní**. Námí využívaný program *Bizagi* používá logicky kvantitativní analýzu prostřednictvím **simulací** a **what-if analýzy**. Té ale nejlépe musí předcházet kvalitativní analýza, protože obecně platí, že až kombinace obou druhů analýzy přináší nejefektivnější výstupy.



Obrázek 5-5: As-is procesní model procesu zpracování závazků.

Zdroj: vlastní zpracování

Postup analýzy procesu zpracování závazků v ukázkové firmě tedy bude následující. Nejdříve se začne **analýzou přidané hodnoty**, která je kvalitativního charakteru a měla by rozřadit procesy na ty, které přináší hodnotu zákazníkovi nebo podniku a ty, nepřinášející hodnotu. Tato analýza pomůže v hledání podnětu pro inovaci a *redesign* ukázkového pro-

cesu, který by měl vyvrcholit ve vybudování to-be procesního modelu. Správnost a účinnost předpokladu, který předcházela vymodelování to-be procesu, bude otestována v programu *Bizagi Modeler*, konkrétně v jeho **simulačním prostředí**. Tím se dokáže vyčíslit možný přínos chystaného *redesignu* a potvrdí se nebo vyvrátí předpoklad vycházející z kvalitativní analýzy.

5.4.1 KVALITATIVNÍ ANALÝZA UKÁZKOVÉHO PROCESU

Analýza
přidané
hodnoty

Jako hlavní technika kvalitativní analýzy byla zvolena analýza přidané hodnoty, neboť její využití je velice široké a může být přínosná téměř v každém případě analýzy procesů.

Obvykle se v první etapě rozkládá proces na jednotlivé kroky a sub-procesy tak, aby se na úrovni pracovního postupu eliminovali zbytečné aktivity. V této případové studii tato úvodní činnost není na místě, protože se operuje na úrovni procesního modelu znázorněného v předchozí podkapitole a další zabředávání a specifikace na konkrétní aktivity by mohla i vzhledem k nízké procesní vyspělosti firmy působit zbytečně složitě a časově náročně. Navíc nejsou žádné další podklady ani procesní popisy či seznamy, proto by se vše muselo tvořit od začátku, a to bude předmět spíše dalších BPM projektů v ukázkové firmě.

Nyní musí následovat určení, o jaký proces se v kontextu analýzy přidané hodnoty jedná. Proces zpracování závazků je **podpůrným procesem** (vychází z procesní architektury) a hodnotu zákazníkovi ze své podstaty nepřináší, jedná se ale o tzv. **business value-adding proces**, který nepřináší hodnotu zákazníkovi, ale firmě umožňuje efektivně pracovat a nakládat s daty dodavatelů nebo zákazníků. Pomyslným **zákazníkem procesu** je dodavatel, který poslal do firmy fakturu (vznikl závazek) a očekává její uhrazení. Pro firmu jsou na druhou stranu důležitá data, která se musí zanést do ERP systému a společně s celkovými daty o závazcích slouží jako základ pro manažerské rozhodování.

Následuje samotný rozklad všech aktivit v procesu a zmapování, které z nich přidávají hodnotu přímo zákazníkovi, které přidávají hodnotu podniku, a které nejsou pro podnik ani zákazníka vůbec přínosné.



ŘEŠENÁ ÚLOHA

Uvažujme nad procesem **požadavek na nákup** vybavení z jedné výrobní haly u velkého kovozpracujícího podniku. Proces zde bude rozložen do vysoké specifčnosti jednotlivých kroků:

Když předák výroby na základě sledovaných hodnot u vybavení, kterým jsou např. pracovní rukavice, špunty do uší a další vybavení spíše spotřebního charakteru (nejedná se o materiál), spatří nedostatek u tohoto typu vybavení, sepíše oficiální požadavek na nákup a přidělení chybního vybavení. Následně tento požadavek pošle skrze email na ekonomické oddělení pro další zpracování. Pracovník na ekonomickém oddělení si tento email otevře,

přečte, následně skrze elektronické rozhraní vybere chybící vybavení, zkontroluje jeho dostupnost nejčastěji telefonátem. Na základě těchto kroků vytvoří požadavek na nákup, který přepošle v emailové podobě vedoucímu výroby na schválení. Ten jej otevře a prozkoumá žádost, v případě potřeby vykomunikuje nejasnosti a přepošle požadavek zpátky pracovníkovi ekonomického oddělení. Ten vytvoří objednávku na vybavení a pošle jej dodavateli vybavení.

Na základě tohoto popisu identifikujte procesní kroky a rozčleňte (popis rozdělení naleznete v kapitole 4.5.1) je mezi:

- *Value-adding (VA)*
 - *Business value-adding (BVA)*
 - *Non-value-adding (NVA)*
-

Proces zpracování závazků se ve sledované firmě skládá z těchto aktivit. Níže je k nim přidán komentář, o jaký typ přidané hodnoty se jedná:

- **Přijetí dokladu** – bez přijetí dokladu by žádné další kroky nebyly možné. Ovšem je zde potřeba jít trochu hlouběji a zaměřit se, jestli je přijetí na recepci, která pouze přepošle fakturu dále, nutné a jestli by fakturu nemohl přímo přijímat finanční asistent. I tak se avšak jedná o **aktivitu přidávající hodnotu zákazníkovi** procesu.
- **Ověření dokladu** – ověřování a kontrolní kroky obecně se často řadí mezi *Business-value-adding* aktivity. Avšak analytik by si měl položit otázku, kolik ověřovacích a kontrolních kroků je potřeba, a jestli jsou skutečně přínosné pro podnik. U této aktivity lze říci, že je **přínosná pro podnik**, protože toto ověření je klíčové kvůli dalšímu zpracování a správnosti faktury. Bez tohoto kroku by dále mohly vzniknout chyby.
- **Schválení produktu/služby** – další schválení následuje, ale nyní již v režii administrativního manažera. Aktivita je nastavena pouze pro vybrané produkty nebo služby, jenž schválením musí projít. Zde je již na zvážení, jaké schválení, jakého produktu je pro **podnik přínosné (BVA)** a co **nepřidává hodnotu**. Tento poznatek může sloužit dále při analýze procesu, kdy se tento schvalovací krok podrobí hlubší inspekci a pokusí se najít opatření pro zefektivnění procesu.
- **Aktualizace v ERP systému** – zde se určitě jedná o **aktivitu přidávající hodnotu zákazníkovi procesu (VA)**, neboť ERP je důležitým komponentem, který i následně generuje platební příkaz a určuje, kdy by měl být závazek uhrazen. Takže bez tohoto kroku by nebylo možné splnění cíle procesu.

- Kontrola splatnosti dokladu – časově nejméně náročná aktivita bude také **přidávající hodnotu**, bez tohoto mezikroku nebude možné pokračovat k vytvoření platebního příkazu. Celý proces po této aktivitě také může skončit kvůli nevyhovujícím podmínkám (splatnost a dostatek volných finančních prostředků) vyplývajících z ERP.
- Vytvoření platebního příkazu – na základě platebního příkazu se uskutečňuje platba závazku dodavateli. Proto se jednoznačně jedná o **aktivitu přidávající hodnotu (VA)**.
- Kontrola a provedení platby – proces skládající se ze dvou kroků, kdy jeden je spíše **přínosný pro podnik (BVA)** a tím je kontrola příkazu a druhý krok je již **přidávající hodnotu pro zákazníka procesu**, protože úhradou závazku se dosahuje cíle celého procesu. U kroku kontroly platebního příkazu je zapotřebí se podívat, jak a proč tato kontrola probíhá a jestli je skutečně potřebná. Tyto pochybné místa budou součástí dalšího zkoumání.

Brainstorming
workshop

Po analýze přidané hodnoty bude následovat **kreativní část** kvalitativní analýzy procesu, kdy se pomocí metody **brainstorming**, bude s majitelem procesu a vybranými procesními účastníky vyhodnocovat rozporuplné místa v procesu, jenž byly výstupem analýzy přidané hodnoty.



PRO ZÁJEMCE

Brainstorming je kreativní technika, která je založena na skupinové kreativitě. Účastníci brainstormingu jsou vyzváni, aby generovali nápady směřující k cíli nebo vyřešení zadaného problému. Musí se přitom dbát na základní pravidla. Brainstorming je o **kvantitě**, a ne o kvalitě nápadů. V průběhu brainstormingu se nápady nehodnotí, aby se předešlo soudu a až po nasbírání dostatečného množství nápadů se na závěr hodnotí realita a proveditelnost vygenerovaných idejí.

Největší otazníky přineslo nadměrné kontrolování a schvalování v procesu, proto tyto místa byly obsahem kreativní porady, jejímž cílem bylo vytvořit **to-be procesní model**. Nejdříve se rozklíčovaly schvalovací aktivity na jednotlivé kroky a spustila se debata, které z kroků jsou přínosné a které naopak ne (podrobnější analýza přidané hodnoty). Procesní analytik celou tuto schůzku **facilitoval** (moderoval). Z porady vyplynulo, že nejvíce zbytečný a zatěžující je proces schvalování platebních příkazů, který je dán zejména velkou pracovní zátěží u účetního, jenž si stěžuje na vysoké množství pracovních úkonů, které musí v průběhu dne vykonávat, a hlavně na nadměrné přecházení mezi aktivitami, což přerušuje **plynulý tok práce** a celková efektivita tak klesá. Oblast pro brainstorming je tedy vydefinovaná a kreativní část pod vedení procesního analytika začala. Výstupem

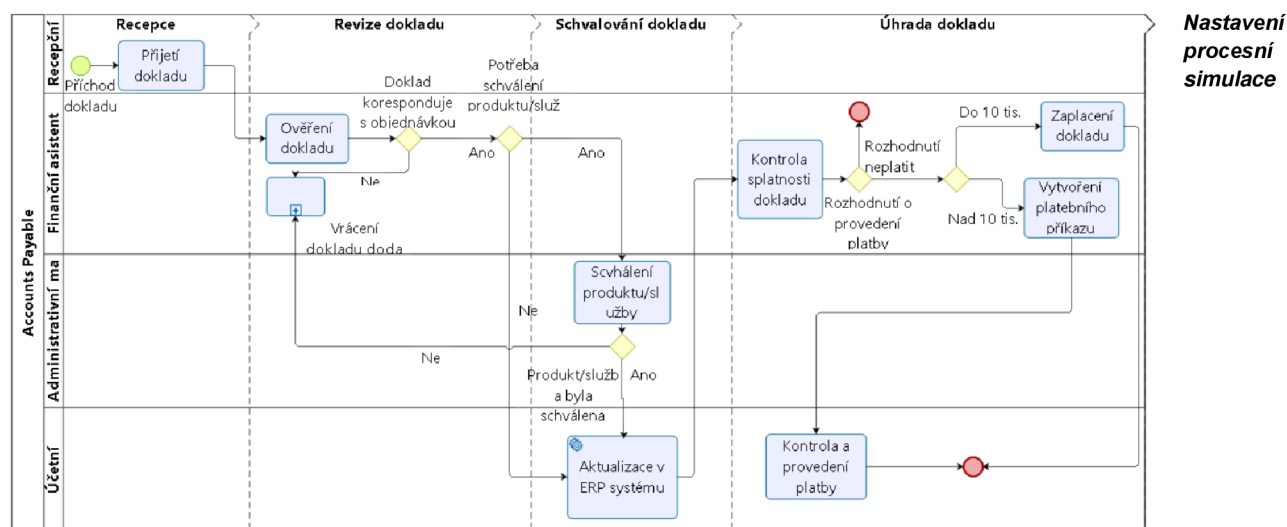
byl předpoklad na vylepšení stávajícího procesu předelegováním části procesu od účetního na finančního asistenta. Tento procesní *redesign* spočívá v přidělení kompetence úhrady závazků pod 10 tisíc korun finančnímu asistentovi, který při splnění této podmínky, vytvoří platební příkaz a rovnou závazek také uhradí. To podpoří plynulost toku práce a mělo by ulehčit účetní. V případě, že závazek bude nad 10 tisíc korun, vše probíhá tak, jako v as-is procesním modelu.

To-be procesní model

Na obrázku 5-6 je znázorněn to-be procesní model, který je až poslední fáze úhrada dokladu stejný jako as-is procesní model. Jediná změna proběhla, jak bylo popsáno výše, po braně rozhodnutí zaplatit, kdy nyní vytvoření příkazu k úhradě a zaplacení dokladu může provést finanční asistent. Nyní je potřeba přistoupit ke **kvantitativní části analýzy procesu**, kdy se pomocí procesní simulace v *Bizagi Modeler*, bude zjišťovat, jak by tento procesní redesign mohl zvýšit efektivitu vykonávaného procesu zpracování závazků.

5.4.2 KVANTITATIVNÍ ANALÝZA UKÁZKOVÉHO PROCESU

Ve stejném programu (*Bizagi Modeler*), ve kterém byly as-is i to-be procesní modely vybudovány, lze testovat vykonávání procesu v tzv. *Simulation view* rozhraní. Velice rychle a jednoduše se dá potvrdit nebo vyvrátit procesní předpoklad. To je tedy další logický krok v případové studii.



Obrázek 5-6: To-be procesní model procesu zpracování závazků.

Zdroj: vlastní zpracování

Samotné nastavení procesní simulace probíhá ve 4 krocích. V prvním se procesní model **validuje** (*Process validation*) v tomto kroku se stanovuje pouze počet přichozích tokenů a přerozdělení v branách. Validace je důležitá na otestování, jestli je proces správně vybudován. V dalším kroku – **časová analýza** (*Time analysis*) se měří časy zpracování u jednotlivých aktivit v procesu. U každé aktivity lze stanovit tzv. **statistická distribuci**, která zpřesňuje simulační výsledky. Počítá se zde s neomezeným množstvím zdrojů. Předposledním

krokem je **analýza zdrojů** (*Resource analysis*). Zdroj zde nabývá podoby lidského nebo hmotného (vybavení, zařízení). Takže se aktivitám přidělují procesní účastníci a popř. i vybavení (v případě, že jeho využívání něco stojí). Cena zdrojů může být fixní nebo variabilní – cena za hodinu nebo za vykonání procesu. Přidělení zdrojů omezuje vykonatelnost procesů, protože se v procesu začíná projevovat čekací doba (*waiting time*), kterou token (doklad) musí čekat na zpracování. Poslední krok je **analýza kalendářů** (*Calendar analysis*), v které se pro jednotlivé zdroje přiřazují pracovní kalendáře. Formují další omezení pro zdroj, protože se zde počítá s pracovní dobou zaměstnanců a mohou se zde projevovat i různé pracovní volna, pauzy, a další prostoje.



PRO ZÁJEMCE

V simulační prostředí *Bizagi Modeler* fungují různé typy **distribucí** pro lepší vyobrazení reality v procesních simulacích. Aktivity v procesu častokrát nemají fixní dobu zpracování, tzn., každý případ se liší. Např. schválení faktury trvá někdy 3 a někdy 5 minut. Proto nám slouží distribuce, aby bylo možné se vypořádat s takovými rozdílnostmi. V případové studii jsou použity dva druhy distribucí. **Poissonova distribuce**, která se používá při zcela náhodném zpracování, které nemá žádné závislé proměnné. Tzn., události se nezávisle na sobě vyskytují pouze s fixní průměrnou mírou. Další typ je **triangulární distribuce**, kdy se stanovuje minimální a maximální hodnota pro dobu zpracování a také pravděpodobná hodnota, která je v mezích minima a maxima a výsledky se poté pohybují v tomto rozmezí.

Z výsledků z rozhovorů a pozorování, které proběhly ve fázi objevování procesů, jsou známy tyto údaje, které slouží jako vstupy do nastavení procesní simulace:

- Aktivita přijetí dokladu i s jeho distribucí finančnímu asistentovi zabere většinu času okolo 2 minut, ať už doklad dorazí v elektronické nebo fyzické podobě. Je zde nastavená doba zpracování na **fixní dobu** 2 minut.
- U ověření dokladu již nelze přesně stanovit kolik aktivita zabere času, protože záleží podobě dokladu a např. množství položek v něm. Proto se nastavuje **Poissonova distribuce** s průměrem 1,5 minuty.
- Z **85 %** doklad koresponduje s objednávkou, což lze vyčíst z ERP softwaru. V **15 %** případů se doklad posílá zpět dodavateli. Sub-proces navrácení dokladu zpět dodavateli se ani v rámci simulací procesů neřeší.
- Potřeba dalšího schválení od administrativního manažera vzniká ve **34 %** případů. Tzn., že **66 %** putuje rovnou do aktivity aktualizace ERP.

- Časová náročnost aktivity schválení produktu/služby se liší v závislosti na typu produktu nebo služby, která má být schválena. Inspekce produktu nebo verifikace služby se zpracovává v rozmezí od 3 minut do 10 minut, ale u většiny produktů to bývá minut 5. Proto se zde zvolila **triangulární distribuce** s minimem 3 minuty maximem 10 minut a nejpravděpodobnějším časem zpracování **5 minut**.
- Z výstupů z ERP softwaru **85 %** produktů/služeb bývá v předcházející aktivitě schváleno. **15 %** tedy putuje zpět dodavateli.
- Aktualizace ERP softwaru se s předchozí schvalovací aktivitou řadí mezi nejnáročnější z pohledu času. Opět zde záleží na množství položek v dokladu, ale průměrně se doba zpracování pohybuje kolem 4 minut na jeden zpracovaný doklad. Opět tedy byla zvolena **Poissonova distribuce** s průměrem 4 minuty.
- Kontrola splatnosti dokladu je otázka **0,5 minuty**, protože finanční asistent pouze rozklikne zvolený doklad a zkontroluje datum splatnosti. Z týdenních reportů již má přehled, jak je na tom firemní cash flow a podle toho fakturu posouvá k zaplacení nebo zde proces končí.
- U rozhodnutí o provedení platby to zde vychází zhruba půl na půl. Tzn., že u **50 %** doklad splňuje podmínku pro zaplacení, u ostatních tato podmínka naplněna není hlavně kvůli vzdálené době splatnosti.
- To-be procesní model zde má další bránu, která vyjadřuje objem dokladů, které jsou **nad 10 tisíc korun**. Nad tuto hranici se dostává pouze **25 %** z celkového počtu dokladů. A u to-be procesního modelu tedy nově vydefinovanou cestou putuje **75 %** dokladů.
- Vytvoření platebního příkazu velmi usnadňuje ERP software, a proto dobrá zpracování u této aktivity je pouze **1,5 minuty**. Zde nezáleží na složitosti faktury, protože ERP systém příkaz generuje automaticky.
- Kontrola a provedení platby přes internetové bankovníctví zabere účetní **2,5 minuty**.
- U to-be procesního modelu aktivita zaplacení dokladu, která obsahuje nejen vytvoření příkazu k úhradě, ale také úhradu samotnou zabere **3 minuty**. Této úspory je dosaženo vynecháním kontroly platebního, která jinak probíhala u účetního.

Sledovaným obdobím byl pracovní týden, a z ERP bylo vysledováno, že průměrné množství zpracovaných závazků za toto období je **215**. Takže úvodní počet vstupujících značek do simulace je nastaven rovněž na 215.

Velice rychle a jednoduše nastavená procesní simulace bude sloužit především jako podklad, zdali a jak se procesní redesign vyplatí. Také majitel procesu a další zájmové skupiny se budou lépe přesvědčovat pro vstup do implementační fáze, s jasnými čísly a benefity pro ně plynoucími.

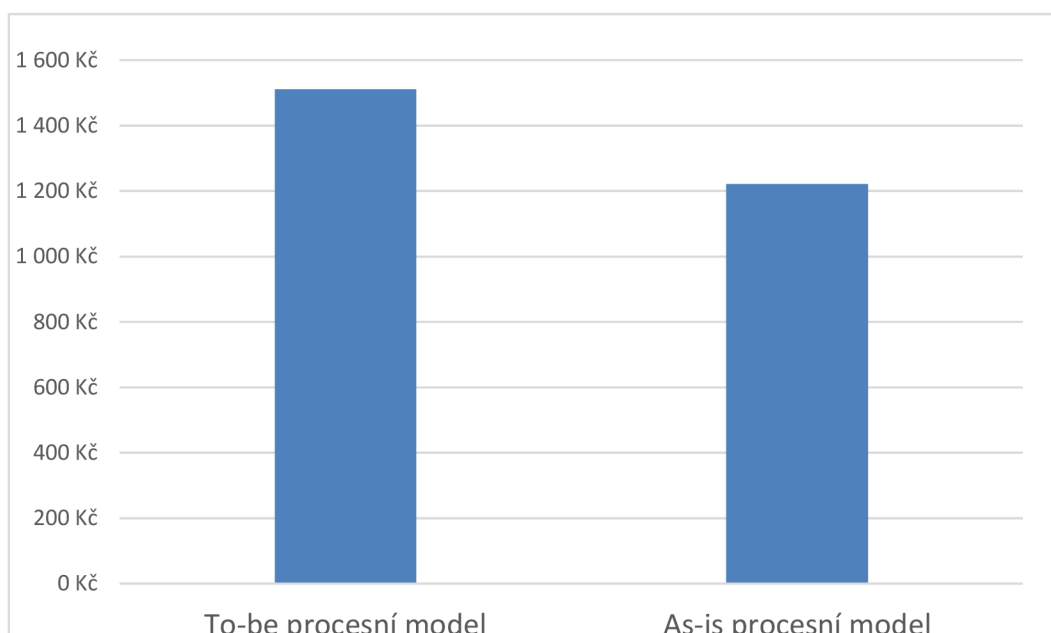
**What-if
analýza**

Pro vyhodnocení výsledků byla použita tzv. **what-if analýza**, která je součástí *Bizagi Modeler*, v které se porovnávají výsledky z dvou scénářů (as-is a to-be procesní model). Výstupy z *what-if* analýzy se vyexportují do excelovské tabulky, kde jsou oba scénáře a jejich výsledky přehledně znázorněny.



DEFINICE

Cílem **What-if analýzy** je prozkoumat chování komplexních systémů, jako třeba korporátní společnosti nebo její části, při stanovených hypotézách, které se nazývají scénáře. V podstatě *what-if* analýza měří, jak změny v množině nezávislých proměnných ovlivňují množinu závislých proměnných s odkazem na specifický simulační model. (Golfarelli, et al., 2006)



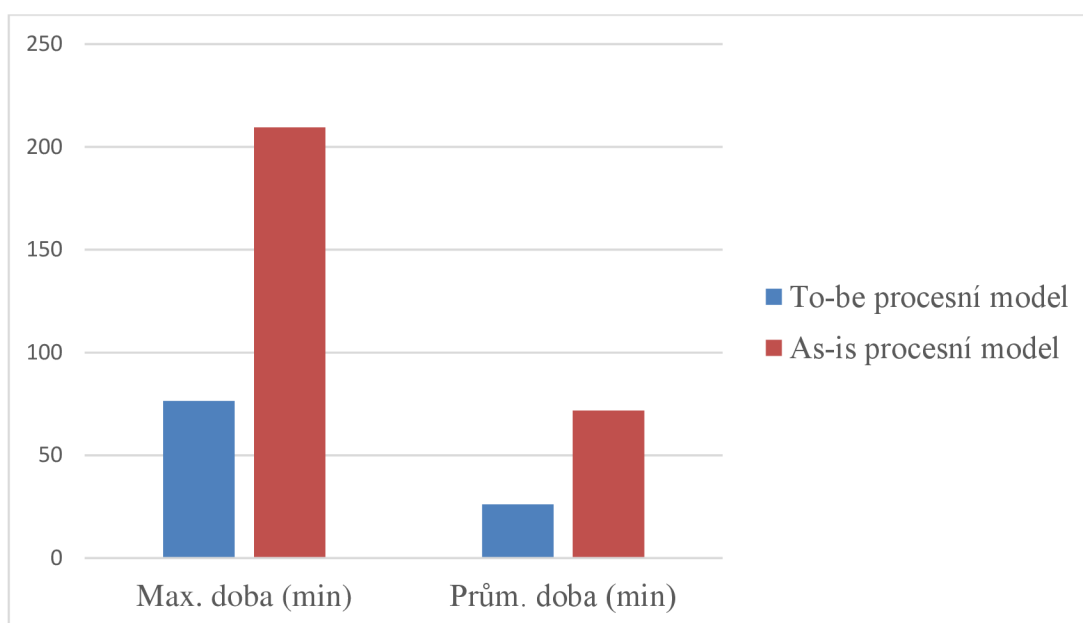
Obrázek 5-7: Náklady na finančního asistenta.

Zdroj: vlastní zpracování

První sešit ve vyexportovaných excelových výsledcích procesních simulací se zabývá využitelností zdrojů a náklady u těchto skupin zdrojů. Z výsledků analýzy tedy logicky vyplývá, že **zátěž** u finančních asistentů v to-be oproti as-is procesnímu modelu stoupla. Přesně z využitelnosti 1,96 % na 2,42 %. Tyto výsledky samozřejmě nejsou brány v kontextu ostatních procesů běžící na ekonomickém oddělení, a jehož jsou součástí procesní

účastníci ze sledovaného procesu. Proto procentuální vyjádření využitelnosti zdrojů odpovídá realitě, kdy mají na starosti i spoustu dalších aktivit. Jak lze logicky vyvodit u účetní využitelnost v to-be procesním modelu poklesla z 3,17 % na 2,4 %. Důležité je ovšem se na tyto rozdíly ve využitelnosti podívat z pohledu financí. Finanční asistent má sazbu 130,- CZK na hodinu a účetní 160,- CZK. Z toho vyplývá, že úspora času u účetního bude více nákladově efektivní než úspora času u finančního asistenta. U finančního asistenta náklady u to-be procesního modelu vzrostly o 289,- CZK, avšak vlivem delegace jedné z aktivit od účetní k finančnímu asistentovi poklesly náklady na účetního o 593,- CZK, což v celkovém součtu dává **úsporu 304,- CZK** u to-be procesního modelu.

Pro hlubší analýzu je potřeba se podívat na simulační výsledky jednotlivých aktivit a podívat se i na další aspekt a tou je doba zpracování závazků.

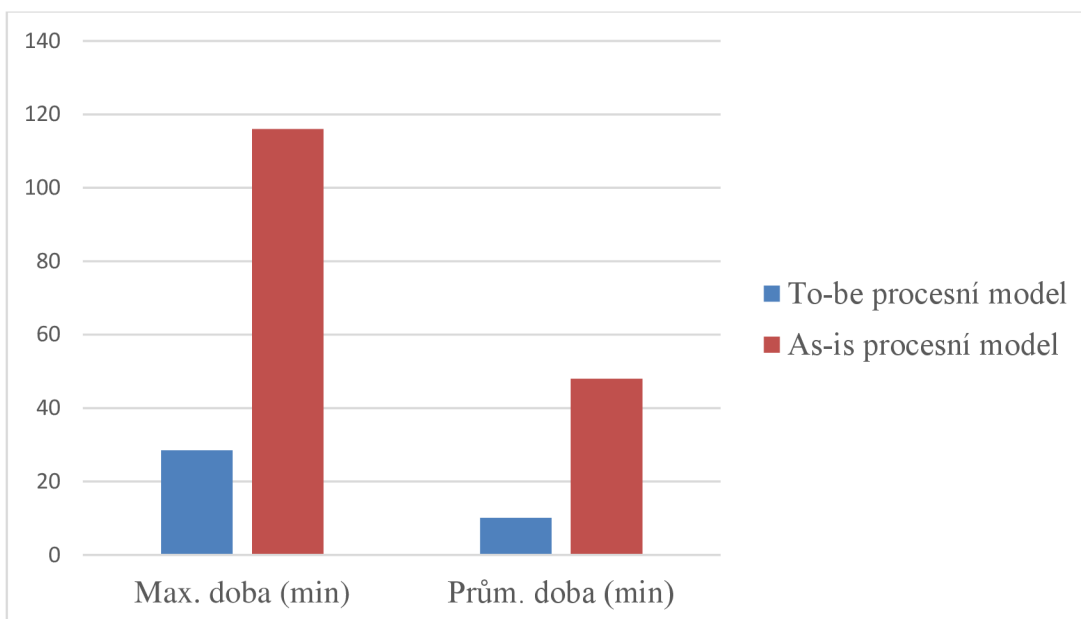


Obrázek 5-8: Doby zpracování u celkového procesu zpracování závazků.

Zdroj: vlastní zpracování

Výsledky jasně ukazují, jaká varianta je z pohledu času více výhodnější. U as-is procesního modelu se maximální doba, za kterou se ve sledované firmě zpracuje závazek, pohybuje na úrovni 209 minut. **Průměrná doba**, jež trvá dokladu od přijetí až po jeho potenciální zaplacení nebo dovršení jiného konce procesu, je **71 minut**. U to-be procesního modelu jsou výsledky o poznání lepší. Maximální doba zde činí 77 minut a **průměrná doba 26 minut**, což je vylepšení výkonu procesu o neskutečných **273 %**. Po další analýze simulačních dat procesní analytik došel k závěru, že za takto rozdílnými výsledky skutečně stojí vysoká míra přetížení účetní, která má na starosti jeden z nejnáročnějších aktivit – aktualizace ERP softwaru a do toho v poslední fázi procesu se stará o kontrolu platebních příkazů a následnou úhradu faktur, proto se častokrát stávalo, že se jedna z těchto aktivit zastavila a tím rostla doba zpracování celého sledovaného procesu. Tím pádem se jednalo o **úzké místo**, což se potvrdilo na datech.

Na posledním obrázku 5.9 lze vidět, jak se úzké místo vytrácí i když se přímo do této aktivity v to-be procesním modelu nezasáhlo. 75 % dokladů je pod částku 10 tis. CZK a tím pádem jej v to-be procesním modelu může uhradit finanční asistent, který nepodléhá tak velké míře vytížení jako účetní. To uvolnilo více času pro účetní na aktivitu aktualizace ERP, a tak se již nestává, že by se právě u tohoto úzkého místa proces zpomalil nebo úplně zastavil. To opět dokládají výsledky, kdy se maximální **doba zpracování dokladu** snížila ze 116 na 29 minut a **průměrná doba** ze 48 na 10 minut.



Obrázek 5-9: Doby zpracování u aktivity aktualizace v ERP systému.

Zdroj: vlastní zpracování

Na datech z procesní simulace se tedy potvrdilo, že to-be procesní model je daleko efektivnější, jak z pohledu času, tak i z pohledu nákladů, a proto se zvolil jako vstup do další fáze životního cyklu BPM, kterou je implementace procesů.

5.5 Implementace procesu zpracování závazků

Implementace bývá nejnáročnější fází v rámci životního cyklu BPM, a proto je potřeba, aby všechny předešlé fáze předcházely, jinak by mohla nastat situace, která již ve sledované firmě dříve nastala, a to je implementace nepřipraveného a neanalyzovaného procesu. Ve firmě se přeskočila jakákoli fáze analýzy procesu a ihned se postoupilo k implementaci a automatizaci as-is procesu. To samozřejmě není správný postup, protože celková efektivita tohoto procesu je nízká a tím pádem bude efektivita nižší i po implementaci či automatizaci tohoto procesu.

Implementace procesů již probíhá v rozhraní druhého programu z balíku *Bizagi* a tím je *Bizagi Studio*. Jak bylo řečeno, tato fáze bývá náročná, a to hlavně z IT pohledu. I když

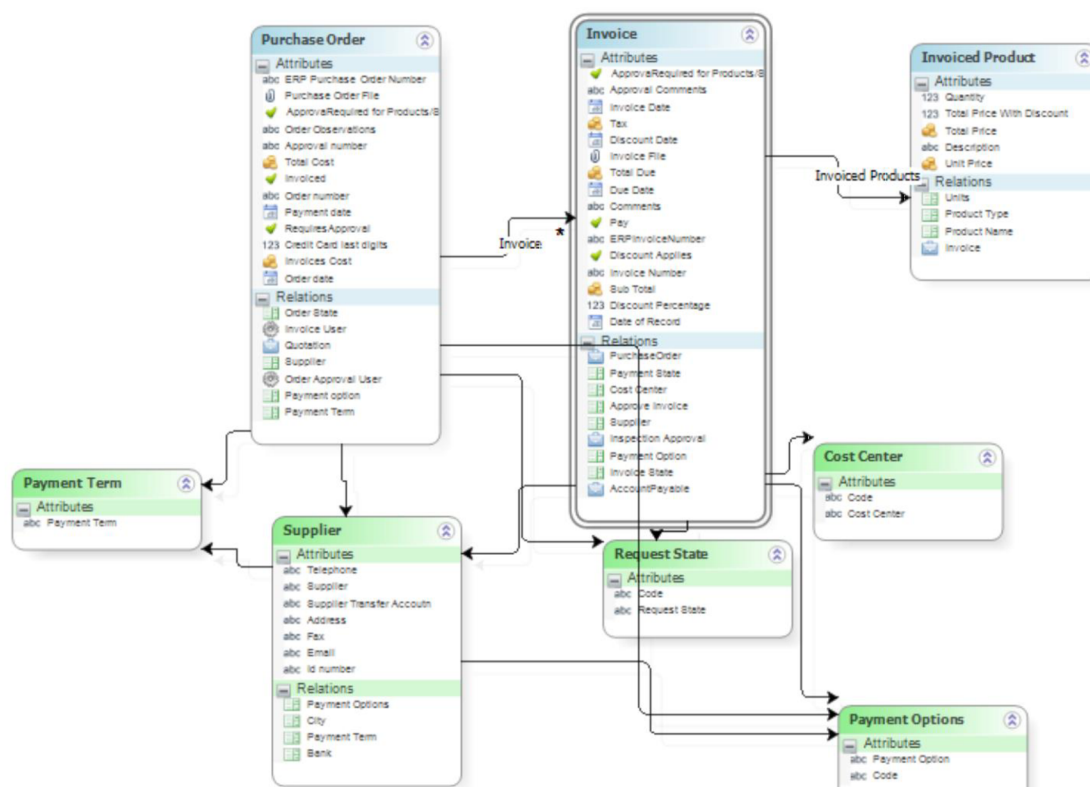
prostředí *Bizagi Studio* je velice přívětivé. Smyslem této části případové studie není vysvětlovat technické detaily implementace ukázkového procesu v *Bizagi Studio*, ale spíše nastínit **postup**, jakým tato implementace probíhá, jaké jsou její výstupy a celkový význam pro podnik.

Celá implementace je přímo navázána na procesní model z *Bizagi Modeler*, který se automaticky exportuje do *Bizagi Studio* a zde samotná implementace začíná. Je rozdělena do několika na sebe navazujících kroků, které budou v této části postupně představeny včetně obecné ukázky implementace procesu zpracování závazků.

5.5.1 DATA MODEL

Prvním krokem je vytvoření tzv. **Data modelu**. Ten obsahuje všechny informace, s kterými bude ukázkový proces dále pracovat a také zobrazovat v pracovním portálu a při samotné exekuci procesu od koncových uživatelů.

Hlavní entita v ukázkovém procesu je doklad (faktura). Hlavními atributy dokladu jsou např.: produkty, celková částka, daně, číslo dokladu, informace o dodavateli, datum vystavení, splatnosti a zdanitelného plnění, informace o způsobu platby atd. Doklad je provázán s objednávkou a produkty, viz obrázek 5-10.



Obrázek 5-10: Data model.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

Další možnost v tomto kroku spočívá ve vytvoření **parametrových tabulek**. Na obrázku 5-10 jsou v zeleném provedení a slouží k používání často se opakujících informací. V případě ukázkového procesu to je např. informace o dodavateli. Kdyby se parametrické tabulky nepoužívaly, musely by se informace o dodavateli vkládat pokaždé manuálně. Takhle se manuálně vytvoří pouze jednou a později se již pouze zvolí dodavatel a ostatní informace o něm se vyplní automaticky.

5.5.2 TVORBA FORMULÁŘŮ

Jakmile je data model připraven postoupí se k tvorbě formulářů spojených s každou jednotlivou aktivitou v procesu vyžadující lidskou interakci. Formuláře utváří rozhraní, které slouží pro zobrazování a vkládání požadovaných informací vyplývajících z data modelu, aby koncový uživatel (účastník procesu) mohl proces vykonávat. Designování formulářů používá pole zadána v data modelu a jednoduchým způsobem si lze přizpůsobovat vzhled formulářů dle vlastní potřeby

V ukázkovém procesu zpracování závazků lze tvorbu formulářů ukázat na příkladu, kdy u druhé aktivity v procesu – ověření dokladu, probíhá porovnání, jestli doklad sedí s objednávkou. Na obrázku 5-11 je zobrazeno navržené rozhraní, pro vyhledávání přijatých dokladů podle předem stanovených kritérií.



The image shows a web form with four input fields. The first field is labeled 'Order number:' and is an empty text box. The second field is labeled 'Order date:' and contains the placeholder text 'dd/MM/yyyy' with a calendar icon to its right. The third field is labeled 'Supplier:' and is a dropdown menu with a downward arrow. The fourth field is labeled 'Total Cost:' and is an empty text box.

Obrázek 5-11: Data model.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

5.5.3 OBCHODNÍ PRAVIDLA

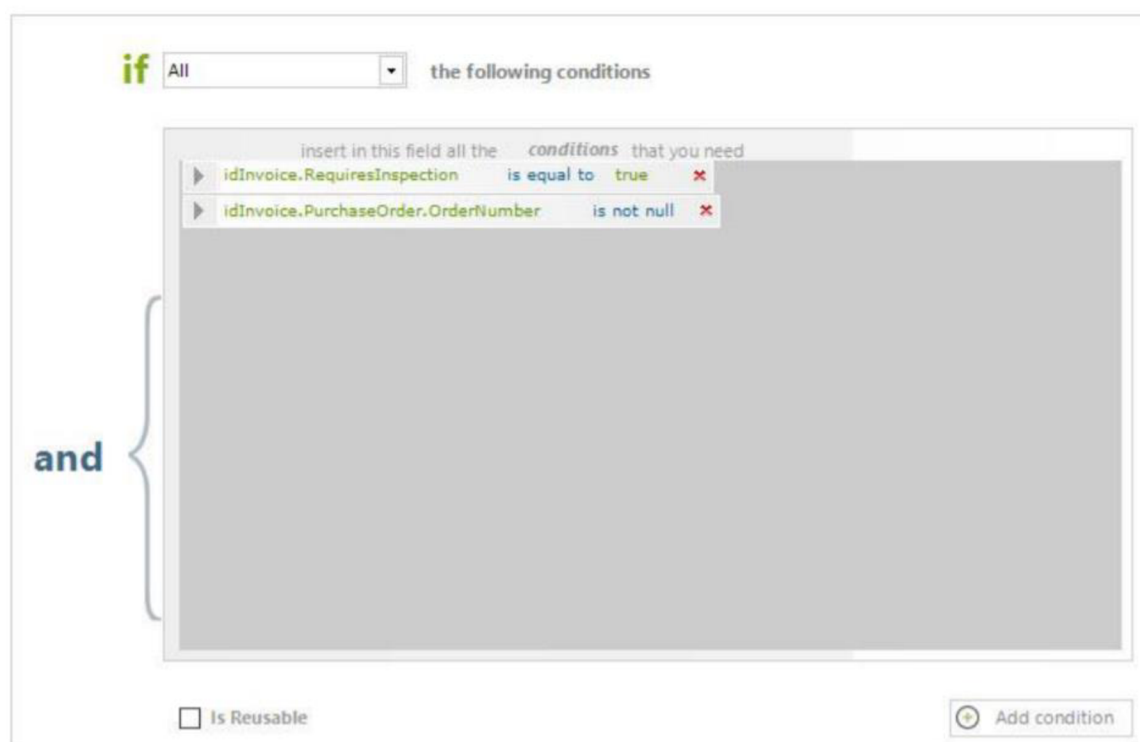
Další krok v implementaci procesu je definice obchodních pravidel, která kontrolují směřování procesu, vznikající u každých **bran**. První a pro tento proces nejdůležitější jsou pravidla přechodová. Určují, za jakých podmínek proces bude směřovat jakou cestou. Souvisí s branami v procesním modelu a mají hodnoty pravda nebo nepravda.

V ukázkovém procesu jsou pouze **exkluzivní brány** (XOR-split brány), proto se zde nastavování podmínek řídí stejnými principy. Na obrázku 5-12 je zobrazeno nastavování

pravidla pro druhou bránu v procesu, která určuje, jestli je potřeba schválení produktu/služby od administrativního manažera. Lze tedy vidět dvě podmínky jednou z nich je potřeba inspekce pro schválení, kde výsledek je buď pravda, nebo nepravda a druhá podmínka značí, že pokud doklad nesouvisí s žádnou objednávkou, aktivita schvalování produktu/služby není k dispozici.

Dále je v tomto kroku možné nastavovat tzv. *Activity actions* (činnosti), které opět slouží pro automatizaci některých mezikroků a pro ulehčení výkonu procesů uživatelům. Tvoří se zde pravidla pro automatické vyplnění požadovaných polí u opakujících se dat. V ukázkovém procesu to např. znamená, že při spárování objednávky s dokladem se některé informace z objednávky automaticky nahrají do faktury a naopak.

Činnosti



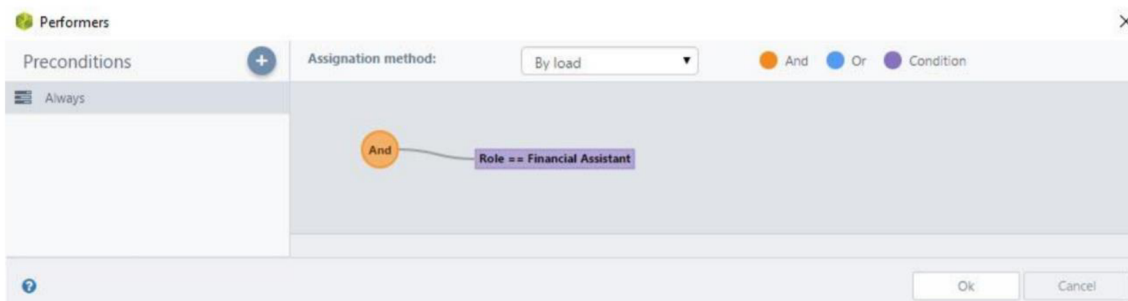
Obrázek 5-12: Tvorba obchodní pravidel.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

5.5.4 UŽIVATELÉ

Přidělování zdrojů je velice důležitá část v implementaci procesu. Zodpovědní vykonavatelé procesu dostávají v tomto kroku přidělené aktivity. Také zde probíhá identifikace vztahů existující mezi různými zaměstnanci, což vychází z charakteristických proměnných, které rozlišují jednotlivé zdroje. Díky této funkcionalitě *Bizagi* inteligentně alokuje aktivity různým zaměstnancům v podniku. Takže nejdříve je důležité určit, jak by se měly vhodně přidělovat aktivity.

V ukázkovém procesu je situace jednodušší, protože na každé pozici existuje právě jeden zdroj (zaměstnanec). Takže úvodní přidělování probíhá podle rolí, znázorněných v procesním modelu. Žádné další podmínky ani proměnné zde nastavovat není potřeba, protože jednotlivé aktivity bude mít k dispozici pro jejich exekuci právě jeden konkrétní zdroj.



Obrázek 5-13: Přidělování zdrojů.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

5.5.5 INTEGRACE

Praktický výkon procesu zpracování závazků by mohl probíhat i bez kroku integrace, ale znamenalo by to nutnost v ukázkovém procesu v průběhu vystoupit z prostředí *Bizagi* a vykonat aktivitu aktualizace ERP systému právě ve firemním ERP systému, což se nejeví jako efektivní. *Bizagi* nabízí možnost integrovat jej s dalšími firemními systémy, ať již formou předefinovaných **konektorů**, které jsou zdarma k stažení na webovém portále *Bizagi.com*, tak také prostřednictvím tzv. **API**, kde je již propojení daleko složitější, avšak může uspokojit individuální potřeby procesních účastníků.



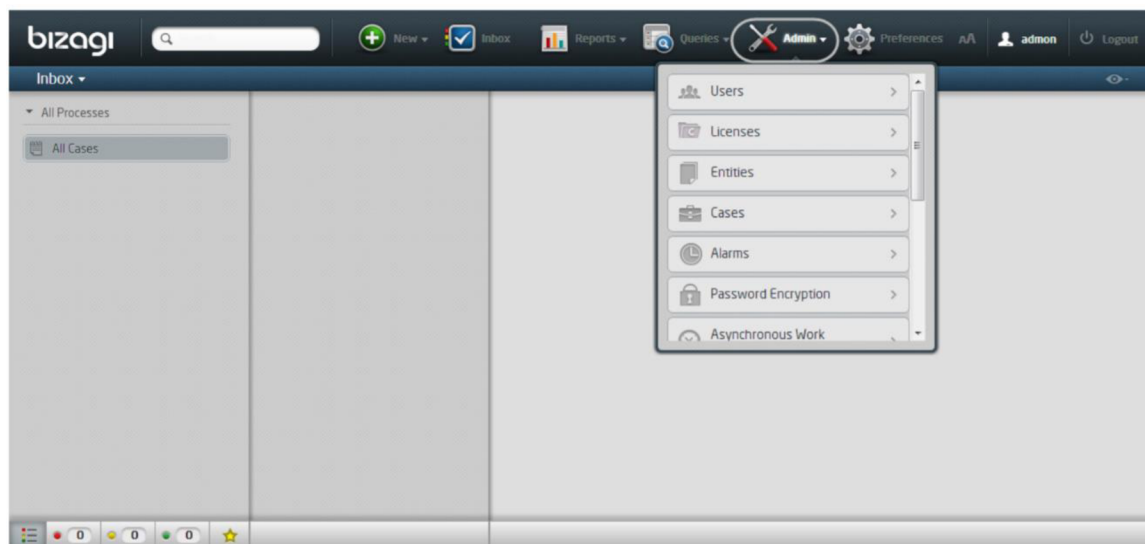
PRO ZÁJEMCE

API (Application Programming Interface) obecně se jedná o soubor přesně definovaných metod, pomocí kterých komunikují různé softwarové komponenty. Uvedeno na příkladu, komunikace mezi dvěma softwary: Budovaná aplikace (může být i *Bizagi*) najde současný stav měnového kurzu tím, že pošle strukturovanou zprávu do API webu *kurzy.cz* (pokud má API) a tato API následně odpoví obdobně strukturovanou zprávou naší aplikaci, která přijatou informaci již může využít dle předem definovaného postupu.

Jak tedy bylo řečeno jediná integrace v ukázkovém procesu je s ERP softwarem, kam se zanáší informace z dokladu. Konkrétně je potřeba, aby informace zanesené do *Bizagi* formuláře se automaticky a správně přenesly do ERP systému. Ve sledované firmě se o toto propojení postaral pracovník z IT oddělení a další podrobnosti zde nebudou rozebrány.

5.5.6 EXEKUCE PROCESU A PRACOVNÍ PORTÁL

Posledním krokem je nastavení pracovního portálu a **testování a validace** implementovaného procesu pro **ostré nasazení**. Testovací exekuce procesu a rozhraní pro procesní účastníky je znázorněno na obrázku 5-14. Pracovní portál tedy slouží pro všechny osoby zúčastněné na procesu a lze jej individuálně přizpůsobovat a definovat dle potřeby jednotlivců. Toto rozhraní ještě neslouží pro finální výkon procesů, to bude probíhat až v rozhraní programu *Bizagi Engine*.



Obrázek 5-14: Pracovní portál Bizagi Studio.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

V tomto kroku se taktéž vytváří **účty** pro jednotlivé zaměstnance a procesní účastníky, pod kterými se do pracovního portálu budou přihlašovat. Testování probíhá tak, že se vytvoří nový **případ (case)**, který značí přijetí dokladu do firmy. Celý proces se tak se všemi procesními účastníky projde ve všech možných scénářích a hledají se možné chyby a špatné či neefektivní nastavení. Pro celkový hladký průběh je v ukázkové firmě zapotřebí zorganizovat školení, které bude představovat způsob práce v pracovním portále *Bizagi*. Nejedná se o složitý systém a v kombinaci se znalostí ukázkového procesu procesními účastníky se adopce nového systému rychle uchytlí. Tím, že zatím žádné další procesy se v rozhraní *Bizagi* nerealizují, efektivita není zatím na tak vysoké úrovni, avšak s přibývajícím množstvím procesů, se bude zlepšovat.

Testování

5.6 Monitorování a kontrola ukázkového procesu

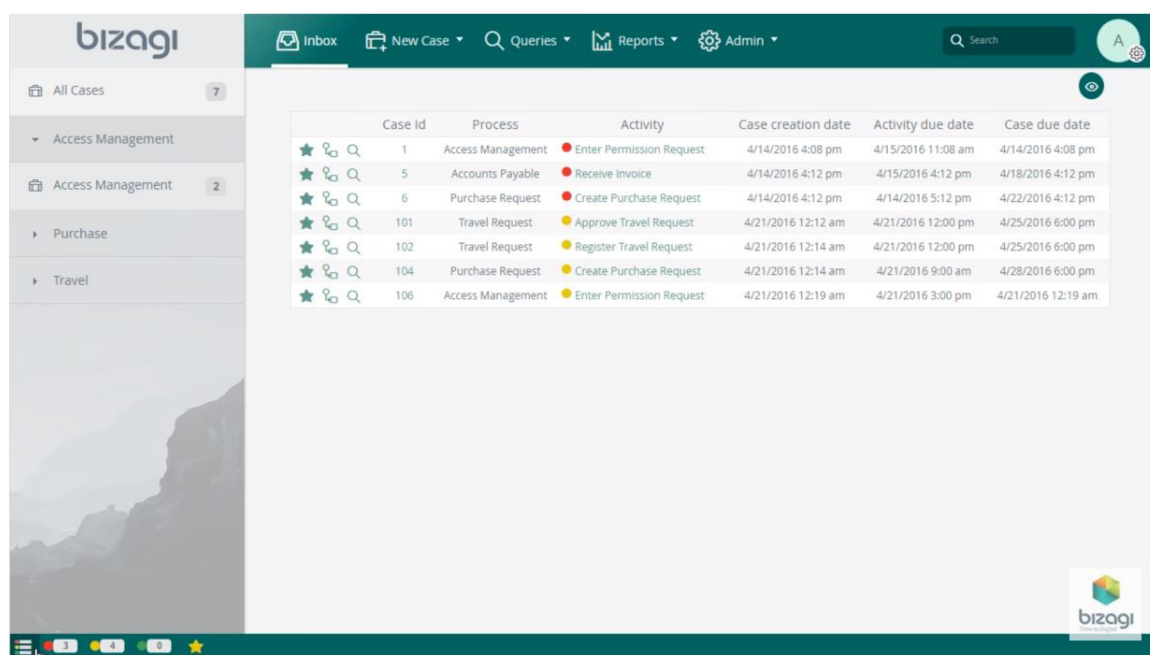
Poslední fáze životního cyklu BPM je zastoupena v posledním programu z balíčku *Bizagi* a tím je **Bizagi Engine**. Jak bylo zmíněno na začátku této kapitoly, jedná se o jediný program z balíku, který je placený. Nacenení probíhá individuálně, takže veřejně není

k dispozici žádný ani orientační ceník. Výhoda takto zvolené metody placení pro firmy spočívá v tom, že až pokud skrze použití *Bizagi Modeler* a *Bizagi Studio*, vidí přínosy pro podnik a procesní řízení v něm, pustí se do ostrého provozu a zavádění funkčních redesignovaných procesů a teprve v této fázi platí za využití služeb *Bizagi*.

Důležité je zmínit, že v této případové studii se již nebude pracovat s reálnými daty ze sledovaného podniku, protože finální implementace a ostrý provoz, se v době tvorby případové studie teprve připravoval – fáze implementace. Avšak v rámci udržení celistvosti případové studie je zapotřebí si ukázat i možnosti a prvky v *Bizagi Engine*, které se budou v budoucnu v případovém podniku používat.

Procesní aplikace

V rámci ukázkového procesu se po kompletní implementaci v *Bizagi Studio* přesuneme v doručení těchto funkčních **procesní aplikací** (rozhraní *Bizagi*) k jednotlivým zaměstnancům, kteří vykonávají roli procesního účastníka nebo procesního majitele u procesu zpracování závazků. *Bizagi Engine* umožňuje, aby tito zaměstnanci proces vykonávali, jak ze svých obvyklých pracovišť (PC) tak i z tabletů či mobilů, a to nejen v jejich stabilním pracovním místě. To proto podporuje mobilitu zaměstnanců včetně možnosti zavádět, v těchto dobách oblíbené, práce z domu nebo to v budoucnu využijí zaměstnanci, často cestující mimo prostory firmy, např. obchodní zástupci.



Obrázek 5-15: Pracovní portál Bizagi engine.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>

Na obrázku 5-15 je zobrazen **pracovní portál** aplikace *Bizagi Engine*, ve kterém, po finálním zavedení, všichni pracovníci operují. Je složen z několika částí a nabízí různé funkce velmi dobře použitelné a ulehčující práci. Na horní liště (hlavní menu) je např. zá-

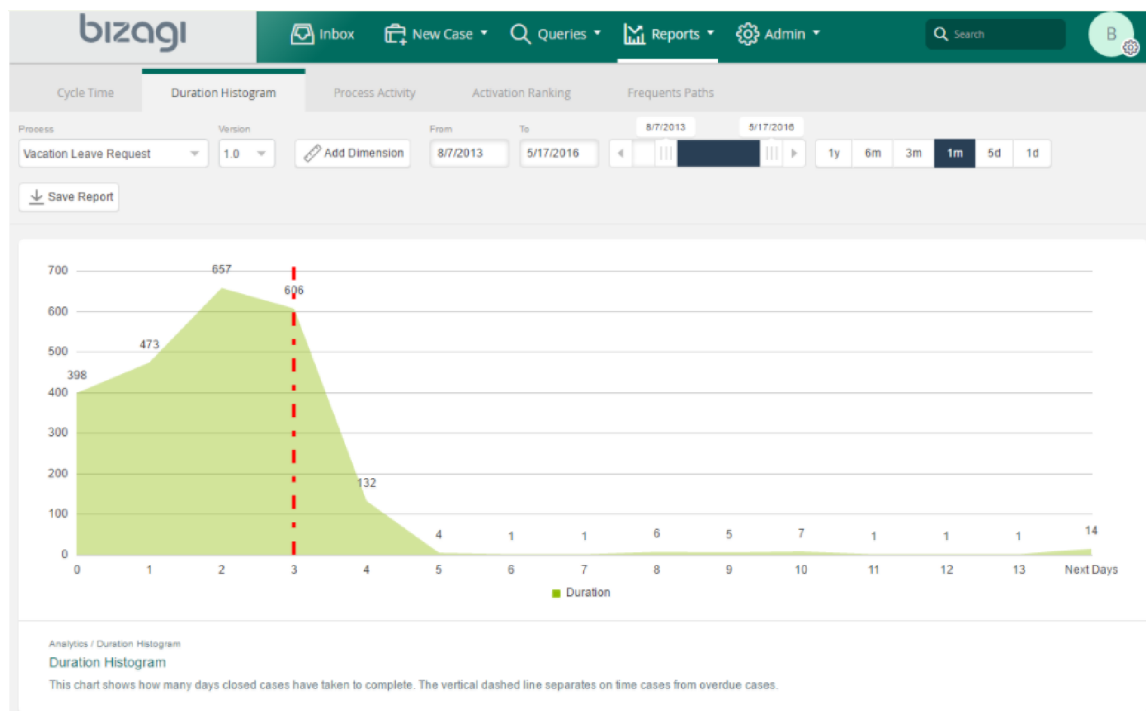
ložka *Inbox* (přímo zobrazená na obrázku 5-15). V ní se zobrazují aktivity čekající na zpracování, schválení nebo další akci od procesního účastníka přihlášeného do portálu. Lze je seřadit i dle priorit a ty se značí barvou. V levém sloupci (případy) lze vidět, do jakých firemních procesů je zaměstnanec zapojen. Jsou seřazeny do záložek např. podle typu nebo jiných kritérií. Samotná exekuce procesu je zde znázorněná v záložce nové případy (*New Case*). Tzn., že každé jedno zpracování dokladu v procesu zpracování závazků bude bráno jako samostatný **případ**.

Hlavním účelem z pohledu monitorování a kontroly procesu je určitě schopnost tvořit **reporty** a vyhodnocovat kompletní průběh sledovaných procesů podle velkého množství předefinovaných ukazatelů. V poli reportů je tedy možné monitorovat tyto výkonové ukazatele:

- **Čas cyklu** – souhrn počtu uzavřených (hotových) případů včetně výčtu dat za celý proces, jakými jsou např. průměrná doba zpracování, standardní odchylka atd.
- **Histogram** – zobrazuje, za jakou dobu se podařilo ve firmě sledovaný proces zpracovat. Vertikální přerušovaná linka (obrázek 5-16) odděluje včasné případy od zpožděných.
- **Procesní aktivita** – tento report pracuje s celkovými čísly. Takže kolik bylo otevřených případů celkem, kolik z nich se dokončilo, kolik bylo přerušeno a také zde lze zobrazit dlouhodobý trend v celkovém množství vykonaných procesů (případů).
- **Analýza frekventovaných cest** – další zajímavá data ukazují (na procesním modelu) jaké jsou nejčastější cesty, kterými se jednotlivé případy ubírají. To samozřejmě ukazuje, jako jsou nejběžnější cesty u většiny procesů a poskytuje to tak důležité informace k dalšímu rozhodování.
- **Analýza aktivit** – pro skutečně cenné vhledy je zapotřebí jít do vyšší úrovně podrobnosti procesu, a to na úroveň aktivit. Zde se tedy zobrazují data z výkonu jednotlivých úkonů a těmi jsou průměrná doba trvání, očekávaná doba trvání, počet úkonů provedených včas apod.
- **Analýza případů** – poslední reportingový výstup jde opět do hloubky procesu a je v něm možné vybrat konkrétní případ a podívat se blíže na jeho průběh z pohledu doby trvání jak celkového případu, tak i konkrétní aktivity v něm. Slouží pro analýzu odchylek a výjimek v procesu.

Veškeré reporty lze vyexportovat do excel formátu a dále s nimi pracovat. Po dokončení první iterace v životním cyklu BPM, by se v podniku měla podrobit celá iniciativa zpětnému vyhodnocení a také by se měl nastartovat další cyklus, který již bude budovat na vzniklé procesní architektuře a zkušenostech z předcházejícího cyklu. Postup zůstane

stejný, jenom se firma může rozhodnout, že do dalšího projektu zapojí více procesů. Síla BPM systému *Bizagi* spočívá v tom, že pokud se všechny operativní procesy podrobí podobnému postupu, jaký byl předveden v této případové studii, může celá firma fungovat na balíku *Bizagi* softwaru a stát se tak procesně orientovanou firmou s vyšší šancí na dosažení maximální provozní efektivity a úspěšné splnění dalších vytyčených cílů.



Obrázek 5-16: Report histogramu trvání.

Zdroj: upravené podle <https://www.bizagi.com/processcentral/Documents/a492d863-2d50-4170-a4d9-3e28e062ff16/docs/Accounts%20Payable%20-%20Const.pdf>



ŘEŠENÁ ÚLOHA

Řešení úlohy z podkapitoly 5.2.1:

Vybrané firemní procesy:

- Nábory zaměstnanců - podpůrný proces
- zpracování dokladů – podpůrný proces
- prototypování nového produktu – klíčový proces
- tvorba plánu výroby – řídicí proces
- vytvoření požadavku na nákup – podpůrný proces

- hodnocení kvartální výkonnosti podniku – řídicí proces

I když by se mohlo zdát, z pohledu vzájemného vztahu, organizační struktura nijak nesouvisí s procesní architekturou. Procesní architektura se buduje za předem jasných pravidel a podle specifické metodiky, oproti tomu organizační struktura může nabývat různých podob v závislosti na zvoleném manažerském přístupu vedení podniku. Organizační struktura slouží jako vhodný materiál vstupující do procesu budování procesní architektury, avšak vzájemný vztah zde nefunguje – tzn. nelze z organizační struktury odvodit procesní architekturu a obráceně.

ŘEŠENÁ ÚLOHA



Řešení úlohy z podkapitoly 5.4.1:

- Vyplnění požadavku (předák výroby) – aktivita přidávající hodnotu
- Poslání požadavku na ekonomické oddělení (předák výroby) – aktivita nepřidávající hodnotu
- Otevření a přečtení požadavku (administrativní pracovník) – aktivita nepřidávající hodnotu
- Výběr chybějícího vybavení (administrativní pracovník) - aktivita přidávající hodnotu
- Kontrola dostupnosti vybavení (administrativní pracovník) - aktivita přidávající hodnotu
- Tvorba požadavků na nákup (administrativní pracovník) - aktivita přidávající hodnotu
- Zaslání požadavku vedoucímu výroby (administrativní pracovník) – aktivita nepřidávající hodnotu
- Komunikace nejasností (vedoucí výroby) – aktivita přidávající hodnotu firmě
- Přeposlání požadavku zpět administrativnímu pracovníkovi (vedoucí výroby) – aktivita nepřidávající hodnotu
- Tvorba objednávky (administrativní pracovník) – aktivita přidávající hodnotu firmě

- Zaslání objednávky dodavateli (administrativní pracovník) – aktivita přidávající hodnotu
-



OTÁZKY

1. Z jakých částí se skládá balík *Bizagi* a k čemu se používá?
 2. Jakým způsobem se přistupuje k budování as-is procesního modelu?
 3. K čemu se v praxi využívají simulace podnikových procesů?
 4. Jak probíhá implementace procesu v *Bizagi*?
 5. Jaké jsou hlavní ukazatele, které je vhodné sledovat při analýze výkonnosti procesů?
-



SHRNUTÍ KAPITOLY

V této kapitole bylo představeno praktické využití životního cyklu BPM prostřednictvím případové studie a za pomoci softwaru *Bizagi*. Každá fáze životního cyklu BPM je zde zastoupena včetně technik a nástrojů používající se v procesním řízení v praxi.



ODPOVĚDI

1. Podkapitola 5.1.
 2. Podkapitola 5.2.
 3. Podkapitola 5.4, str. 84.
 4. Podkapitola 5.4.2.
-

6 PROCESS MINING

RYCHLÝ NÁHLED KAPITOLY



V posledních letech si podniky uvědomily, kolik užitečných informací se skrývá ve velkých objemech dat, které buďto samy produkují, anebo ke kterým mají přístup. Z tohoto důvodu se podniky uchýlily k zařazení datových věd do svého arzenálu nástrojů, kterými se snaží zajistit prosperitu sobě i svému okolí, zvýšit svou konkurenceschopnost a efektivitu, snížit své náklady, zabránit plýtvání zdroji apod. To vše nás přivádí k pojmu process mining.

Process mining je disciplína sjednocující BPM a datové vědy. Tvoří tak jakýsi most mezi procesní a datovou vědou. Jednoduše řečeno, process mining se snaží získat informace o podnikových procesech ukryté v podnikových datech. Tato data neboli tzv. události jsou zaznamenávány podnikovými informačními systémy, které jsou stále více a více propojeny s operačními procesy podniků. Process mining tak poskytuje celou řadu nových metod, které lze využít pro modelování či redesign podnikových procesů, a je schopen odstranit některé nedostatky metod z předchozích kapitol. V této kapitole se tak seznámíme se základními pojmy vážícími se na process mining.

CÍLE KAPITOLY



Po prostudování této kapitoly budete umět:

- Objasnit podstatu process miningu.
- Popsat data potřebná pro process mining a určit zdroje těchto dat.
- Objasnit základní typy process miningu a využít Petriho sítě pro reprezentaci modelů.
- Vysvětlit podstatu vybraných process miningových metod.



KLÍČOVÁ SLOVA KAPITOLY



Process mining, objevování procesů, zjišťování shodnosti, Petriho sítě, záznamy, stopy, události.

V následující podkapitole si představíme základní pojmy proces miningu. Tyto základní pojmy si uvedeme v kontextu BPM životního cyklu (více kapitola 4). Představíme si základní typy proces miningu – **objevování, zjišťování shodnosti a zlepšování podnikových procesů**. Vzhledem k účelu tohoto textu se budeme do detailu věnovat pouze prvním dvěma typům, a sice objevování a zjišťování shodnosti. Zlepšování podnikových procesů zmíníme jen okrajově.

6.1 Podstata a základní pojmy process miningu

PAIS

Process mining je relativně mladá disciplína, jejíž počátky sahají do konce devadesátých let minulého století. Účelem process miningu bylo vyplnit pomyslnou díru mezi přístupy *machine learningu* a *data miningu* na jedné straně a procesního modelování a analýzy na straně druhé. Jelikož zatímco *machine learning* a *data mining* nebraly v potaz podnikové procesy, procesní modelování a analýza nepracovaly s dostupnými daty. Process mining tak spojuje procesní a datový přístup v jeden. Tento přístup nabyl na významu zejména v posledních letech v důsledku nových technologií (především ICT) a jejich penetrace napříč podnikatelskými jednotkami. Tyto podnikatelské jednotky generují velké objemy dat, u kterých roste potřeba jejich zpracování. Data potřebná pro process mining jsou generována skupinou informačních systémů zvanou **Process-Aware Information Systems (PAIS)**.



DEFINICE

PAIS zahrnují takové informační systémy, které podporují celý proces namísto pouze izolovaných aktivit. Příklady takových informačních systémů mohou být ERP (Enterprise Resource Planning) systémy jako notoricky známý SAP, CRM (Customer Relationship Management) systémy, WFM systémy a další. PAIS systémy produkují potřebná data přímo, zatímco u ostatních typů informačních systémů může být zapotřebí je předzpracovat do vhodné podoby.



PRO ZÁJEMCE

Pro zájemce, kteří by se chtěli o PAIS dozvědět více, odkazujeme na práci Dumas, Aalst a Hofstede (2005).

Za průkopnické v oblasti process miningu můžeme označit práce Cooka a Wolfa (1996) a také Agrawala, Gunopulose a Leymanna (1999). Cook a Wolf se věnovali objevování softwarových procesů z na událostech založených dat, a právě oni přišli s technikou objevování procesů. Ve svém článku srovnávali jimi vyvinutou statisticko-algoritmickou techniku zvanou **Markovova metoda**, kterou porovnávali se dvěma dalšími metodami přebíranými z dalších dvou domén, které byly upraveny pro potřeby objevování procesů. Dalšími dvěma metodami byla tak zvaná KTail metoda a RNet metoda. Zatímco KTail metoda byla čistě algoritmická, RNet metoda byla čistě statistická. Autoři dospěli k názoru, že KTail metoda a Markovova metoda jsou slibným začátkem, zatímco RNet metoda založená na **neuronových sítích**, ještě není zralá pro praktickou aplikaci. Autoři také předpověděli, že největší potenciál vidí v algoritmickém přístupu. Tuto jejich domněnku můžeme s odstupem času potvrdit, jelikož naprostá většina doposud vyvinutých metod je založena na algoritmickém přístupu nebo je jeho kombinací. Agrawal, Gunopulos a Leymann (1999) představili ve své práci algoritmus, který byl schopen modelovat podnikový proces v podobě grafu na základě existujících logů. Autory výše můžeme označit za průkopníky, kteří stáli u zrodu této disciplíny. Nicméně nejvlivnější postavou v oblasti process miningu byl a stále je **profesor Aalst**, a to zejména kvůli komplexnímu přístupu k tomuto tématu, a to nejen jeho přístup, ale i přístup skupiny, kterou v této oblasti vede na Technické univerzitě v Eindhoven. Pojem process miningu poprvé představili ve své práci Weijters a Aalst (2001).

DEFINICE



Process mining, je výpočetní proces zaměřený na identifikaci skrytých vzorů a trendů vyskytujících se v datech, umožňující modelování a analýzu podnikových procesů na základě dat ve formě záznamů. Process mining se zaměřuje na **zvyšování efektivity a porozumění podnikových procesů**.

Ještě než si představíme tři základní typy process miningu, pojdme si přiblížit, na čem jsou vlastně process miningové analýzy zpracovávány. Abychom byli schopni provést process miningovou analýzu, potřebujeme, aby se data vyskytovala v potřebném formátu. Taková data jsou v rámci process miningu označována jako **záznamy**. Předpokládejme, že je možné sekvenčně zachytit události tak, že každá událost odpovídá jedné aktivitě a je spojena s právě jedním **případem** neboli **instancí procesu**. Skupině událostí jedné instance procesu poté říkáme **stopa**. Uvědomme si, že tatáž stopa může reprezentovat několik případů. Množinu případů reprezentovaných jednotlivými vykonanými aktivitami poté označujeme jako záznam.



PŘÍPADOVÁ STUDIE

Uvažujme jednoduchý proces výběru hotovosti z bankomatu a následující množinu čtyř aktivit, které nám tento bankomat umožňuje provádět {Přihlášení, Výběr hotovosti, Kontrola zůstatku, Odhlášení}. Příkladem stopy poté může být $\sigma_1 = \langle \text{Přihlášení, Výběr hotovosti, Odhlášení} \rangle$ nebo $\sigma_2 = \langle \text{Přihlášení, Výběr hotovosti, Kontrola zůstatku, Odhlášení} \rangle$. Jak můžete vidět v našem jednoduchém procesu, stopa σ_1 se nejspíše vyskytne v několika případech, neboli u několika zákazníků vybírajících hotovost z bankomatu. Množina všech stop registrovaných bankomatem potom bude tvořit záznam.

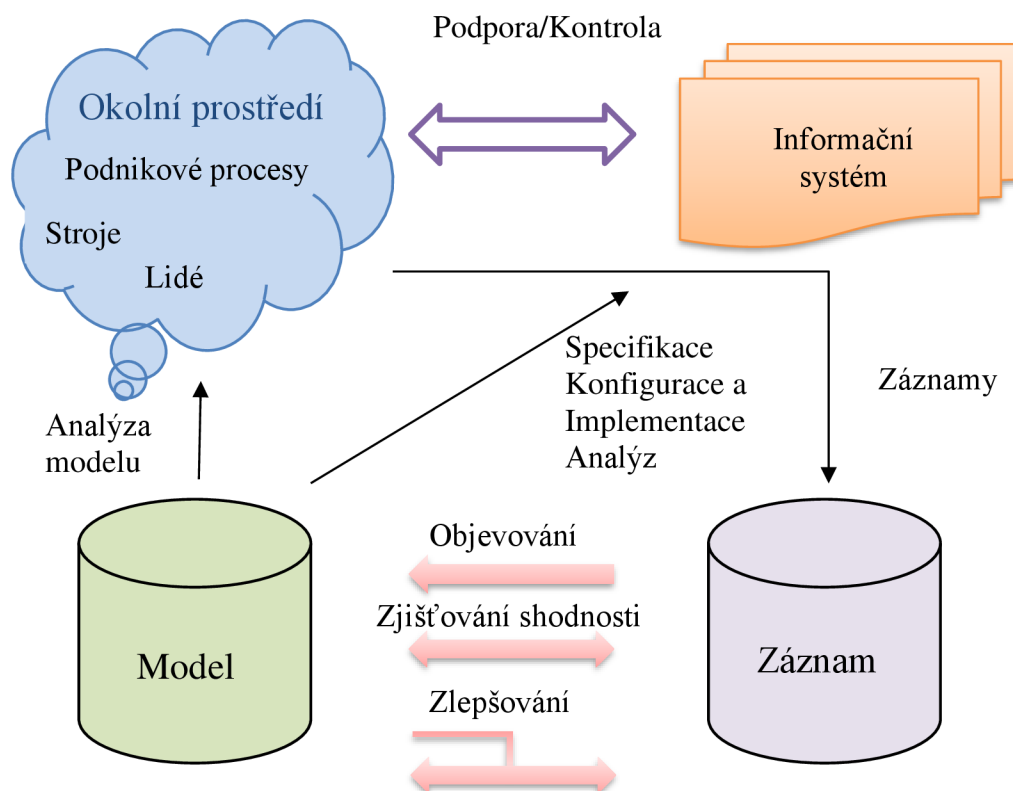
Otázce dat a záznamů se budeme podrobněji věnovat v následující podkapitole, ale jelikož již máme alespoň základní pojetí, s jakými typy dat se v process miningu budeme setkávat, pojďme si představit tři základní typy process miningu – objevování podnikových procesů, zjišťování shodnosti podnikových procesů a zlepšování podnikových procesů.

Objevo-
vání pro-
cesů

Prvním typem process miningu je tzv. **objevování podnikových procesů**. V tomto případě je na základě záznamu vyprodukovan model procesu, a to bez jakékoliv další informace o daném procesu. V následujících podkapitolách si do detailu představíme jeden z prvních algoritmů tohoto typu – **alfa-algoritmus**. Tento algoritmus je schopen vygenerovat model procesu v podobě Petriho sítě na základě použitého záznamu. Další přístupy k objevování podnikových procesů představíme již jen v principu a méně detailně.

Zjišťování
shodnosti

Druhým typem process miningu je **zjišťování shodnosti**. V tomto případě se již existující model porovnává s reálným záznamem a zjišťuje se, jestli tento existující model odpovídá realitě a obráceně. Zjišťování shodnosti je tak využíváno zejména pro detekci, lokalizaci a objasnění odchylek a měření závažnosti těchto odchylek. Představte si, že management podniku stanovil pravidlo, aby při reklamaci u výrobků v hodnotě nad 5 000 Kč nebyly v případě jeho poškození vráceny peníze bez souhlasu technického pracovníka. S využitím metod zjišťování shodnosti lze ověřit, zdali je toto pravidlo zaměstnanci podniku opravdu dodržováno či nikoliv.



Obrázek 6-1: Pozice tří základních typů process miningu: objevování, zjišťování shodnosti a zlepšování podnikových procesů

Zdroj: upraveno podle Aalsta (2016)

Třetím typem process miningu je **zlepšování**. Podstatou tohoto přístupu je rozšíření či vylepšení existujícího procesu s využitím informací o tomto modelu vyskytujících se v záznamu o tomto procesu. V rámci zlepšování podnikových procesů jsou tedy využívány i výsledky předchozích dvou typů process miningu. V následující kapitole si ukážeme, že záznamy, kromě dat nutných pro process mining mohou obsahovat také záznamy jako časová značka, zdroj a podobně. Tato dodatečná data umožňují zkoumat další perspektivy – jako časová, organizační či případová, které lze využít při zlepšování podnikových procesů. Tyto perspektivy lze samozřejmě využít i u předchozích dvou typů process miningu. Některým perspektivám, jako například analýze sociálních sítí, se budeme v pozdějších kapitolách.

Zlepšování

Existuje ještě čtvrtý typ process miningu zvaný **operační podpora**, který začíná nabývat na významnosti zejména v posledních letech. V předchozích typech se pracuje s daty již ukončených procesů, tedy s daty z minulosti. Zatímco v operační podpoře se pracuje s daty z právě probíhajících procesů. Tato data bývají často doplněna také o data z minulosti, jelikož je často vhodné v rámci operační podpory pracovat také s informacemi získanými z předchozích tří typů. Nicméně operační podpoře se v tomto skriptu již více věnovat nebudeme. Jak můžete vidět z dosavadního povídání, process mining dnes zdaleka není pouze

Operační podpora

o objevování procesů, nýbrž process mining je schopen nabídnout vhodné nástroje v prakticky všech fázích životního cyklu BPM. Operační podpoře se však hlouběji věnovat nebudeme, jelikož je nad rámec tohoto textu.



K ZAPAMATOVÁNÍ

V rámci process miningu je velmi důležité odlišovat mezi modelem procesu a realitou zachycenou ve formě záznamu, proto rozlišujeme tři základní pojmy, které nemají v českém jazyce obdobu, a proto je pro naše potřeby přebereme z angličtiny.

*Play-In a
Play-Out*

Prvním z pojmů je **Play-Out**, který referuje klasické použití modelů procesů. Budeme-li uvažovat například Petriho síť, potom je možnost vygenerovat chování. Například budeme-li mít Petriho síť N_1 z kapitoly 6.3 přehráváním značek bychom byly schopni získat záznam L_1 . Play-Out se tak využívá pro analýzu a ustanovení správných podnikových procesů. Například jde s využitím vhodného softwaru, jako je Disco či ProM, simulovat chování daného modelu. Model bude opakovaně exekurovaný a v rámci simulace budou zachycovány různé statistiky.

Druhým pojmem je **Play-In**. V tomto případě máme k dispozici data například v podobě záznamu a cílem je zkonstruovat model. Play-In je tedy opakem Play-Out. pod pojmem Play-In si tak můžeme představit různé přístupy k objevování podnikových procesů, jako například Alfa-algoritmus.

Pojem **Replay** je svým způsobem kombinací předchozích dvou pojmů, jelikož v tomto případě je k dispozici nejen záznam, ale i daný model. Daný záznam je přehráván na daném modelu. Replay je používán zejména v případě zjišťování shodnosti, konstruování prediktivních modelů či v případě operační podpory.



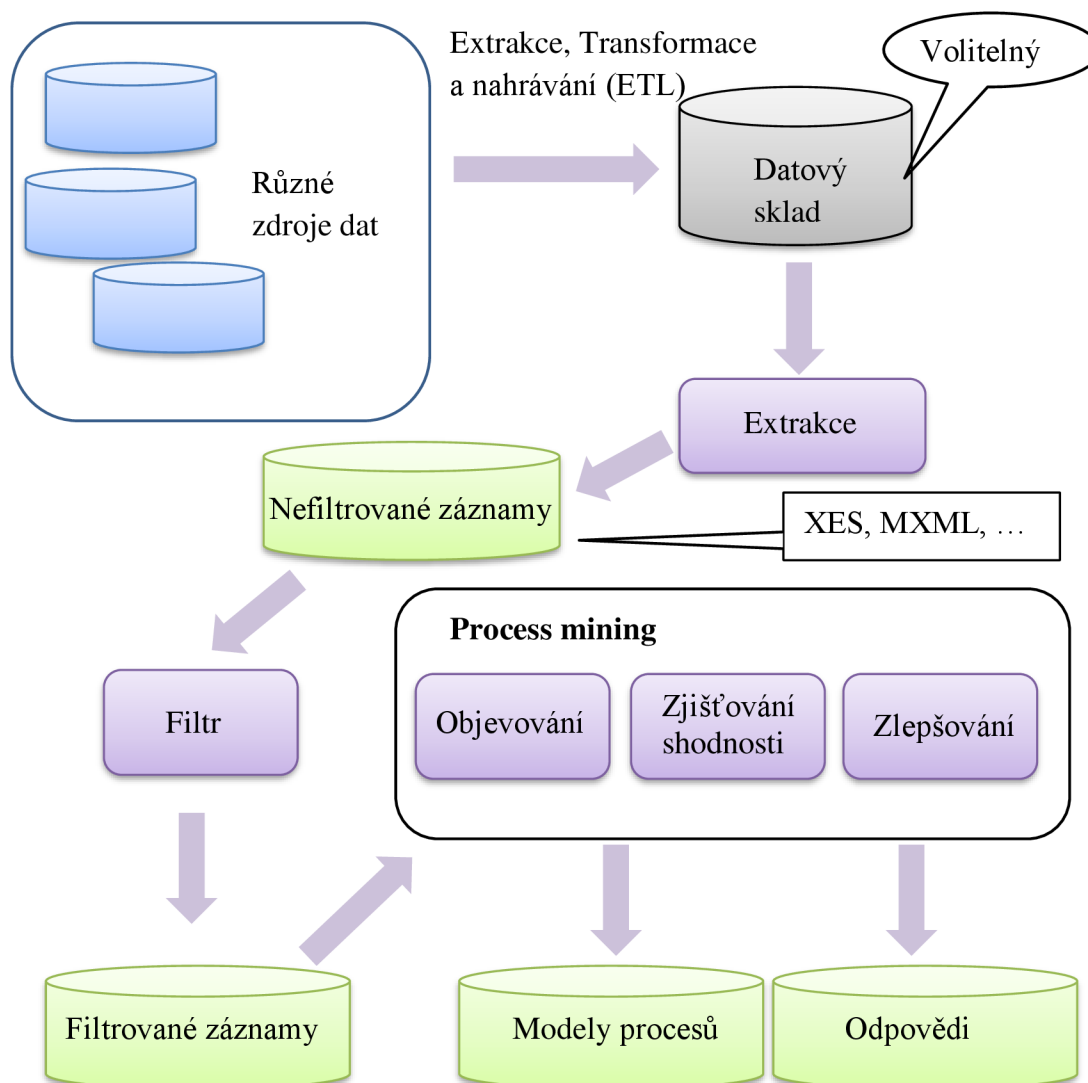
PRO ZÁJEMCE

V anglické literatuře jsou pojmy používané v tomto skriptu označovány následovně: záznamy jsou označovány slovy *log* nebo *event log*, události jsou označovány jako *event*, aktivity jako *activity*, případy neboli instance procesu jako *case* a stopa jako *trace*.

6.2 Data potřebná pro process mining

V předchozí kapitole jsme si přiblížili vznik process miningu a představili jsme si základní pojmy z této oblasti. Ještě než se pustíme do samotných metod, řekneme si něco více k datům potřebným pro process mining, co reprezentují, jakých formátů mohou nabývat,

co musí a nemusí obsahovat a další potřebné náležitosti. Je potřeba si uvědomit, že jelikož má process mining své kořeny v datových vědách, kvalitní data jsou základním pilířem process miningu. Kvalita dat tak značně ovlivní kvalitu dosažených výsledků. Budete-li tedy praktikovat process mining, musíte mít na paměti, že přípravě a práci s daty budete muset věnovat značnou část Vašeho časového fondu v rámci process miningových analýz.



Obrázek 6-2: Workflow od heterogenních dat k process miningovým výsledkům

Zdroj: upraveno podle Aalsta (2016)

Počátečním bodem pro process miningovou analýzu je extrakce často **nestrukturovaných dat** ukrytých ve všech možných datových zdrojích. Zdroji dat mohou být prosté databázové soubory, excelovské tabulky nebo relační databáze. Tyto datové zdroje se však mohou nacházet napříč několika odděleními či dokonce napříč několika firmami. Nestruk-

turovaná data mohou pocházet z například z webových stránek, emailů atd., ale i strukturovaná data z podnikových informačních systémů mohou být často oříškem vzhledem ke komplexnosti těchto systémů. Obrázek výše popisuje workflow spojené se získáváním process miningových výsledků z heterogenních dat. **ETL** (neboli *Extraction, Transformation and Loading*) popisuje v oblasti zvané **Business Intelligence** (BI) proces složený ze tří fází: (1) **extrakce** dat z vnějších zdrojů, (2) **transformace** dat do podoby potřebné pro další zpracování a (3) **nahrání** dat do cíleného systému (jako je například datový sklad). Datové sklady umožňují uskladnit operační a transakční data. Je potřeba si uvědomit, že tyto datové zdroje častokrát budou obsahovat i data nevztahující se k analyzovanému procesu, určení správného rozsahu dat, které z těchto zdrojů využijeme při tvorbě záznamů, je proto velmi důležité. Záznamy extrahované z různých datových zdrojů musejí být často podrobeny předzpracování v podobě různých filtrů a dalších metod ještě před samotnou process miningovou analýzou v důsledku jejich nestrukturovanosti. Některé filtry a další metody využívané pro předzpracování záznamů si představíme až v kapitolách věnovaných process miningovým softwarům Disco a ProM. Filtry jsou používány zejména v souvislosti s tzv. **špagetovými procesy**. Kromě špagetových procesů existují také **lasagna procesy**. Oba případy si představíme v podkapitole věnované softwaru Disco.

6.2.1 ZÁZNAMY

Tabulka níže reprezentuje typické informace obsažené v **záznamu** pro potřeby process miningové analýzy. Obecně je předpokládáno, že jeden záznam obsahuje data týkající se jednoho procesu. Toto souvisí s výše debatovaným rozsahem použitých dat. V tabulce níže jsou obsažena data generovaná v rámci reklamačního procesu. Každá událost, v našem případě každý řádek, musí připadat právě jednomu případu neboli instanci procesu (v tabulce níže každé je každá událost (ID události) přiřazena právě jednomu případu (ID případu). Dále předpokládáme, že každá událost může být reprezentována vykonanou aktivitou. Reprezentace událostí aktivitami je v rámci process miningu vcelku přirozená. Dalším nutným předpokladem je, že události jsou v rámci jednotlivých případů seřazeny, nejčastěji s využitím časových značek. Bez seřazení událostí samozřejmě není možné určit příčinné nebo kauzální závislost a tím model daného procesu.

Tabulka 6-1: Část záznamu reklamačního procesu - každý řádek reprezentuje jednu událost

ID případu	ID události	Časová značka	Aktivita	Zdroj	...
1	138521	15-10-2016:11.33	Registrace požadavku	Petr	...
	138522	16-10-2016:10.55	Důkladné prověření	Monika	...
	138523	16-10-2016:14.15	Kontrola účtenky	Marcel	...
	138524	16-10-2016:14.40	Vydání rozhodnutí	Sára	...
	138525	17-10-2016:08.30	Oznámení o zamítnutí	Petr	...
2	138531	15-10-2016:12.15	Registrace požadavku	Marcel	...
	138532	15-10-2016:12.45	Kontrola účtenky	Marcel	...

	138533	15-10-2016:13.05	Zběžné prověření	Monika	...
	138534	15-10-2016:13.35	Vydání rozhodnutí	Sára	...
	138535	16-10-2016:09.45	Vyplacení hotovosti	Agáta	...
3	138625	25-10-2016:14.32	Registrace požadavku	Petr	...
	138626	25-10-2016:15.06	Zběžné prověření	Monika	...
	138627	25-10-2016:16.34	Kontrola účtenky	Marcel	...
	138628	28-10-2016:09.18	Vydání rozhodnutí	Sára	...
	138629	28-10-2016:12.18	Oznámení o zamítnutí	Petr	...
	138630	28-10-2016:13.06	Znovuotevření požadavku	Sára	...
	138631	29-10-2016:11.35	Důkladné prověření	Monika	...
	138632	29-10-2016:12.26	Kontrola účtenky	Marcel	...
	138633	30-10-2016:12.23	Vydání rozhodnutí	Sára	...
	138634	30-10-2016:12.45	Vyplacení hotovosti	Agáta	...
4	138675	28-10-2016:15.02	Registrace požadavku	Petr	...
	138676	29-10-2016:12.06	Důkladné prověření	Jozef	...
	138677	29-10-2016:14.44	Kontrola účtenky	František	...
	138678	29-10-2016:15.44	Vydání rozhodnutí	Sára	...
	138679	30-10-2016:09.02	Oznámení o zamítnutí	Petr	...
...

Zdroj: upraveno podle Aalsta (2016)

Tabulka 6-1 tedy reprezentuje typickou strukturu záznamu. Mimo to lze z tabulky výše vyvodit základní předpoklady očekávané u každého záznamu:

- Každý proces je tvořen z případů,
- případy jsou tvořeny událostmi a to tak, že každá událost (ID události) se vyskytuje pouze v rámci jednoho případu (ID případu),
- události v rámci případů jsou seřazeny,
- události mohou mít atributy, jako například aktivita, časová značka, zdroj či cena.

Minimální informace, kterou musí záznamy obsahovat, jsou tedy jednotlivé případy a v rámci nich seřazené události. Nicméně události mohou mít i další vlastnosti a tyto vlastnosti jsou nazývány atributy. **Atributy** mohou poskytnout další zajímavé perspektivy na zkoumaný proces. Například časové značky nám umožňují náhled do výkonnostní složky procesu, zatímco informace o zdrojích jednotlivých událostí nám umožňují náhled do organizační perspektivy. Ne všechny události musejí mít stejnou množinu atributů, ale typicky události vázící se k jedné a té samé aktivitě mívají stejnou množinu atributů.

Df

DEFINICE

Nechť \mathcal{E} je množina všech možných událostí. Události mohou být charakterizovány atributy (např. událost může mít časovou značku, korespondovat s jistou aktivitou, být vykonána nějakou osobou atd.) Nechť AN je množina jmen atributů. Potom pro jakoukoliv událost $e \in \mathcal{E}$ a jméno $n \in AN$ je $\#_n(e)$ hodnota atributu n události e . Jestliže událost e nemá atribut n , potom $\#_n(e) = \perp$ (nulová hodnota).

Posloup-
nosti

Použijeme-li záznam v tabulce výše, pak $\#_{Aktivita}(138521) = Registrace\ požadavku$, $\#_{Zdroj}(138521) = Petr$, atd. Události jsou často referovány názvy příslušných aktivit, které reprezentují. Proto je nutné rozlišovat jednotlivé instance těchto aktivit, protože jak si za chvíli řekneme, každá událost se může v záznamu vyskytovat pouze jednou. Dále je třeba mít na paměti, že aktivity často, nějaký čas trvají a proto jedna aktivita může být reprezentována několika událostmi.

Ještě než si řekneme formální definici záznamu, povíme si něco blíže k posloupnostem. Posloupnosti jsou vhodným nástrojem pro reprezentaci stop v záznamech. Pro danou množinu A , je A^* množina všech možných konečných posloupností nad množinou A . Konečná posloupnost nad množinou A délky n je funkce $\sigma \in \{1, 2, \dots, n\} \rightarrow A$. Taková posloupnost může být reprezentována řetězcem, např.: $\sigma = \langle a_1, a_2, \dots, a_n \rangle$, kde $a_i = \sigma(i)$ pro $1 \leq i \leq n$. $|\sigma|$ označuje délku posloupnosti, např.: $|\sigma| = n$. $\sigma \oplus a' = \langle a_1, a_2, \dots, a_n, a' \rangle$ přidá prvek a' na konec posloupnosti σ , podobně $\sigma_1 \oplus \sigma_2$ sloučí obě posloupnosti v posloupnost délky $|\sigma_1| + |\sigma_2|$.

$hd^k(\sigma) = \langle a_1, a_2, \dots, a_{\min(k,n)} \rangle$ je posloupnost prvních k prvků posloupnosti σ , tzv. předpona. Pro $k = 0$ je $hd^k(\sigma)$ prázdná posloupnost a pro $k \geq n$: $hd^k(\sigma) = \sigma$. Množina $pref(\sigma) = \{hd^k(\sigma) \mid 0 \leq k \leq n\}$ je množina všech předpon. $tl^k(\sigma) = \langle a_{\max(n-k+1,1)}, \dots, a_n \rangle$ je posloupnost posledních k prvků posloupnosti σ . Pro $k = 0$ je $tl^k(\sigma)$ prázdná posloupnost a pro $k \geq n$: $tl^k(\sigma) = \sigma$.

Pro jakoukoliv posloupnost $\sigma = \langle a_1, a_2, \dots, a_n \rangle$ nad množinou A , $\partial_{množina}$ konvertuje posloupnost v množinu, např.: $\partial_{množina}(\langle a, a, b, b, a, b, b, b \rangle) = \{a, b\}$, a $\partial_{multimnožina}$ konvertuje posloupnost v multimnožinu, např.: $\partial_{multimnožina}(\langle a, a, b, b, a, b, b, b \rangle) = [a^3, b^5]$. Nyní, když víme co je to posloupnost a víme jak s nimi pracovat, pojďme si formálně nadefinovat případ, stopu a záznam.

Df

DEFINICE

Nechť \mathcal{C} je množina všech možných případů. **Případy** podobně jako události mohou mít různé atributy. Pro jakýkoliv případ $c \in \mathcal{C}$ a jméno $n \in AN$ platí, že $\#_n(c)$ je hodnota

atributu n případu c ($\#_n(c) = \perp$, jestliže případ c nemá atribut n). Každý případ má speciální povinný atribut zvaný stopa $\#_{stopa}(c) \in \mathcal{E}^*$. $\hat{c} = \#_{stopa}(c)$ ¹⁴ označuje ve zkratce stopu.

Stopu tedy definujeme jako konečnou posloupnost událostí $\sigma \in \mathcal{E}^*$, ve které se každá událost objeví pouze jednou, tzn. pro $1 \leq i \leq j \leq |\sigma|: \sigma(i) \neq \sigma(j)$.

A konečně **záznam** definujeme jako množinu případů $L \subseteq \mathcal{C}$, pro něž platí, že každá událost se v logu vyskytne nejvýše jednou, tzn. pro každý případ $c_1, c_2 \in L$ platí $c_1 \neq c_2: \partial_{množina}(\hat{c}_1) \cap \partial_{množina}(\hat{c}_2) = \emptyset$. V případě, že záznam obsahuje atribut časové stopy, pak by řazení událostí v jednotlivých případech mělo tento atribut respektovat, tzn. pro $c \in L$ a pro $1 \leq i \leq j \leq |\hat{c}|: \#_{časová\ stopa}(\hat{c}(i)) \leq \#_{časová\ stopa}(\hat{c}(j))$.

SAMOSTATNÝ ÚKOL



Ověřte dle definice záznamu, že v tabulce 1 se skutečně nachází část záznamu.

Pro lepší práci se záznamy se používají tzv. **identifikátory**. Tyto identifikátory nám umožňují jednoznačně určit specifickou událost a specifický případ. Proto tyto identifikátory musejí být unikátní. Toto je velmi důležité, jelikož záznam většinou obsahuje události a případy se stejnými atributy neboli vlastnostmi, a my musíme být schopni rozlišit jednotlivé instance. Tyto identifikátory jsou technickou záležitostí a v původním zdroji dat nemusí být obsaženy.

ÚKOL K ZAMYŠLENÍ



Který sloupec v tabulce 1 můžeme považovat za jedinečný identifikátor případů, a který sloupec můžeme identifikovat za jedinečný identifikátor událostí. Pokud jste kapitolu 6.2.1 doposud četli pořádně, odpověď by pro Vás měla být velmi snadná.

Definice záznamu v podobě, kterou jsme uváděli výše je velmi detailní. A tato forma je v praxi velmi užívaná, jak si ukážeme v kapitolách věnovaných softwarům Disco a ProM, kde si představíme i různé formáty se kterými tyto softwary pracují. Nicméně pro naše

¹⁴ Předpokládáme, že $\hat{c} = \#_{stopa}(c) \neq \langle \rangle$, tedy že každá stopa v záznamu obsahuje alespoň jednu událost.

účely této teoretické kapitoly je tato definice až příliš detailní a proto si definujeme také tak zvaný zjednodušený záznam.



DEFINICE

Nechť je \mathcal{A} množina jmen aktivit, potom **zjednodušená stopa** σ je posloupnost aktivit, tzn. $\sigma \in \mathcal{A}^*$. Zjednodušený záznam L poté definujeme jako multimnožinu stop nad množinou \mathcal{A} , tzn. $L \in \mathbb{B}(\mathcal{A}^*)$, kde $\mathbb{B}(\mathcal{A}^*)$ je zobrazení $\mathcal{A}^* \rightarrow \mathbb{N}$, například mějme multimnožinu $X \in \mathbb{B}(D)$ pak je-li $X = [a^2, b^3, c^1]$ potom $X(b) = 3$.

Jak je možné vidět z definice výše, u zjednodušeného záznamu nepracujeme s atributy událostí, a události referujeme názvy odpovídajících aktivit. V následujícím textu budeme ve chvílích, kdy bude z kontextu jasné, že mluvíme o zjednodušeném záznamu, označovat jen jako záznam. Po zbytek kapitoly 6.1 budeme pracovat téměř výhradně se zjednodušeným záznamem.



ŘEŠENÁ ÚLOHA

Naším úkolem je transformovat záznam z tabulky 1 na zjednodušený záznam.

Jak jsme si řekli v definici zjednodušeného záznamu, referujeme jednotlivé události názvy aktivit, které reprezentují. V první řadě si tedy sestavíme množinu aktivit následovně (pro zjednodušení budeme aktivity označovat zkratkami tvořenými z prvních písmen slov, tedy zběžné prověření bude značeno jako ZP , zatímco znovuotevření požadavku ZnP , atd.): $\{RP, DP, KÚ, VR, OZ, ZP, VH, ZnP\}$. Máme-li sestavenou množinu aktivit, sestavíme si jednotlivé stopy dle množiny případů. Dle případu 1 (ID případu) dostaneme $\sigma_1 = \langle RP, DP, KÚ, VR, OZ \rangle$, dále dle případu 2 dostaneme $\sigma_2 = \langle RP, KÚ, ZP, VR, VH \rangle$, $\sigma_3 = \langle RP, ZP, KÚ, VR, OZ, ZnP, DP, KÚ, VR, VH \rangle$ a $\sigma_4 = \sigma_1 = \langle RP, DP, KÚ, VR, OZ \rangle$. Dle definice tedy zjednodušený záznam vypadá následovně: $L = [\langle RP, DP, KÚ, VR, OZ \rangle^2, \langle RP, KÚ, ZP, VR, VH \rangle, \langle RP, ZP, KÚ, VR, OZ, ZnP, DP, KÚ, VR, VH \rangle]$

Pozn.: Pro definici zjednodušeného záznamu nemusejí být vždy použity pouze jména aktivit, ale je možné použít také například jména zdrojů, v tom případě by $\sigma_1 = \langle Petr, Monika, Marcel, Sára, Petr \rangle$.

K datům se ještě znova vrátíme v kapitolách 7 a 8 věnovaným softwarům Disco a ProM, kde se na data zaměříme spíše z praktického hlediska a kde si ukážeme reálné záznamy. Nicméně pro pokračování v našem teoretickém úvodu k process miningu nám výše definované záznamy zcela postačí.

6.3 Procesní modely

Ať již procesní modely tvoříme anebo objevujeme, je potřeba je nějakým způsobem reprezentovat či zobrazit. V tomto textu jste se již ve 3. kapitole seznámili s BPMN notací. Kromě notace BPMN existují ještě také kauzální sítě, procesní stromy, EPC notaci, přechodové systémy či Petriho sítě. My se blíže seznámíme s pojmy **přechodový systém** a **Petriho sítě**.

DEFINICE

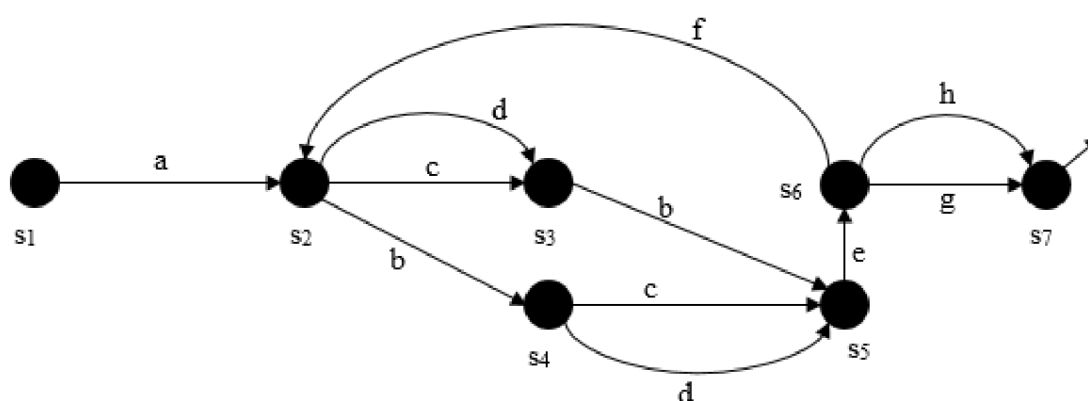


Přechodový systém je trojice $TS = (S, A, T)$, kde S je množina stavů, $A \subseteq \mathcal{A}$ je množina aktivit a $T \subseteq S \times \mathcal{A} \times S$ je množina přechodů. $S^{\text{začátek}} \subseteq S$ je množina počátečních stavů a $S^{\text{konec}} \subseteq S$ je množina konečných stavů.

Množina stavů S může být nekonečná, nicméně u praktických aplikací je množina stavů konečná. V takovém případě je přechodový systém označován také jako *Finite-State Machine* (FSM) nebo *finite-state automaton*.

6.3.1 PŘECHODOVÝ SYSTÉM

Přechodový systém je jednou z nejzákladnějších modelovacích notací. Přechodové systémy sestávají ze stavů a přechodů. Ukázkou přechodového systému lze vidět na obrázku níže. Stavů jsou reprezentovány uzly s počátečním stavem s_1 a koncovým stavem s_2 , zatímco přechody jsou reprezentovány hranami.



Obrázek 6-3: Přechodový systém

Zdroj: upraveno podle Aalsta (2016)



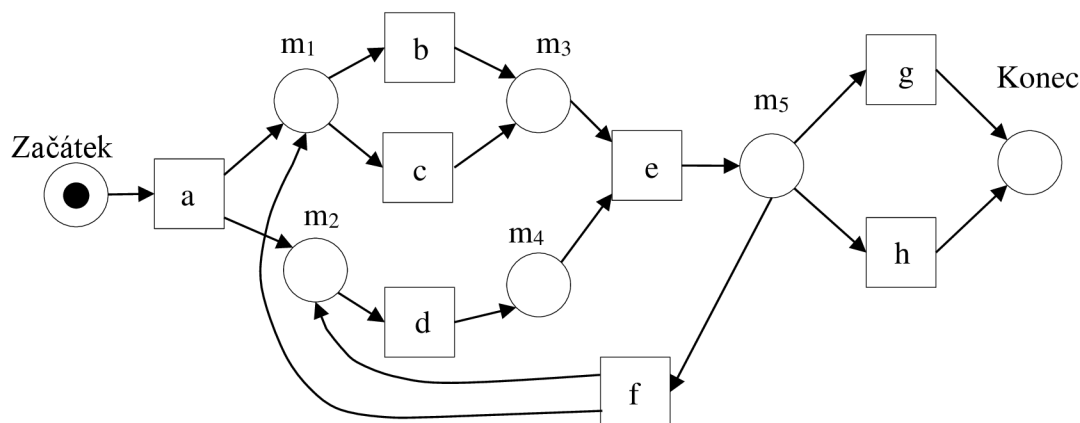
SAMOSTATNÝ ÚKOL

Sestavte množiny $S, A, T, TS, S^{\text{začátek}}, S^{\text{konec}}$ pro přechodový systém na obrázku 6-3.

Výhodou přechodových systémů je, že jsou velmi jednoduché a lze je velmi jednoduše transformovat do vyšších jazyků, které mají definovanou spustitelnou sémantiku, jako jsou Petriho sítě či BPMN. Nicméně přechodové systémy mají problémy vyjádřit konkurenční chování a to kvůli pojmu, zvanému *state explosion*. Proto byly představeny Petriho sítě, které modelování konkurenčního chování značně zjednoduší. Ovšem je potřeba si pamatovat, že ani Petriho sítě nejsou dokonalé. V podstatě každý modelovací jazyk má nějaký svůj „*representational bias*“ neboli nějaké limitace, avšak toto už je nad rámec tohoto textu.

6.3.2 PETRIHO SÍŤ

Petriho síť sice patří ke starším nástrojům umožňujícím modelování paralelního chování, nicméně stále patří mezi jedny z nejlepších. Přestože je jejich grafická notace velmi intuitivní a dobře čitelná, mají Petriho sítě také spustitelnou sémantiku, což umožňuje mnoho způsobů jak je analyzovat. Podobně jako tomu bylo u přechodového systému, i zde se jedná jen o úvod do základů Petriho sítí. Jelikož Petriho sítě mají dlouholetou tradici a jedná se o samostatnou vědní disciplínu se stovkami teorémů. Za vznikem Petriho sítí stál Carl Adam Petri, už je asi jasné, proč je nazýváme Petriho sítě.



Obrázek 6-4: Značená Petriho síť

Zdroj: upraveno podle Aalsta (2016)

Petriho síť jak ji můžeme vidět na obrázku 6-4, sestává z **míst** a **přechodů**, kde místa by se dala přirovnat ke stavům u přechodového systému a přechody k aktivitám přechodo-

vého systému. Struktura sítě nebo jinak samotný graf je sice statická, nicméně síť lze přehrát s využitím **značek** a **přechodového pravidla** (v anglické literatuře se můžete setkat s pojmy *token* a *firing rule*). Stav Petriho se je určen **značením** Petriho sítě, takové Petriho sítě nazýváme značené Petriho sítě. Na obrázku 6-4 se tedy nachází značená Petriho síť se značkou v místě začátek.

DEFINICE

Df

Petriho síť je trojice $N = (P, T, F)$, kde P je konečná množina míst (v angličtině *places*), T je konečná množina přechodů (v angličtině *transitions*) takových, že $P \cap T = \emptyset$ a F je konečná množina orientovaných hran mezi místy a přechody $F = (P \times T) \cup (T \times P)$ popisujících toky v síti.

Značená Petriho síť je potom dvojice (N, M) , kde $N = (P, T, F)$ je Petriho síť a $M \subseteq \mathbb{B}(P)$ je multimnožina nad množinou P tzv. značení sítě. \mathcal{N} označujeme jako množinu všech značených sítí.

S využitím značek a přechodového pravidla jsme tedy do jinak statických sítí schopni vnést jistou úroveň dynamiky. Přechod je **umožněn**, jestliže se v každém vstupním místě nachází alespoň jedna značka. Na obrázku 6-4 je tedy umožněn přechod a , chtěli-li bychom umožnit přechod e , museli bychom dostat značky do míst m_3, m_4 (mějme na paměti, že v jednotlivých místech se může vyskytovat i více než jedna značka). Ve chvíli, kdy je přechod umožněn, může být přechod **proveden**. V tomto případě, by byla spotřebována značka z místa *začátek* a vyprodukovány značky v místech m_1, m_2 , tímto by došlo k umožnění přechodů b, c, d , ale přechod a by již umožněn nebyl. Při provedení přechodu tedy dojde ke spotřebování značek ve všech vstupních místech daného přechodu a vyprodukování značek ve všech výstupních místech daného přechodu.

Abychom byly schopni formalizovat přechodové pravidlo, potřebujeme definovat vstupní a výstupní místa, respektive přechody. Mějme tedy síť $N = (P, T, F)$ a množinu uzlů $P \cup T$. Uzel x je vstupním uzlem uzlu y jen a pouze tehdy, je-li mezi těmito uzly orientovaná hrana z x do y . Obdobně platí, že uzel x je výstupním uzlem uzlu y pouze tehdy, je-li mezi těmito uzly orientovaná hrana y do x . Pro uzel $x \in P \cup T$ je množina vstupních a výstupních uzlů následující $\bullet x = \{y \mid (y, x) \in F\}$ a $x \bullet = \{y \mid (x, y) \in F\}$.

DEFINICE

Df

Necht' (N, M) je značená Petriho síť, kde $N = (P, T, F)$ a $M \subseteq \mathbb{B}(P)$. Přechod $t \in T$ je umožněn, značíme $(N, M)[t]$, jen a pouze tehdy platí-li $\bullet t \leq M$. **Přechodové pravidlo** $_[-]_ \subseteq \mathcal{N} \times T \times \mathcal{N}$ je nejmenší možná relace uspokojující pro jakoukoliv síť (N, M) a jakýkoliv přechod $t \in T$, $(N, M)[t] \Rightarrow (N, (M \setminus \bullet t) \uplus t \bullet)$.

Podíváme-li se opět na obrázek 6-4 pak provedení umožněného přechodu a u značené sítě $(N, [začátek])$ vyústí ve značenou síť $(N, [začátek])[a](N, [m_1, m_2])$. Samozřejmě že toto můžeme napsat i pro celou posloupnost postupně prováděných přechodů, např. vrátíme-li se opět k síti na obrázku 6-4 $(N, [začátek])[a, b](N, [m_2, m_3])$, či $(N, [začátek])[a, b, d, e, g](N, [konec])$.

Při modelování podnikových procesů je často využívána podtřída Petriho sítí, zvaná **WF-sítě** (*WorkFlow nets*). Pro WF-sítě je typické, že se mají jasně určené vstupní a výstupní místo a všechny uzly v síti jsou na cestě vedoucí těmito dvěma místy.

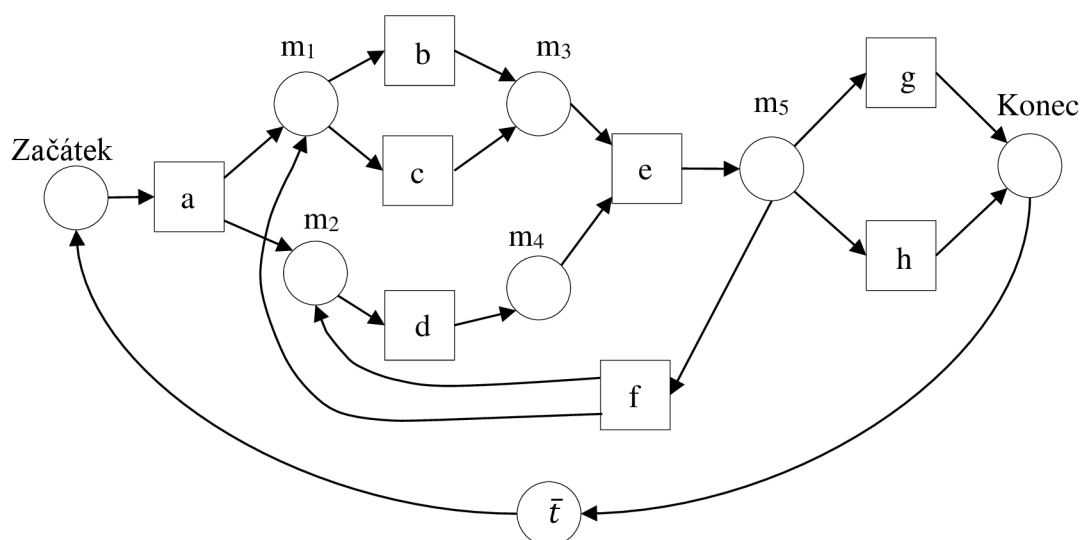


DEFINICE

Označená Petriho síť je pětice $N = (P, T, F, A, l)$, kde $N = (P, T, F)$ je Petriho síť tak, jak ji máme definovanu výše, $A \subseteq \mathcal{A}$ je množina názvů aktivit a $l \in T \rightarrow A$ je funkce přiřazující přechodu t název dané aktivity.

WF-síť poté můžeme definovat následovně. Mějme síť $N = (P, T, F, A, l)$ a $\bar{t} \notin P \cup T$. N je WF-síť jen a pouze tehdy platí-li“:

- P obsahuje vstupní místo i takové, že $\bullet i = \emptyset$
- P obsahuje výstupní místo o takové, že $o \bullet = \emptyset$
- a síť $\bar{N} = (P, T \cup \{\bar{t}\}, F \cup \{(o, \bar{t}), (\bar{t}, i)\}, A \cup \{\tau\}, l \cup \{(\bar{t}, \tau)\})$ je silně souvislá, což znamená, že je v síti cesta mezi každými dvěma uzly (a to jak z x do y tak z y do x).



Obrázek 6-5: Definice WF-sítě

Zdroj: vlastní zpracování

Petriho síť na obrázku 6-4 je WF-síť, což lze jednoduše vidět z obrázku 6-5. Jelikož po přidání přechodu \bar{t} mezi místa *začátek* (vstupní místo) a *konec* (výstupní místo) dostaneme silně souvislou síť ve smyslu silně souvislého orientovaného grafu.

Spolehlivá síť

Petriho sítě a stejně tak i WF-sítě mohou korespondovat velkou spoustou vlastností, které tu nebudeme rozebírat, ani si je nebudeme uvádět. Nicméně uvedeme si alespoň jednu, kterou využijeme v kapitole 6.4, a sice **spolehlivost sítě** či **spolehlivá síť**. Uvažujme tedy opět WF-síť $N = (P, T, F, A, l)$ se vstupním místem i a výstupním místem o . O síti $N = (P, T, F, A, l)$ můžeme říci, že je spolehlivá jen v případě, splňuje následující podmínky:

- **Bezpečnost** - $(N, [i])$ musí být bezpečná, tzn., že každé místo může v danou chvíli obsahovat nanejvýš jednu značku.
- **Řádné dokončení** – pro jakékoliv značení $M \in [N, [i])$, $o \in M$ implikuje $M = [o]$.
- **Možnost dokončit** - pro jakékoliv značení $M \in [N, [i])$, $o \in [N, M)$.
- **Absence mrtvých částí** - $(N, [i])$ neobsahuje žádné mrtvé přechody, tzn., že pro každý přechod t existuje posloupnost přechodů umožňující přechod t .

Množina $[N, M_0)$ je množina dosažitelných značení značené sítě (N, M_0) . Značení M je dosažitelné z původního značení jen a pouze tehdy, existuje-li posloupnost umožněných značení, jejichž provedení vede ze značení M_0 ke značení M . Dále si všimněte, že vlastnost možnost dokončit implikuje vlastnost řádného dokončení.

6.4 Objevování podnikových procesů

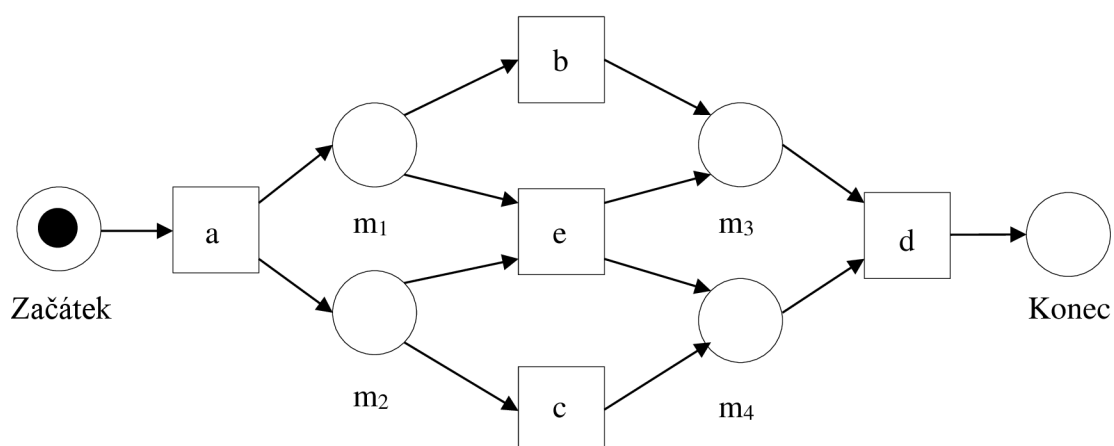
Objevování podnikových procesů je velmi vyzývajícím úkolem. Cílem objevování podnikových procesů je na základě záznamu zkonstruovat model procesu, který bude zachycovat chování pozorované v daném záznamu. Objevování podnikových procesů si představíme na jednom z prvních process miningových algoritmů zvaném **alfa-algoritmus**. Alfa-algoritmus ilustruje obecné myšlenky využívané při objevování procesů. Mimo to je prvním krokem v diskuzi o výzvách, kterým objevování procesů musí čelit. Nicméně je nutné zmínit, že jelikož se jedná o jeden z prvních algoritmů, má alfa-algoritmus relativně hodně nedostatků. Než se pustíme dále, možná bude vhodné připomenout si, co je to vlastně algoritmus.



DEFINICE

Neformálně je **algoritmus** definován jako soubor detailních instrukcí definujících výpočetní proces, který plně determinuje výstup na základě vstupu.¹⁵

Pro naše potřeby však Aalst (2016) definuje algoritmus jako funkci γ , která zobrazí záznam $L \in \mathbb{B}(\mathcal{A}^*)$ na značenou Petriho síť $\gamma(L) = (N, M)$. V ideálním případě je síť N spolehlivá WF-síť a všechny stopy obsažené v záznamu korespondují s posloupnostmi provedených přechodů značené Petriho sítě (N, M) .



Obrázek 6-6: WF-síť N_1

Zdroj: upraveno podle Aalsta (2016)

Podíváme-li se na N_1 na obrázku 6-3 a vezmeme-li v úvahu záznam $L_1 = [\langle a, b, c, d, \rangle^3, \langle a, c, b, d \rangle^2, \langle a, e, d \rangle]$, pak je možné, aby algoritmus γ objevil síť N_1 na obrázku 3. Není problém ověřit, že každá stopa obsažená v záznamu je možná posloupnost provedených přechodů sítě N_1 .

¹⁵ Definition of algorithm. *Encyclopedia of Mathematics* [online]. 2017 [cit. 2017-5-11]. Dostupné z: <https://www.encyclopediaofmath.org/index.php/Algorithm>

SAMOSTATNÝ ÚKOL

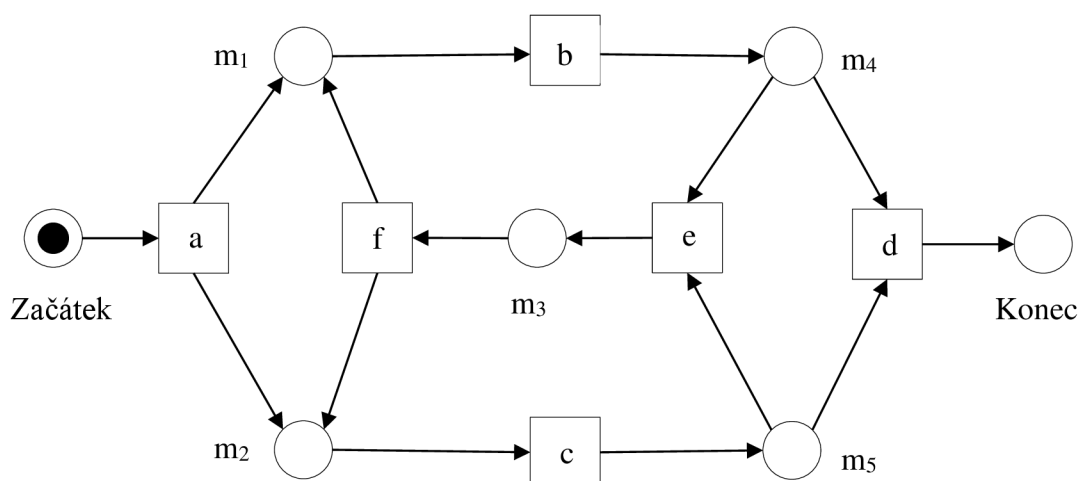


Určete z kolika případů a z kolika různých stop sestává záznam $L_1 = [\langle a, b, c, d, \rangle^3, \langle a, c, b, d \rangle^2, \langle a, e, d \rangle]$ a ověřte, že je záznam L_1 vskutku přehratelný na síti N_1 . To samé poté proveďte pro záznam L_2 a síť N_2 níže.

Nyní uvažujme záznam:

$$L_2 = \left[\begin{array}{l} \langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^4, \langle a, b, c, e, f, b, c, d \rangle^2, \langle a, b, c, e, f, c, b, d \rangle, \\ \langle a, c, b, e, f, b, c, d \rangle^2, \langle a, c, b, e, f, b, c, e, f, c, b, d \rangle \end{array} \right]$$

Na základě záznamu L_2 mohl nějaký algoritmus objevit model reprezentovaný WF-sítí N_2 na obrázku níže. Síť N_2 je podobně jako v předchozím případě opět schopna přehrát záznam L_2 . Nicméně mezi oběma sítěmi je jeden zásadní rozdíl. A sice zatímco u sítě N_1 obsahuje záznam L_1 všechny možné posloupnosti provedených přechodů, které síť N_1 umožňuje, síť N_2 umožňuje provedení posloupnosti přechodů $\langle a, c, b, e, f, c, b, d \rangle$, tato stopa ovšem není přítomna v záznamu L_2 . Ve skutečnosti kvůli existenci smyčky v daném modelu existuje nekonečně mnoho takových stop.



Obrázek 6-7: WF-síť N_2

Zdroj: upraveno podle Aalsta (2016)

Modely procesů hodnotíme na základě čtyř kritérií. Obecně vzato žádný reálný model nebude perfektně uspokojovat všechny čtyři kritéria, ale některá kritéria budou uspokojena na úkor jiných. Čtyři kritéria týkající se kvality modelů procesů jsou následující:

- **Shodnost** – je kritérium hodnotící, do jaké míry objevený model zachycuje chování vyskytující se v záznamu, neboli jeli možno na daném modelu přehrát celý záznam.
- **Přesnost** – objevený model by neměl povolit chování radikálně odlišné od toho, které se vyskytuje v záznamu. Přesnost úzce souvisí s pojmem nedoučení modelu (v anglické literatuře *underfitting*), ve chvíli kdy model vykazuje nízkou přesnost, říkáme, že je nedoučený, jelikož je příliš obecný a umožňuje chování neregistrované v záznamu.
- **Generalizace** – na druhou stranu v rámci kritéria generalizace je vyžadováno jisté zobecnění daného logu, a tedy možnosti přehrát na daném modelu i chování v záznamu se nevyskytující. Generalizace úzce souvisí s pojmem přeučení modelu (v anglické literatuře *overfitting*), ve chvíli, kdy model vykazuje nízký stupeň generalizace, říkáme, že je přeučení, jelikož je příliš specifický a umožňuje přehrát pouze případy vyskytující se v záznamu.
- **Jednoduchost** – toto kritérium souvisí s principem známé *Occamovy břitvy*, podstatou tohoto kritéria je fakt, že objevený model by měl být co možná nejjednodušší.



PRO ZÁJEMCE

Princip Occamovy břitvy krásně shrnuje citát Alberta Einsteina: „*Everything should be made as simple as possible, but not simpler*“. Volně přeloženo to znamená, že vše by mělo být tak jednoduché, jak jen to jde, ale ani o kousek jednodušší.

V anglické literatuře jsou jednotlivá kritéria označována následovně: shodnost – *fitness*, přesnost – *precision*, generalizace – *generalization* a jednoduchost – *simplicity*.

6.4.1 ALFA-ALGORITMUS

Alfa-algoritmus je příkladem výše zmíněné funkce γ . Vstupem alfa-algoritmu je zjednodušený záznam a jeho výstupem je Petriho síť umožňující přehrání záznamu. Alfa-algoritmus byl jedním z prvních, který si dokázal jistým způsobem poradit s konkurenčním chováním. Konkurenční chování můžeme vidět například u sítě N_2 výše, konkrétně se jedná

o přechody b a c . Přestože je alfa-algoritmus pro objasnění základních principů objevování procesů, v praxi je dnes již zastíněn daleko sofistikovanějšími přístupy. To je dáno zejména problémy, se kterými si alfa-algoritmus nedokáže poradit jako hluk, neúplnost či komplexní chování (všem těmto problematickým částem se budeme věnovat později).

ÚKOL K ZAMYŠLENÍ



Uvažujte síť N_1 a zdůvodněte, proč nejsou přechody (b, e) ani (c, e) vzájemně konkurenčními přechody, zatímco (b, c) konkurenční chování vykazují, i když by se při prvním pohledu na daný model mohlo zdát, že ano.

Mějme tedy zjednodušený záznam $L \in \mathbb{B}(\mathcal{A}^*)$ a jednotlivé elementy množiny \mathcal{A} označujeme jako aktivity. Tyto aktivity korespondují s přechody v objevené Petriho síti. Dále označujeme množiny aktivit velkými písmeny, $A, B \subseteq \mathcal{A}$, a jednotlivé aktivity malými písmeny, $a, b \in \mathcal{A}$. Výstupem alfa-algoritmu je značená WF-síť, $\alpha(L) = (N, M)$. Principem alfa-algoritmu je hledání jistých vzorců chování v záznamu (například je-li aktivita a vždy následována aktivitou b , ale nikdy ne opačně, pak se dá předpokládat kauzální závislost mezi oběma aktivitami. Rozlišujeme čtyři vztahy, které vyjadřují relevantní vzorce nacházející se v záznamu.

DEFINICE



Nechť L je záznam nad množinou aktivit \mathcal{A} , tedy $L \in \mathbb{B}(\mathcal{A}^*)$ a mějme dvě aktivity $a, b \in \mathcal{A}$, potom definujeme následující čtyři vztahy mezi aktivitami v daném záznamu (Aalst, 2011):

- $a >_L b$ jen a pouze existuje-li stopa $\sigma = \langle t_1, t_2, \dots, t_n \rangle$ a $i \in \{1, \dots, n-1\}$ takové, že $\sigma \in L$ a $t_i = a$ a $t_{i+1} = b$ (tzv. přímá následnost),
- $a \rightarrow_L b$ jen a pouze tehdy, platí-li $a >_L b$ a $b \not>_L a$,
- $a \#_L b$ jen a pouze tehdy, platí-li $a \not>_L b$ a $b \not>_L a$,
- $a \parallel_L b$ jen a pouze tehdy, platí-li $a >_L b$ a $b >_L a$.

Vezmeme-li v úvahu znovu záznam $L_1 = [\langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^2, \langle a, e, d \rangle]$, pak se doptáme toho, že obsahuje následující vztahy:

- $>_{L_1} = \{(a, b), (a, c), (a, e), (b, c), (c, b), (b, d), (c, d), (e, d)\}$

- $\rightarrow_{L_1} = \{(a, b), (a, c), (a, e), (b, d), (c, d), (e, d)\}$
- $\#_{L_1} = \{(a, a), (a, d), (b, b), (b, e), (c, c), (c, e), (d, a), (d, d), (e, b), (e, c), (e, e)\}$
- $\parallel_{L_1} = \{(b, c), (c, b)\}$

Vztah $>_{L_1}$ je tedy množina obsahující všechny páry aktivit, které jsou ve vztahu přímé následnosti. Vezmeme-li v úvahu stopu $\langle a, b, c, d \rangle$, pak vidíme, že aktivita b přímo následuje aktivitu a . Navíc, jelikož aktivita a není v žádné ze tří různých stop záznamu L_1 ve vztahu přímé následnosti s aktivitou b , je mezi aktivitami (a, b) také vztah \rightarrow_{L_1} .

Tabulka 6-2: Vztahová matice záznamu L_1

	a	b	c	d	e
a	$\#_{L_1}$	\rightarrow_{L_1}	\rightarrow_{L_1}	$\#_{L_1}$	\rightarrow_{L_1}
b	\leftarrow_{L_1}	$\#_{L_1}$	\parallel_{L_1}	\rightarrow_{L_1}	$\#_{L_1}$
c	\leftarrow_{L_1}	\parallel_{L_1}	$\#_{L_1}$	\rightarrow_{L_1}	$\#_{L_1}$
d	$\#_{L_1}$	\leftarrow_{L_1}	\leftarrow_{L_1}	$\#_{L_1}$	\leftarrow_{L_1}
e	\leftarrow_{L_1}	$\#_{L_1}$	$\#_{L_1}$	\rightarrow_{L_1}	$\#_{L_1}$

Zdroj: upraveno podle Aalsta (2016)

Budeme-li nyní uvažovat stopu $\langle a, c, b, d \rangle$ a v ní vyskytující se aktivity (c, b) , vidíme, že je mezi nimi vztah $>_{L_1}$, avšak jelikož ve stopě $\langle a, b, c, d \rangle$ aktivita c přímo následuje aktivitu b , a proto mezi nimi není vztah \rightarrow_{L_1} . Nicméně jelikož mezi aktivitami b a c platí vztahy $b >_{L_1} c$ a $c >_{L_1} b$, pak mezi nimi platí vztah \parallel_{L_1} . $\#_{L_1}$ vztah je množina obsahující všechny páry aktivit pro které platí $a \not>_L b$ a $b \not>_L a$. Vztahy mezi jednotlivými aktivitami lze také zobrazit pomocí tzv. matice vztahů, viz obrázek výše.



SAMOSTATNÝ ÚKOL

Sestrojte vztahovou matici pro záznam $L_2 = [\langle a, b, c, d \rangle^3, \langle a, c, b, d \rangle^4, \langle a, b, c, e, f, b, c, d \rangle^2, \langle a, b, c, e, f, c, b, d \rangle, \langle a, c, b, e, f, b, c, d \rangle^2, \langle a, c, b, e, f, b, c, e, f, c, b, d \rangle]$.

Nyní si tedy konečně představíme samotný alfa-algoritmus. Nejste-li si jisti některými definicemi uvedenými v kapitole 6.3, doporučujeme, ještě než se vrhnete na studování následující definice věnující se alfa-algoritmu, abyste si dřívější definice znova prošli. Zejména je potřeba vědět co je to záznam, výše uvedené čtyři vztahy mezi událostmi, co je to WF-síť a čím je charakteristická, mít jasno mezi místy a přechody. Dále je potřeba jistý

matematický aparát z teorie množin a výrokové logiky, které v tomto textu neprocházíme, musíme vědět co je to množina, co je to podmnožina, ale to již necháme na Vás, jelikož jste jistě poctivě navštěvovali všechny povinné kurzy matematiky, které naše fakulta nabízí.

DEFINICE

Df

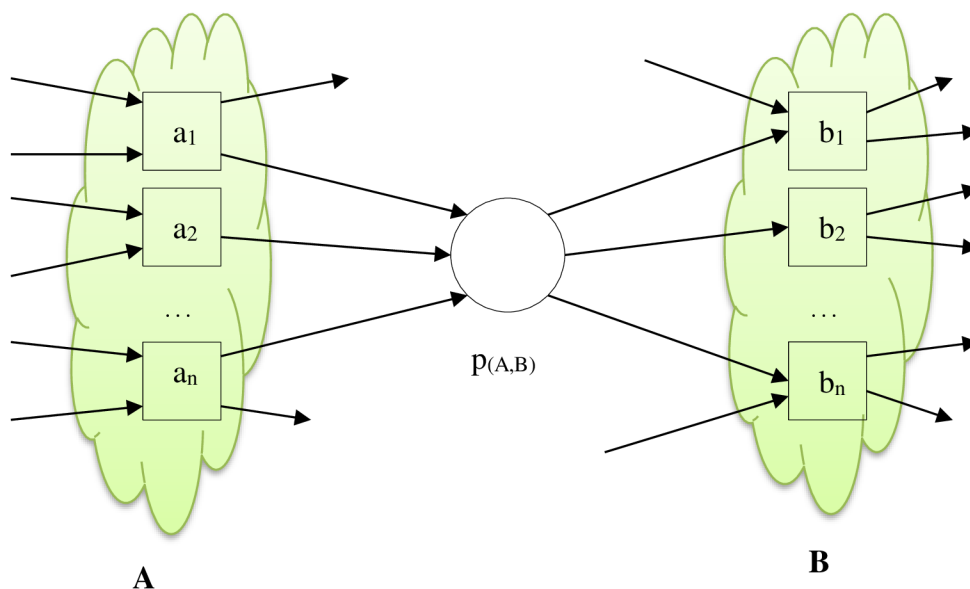
Nechť L je záznam nad množinou $T \subseteq \mathcal{A}$, poté je $\alpha(L)$ definován následovně (Aalst, 2016):

1. $T_L = \{t \in T \mid \exists \sigma \in L t \in \sigma\}$,
2. $T_I = \{t \in T \mid \exists \sigma \in L t = \text{první}(\sigma)\}$,
3. $T_O = \{t \in T \mid \exists \sigma \in L t = \text{poslední}(\sigma)\}$,
4. $X_L = \{(A, B) \mid A \subseteq T_L \wedge A \neq \emptyset \wedge B \subseteq T_L \wedge B \neq \emptyset \wedge \forall a \in A \forall b \in B a \rightarrow_L b \wedge \forall a_1, a_2 \in A a_1 \#_L a_2 \wedge \forall b_1, b_2 \in B b_1 \#_L b_2\}$,
5. $Y_L = \{A, B \in X_L \mid \forall (A', B') \in X_L A \subseteq A' \wedge B \subseteq B' \Rightarrow (A, B) = (A', B')\}$
6. $P_L = \{p_{(A,B)} \mid (A, B) \in Y_L\} \cup \{i_L, o_L\}$,
7. $F_L = \{(a, p_{(A,B)}) \mid (A, B) \in Y_L \wedge a \in A\} \cup \{(p_{(A,B)}, b) \mid (A, B) \in Y_L \wedge b \in B\} \cup \{(i_L, t) \mid t \in T_I\} \cup \{(t, o_L) \mid t \in T_O\}$,
8. $\alpha(L) = (P_L, T_L, F_L)$.

Pojďme si nyní projít, co znamenají jednotlivé body alfa-algoritmu. V prvním kroku zkontrolováno, které aktivity se nacházejí v záznamu L . Jednotlivé aktivity množiny T_L poté korespondují s přechody dané konstruované WF-sítě. Ve druhém kroku je sestavena množina T_I aktivit, které se vyskytují ve všech stopách napříč celým záznamem na prvním místě. Ve třetím kroku je sestavena množina T_O aktivit, které se vyskytují ve všech stopách napříč celým záznamem na posledním místě. Kroky čtyři a pět jsou těmi hlavními kroky, celého algoritmu. V těchto dvou krocích jsou determinovány místa objevené WF-sítě. Tyto místa jsou označována $p_{(A,B)}$, kde A označuje množinu vstupních přechodů ($\bullet p_{(A,B)} = A$) a množina B je množina výstupních přechodů ($p_{(A,B)} \bullet = B$) místa $p_{(A,B)}$.

V rámci čtvrtého kroku jsou identifikována jednotlivá potenciální místa objevené WF-sítě. Myšlenku stojící za objevením jednotlivých míst u alfa-algoritmu je vidět na obrázku výše. To znamená, že všechny prvky množiny A by měly mít kauzální závislost na všech prvcích množiny B , neboli má platit pro všechny $(a, b) \in A \times B: a \rightarrow_L b$. Mimo to

je požadováno, aby prvky v rámci množiny A nebyly vzájemně ve vztahu přímé následnosti, tedy pro všechny $a_1, a_2 \in A: a_1 \#_L a_2$. Podobně pro množinu je požadováno, aby pro všechny prvky množiny B platilo $b_1, b_2 \in B: b_1 \#_L b_2$. V pátém kroku dojde k vyloučení menších než maximální párů, jelikož v opačném případě by bylo v objevené síti příliš mnoho míst, a dojde tak ke vzniku množiny Y_L . V šestém kroku jsou určeny jednotlivá místa korespondující prvkům $(A, B) \in Y_L$, a také vstupní a výstupní místo typické pro WF-sítě. V sedmém kroku jsou vygenerovány hrany WF-sítě mezi všemi místy $p_{(A,B)}$ a vstupními přechody množiny A a výstupními přechody množiny B , a mezi vstupním a výstupním místem a příslušnými přechody příslušných množin T_I a T_O .



Obrázek 6-8: Základní myšlenka stojící za místy u čtvrtého kroku alfa-algoritmu

Zdroj: upraveno podle Aalsta (2016)



ŘEŠENÁ ÚLOHA

Nyní aplikujeme alfa-algoritmus v praxi, uvažujme záznam $L_3 = [\langle a, b, e, f \rangle^3, \langle a, b, e, c, d, b, f \rangle^5, \langle a, b, c, e, d, b, f \rangle^2, \langle a, b, c, d, e, b, f \rangle^2, \langle a, e, b, c, d, b, f \rangle^3]$

Potom jednotlivé kroky alfa algoritmu jsou následující:

1. $T_{L_3} = \{a, b, c, d, e, f\}$

V prvním kroku byly nejprve identifikovány všechny aktivity vyskytující v záznamu L_3 reprezentující události s podobnými vlastnostmi, což dalo vznik množině T_{L_3} .

2. $T_I = \{a\}$

Ve druhém kroku jsou identifikovány všechny aktivity, které se vyskytují na první pozici napříč všemi stopami v záznamu, což není v našem případě problém, jelikož v případě záznamu L_3 se jedná pouze o aktivitu a .

$$3. T_0 = \{f\}$$

Ve třetím kroku byly identifikovány všechny aktivity, které se vyskytují na poslední pozici napříč všemi stopami záznamu L_3 , což opět není problém, jelikož se jedná pouze o aktivitu f .

$$4. X_{L_3} = \left\{ \begin{array}{l} (\{a\}, \{b\}), (\{a\}, \{e\}), (\{b\}, \{c\}), (\{b\}, \{f\}), (\{c\}, \{d\}), \\ (\{d\}, \{b\}), (\{e\}, \{f\}), (\{a, d\}, \{b\}), (\{b\}, \{c, f\}) \end{array} \right\}$$

Ve čtvrtém kroku dojde k identifikaci potencionálních míst WF-sítě prostřednictvím vztahů \rightarrow_L a $\#_L$. V praxi, kde budeme pracovat s desetitisíci řádky v záznamu, to za nás udělá technika, ale u takto jednoduchých záznamů lze využít matici vztahů k odhalení množiny X_{L_3} .

Tabulka 6-3: Matice vztahů L_3

	a	b	c	d	e	f
a	$\#_{L_3}$	\rightarrow_{L_3}	$\#_{L_3}$	$\#_{L_3}$	\rightarrow_{L_3}	$\#_{L_3}$
b	\leftarrow_{L_3}	$\#_{L_3}$	\rightarrow_{L_3}	\leftarrow_{L_3}	\parallel_{L_3}	\rightarrow_{L_3}
c	$\#_{L_3}$	\leftarrow_{L_3}	$\#_{L_3}$	\rightarrow_{L_3}	\parallel_{L_3}	$\#_{L_3}$
d	$\#_{L_3}$	\rightarrow_{L_3}	\leftarrow_{L_3}	$\#_{L_3}$	\parallel_{L_3}	$\#_{L_3}$
e	\leftarrow_{L_3}	\parallel_{L_3}	\parallel_{L_3}	\parallel_{L_3}	$\#_{L_3}$	\rightarrow_{L_3}
f	$\#_{L_3}$	\leftarrow_{L_3}	$\#_{L_3}$	$\#_{L_3}$	\leftarrow_{L_3}	$\#_{L_3}$

Zdroj: upraveno podle Aalsta (2016)

$$5. Y_{L_3} = \{(\{a\}, \{e\}), (\{c\}, \{d\}), (\{e\}, \{f\}), (\{a, d\}, \{b\}), (\{b\}, \{c, f\})\}$$

V pátém kroku odebereme v z množiny X_{L_3} všechny nemaximální prvky a dáme vzniknout množině Y_L . Jak lze vidět, budeme-li uvažovat prvky $(\{a\}, \{b\})$, $(\{d\}, \{b\})$ a $(\{a, d\}, \{b\})$ vzniklo by v síti zbytečně mnoho míst, zatímco, ponecháme-li jen maximální prvek $(\{a, d\}, \{b\})$ dostaneme stejné chování s využitím pouze jednoho místa. Podobně je tomu s prvky $(\{b\}, \{c\})$, $(\{b\}, \{f\})$ a $(\{b\}, \{c, f\})$.

$$6. P_{L_3} = \{p_{(\{a\}, \{e\})}, p_{(\{c\}, \{d\})}, p_{(\{e\}, \{f\})}, p_{(\{a, d\}, \{b\})}, p_{(\{b\}, \{c, f\})}, i_{L_3}, o_{L_3}\}$$

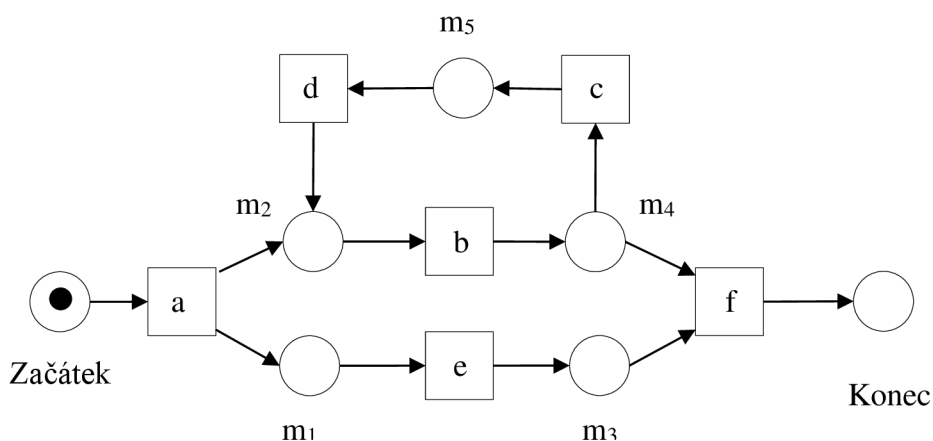
V šestém kroku tedy dojde k formální definici míst a tedy vzniku množiny míst P_{L_3} .

$$7. F_{L_3} = \left\{ \begin{array}{l} (i_L, a), (a, p_{\{\{a\},\{e\}\}}), (a, p_{\{\{a,d\},\{b\}\}}), (p_{\{\{a\},\{e\}\}}, e), (p_{\{\{a,d\},\{b\}\}}, b), \\ (e, p_{\{\{e\},\{f\}\}}), (b, p_{\{\{b\},\{c,f\}\}}), (p_{\{\{b\},\{c,f\}\}}, c), (c, p_{\{\{c\},\{d\}\}}), (p_{\{\{c\},\{d\}\}}, d), \\ (d, p_{\{\{a,d\},\{b\}\}}), (p_{\{\{e\},\{f\}\}}, f), (p_{\{\{b\},\{c,f\}\}}, f), (f, o_L) \end{array} \right\}$$

V sedmém kroku doplníme množinu hran WF-sítě. V tomto bodě jsme připraveni na osmý krok, ve kterém již jsme schopni definovat objevenou WF-sít'.

$$8. \alpha(L_3) = (P_{L_3}, T_{L_3}, F_{L_3})$$

Model záznamu objevený s využitím alfa-algoritmu reprezentovaný WF-sítí můžete vidět na obrázku níže.



Obrázek 6-9: WF-sít' N₃ objevená alfa-algoritmem ze záznamu L₃

Zdroj: Aalst (2016)

Ted' když jsme si představili, jak alfa-algoritmus funguje, pojd' me si představit oblasti, ve kterých se s alfa-algoritmem můžeme dostat do značných potíží. Toto je jeden z dalších důvodů, proč je alfa-algoritmus vhodný jako vstupní přístup v rámci objevování podnikových procesů, jelikož jeho nedostatky jsou výzvami, kterým je v rámci process miningu potřeba čelit a to obecně. To znamená, že nejsou pouze otázkou alfa-algoritmu, ale těmto výzvám bylo a bude potřeba čelit i v nově vznikajících metodách zaměřujících se na objevování procesů.

Aniž bychom to explicitně konstatovali, tak jsme u alfa-algoritmu pracovali s předpokladem, že je záznam **kompletní**. Což znamená, že záznam obsahuje všechny informace popisující chování v rámci daného procesu. Jelikož není-li jisté chování reprezentováno případem v záznamu, pak z principu věci ani toto chování nemůže být objeveno. Jenomže v mnoha případech tohoto není možné docílit. Například budeme-li uvažovat model, ve kterém je 10 paralelních přechodů, jenom pro pokrytí těchto přechodů v rámci jednoho případu je zapotřebí $10! = 3628800$ stop. A i za předpokladu kompletního záznamu, může existovat několik sítí, které se budou lišit svou strukturou, ale které budou schopny přehrát jeden a ten samý záznam, tzn., budou stopově ekvivalentní.

Nekompletní záznam

V jistém smyslu opakem nekompletního záznamu je záznam obsahující **hluk**. V takovém případě záznam obsahuje vzácné a nefrekventované chování, které se velmi liší od typického chování, které lze pozorovat v daném záznamu. S tímto typem problému si dokáží velice dobře poradit pokročilé techniky jako *heuristic mining*, *genetic mining* nebo *fuzzy mining*.

Hluk

Dalším problémem u původního alfa-algoritmu bylo to, že si nedokázal poradit s **krátkými smyčkami** – konkrétně se jedná o smyčky délky jedna a dva. Toto plyne z vtažů používaných k hledání míst v objevené síti a vzorům, kterými jsou reprezentovány smyčky délky jedna a dva. Tento problém je poměrně jednoduše odstranitelný s pomocí předzpracování záznamu ještě před aplikací alfa-algoritmu, kde jsou tyto vzory identifikovány a zpracovány. Zájemce odkazujeme na práci Medeirosové, Aalsta a Weijterse (2003), kde detailněji rozebírají všechny námi zmíněné i nezmněné nedostatky, zatímco Medeirosová et al. (2004, 2005) se detailněji věnují přímo krátkým smyčkám a navrhují rozšířenou verzi tak zvaný +alfa-algoritmus. Se smyčkami délky tři a více si poradí také původní algoritmus.

Krátké smyčky

6.4.2 POKROČILÉ METODY OBJEVOVÁNÍ

V této podkapitole si ukážeme i některé další metody objevování procesů. Nicméně nepůjdeme ani zdaleka tak do hloubky jako u alfa-algoritmu, tyto metody si jen stručně představíme v několika slovech. Zájemce o tyto pokročilé metody vždy nasměrujeme k vhodným zdrojům v případě zájmu o jejich bližší nastudování.

První z pokročilých metod objevování procesů se nazývá **heuristic mining**. Na rozdíl od alfa-algoritmu využívá heuristic mining k reprezentaci modelu sítě, podobné kauzálním sítím, nicméně je možné využít také kauzální sítě. Jak jsme si již řekli, heuristic mining si svým způsobem dokáže poradit s hlučkem vyskytujícím se v záznamu, jelikož pracuje také s frekvencemi, s jakými se jednotlivé případy vyskytují v daném záznamu. Pro více informací odkazujeme čtenáře na následující práce Weijters a Ribeiro (2011), Weijters a Aalst (2003), Aalst (2016), Aalst (2011).

Heuristic mining

Genetic mining

Další metodou je **genetic mining**. Genetic mining patří mezi evoluční přístupy, využívající iterativní proceduru pro napodobení procesu evoluce. Podobně jako u ostatní genetických algoritmů, i genetic mining sestává z následujících čtyř kroků: **inicializace**, **výběr**, **reprodukce** a **terminace**. Ve fázi inicializace jsou na základě jmen aktivit náhodně vytvořeny procesní modely. V každé generaci se může objevit stovky až tisíce jednotlivých modelů. Ve fázi výběru se pro všechny modely vypočte hodnota kritéria shodnosti (toto kritérium může být definováno různými způsoby). A na základě tohoto jsou vybrány modely s nejlepšími hodnotami. U složitějších přístupů bývají dokonce brány v potaz všechny čtyři kritéria určující kvalitu modelu – to znamená kromě shodnosti také přesnost, generalizace a jednoduchost. V další fázi dojde k reprodukci nových modelů z generace z fáze výběru. Tyto nové modely jsou reprodukovány prostřednictvím operací: **křížení** a **mutace**. Nejprve jsou dva modely zkříženy za účelem získání nových dvou modelů. Na tyto nově získané modely je poté aplikována operace mutace, kdy dojde k náhodnému přidání či odebrání kauzálních závislostí. Tento postup se opakuje, dokud nedojde k nalezení uspokojujících modelů v rámci fáze terminace. Pro více informací opět odkazujeme čtenáře na Medeirosová, Weijters a Aalst (2005), Aalsta (2016), Aalsta (2011).

Region-based mining

Region-based mining je v sobě obsahuje dva přístupy. Prvním z nich je **state-based mining** a druhým z nich je **language-based mining**. Podstatou region-based mining je konstruování Petriho sítí, avšak v případě state-based mining jsou Petriho sítě tvořeny z přechodových systémů, zatímco v případě language-based mining je Petriho síť objevována na základě předpon vyskytujících se v záznamu. Language-based mining je tedy přímo aplikovatelný na záznam, zatímco v případě state-based mining je nejprve potřeba vytvořit přechodový systém. Oba přístupy hledají regiony, které poté odpovídají jednotlivým místům v objevované Petriho síti. Pro více informací opět odkazujeme čtenáře na Berghem et al. (2007), Aalst et al. (2010), Aalsta (2016), Aalsta (2011).

Inductive mining

Posledním přístupem, který si představíme je **inductive mining**. Inductive mining reprezentuje procesní modely prostřednictvím procesních stromů. Zatímco Petriho a WF-sítě, BPMN modely a další mohou obsahovat **deadlocky** a podobné nežádoucí anomálie, procesní stromy jsou spolehlivé z podstaty jejich tvorby a navíc dokáží vždy přehrát celý záznam¹⁶. Inductive mining je v současné době jedním z nejpoužívanějších přístupů v objevování procesů zejména díky jeho flexibilitě, formálním zárukám a škálovatelnosti. Pro více informací opět odkazujeme čtenáře na Leemans, Fahland a Aalst (2013a,b; 2014), Aalst (2016).

6.5 Zjišťování shodnosti modelů procesů

V následující podkapitole se zaměříme na zjišťování toho, zdali daný model je ve shodě s daným záznamem. Jaksi jistě dobře vzpomínáte, doposud probírané objevování procesů

¹⁶ Například je matematicky dokázáno, že alfa-algoritmus je schopen objevit spolehlivou síť, ale jen pro velmi specifickou skupinu sítí, což je v případě inductive mining značný posun vpřed.

souviselo s pojmem Play-In, zatímco jak jsme si již řekli, zjišťování shodnosti souvisí s pojmem Replay. To znamená, že zatímco v prvním případě jsme měli k dispozici pouze záznam, zde je potřeba jak záznamu, tak modelu. Tento model jsme mohli získat objevováním procesů, ale nemusí tomu vždy být takto, model může být udělán i ručně. Cílem zjišťování shodnosti je tak nalezení shody nebo rozporu mezi záznamem a modelem, neboli mezi pozorovaným chováním a modelovaným chováním. Zjišťování shodnosti tak může být v podnicích využito, jsou-li dodržována pravidla stanovená managementem podniku, dále pro hodnocení objevených modelů a tedy i hodnocení objevovacích algoritmů.

K ZAPAMATOVÁNÍ



V závislosti od toho, je-li analyzovaný model **deskriptivní** anebo **normativní**, se bude odvíjet, jakým způsobem bude zjišťování shodnosti využito. Jak jsme si již řekli, výše alfa-algoritmus není zdaleka perfektní, a perfektní nejsou ani pokročilejší metody, které jsme si představili. Proto je vhodné zjišťovat, na kolik se objevený deskriptivní model opravdu shoduje s použitým záznamem. Jedná se tedy o situaci, kdy je zjišťováno, zdali model odpovídá záznamu. Ale může nastat také opačná situace. Představte si například reklamační proces, u něhož management požaduje, aby u reklamací nad 10 000 Kč nikdy nebyly vráceny peníze, aniž by toto odsouhlasil vedoucí reklamačního oddělení. S využitím zjišťování shodnosti se tak lze dopátrat, zdali je toto pravidlo pracovníky reklamačního oddělení skutečně dodržováno, tedy zdali záznam odpovídá normativnímu modelu.

ÚKOL K ZAMYŠLENÍ



Mohou nastat situace, kdy by mohly být tyto odchylky přínosem pro podnik? Jak by šly zjištěné odchylky v podniku využít?

V předchozí kapitole, když jsme diskutovali kvalitu modelů a řekli jsme si, že je kvalita hodnocena v rámci čtyř kritérií (dokážete vyjmenovat všechny čtyři?). Jedním z těchto kritérií byla shodnost a v této kapitole si ukážeme, jak lze kritérium shodnosti kvantifikovat. Pro tento účel použijeme log L_4 níže, který je sestaven z 1391 případů a 21 různých stop.

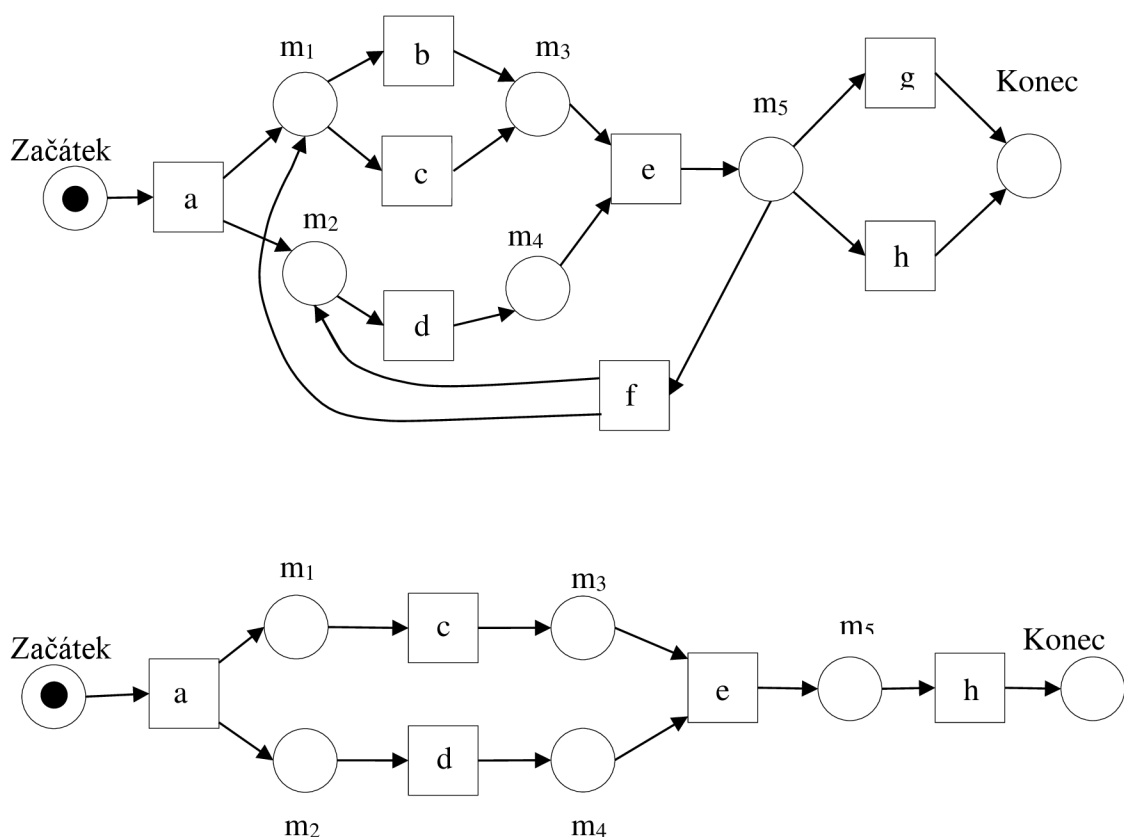
Tabulka 6-4: Záznam L_4

Frekvence	Jméno	Stopa
455	σ_1	<a,c,d,e,h>
191	σ_2	<a,b,d,e,g>
177	σ_3	<a,d,c,e,h>
144	σ_4	<a,b,d,e,h>

111	σ_5	<a,c,d,e,g>
82	σ_6	<a,d,c,e,g>
56	σ_7	<a,d,b,e,h>
47	σ_8	<a,c,d,e,f,d,b,e,h>
38	σ_9	<a,d,b,e,g>
33	σ_{10}	<a,c,d,e,f,d,b,e,h>
14	σ_{11}	<a,c,d,e,f,b,d,e,g>
11	σ_{12}	<a,c,d,e,f,d,b,e,g>
9	σ_{13}	<a,d,c,e,f,c,d,e,h>
8	σ_{14}	<a,d,c,e,f,d,b,e,h>
5	σ_{15}	<a,d,c,e,f,b,d,e,g>
3	σ_{16}	<a,c,d,e,f,b,d,e,f,d,b,e,g>
2	σ_{17}	<a,d,c,e,f,d,b,e,g>
2	σ_{18}	<a,d,c,e,f,b,d,e,f,b,d,e,g>
1	σ_{19}	<a,d,c,e,f,d,b,e,f,b,d,e,h>
1	σ_{20}	<a,d,b,e,f,b,d,e,f,d,b,e,g>
1	σ_{21}	<a,d,c,e,f,d,b,e,f,c,d,e,f,d,b,e,g>

Zdroj: upraveno podle Aalsta (2016)

K problému kvantitativní charakterizace kritéria shodnosti v rámci zjišťování shodnosti, lze poměrně naivně přistupovat například způsobem, že kritérium shodnosti vyjádříme jako podíl přehratelných stop v rámci celého záznamu. Uvažovali-li bychom záznam L_4 a síť $N_{4.1}$ a $N_{4.2}$, pak bychom mohli určit shodnost sítě $N_{4.1}$ následovně $S = 1391/1391 = 1$ a shodnost sítě $N_{4.2}$ by byla $S = 948/1391 = 0,6815$. Sami si vyzkoušejte, že síť $N_{4.1}$ je vskutku schopna přehrát celý záznam, zatímco $N_{4.2}$ není schopna přehrát celý záznam (pozn. již na první pohled je vidět, že jelikož v síti $N_{4.2}$ chybějí přechody b, f, g , nebude možno přehrát stopy, v nichž se vyskytují aktivity $\{b, f, g\} \subset T_{L_4}$). Nicméně tato naivní metrika není zdaleka vhodná pro více realistické procesy. Například došlo-li by ke spojení míst m_1 a m_2 v síti $N_{4.1}$ v jedno, nebylo by možno přehrát žádnou ze stop v záznamu L_4 a tedy ani jeden případ z 1391 případů a kritérium shodnosti by bylo $S = 0/1391 = 0$, což se ale zdá být příliš svazující vzhledem k tomu, že zbylé chování by taková síť vystihovala vcelku dobře.

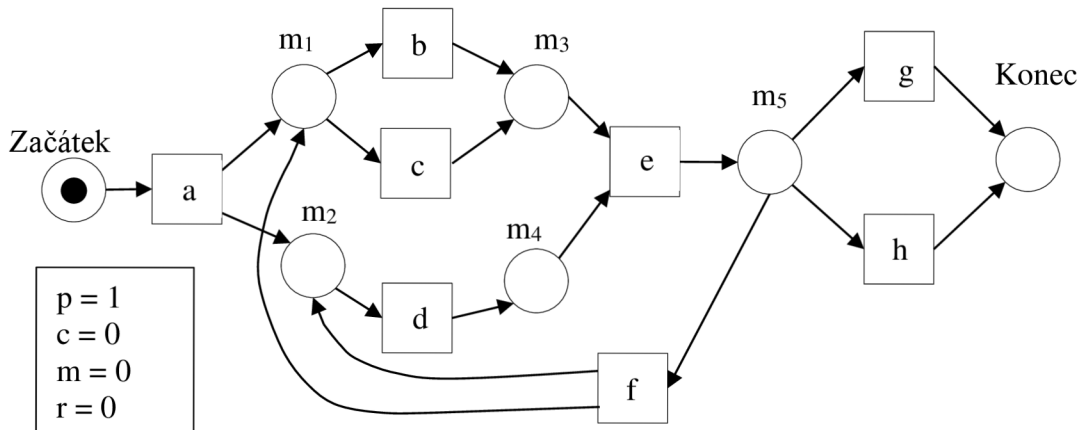


Obrázek 6-10: Sítě N_{4.1} (horní) a N_{4.2} (dolní)

Zdroj: upraveno podle Aalsta (2016)

Z tohoto důvodu je kritérium shodnosti spíše řešeno na úrovni událostí a schopnosti přehrát jednotlivé události, než na úrovni stop. Budeme-li se držet úrovně událostí, stačí jen trochu modifikovat předchozí případ, a sice v případě, že nějaká událost nebude přehratelná, neskončíme a neklasifikujeme stopu jako nepřehratelnou, ale budeme pokračovat a tyto nejasnosti zaznamenávat. V takovém případě nás budou zajímat následující charakteristiky: **vyprodukované značky** (p), **spotřebované značky** (c), **chybějící značky** (m) a **přebývající značky** (r).

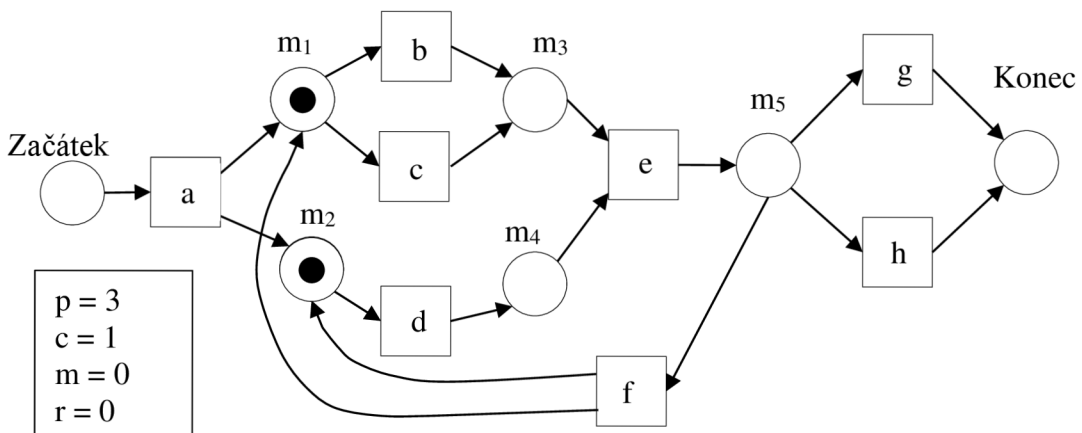
Uvažujme síť $N_{4.1}$ a stopu $\sigma_1 = \langle a, c, d, e, h \rangle$ záznamu L_4 . Na počátku je $p = c = 0$ a žádné místo neobsahuje značku. Následně prostředí vyprodukuje značku v místě *Začátek* a v důsledku toho $p = 1$, viz obrázek níže.



Obrázek 6-11: Síť N_{4.1} – značka v začátku

Zdroj: upraveno podle Aalsta (2016)

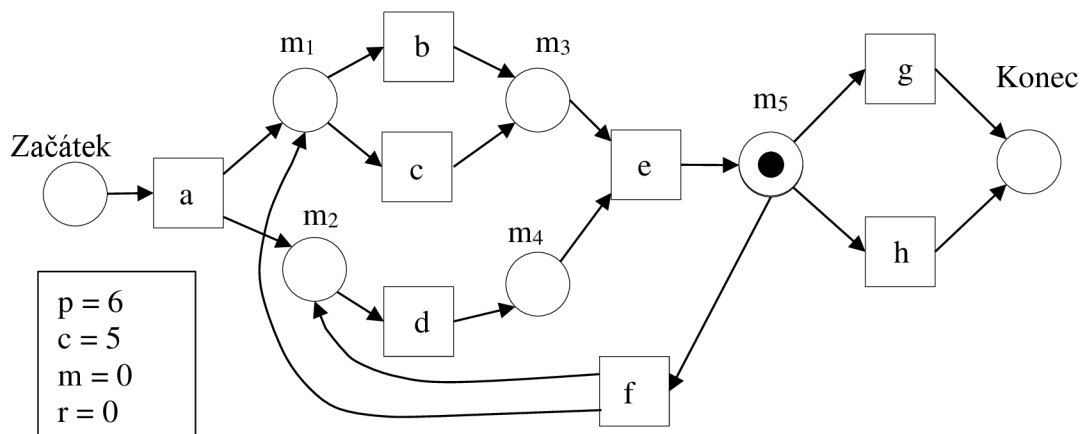
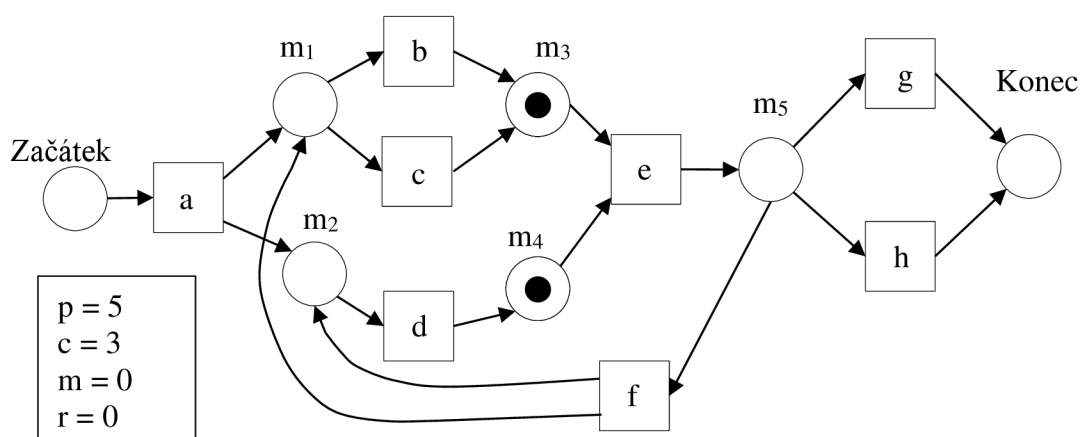
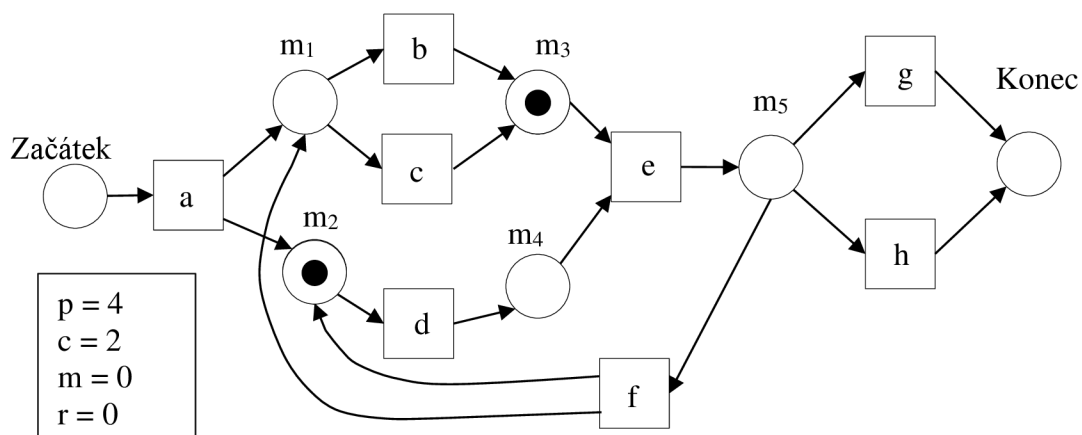
V následujícím kroku je proveden přechod a a tím dojde ke spotřebování jedné značky a vyprodukování dvou nových značek, dostaneme tedy značenou síť na obrázku níže.

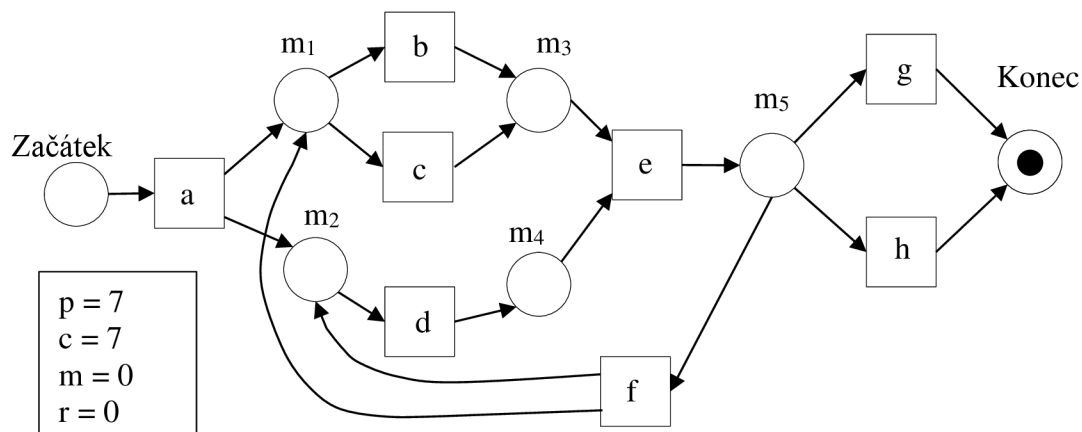


Obrázek 6-12: Síť N_{4.1} – značka v m_1 a m_2

Zdroj: upraveno podle Aalsta (2016)

Dle stopy $\sigma_1 = \langle a, c, d, e, h \rangle$ následuje po provedení přechodu a přechod c . Obrázek níže zobrazuje provedení zbylých aktivit c až h dle stopy σ_1 s příslušnými charakteristikami p, c, m, r .





Obrázek 6-13: Síť $N_{4.1}$ - značky m_2, m_3 až m_{Konec}

Zdroj: upraveno podle Aalsta (2016)

V tomto případě můžeme definovat shodnost následovně:

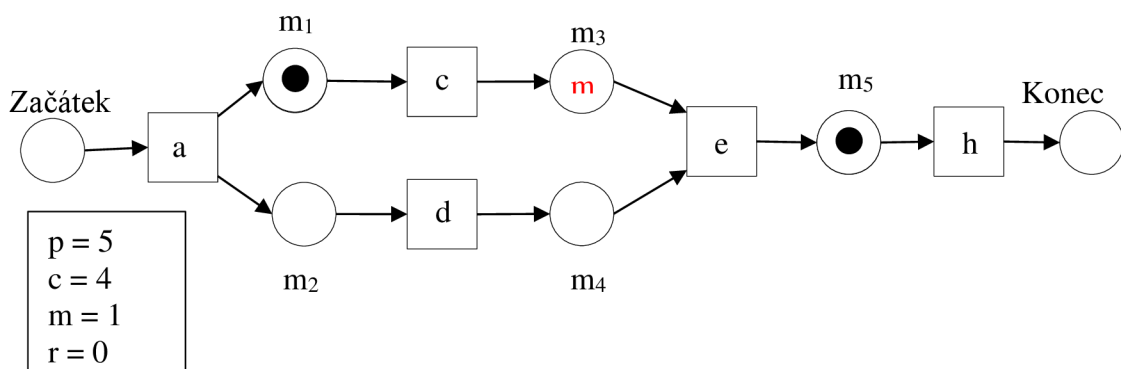
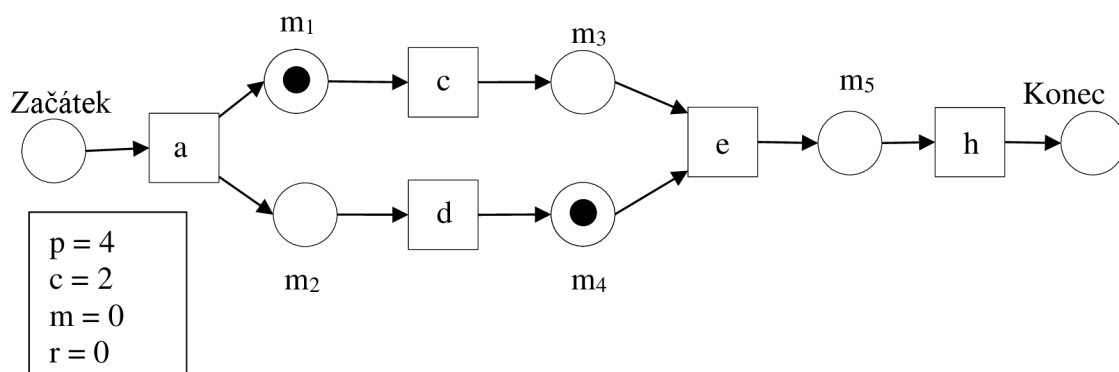
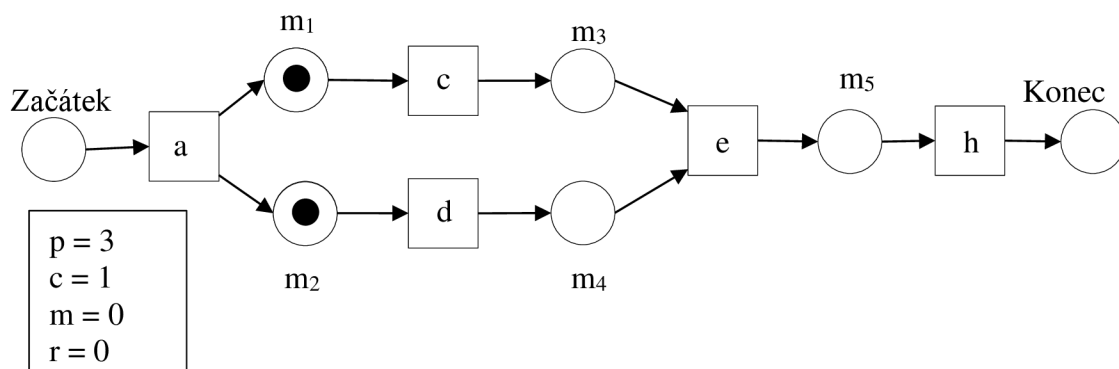
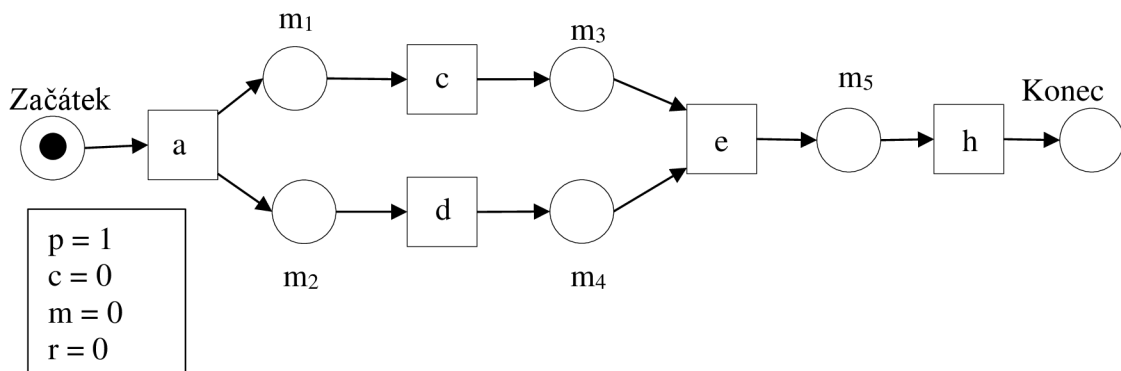
$$shodnost(\sigma, N) = \frac{1}{2} \left(1 - \frac{m}{c}\right) + \frac{1}{2} \left(1 - \frac{r}{p}\right) \quad (1)$$

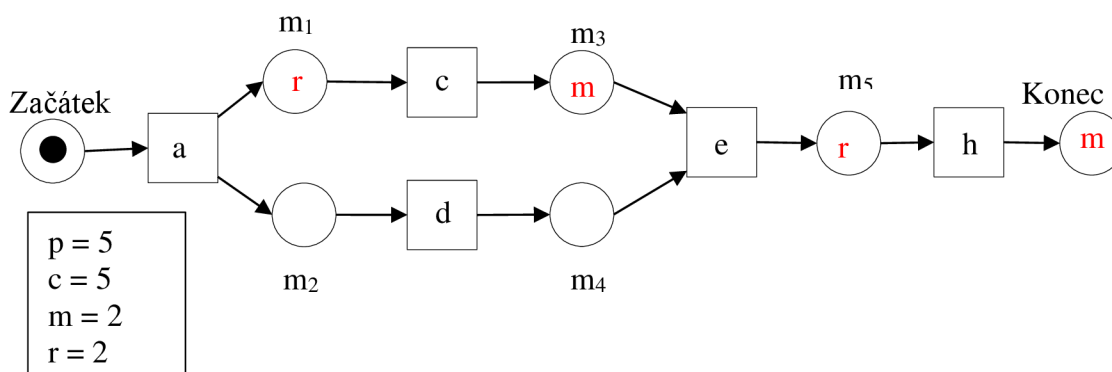
Jelikož jsme byli schopni přehrát v síti $N_{4.1}$ bez problémů celou stopu σ_1 a neměli jsme chybějící ani přebývající značky, je $shodnost(\sigma_1, N_{4.1}) = 1$, což je stejný výsledek jako u naší naivní definice shodnosti.



ŘEŠENÁ ÚLOHA

Uvažujme nyní síť $N_{4.2}$ a stopu $\sigma_2 = \langle a, b, d, e, g \rangle$ a vypočítejme kritérium shodnosti $shodnost(\sigma_2, N_{4.2})$. Celý postup shrnuje obrázek níže. Opět začínáme s neznačenou sítí a tedy $p = c = 0$.





Obrázek 6-14: Síť $N_{4.2}$ - značky $m_{\text{Začátek}}$ až m_{Konec}

Zdroj: upraveno podle Aalsta (2016)

Zde nám nastává problém, jelikož nám se neshodují aktivity v síti a stopě. V tomto případě abstrahujeme od chybějících událostí a přehrajeme $\sigma'_2 = \langle a, d, e \rangle$. Při přehrání aktivity a dojde ke spotřebě značky v místě *Začátek* a vyprodukování značek v místech m_1 a m_2 . Poté je dle stopy $\sigma'_2 = \langle a, d, e \rangle$ provedena aktivita d spotřebou značky v místě m_2 a vyprodukováním značky v místě m_4 . Problém nastane, ve chvíli, kdy se snažíme provést aktivitu e , jelikož nám v místě m_3 chybí značka a provedení aktivity e tak není umožněno. Místo m_3 tedy označíme značkou m , a přehrajeme aktivitu e . Po přehrání $\sigma'_2 = \langle a, d, e \rangle$ je značení $[p_1, p_5]$, jenomže prostředí potřebuje spotřebovat značku z místa *Konec*, a tedy i místo *Konec* je označeno značkou m . Jelikož nám po spotřebování značky v místě *Konec* stále přebývají značky v místech $[p_1, p_5]$, označíme je jako r . A po dosazení je tedy

$$\text{shodnost}(\sigma_3, N_{4.2}) = \frac{1}{2} \left(1 - \frac{m}{c}\right) + \frac{1}{2} \left(1 - \frac{r}{p}\right) = \frac{1}{2} \left(1 - \frac{2}{5}\right) + \frac{1}{2} \left(1 - \frac{2}{5}\right) = 0.6$$

Shodnost dle vzorce (1) výše ovšem řeší pouze jeden případ respektive stopu, abychom zjistili shodnost pro celý záznam je potřeba vzorec (1) následovně modifikovat:

$$\begin{aligned} \text{shodnost}(\sigma, N) &= \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times c_{N,\sigma}}\right) \\ &+ \frac{1}{2} \left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_{N,\sigma}}{\sum_{\sigma \in L} L(\sigma) \times p_{N,\sigma}}\right) \end{aligned} \quad (2)$$

Jelikož $\sum_{\sigma \in L} L(\sigma) \times m_{N,\sigma}$ představuje celkový počet chybějících značek při přehrávání celého záznamu, pak $L(\sigma)$ reprezentuje frekvenci s jakou se stopa σ vyskytuje v záznamu.

6.6 Další perspektivy process miningu

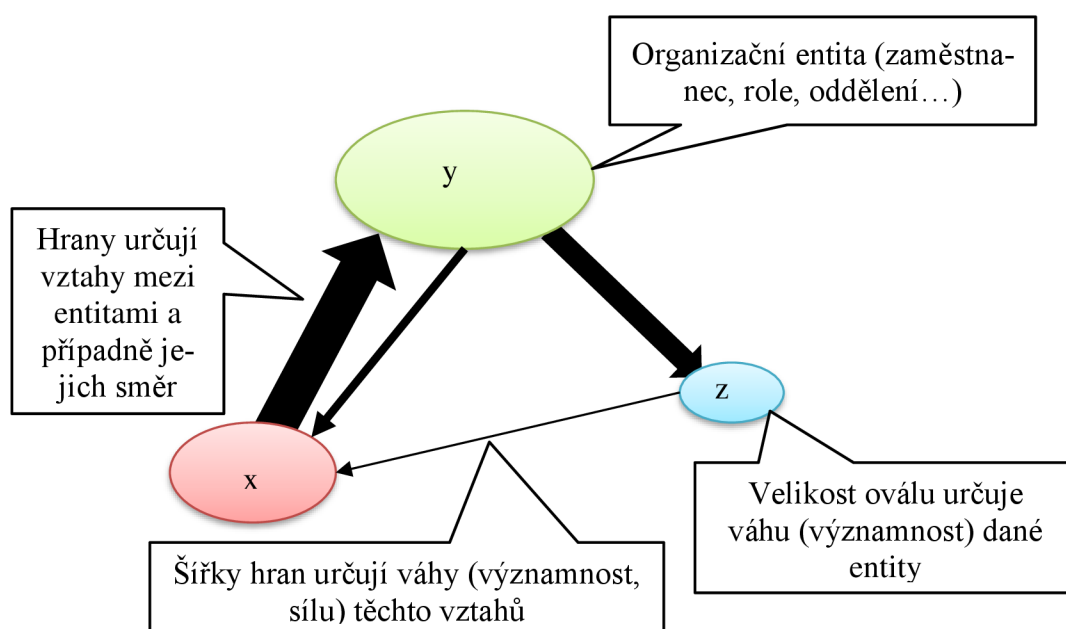
V této podkapitole si představíme několik **různých perspektiv**, na které je možno v rámci process miningu zaměřit. Jak jsme si již řekli, záznamy obvykle obsahují informace, které umožňují obohatit modely procesů, které jsme doposud viděli. V první řadě se podíváme na **organizační perspektivu**, dále na **časovou**.

6.6.1 ORGANIZAČNÍ PERSPEKTIVA

Na organizační perspektivu se soustředí tzv. **organizational mining**. Potřebným atributem pro organizační perspektivu je atribut zdroj ($\#_{zdroj}(e)$), který bývá v záznamech často přítomen. Organizační perspektiva nabízí techniky, díky kterým se lze dovědět více o zaměstnancích, organizační struktuře, dělení práce a pracovních vzorech. Organizační perspektivy lze dle toho, co chceme zjistit dále dělit na: analýzu sociálních sítí či objevování organizačních struktur.

Analýza sociálních sítí (Social Network Analysis – SNA)

Jedná se o jednu ze sociometrických metod, která se zabývá analýzou interpersonálních vztahů v podobě grafů či matic. V minulosti byla data pro sociometrický výzkum získávána prostřednictvím dotazníkových šetření anebo rozhovorů. Nicméně v dnešní době lze tato data získávat prostřednictvím elektronických zdrojů.



Obrázek 6-15: Jednoduchá sociální síť

Zdroj: upraveno podle Aalsta (2016)

Uzly grafů sociálních sítí v našem případě korespondují s entitami vyskytujícími se v organizacích. Velmi často se jedná o zaměstnance organizace či obecně lidi, avšak uzly mohou reprezentovat také agregované organizační entity jako role, skupiny pracovníků či celá oddělení. Hrany v grafu určují vztahy mezi uzly, tedy mezi jednotlivými organizačními entitami. Jak hranám, tak uzlům, mohou být přiřazeny váhy určující jejich významnost či sílu. V případě obrázku výše je vidět, že nejvýznamnějším uzlem je uzel y a nejvýznamnější vztah je mezi uzly (x, y) . Konkrétní interpretace významnosti či síly je závislá na konkrétní sociální síti, která je analyzována. Například budou-li jednotlivé uzly reprezentovat oddělení ve firmě, je možno analyzovat předávání práce, kde velikost uzlu by mohla reprezentovat, s jakým podílem vykonaných se jednotlivá oddělení podílejí na daném procesu, a velikost hrany zase, která oddělení mezi sebou nejvíce spolupracují.

Objevování organizačních struktur

Chování jednotlivých zdrojů může být charakterizováno profily těchto zdrojů, které mohou mít například podobu vektorů, kde jednotlivé složky budou tvořeny frekvencemi, s jakými daný zdroj vykonával jednotlivé aktivity. Na takový typ dat lze poté aplikovat různé metody **clusterové analýzy** pro objevení podobně se chovajících zdrojů. Mezi ty nejznámější clusterové analýzy patří *k-means clustering* a *agglomerative hierarchical clustering*.

6.6.2 ČASOVÁ A PRAVDĚPODOBNOSTNÍ PERSPEKTIVA

Časové perspektiva je zaměřena na analýzu doby výskytu událostí a frekvencí s jakou se dané události vyskytují. Časové značky jsou atributy vyskytující se prakticky v každém záznamu. Tyto časové značky se mohou vyskytovat s různou granularitou nebo jinak zrnitostí. Některé záznamy mohou poskytovat informace o výskytu událostí s přesností na dny jiné zase s přesností na milisekundy. Přítomnost časových značek umožňuje analýzu úzkých míst, monitorování využití zdrojů či predikci časových značek ještě neproběhlých aktivit a další s výkonností související charakteristiky.

Tímto jsme uzavřeli povídání o process miningu a v následujících dvou kapitolách se budeme věnovat praktické aplikaci získaných poznatků. Představíme si komerční a uživatelsky přívětivější software Disco a akademický a méně přívětivější, ale o to mocnější nástroj ProM.



OTÁZKY

1. Co je podstatou process miningu?
2. Vyjmenujte tři základní typy process miningu?
3. Co je to událost a případ?
4. Jaký je rozdíl mezi případem a instancí procesu?
5. Co je to záznam?
6. Jak definujeme přechodový systém?

7. Jak definujeme Petriho síť?
 8. Jaký je rozdíl Petriho sítí a značenou Petriho sítí?
 9. Jaké tři vlastnosti musí Petriho síť splňovat, aby se jednalo o WF-síť?
 10. Jaké tři vlastnosti musí Petriho síť splňovat, abychom o ní mohli říci, že je spolehlivá?
 11. Co je podstatou process miningového přístupu zjišťování shodnosti?
 12. Jaký je rozdíl mezi deskriptivním a normativním modelem?
 13. Která čtyři kritéria používáme pro hodnocení kvality modelů?
 14. Jaké process miningové perspektivy znáte?
-

SHRNUTÍ KAPITOLY



V této kapitole jsme představili základní pojmy z oblasti process miningu. Věnovali jsme se zejména objevování procesů a zjišťování shodnosti, nicméně neopomněli jsme zmínit ani zlepšování procesů a operační podporu. Jelikož je process mining z části datová věda, neopomněli jsme ani datovou část, která může často zabrat i více jak polovinu času celé analýzy. Ještě než jsme začali objevovat procesy, ukázali jsme si dva modelovací nástroje, a sice přechodový systém a Petriho síť. Kapitulu jsme uzavřeli stručným náhledem do jiných než k případům vztažených perspektiv.

ODPOVĚDI



1. Podkapitola 6.1, str. 105 až 107.
 2. Podkapitola 6.1, str. 109.
 3. Podkapitola 6.2.1, str. 113 až 115.
 4. Podkapitola 6.1 a 6.2.
 5. podkapitola 6.2.1.
 6. Podkapitola 6.3.1.
 7. Podkapitola 6.3.2, str. 117 až 119.
 8. Podkapitola 6.3.2, definice na str. 118.
 9. Podkapitola 6.3.2, str. 119.
 10. Podkapitola 6.3.2, str. 120.
 11. Podkapitola 6.5.
 12. Podkapitola 6.5, definice na str. 133.
 13. Podkapitola 6.5.
 14. Podkapitola 6.6.
-

7 PŘÍPADOVÁ STUDIE DISCO



RYCHLÝ NÁHLED KAPITOLY

Tato kapitola bude seznámením s praktickou aplikací process miningových metod na skutečných datech. Dále představíme použitelné formáty záznamů. To vše bude provedeno v prostředí softwarového produktu Disco. Věnovat se budeme zejména objevování procesů, nicméně najde se prostor i pro zlepšování procesů.



CÍLE KAPITOLY

Po prostudování této kapitoly budete umět:

- Provádět základní process miningové analýzy v prostředí Disco.
 - Orientovat se v nástroji Disco.
 - Pracovat se základními formáty dat pro process miningovou analýzu.
-



KLÍČOVÁ SLOVA KAPITOLY

Process mining, Disco, případová studie, formát dat, XES, MXML

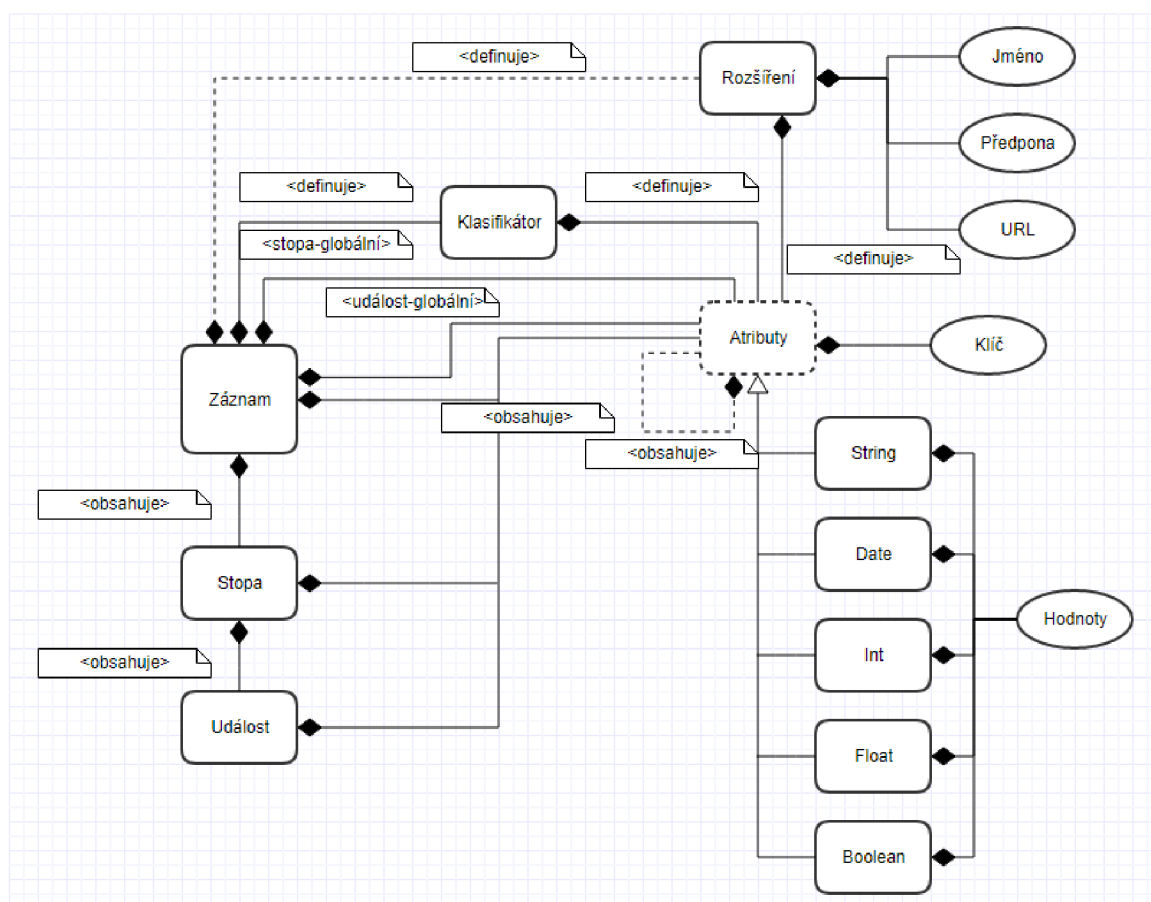
Ještě než se pustíme do samotné případové studie, pojďme si představit základní záznamů, které jsou přijatelné pro process miningovou analýzu, konkrétně se jedná o XML, MXML, CSV a XES.

7.1 XES

Do roku 2010 byl standardem pro ukládání a výměnu záznamů jazyk MXML (*Mining eXtensible Markup Language*), který se objevil v roce 2003, a který prvně adoptoval softwarový nástroj ProM. Syntaxe MXML je založena na syntaxi jazyka XML (*eXtensible Markup Language*). MXML má standardní notaci pro ukládání časových značek, zdrojů a typů transakcí. Možností je také přidávání libovolných vlastností událostem a případům v záznamu. Přestože MXML v praxi fungoval velmi dobře, různá rozšíření, která se v rámci process miningových analýz používala, odhalila mnoho nedostatků. Tyto nedostatky zapříčinily vývoj XES¹⁷ (*eXtensible Event Stream*). V roce 2010 byl formát XES adoptován

¹⁷ XES. *XES-Standard* [online]. 2017 [cit. 2017-11-10]. Dostupné z: <http://www.xes-standard.org/>

IEEE Task Force on Process Mining. V té době se stal XES v podstatě základním formátem záznamů pro process miningové analýzy. V roce 2016 se pak stal oficiálním standardem organizace IEEE Standards Organization.



Obrázek 7-1: Meta model XES

Zdroj: upraveno podle Aalsta (2016)

Jak ukazuje meta model XES na obrázku 7-1, XES dokument obsahuje jeden záznam skládajícího se z jakéhokoliv počtu stop. Každá stopa popisuje posloupnost událostí připadající na každý případ. Záznam, jeho stopy a události mohou mít jakýkoliv vlastností a tyto vlastnosti mohou být vnořené. XES obsahuje pět základních typů a sice: *string*, *date*, *int*, *float* a *boolean*. XES nepředepisuje množinu nutných vlastností pro každý element (záznam, stopa nebo událost). Nicméně pro poskytnutí sémantiky každému atributu – vlastnosti, záznam obsahuje tzv. **rozšíření**, která poskytují sémantiku jednotlivým atributům. Ukázkou části záznamu ve formátu XES můžete vidět na obrázku níže. Pro více informací odkazujeme čtenáře na Aalsta (2016) případně Aalsta (2011).

Základní typy XES

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!-- This file has been generated with the OpenXES library. It conforms -->
3 <!-- to the XML serialization of the XES standard for log storage and -->
4 <!-- management. -->
5 <!-- XES standard version: 1.0 -->
6 <!-- OpenXES library version: 1.0RC7 -->
7 <!-- OpenXES is available from http://www.openxes.org/ -->
8 <log xes:version="1.0" xes:features="nested-attributes" openxes:version="1.0RC7" xmlns="http://www.xes-standard.org/">
9   <extension name="Lifecycle" prefix="lifecycle" uri="http://www.xes-standard.org/lifecycle.xesext"/>
10  <extension name="Organizational" prefix="org" uri="http://www.xes-standard.org/org.xesext"/>
11  <extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext"/>
12  <extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.xesext"/>
13  <extension name="Semantic" prefix="semantic" uri="http://www.xes-standard.org/semantic.xesext"/>
14  <global scope="trace">
15    <string key="concept:name" value="__INVALID__"/>
16  </global>
17  <global scope="event">
18    <string key="concept:name" value="__INVALID__"/>
19    <string key="lifecycle:transition" value="complete"/>
20  </global>
21  <classifier name="MXML Legacy Classifier" keys="concept:name lifecycle:transition"/>
22  <classifier name="Event Name" keys="concept:name"/>
23  <classifier name="Resource" keys="org:resource"/>
24  <classifier name="Event Name AND Resource" keys="concept:name org:resource"/>
25  <string key="source" value="CPN Tools simulation"/>
26  <string key="concept:name" value="repairExample.xml"/>
27  <string key="lifecycle:model" value="standard"/>
28  <string key="description" value="Simulated process"/>
29  <trace>
30    <string key="concept:name" value="1"/>
31    <string key="description" value="Simulated process instance"/>
32    <event>
33      <string key="org:resource" value="System"/>
34      <date key="time:timestamp" value="1970-01-02T12:23:00.000+01:00"/>
35      <string key="concept:name" value="Register"/>
36      <string key="lifecycle:transition" value="complete"/>
37    </event>
38    <event>
39      <string key="org:resource" value="Tester3"/>
40      <date key="time:timestamp" value="1970-01-02T12:23:00.000+01:00"/>
41      <string key="concept:name" value="Analyze Defect"/>
42      <string key="lifecycle:transition" value="start"/>
43    </event>

```

Obrázek 7-2: Záznam v XES

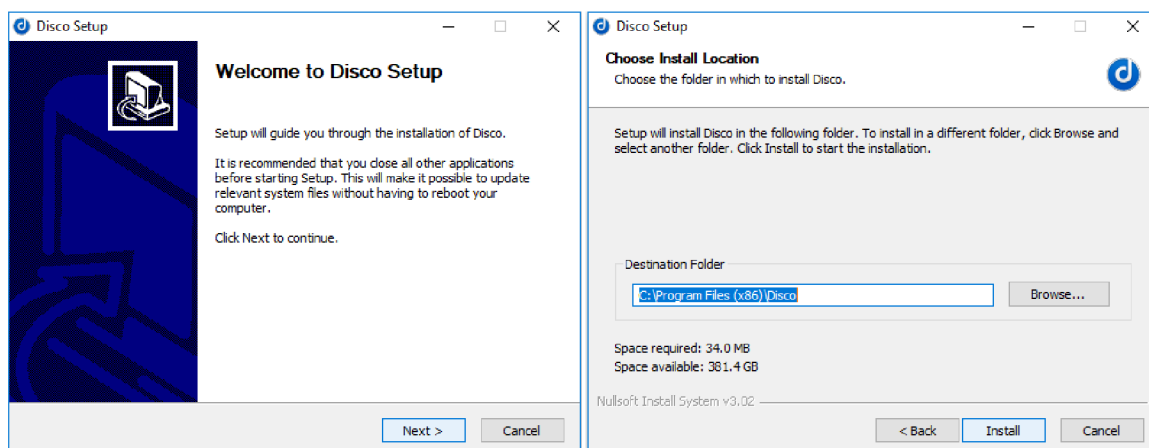
Zdroj: Vlastní zpracování

Kromě XES a MXML souborů si Disco poradí také s CSV (*Comma Separated Values*), TXT (*Text files*), XLS/XLSX (*Microsoft Excel*) či FBT soubory.

7.2 Instalace softwarového produktu Disco

Disco¹⁸ je produktem Nizozemské společnosti Fluxicon. Tuto společnost založili bývalí studenti ve skupině profesora Aalsta na Technické univerzitě v Eindhovenu, Anne Rozinat a Christian W. Günther. Oba mají zázemí v oblasti softwarového inženýrství a process miningu. Ještě než se popíšeme prostředí produktu Disco, rychle si projdeme proces instalace. Disco je k dispozici ke stažení online (viz poznámka pod čarou číslo 16). Disco je k dispozici jak pro uživatele systému Windows tak Mac OS. Stáhněte si příslušnou verzi *.exe* souboru a dvojklikem na soubor *Disco-Setup.exe* spusťte instalaci. Poté klikněte na tlačítko „Next“ (obrázek 7-3 vlevo) a následně „Install“ (obrázek 7-3 vpravo).

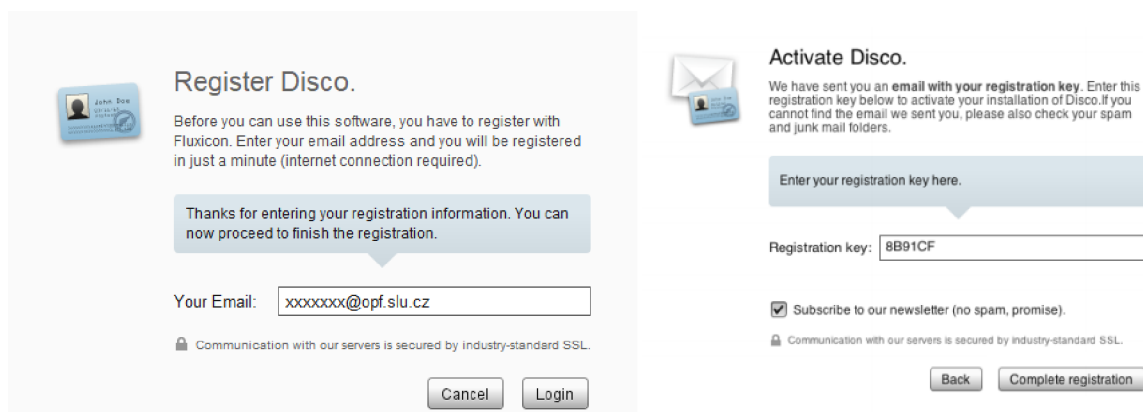
¹⁸ Disco. Fluxicon: process mining for professionals [online]. 2017 [cit. 2017-11-10]. Dostupné z: <https://fluxicon.com/disco/>



Obrázek 7-3: Instalace Disco

Zdroj: Vlastní zpracování

Ve chvíli, kdy Disco spustíte poprvé, budete dotázáni pro souhlas s licenčními podmínkami, zakřížkujte pole „*I have read and understood this license.*“ a potvrďte stisknutím „*I accept these terms*“.



Obrázek 7-4: Registrace Disco

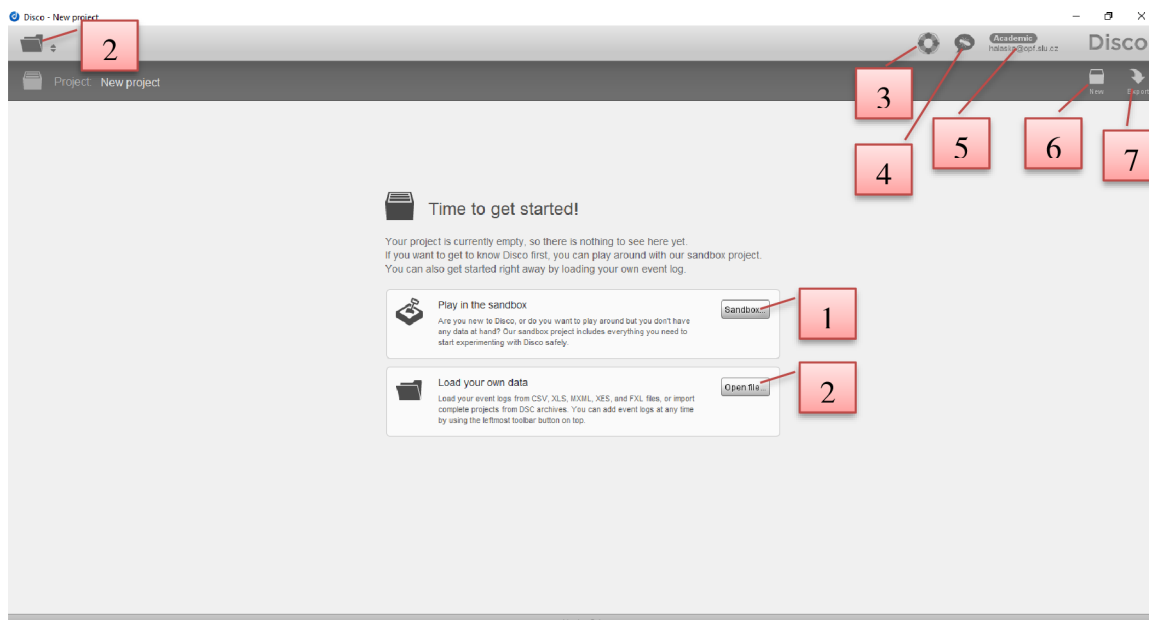
Zdroj: vlastní zpracování

Po vyjádření souhlasu s licenčními podmínkami je potřeba kopii Disco registrovat. Na OPF je pořízena licence pro software Disco, a registrace probíhá prostřednictvím školního emailu ve formátu xxxxxxx@opf.slu.cz, namísto xxxxxxx použijte své uživatelské jméno pro přihlášení (viz obrázek 7-4 vlevo). Na email Vám poté bude zaslán email s registračním klíčem, který vyplníte do pole „*Registration key*“ a kliknete na políčko „*Complete registration*“ (viz obrázek 7-4 vpravo). V případě problémů odkazujeme studenty na vyučující ať již osobně či prostřednictvím emailu, případně na uživatelskou příručku¹⁹.

¹⁹ Disco. Disco User's Guide [online]. 2017 [cit. 2017-11-10]. Dostupné z: <https://fluxicon.com/disco/files/Disco-User-Guide.pdf>

7.3 Analýza procesů v Disco

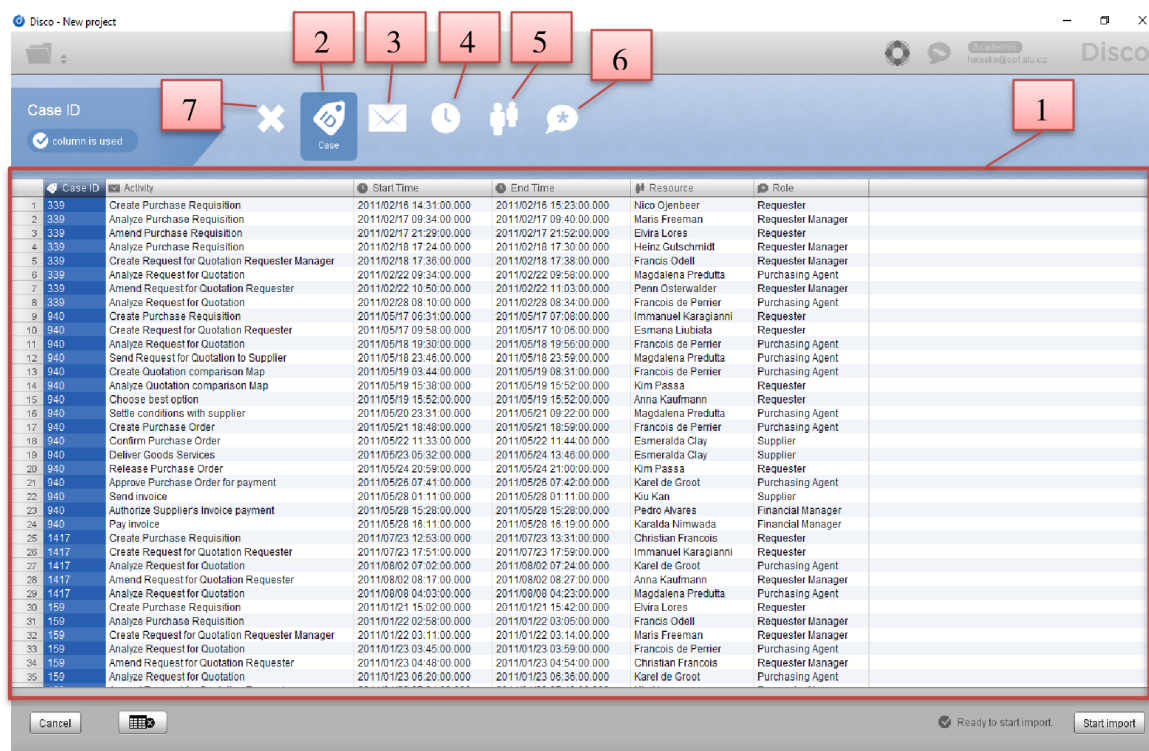
Vše máme připraveno a můžeme analyzovat. V rámci této kapitoly budeme analyzovat vybraný proces a zároveň si i představíme prostředí Disco. Po provedení registrace se nám objeví úvodní obrazovka, kterou můžete vidět na obrázku 7-5 níže.



Obrázek 7-5: Disco úvodní obrazovka

Zdroj: vlastní zpracování

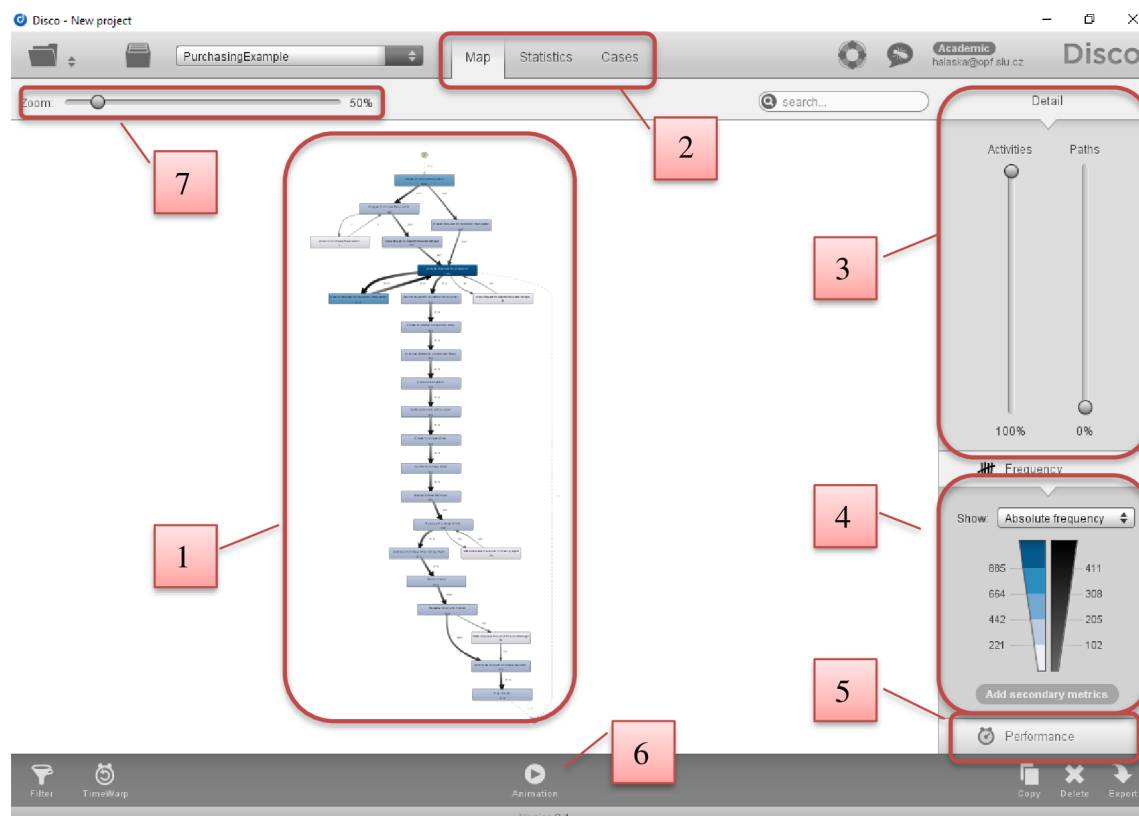
1. Při kliknutí na políčko „*Sandbox*“ se vytvoří úvodní projekt pro nové uživatele s předem nahranými daty a nápovědou pro vyzkoušení si softwaru Disco.
2. V případě, že máme svá vlastní data, můžeme je importovat buď stisknutím tlačítka „*Open file*“ anebo kliknutím na ikonu v levém horním rohu na obrázku 7-5.
3. Kliknutím na ikonu číslo 3 na obrázku 7-5 si můžeme zobrazit nápovědu.
4. Kliknutím na ikonu číslo 4 na obrázku 7-5 lze nahlásit problém se softwarem.
5. Kliknutím na ikonu číslo 5 na obrázku 7-5 lze změnit uživatele.
6. Kliknutím na ikonu číslo 6 na obrázku 7-5 lze vytvořit nový projekt.
7. Kliknutím na ikonu číslo 7 na obrázku 7-5 lze uložit stávající projekt.



Obrázek 7-6: Import dat v Disco

Zdroj: vlastní zpracování

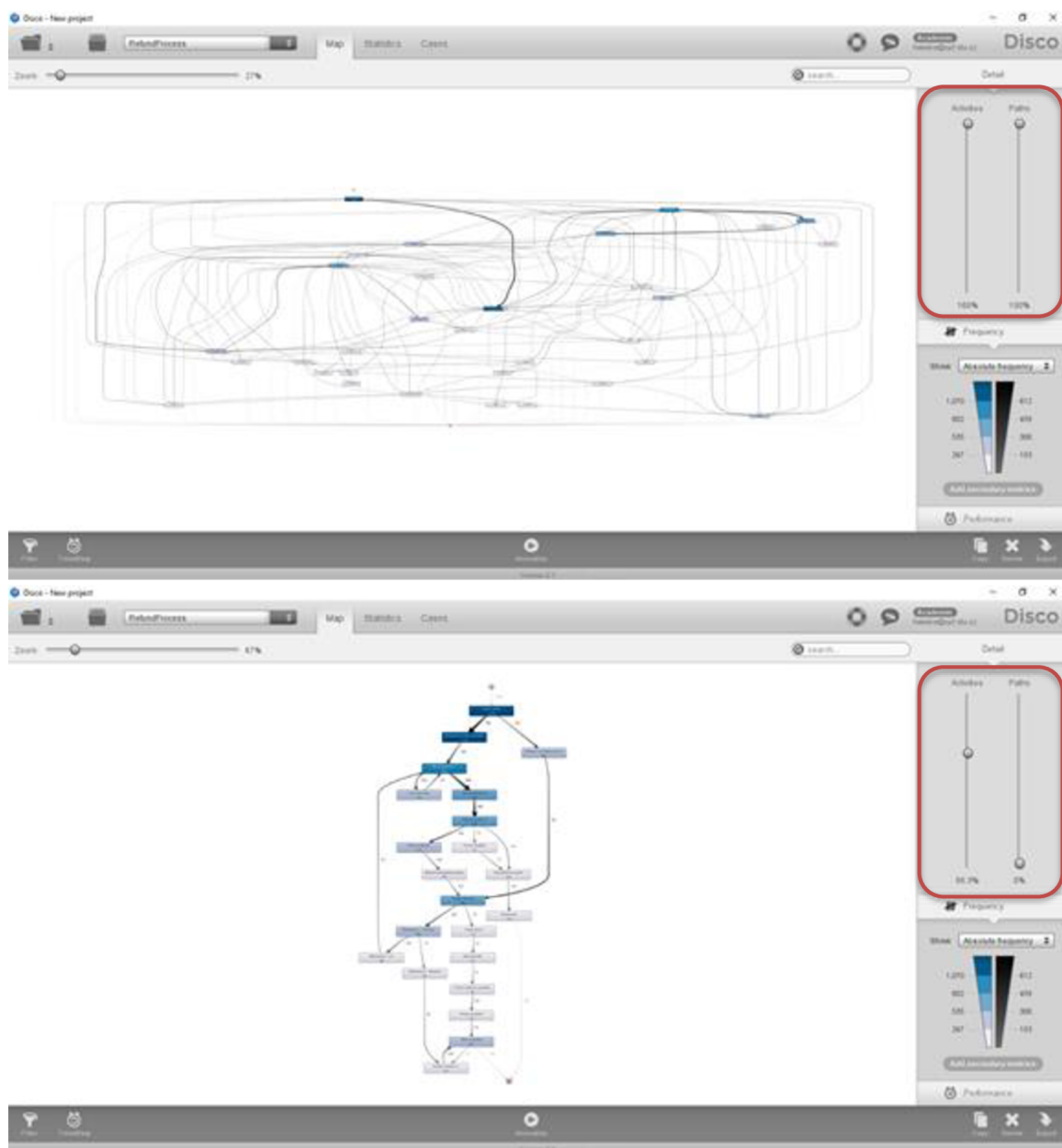
Již jsme seznámeni s úvodní obrazovkou v Disco, pojdme si do Disca importovat nějaká data. V naší případové studii se budeme zabývat nákupním procesem, nahrajeme si tedy data s názvem „*PuchasingExample.ftb*“. Kliknutím na tlačítko či ikonu 2 v obrázku 7-5 se nám objeví klasická nabídka pro zadání cesty k požadovanému souboru, soubor vyhledáme, označíme a klikneme na otevřít. U souborů typu .ftb, .csv, .xls či .xlsx je potřeba určit, který sloupec reprezentuje případ, který aktivitu atd. Na rozdíl od souborů typu .xes či .mxml, u nichž to bývá určeno v meta datech daného souboru. V našem případě sice Disco automaticky rozpozná jednotlivé sloupce v našich datech (obrázek 7-6 popisek 1). V opačném případě se postupuje následovně. Klikneme na sloupec, který chceme určit a poté na jednu z ikon popisků 2 až 6 na obrázku 7-6. V našem případě máme vybrán první sloupec nazvaný „*Case ID*“ a poté jsme klikli na ikonku s popiskem 2. Tím jsme Disco řekli, že ve sloupci „*Case ID*“ se nacházejí jednotlivé případy (anglicky *case*). Pod popisem 3 jsou aktivity (anglicky *activity*), pod popisem 4 jsou časové značky (anglicky *timestamp*), pod popisem 5 jsou zdroje (anglicky *resources*), pod popisem 6 jsou ostatní (anglicky *other*) a pod popisem 7 je odstranění (anglicky *remove*), čímž Disco můžeme říci, aby daný sloupec ignoroval. Ve chvíli kdy máme určen, které sloupce určují případy, které aktivity, které časové značky atd., stiskneme tlačítko „*Start import*“ v pravém spodním rohu obrazovky (viz obrázek 7-6).



Obrázek 7-7: Úvodní analýza Disco

Zdroj: vlastní zpracování

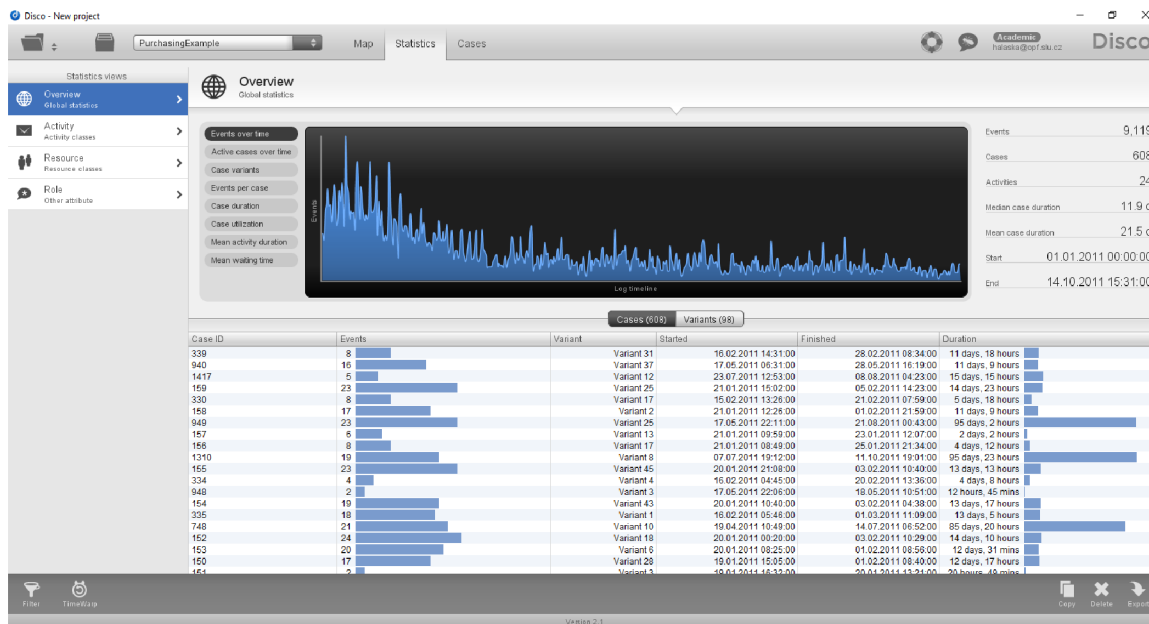
Po importování dat nám Disco automaticky vygeneruje procesní mapu, jak lze vidět na obrázku 7-7 popisek 1. Mapu si lze přiblížit ať už prostřednictvím posuvníku (popisek 7) tak kolečkem myši. Nákupní proces na obrázku 7-7 je **strukturovaný proces**, ve kterém už se lze orientovat a takto zobrazený proces lze analyzovat, tyto typy procesů se nazývají **lasagna procesy**, jak jsme si již řekli v kapitole 6.2. Říká se jim tak proto, že připomínají lazagně, které také mají jednotlivé vrstvy strukturované a nepřekrývající se. V případě, že bychom neměli takto pěkně strukturovaný proces je možno využít panel „Detail“ (popisek 3), ve kterém lze objevenou procesní mapu zjednodušit odebráním méně frekventovaných aktivit prostřednictvím posuvníku „Activities“ anebo odebráním méně frekventovaných cest prostřednictvím posuvníku „Paths“. Obecně je lépe při zjednodušování procesní mapy začít skrze cesty a v případě, že to nestačí, začít odebírat aktivity. Nicméně záleží případ od případu a tedy na použitých datech, v některých částech je process mining stále spíše umění než věda, a jde o to najít ten správný pohled. Zajímá-li nás například hlavní tok daného procesu, je lépe se rovnou soustředit na aktivity. Na obrázku 7-8 níže je nahoře ukázka nestrukturovaného procesu vrácení peněz, a dole je tento proces strukturovaný s využitím panelu „Detail“. Vlevo dole lze aplikovat různé filtry („Filter“) a vpravo dole lze kopírovat data („Copy“), odstranit data („Delete“) anebo exportovat data („Export“).



Obrázek 7-8: Špagetový nestrukturovaný proces (nahore) a lazagna strukturovaný process (dole)

Zdroj: vlastní zpracování

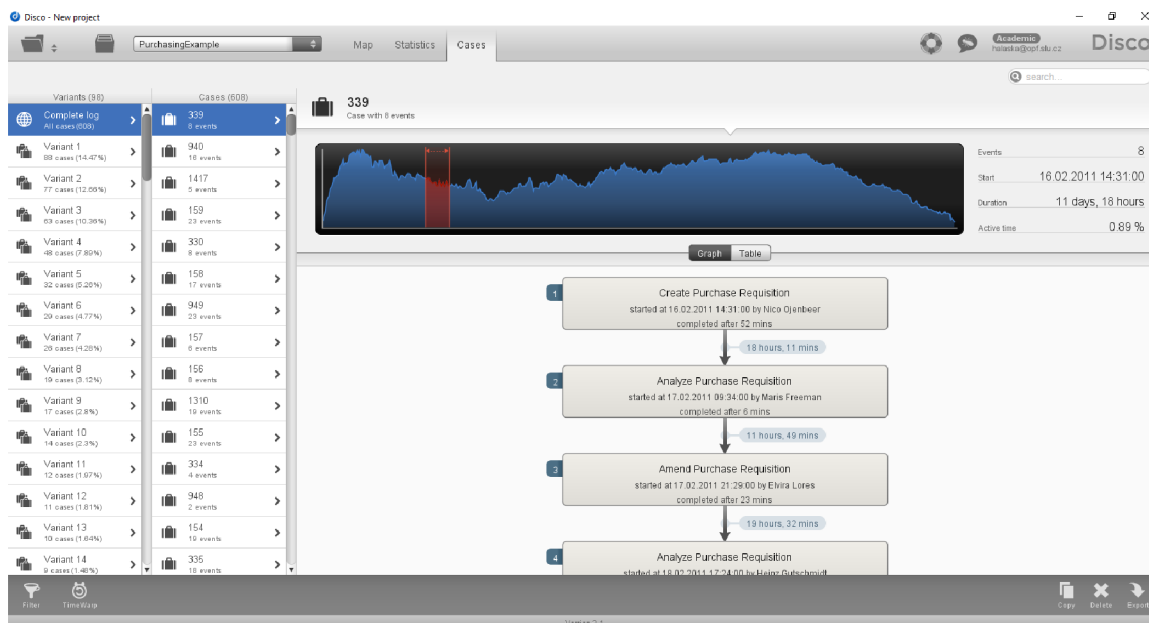
Nicméně vraťme se opět k našemu nákupnímu procesu. V rámci popisku 2 obrázku 7-7 si máme tři zobrazení, a sice procesní mapu („*Map*“) tak ji vidíme na obrázku. Dále je možné si zobrazit různé statistiky kliknutím na políčko „*Statistics*“ viz obrázek 7-9 níže. Můžeme zjistit, že v našem záznamu se vyskytuje 9119 událostí a 608 případů, což jsou globální statistiky („*Overview*“), nicméně lze si zobrazit i statistiky vázané na aktivity („*Activity*“), zdroje („*Resource*“) či role („*Roles*“) spolu se statistikami jako průměr, medián, rozložení v čase apod.



Obrázek 7-9: Statistika záznamu

Zdroj: vlastní zpracování

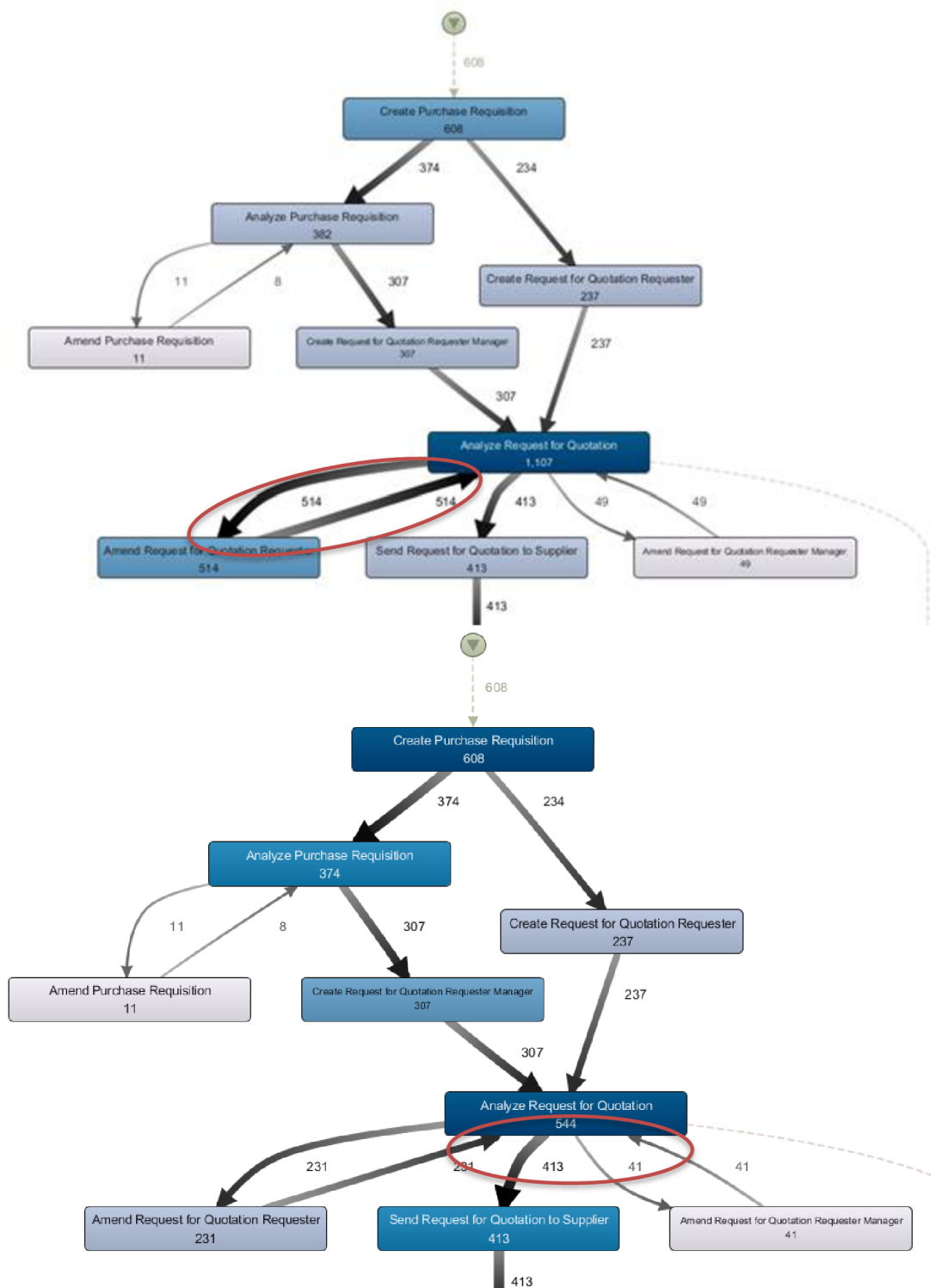
Kromě statistik se lze také zaměřit přímo na jednotlivé případy kliknutím na položku „Cases“. Tento náhled lze vidět na obrázku 7-10 níže. Jednotlivé případy lze analyzovat položku po položce, či v rámci jednotlivých stop, což jsou skupiny případů se stejným sledem aktivit či událostí. V námi analyzovaném procesu máme 608 případů, které se vyskytují v 98 variantách.



Obrázek 7-10: Detail případů a stop

Zdroj: vlastní zpracování

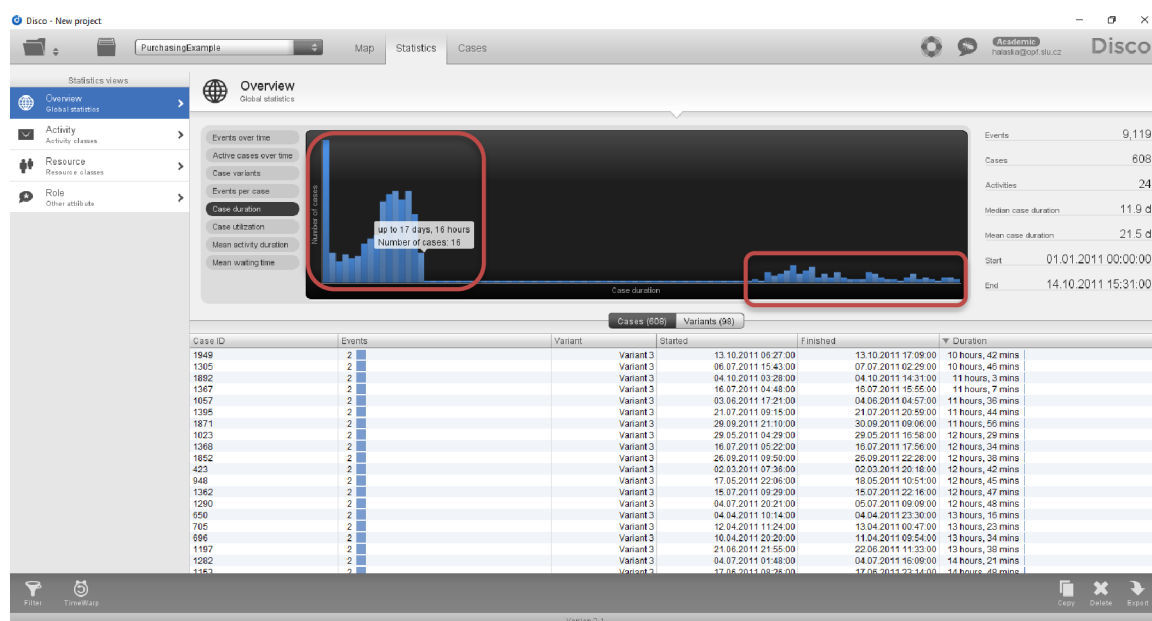
Vrátíme-li se opět k obrázku 7-7 popiskům 4 a 5, vidíme, že nám Disco umožňuje analyzovat daný proces ze dvou perspektiv, a sice z pohledů toků jednotlivých případů („Frequency“) a časové perspektivy („Performance“), která nám umožňuje sledovat výkony v rámci daného procesního modelu.



Obrázek 7-11: Počáteční část nákupního procesu

Zdroj: vlastní zpracování

Nyní se vraťme k naší procesní mapě z obrázku 7-7 a soustředíme se na horní část naší procesní mapy. I na procesní mapě je vidět, že máme 608 případů vycházejících ze vstupního místa (trojúhelník v kolečku na vrcholu procesní mapy). Všechny 608 případů začíná aktivitou „Vytvoření nákupního požadavku“ („*Create purchase requisition*“). Po vytvoření nákupního požadavku můžeme vidět, že se nám celý proces dělí. V 374 případech je provedena aktivita „Analýza požadavku na nákup“ („*Analyze purchase requisition*“) a v 234 případech byla provedena aktivita „Vytvoření požadavku pro žadatele o nabídku“ („*Create request for quotation requester*“). Aktivita „Analýza požadavku na nákup“ byla ve skutečnosti provedena 382, jelikož 11 požadavků na nákup bylo po analýze upraveno v rámci aktivity „Pozměnění požadavku na nákup“ („*Amend purchase requisition*“) a 8 z nich bylo vráceno zpět k analýze. Jistě se ptáte, co se stalo se třemi pozměněnými požadavky, které se nevrátily k analýze. Tato část cesty byla odstraněna z důvodu nízkého počtu výskytů, takže pozměněné požadavky se nikam neztratily, jen zrovna nejsou brány v potaz a nemáme je zobrazeny. Podobně aktivita „Vytvoření požadavku pro žadatele o nabídku“ nebyla vykonána jen 234, ale 237. Všimněme si také, že šířka hran mezi jednotlivými aktivitami souvisí s frekvencí, čím hrubší hrana, tím častější výskyt.



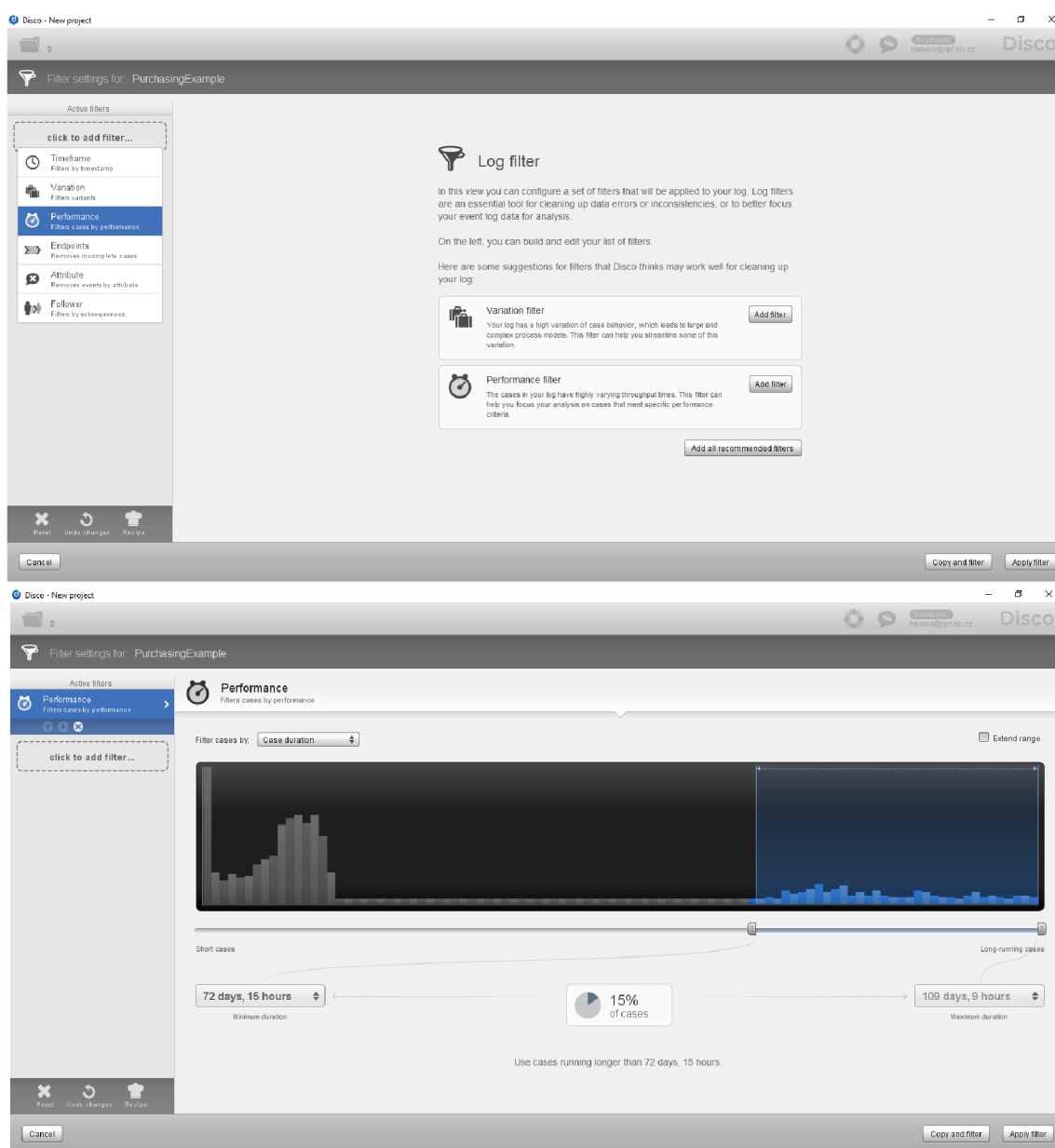
Obrázek 7-12: Délky trvání jednotlivých případů

Zdroj: vlastní zpracování

Co nás ale zaujme na první pohled na dané části procesu je tok vyskytující se v červeném oválu mezi aktivitami „Analýza požadavku o nabídku“ („*Analyze request for quotation*“) a „Pozměnění požadavku o nabídku žadatele“ („*Amend request for quotation requester*“). U aktivity „Pozměnění požadavku na nákup“ se jednalo o 11 případů z 382, zatímco u „Pozměnění požadavku o nabídku žadatele“ se jedná o 514 případů z našich 608 celkových případů. Tato smyčka znamená, že aktivita „Analýza požadavku o nabídku“ je v důsledku nutných změn vykonána namísto řekněme 570-krát 1107-krát, což je velice neefektivní (obrázek 7-11 dole, kde nejsou zobrazeny absolutní frekvence výskytu jednotlivých aktivit,

ale frekvence výskytu v rámci případů, tzn., vyskytla-li se v rámci případu daná aktivita třikrát, bude započítána jen jednou).

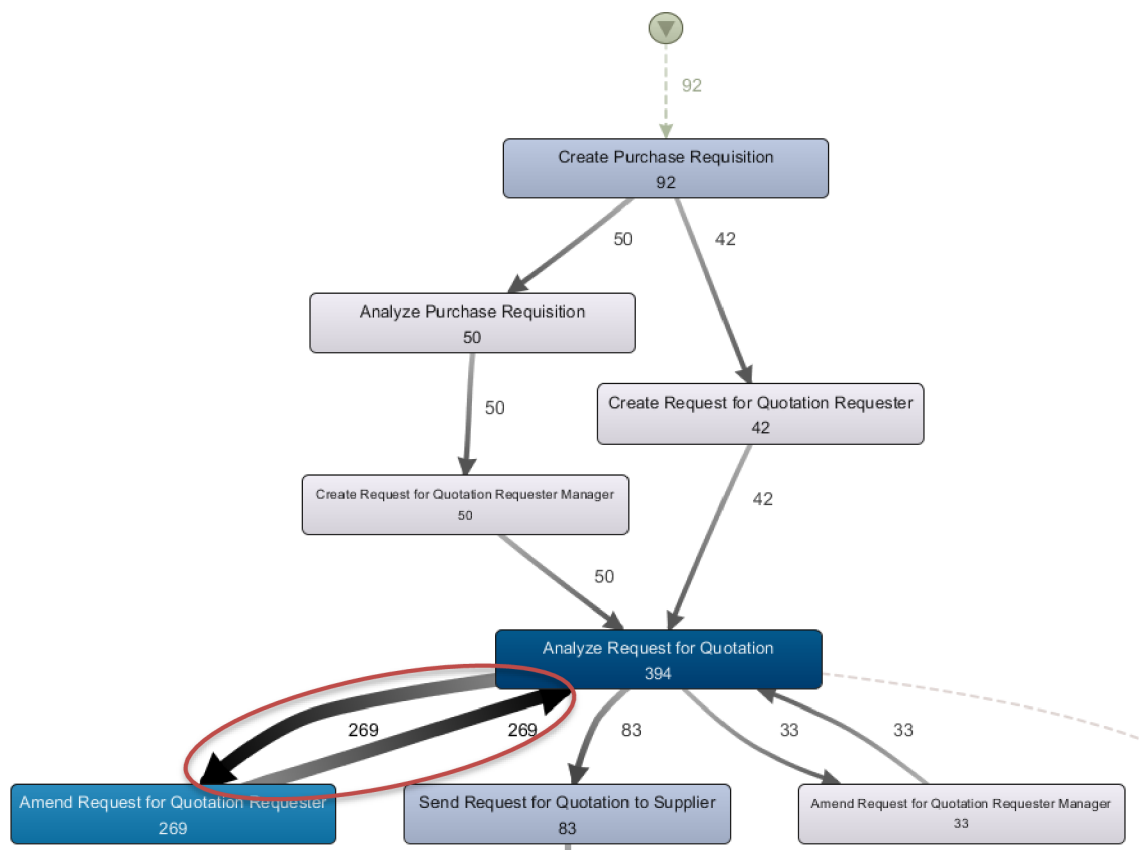
Podíváme-li se v náhledu statistik na dobu trvání jednotlivých případů („Case duration“) v rámci globálních statistiky, zjistíme, že 80 případů bylo vyřízeno v době jednoho a půl dne či kratší (první sloupec v grafu na obrázku 7-12 červený rámeček vlevo). Ve skutečnosti naprostá většina 516 případů byla vyřízena v době kratší než 17 dní a 16 hodin. Přesto zatímco medián doby trvání jednoho případu je 11,9 dne, průměrná délka doby trvání jednoho případu je 21,5 dní. Toto je způsobeno odlehlými hodnotami na grafu na obrázku 7-12 červený rámeček vpravo, jelikož zde máme několik případů trvajících více než 74 dní.



Obrázek 7-13: Aplikace výkonnostního filtru

Zdroj: vlastní zpracování

Můžeme tedy aplikovat filtr a podíváme se na tyto odlehlé případy blíže. V levém dolním rohu tedy klikneme na položku filtr („Filter“). Na další obrazovce klikneme na políčko přidat další filtr („Click to add filter“) a vybereme výkonnostní filtr („Performance“), viz obrázek 7-13 nahoře. A poté navolíme na posuvníku případy, které trvaly 72 a více dní a klikneme na aplikaci filtru („Apply filter“), viz obrázek 7-13 dole.

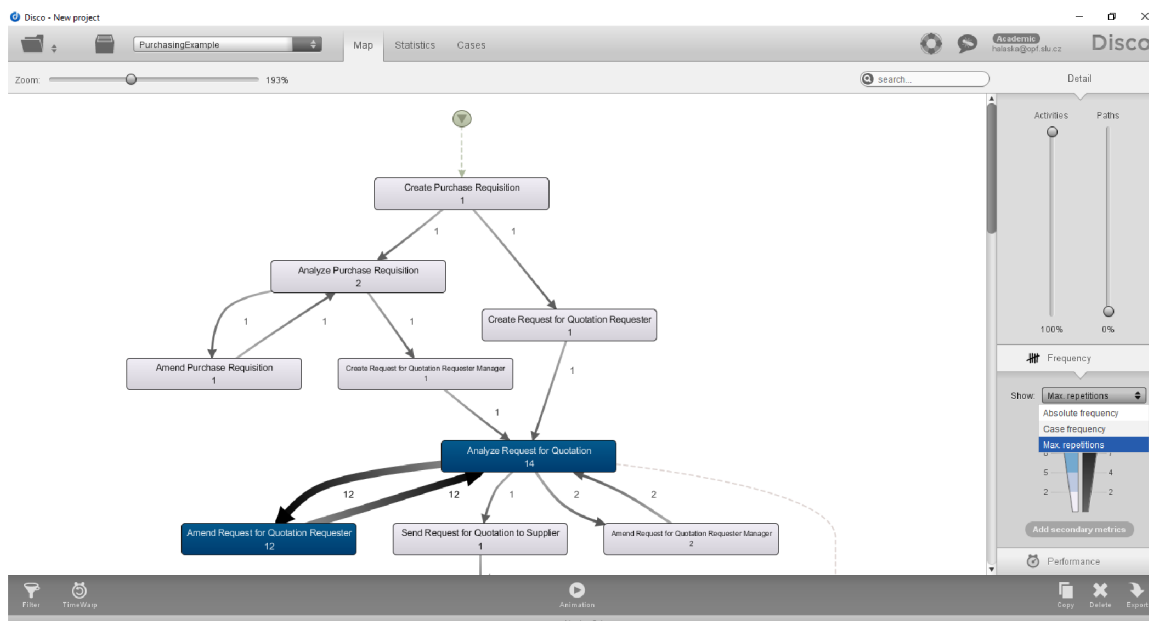


Obrázek 7-14: Počáteční část nákupního procesu po aplikaci filtru

Zdroj: vlastní zpracování

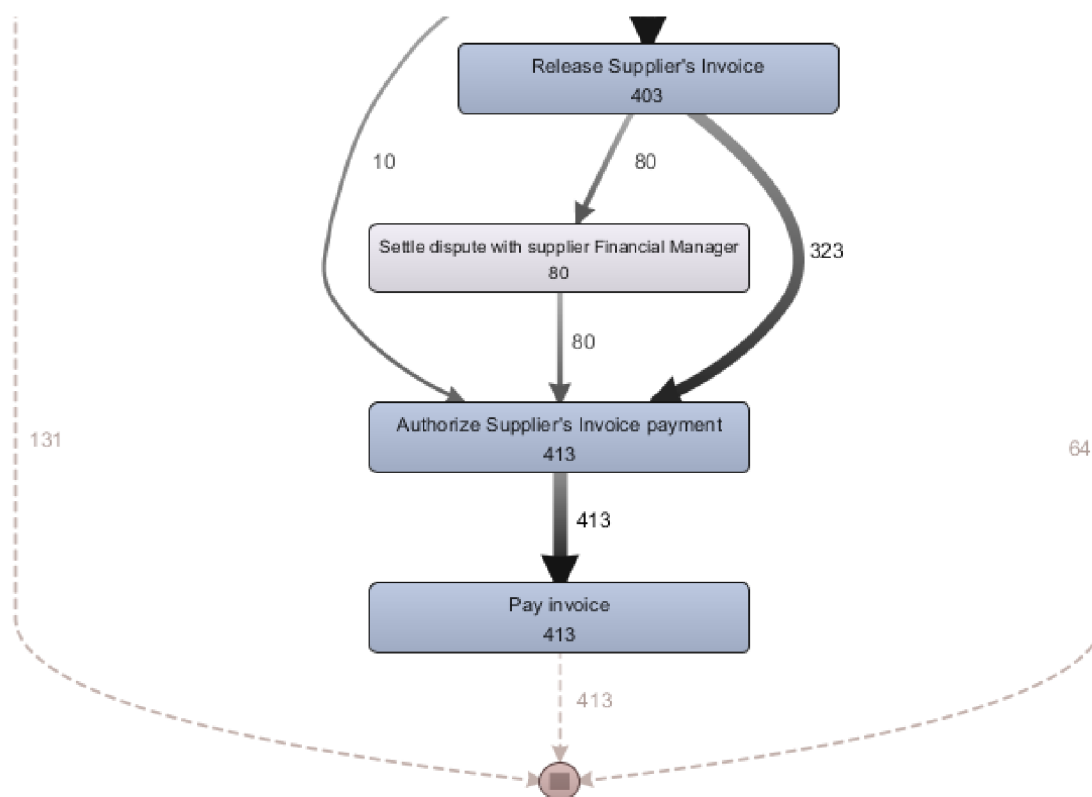
Po aplikaci filtru již pracujeme jen s 92 případy trvajících déle než 72 dní. Před aplikací filtru jsme viděli, že v rámci každého případu byla aktivita „Pozměnění požadavku o nabídku žadatele“ („Amend request for quotation requester“) vykonána v průměru 1,8-krát ($\frac{1107}{608} = 1,8$), zatímco po aplikaci filtru byla aktivita „Pozměnění požadavku o nabídku žadatele“ („Amend request for quotation requester“) vykonána v průměru 2,9-krát ($\frac{269}{92} = 2,9$). Navíc nastavíme-li si místo absolutních frekvencí („Absolute frequency“) zobrazení maximálního počtu opakování v rámci případu (obrázek 7-15 červený rámeček), uvidíme, že v některých případech se aktivita „Pozměnění požadavku o nabídku žadatele“ mohla opakovat až dvanáctkrát a aktivita „Analýza požadavku na nákup“ až čtrnáctkrát. Na základě process miningové analýzy jsme tedy zjistili, že se v našem procesu vyskytuje značná

neefektivita vyúsťující v nadměrné opakování aktivity „Analýza požadavku o nabídku“ způsobené nutností změn.



Obrázek 7-15: Maximální opakování aktivit v procesní mapě

Zdroj: vlastní zpracování

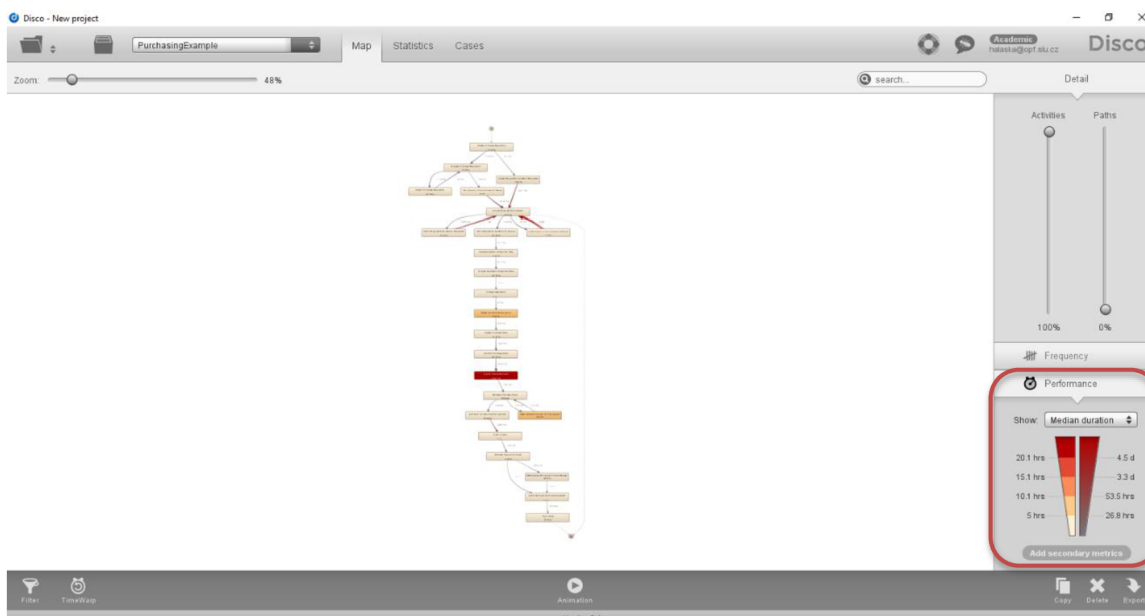


Obrázek 7-16: Koncová část nákupního procesu

Zdroj: vlastní zpracování

Process miningová analýza je založená na datech a pro tuto v podstatě nic jiného nepotřebujeme, nicméně ve chvíli, kdy najdeme úzká místa, velké časové prodlevy apod., a chceme tyto a další problémy v procesu odstranit, či daný proces vylepšit, musíme brát v potaz i samotný proces a hledat příčiny těchto problémů v podniku samotném. V našem případě bychom se tedy soustředili na otázku, jak předejít nutnosti neustálých změn v požadavcích o nabídku.

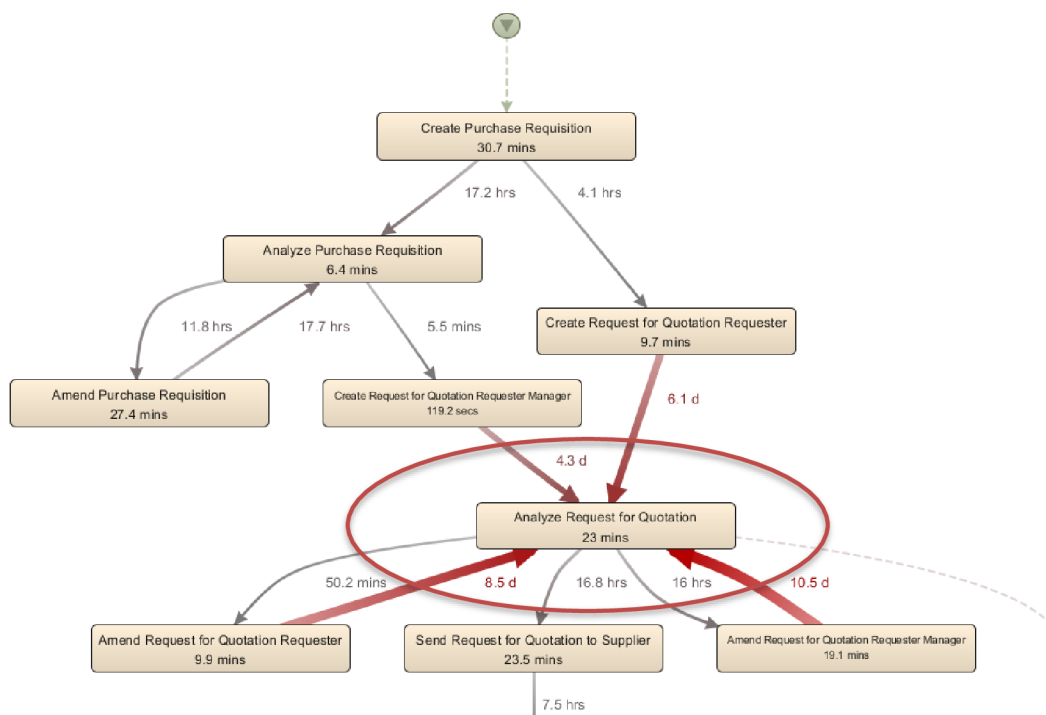
Nyní pozměníme nastavení procesní mapy z obrázku 7-7 v panelu „Detail“ a zobrazíme si kromě všech aktivit i všechny cesty. Další věcí, která by nás měla zaujmout, a které bychom se měli věnovat je, že z 608 případů, které se v záznamu objevily, jich končí uhrazením faktury („Pay invoice“) pouze 413 jak můžeme vidět na obrázku 7-16. Zatímco vstupem procesu byl trojúhelník v kolečku, výstupem procesu je čtverec v kolečku. 131 případů je ukončeno po vykonání aktivity „Analýza požadavku o nabídku“ a 64 jich bylo ukončeno po provedení aktivity „Analýza požadavku na nákup“. Otázkou tedy je proč tomu tak je a jak by se to dalo řešit.



Obrázek 7-17: Procesní mapa v časové perspektivě

Zdroj: vlastní zpracování

Do teď jsme analyzovali toky případů v procesu nákupu, nyní se pojdme podívat na časovou perspektivu a s ní spojenou výkonnostní složku. Tu si zobrazíme, vrátíme-li se k obrázku 7-7 popisku 5 kliknutím do zvýrazněné oblasti výkonu („Performance“). Jak vidíme na obrázku 7-17, procesní mapa zůstane stejná, jen dojde ke změně sledovaných veličin z frekvencí případů, událostí atd., na sledování dob trvání, průměry dob trvání atp.



Obrázek 7-18: Průměrné doby trvání v rámci procesní mapy

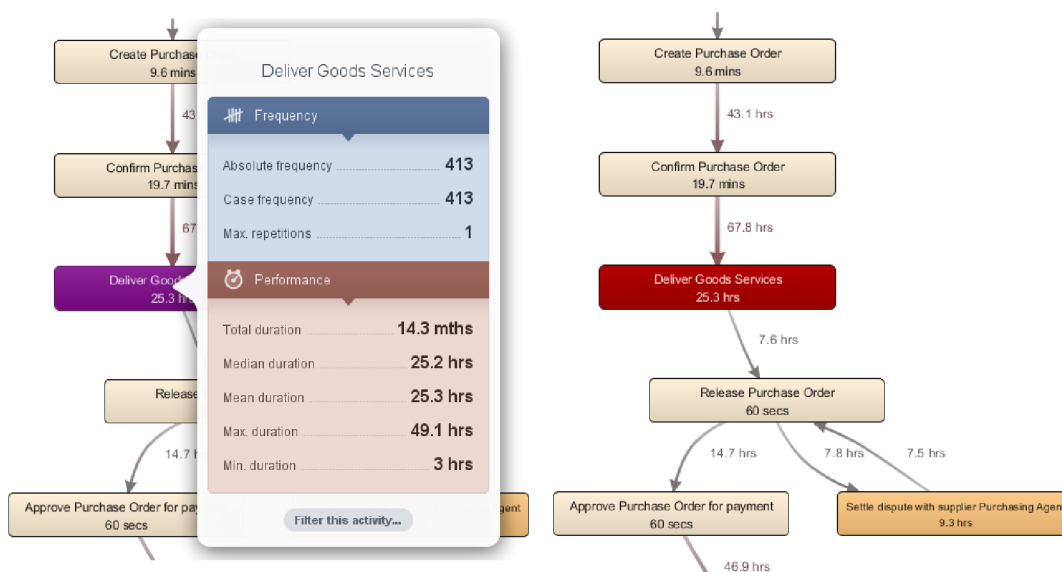
Zdroj: vlastní zpracování

Jak můžeme vidět na obrázku 7-18 máme zde problémovou oblast u vstupních aktivit aktivity „Analýza požadavku o nabídku“ („Analyze request for quotation“), kde jednotlivé aktivity trvají v průměru řádově několik minut, avšak přechody ze vstupních aktivit aktivity „Analýza požadavku o nabídku“ do této aktivity trvají v průměru řádově 4 až 10 dní. U předchozí perspektivy se zdála být problémová zejména aktivita „Pozměnění požadavku o nabídku žadatele“ („Amend request for quotation requester“) nicméně při přechodu na perspektivu časovou vidíme, že problémem je hned několik aktivit. Rozhodně se jeví, že by stálo za bližší prozkoumání všech vstupních aktivit aktivity „Analýza požadavku o nabídku“.

ÚKOL K ZAMYŠLENÍ



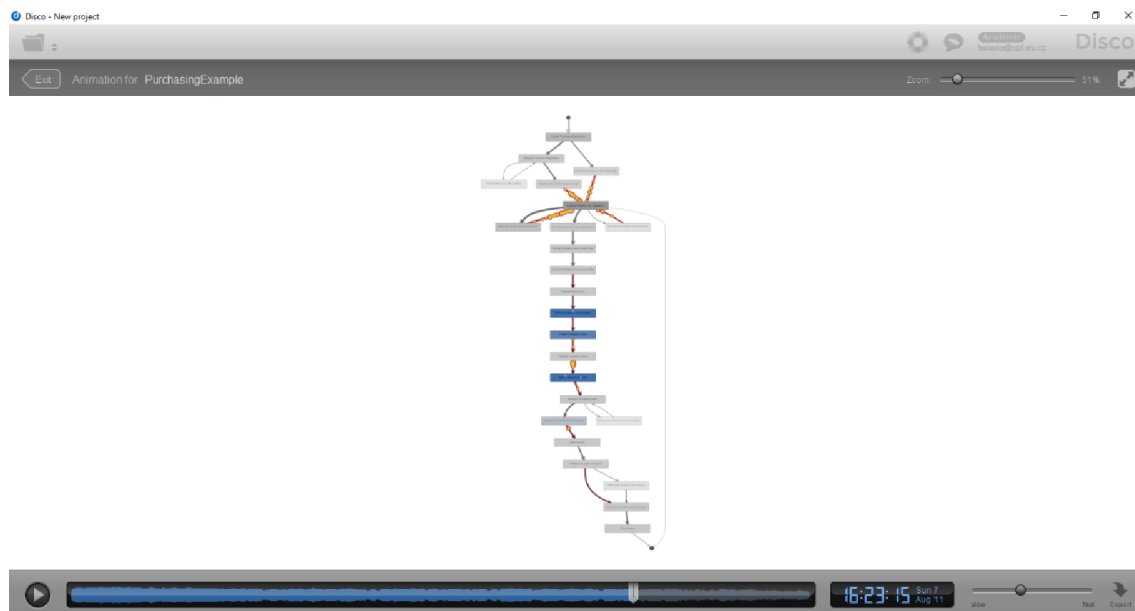
V rámci námi analyzovaného nákupního procesu se z pohledu časové perspektivy vyskytl další potenciální problém, a sice s dobou trvání dovozu zboží („*Deliver goods services*“), jak lze vidět na obrázku 7-19.



Obrázek 7-19: Doba trvání dovozu zboží

Zdroj: vlastní zpracování

Otázkou je, jedná se skutečně o problémovou aktivitu? Upozorňujeme, že Disco upozorňuje na potenciální chyby v kontextu daného záznamu, tzn., že trvá-li většina aktivit řádově minuty, bude logicky aktivita trvající hodiny, bude taková hodnota automaticky označena jako potenciálně problémová.



Obrázek 7-20: Animace nákupního procesu

Zdroj: vlastní zpracování

Posledním prvkem, který si představíme, a který je velmi užitečným nástrojem, je animace. Tohoto docílíme, kliknutím na ikonu animace („Animation“), viz obrázek 7-7 popisek 6. Pomocí tohoto prvku si lze přehrát celý záznam. Jen upozorníme, že se nejedná simulaci. Tímto ukončíme kapitolu 7, zájemce nabádáme k ještě detailnějšímu prozkoumání softwaru Disco. Více dat je k dostání na stránkách *Centre for research data* Technické univerzity v Eindhovenu²⁰.

OTÁZKY



1. Co je to MXML?
2. Co je to XES?
3. Syntaxe XES je založená na jazyku XML – ano či ne?
4. Jaké další formáty dat jsou použitelné?

SHRNUTÍ KAPITOLY



V této kapitole jsme navázali na kapitolu 6.2, která se věnovala datům potřebným pro process mining. V této kapitole jsme si přiblížili formáty dat používané v praxi. Věnovali jsme se zejména formátu XES a MXML, nicméně můžeme se setkat i s formáty CSV či excelovskou tabulkou. Následně jsme si ukázali, jak lze použít process mining v praxi na případové studii týkající se nákupního procesu. V rámci této studie jsme si také blíže představili prostředí Disco. Nicméně jsme nebyli schopni projít všechny možnosti, které Disco nabízí. Zájemce nabádáme k detailnějšímu prozkoumání tohoto softwaru.

ODPOVĚDI



1. Podkapitola 7.1, str. 144-145.
2. Podkapitola 7.1, str. 145.
3. Ano
4. Podkapitola 7.1, str. 146.

²⁰ Repozitář dat. Centre for research data – University of technology Eindhoven [online]. 2017 [cit. 2017-11-12]. Dostupné z: <http://data.4tu.nl/repository/>

8 PŘÍPADOVÁ STUDIE PROM



RYCHLÝ NÁHLED KAPITOLY

V této kapitole dojde k prohloubení zkušeností studentů s praktickou aplikací process miningových metod na skutečných datech. Dále představíme použitelné formáty záznamů v softwaru ProM. Věnovat se budeme zejména objevování procesů, nicméně najde se prostor i pro zlepšování procesů.



CÍLE KAPITOLY

Po prostudování této kapitoly budete umět:

- Provádět základní process miningové analýzy v prostředí ProM.
 - Orientovat se v nástroji ProM.
 - Rozeznat základní formáty dat pro process miningovou analýzu v prostředí ProM.
-



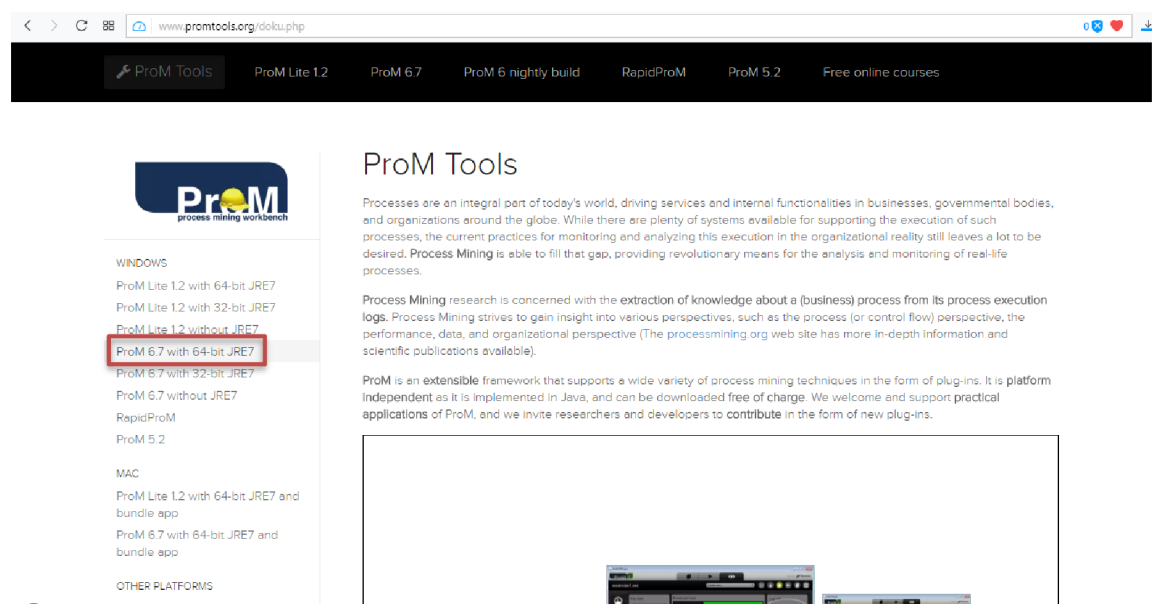
KLÍČOVÁ SLOVA KAPITOLY

Process mining, ProM, případová studie, formáty dat.

Podobně jako v sedmé kapitole, ještě než se pustíme do praktických analýz v ProMu, nejprve si jej nainstalujeme. Poté si představíme prostředí ProMu, se kterým se seznámíme v rámci jeho praktické aplikace. Zatímco v Discu jsme si představili v podstatě všechny klíčové funkce, i když i tam jsme nebyly schopni ukázat vše, v ProMu si toho budeme schopni ukázat ještě méně, jelikož se jedná o akademický a ne komerční software, což znamená, že není kladen takový důraz na komfort uživatele, ale na druhou stranu je ProM nabitý mnoha i těmi nejaktuálnějšími technikami ve formě různých pluginů. Další výhodou je distribuovaný pod *GNU Public License (GPL)* a je tedy osvobozen od jakýchkoliv poplatků.

8.1 Instalace softwaru ProM

Instalace ProM je opět poměrně přímočará podobně, jako tomu bylo u softwaru Disco. Instalační soubor lze nalézt na příslušných stránkách softwaru ProM²¹. Na těchto webových stránkách jsou dostupné dvě verze ProM a ProM lite. Rozdíl mezi těmito verzemi je ProM obsahuje i ty nejnovější a ještě ne vždy úplně stabilní pluginy, zatímco ProM lite v sobě skrývá méně funkcí, ale obsahuje jen otestované pluginy. Každá z těchto verzí je dále ještě dostupná s anebo bez JRE7 (*Java Runtime Environment 7*). Jelikož ProM je vyvíjen v programovacím jazyce Java. Doporučujeme zvolit verzi, ve které je obsaženo „with JRE7“, jelikož v případě, že JRE v počítači nainstalováno nemáte, tak vám jej tento balík rovnou nainstaluje, a v případě že JRE nainstalováno máte, tak se nic nestane.

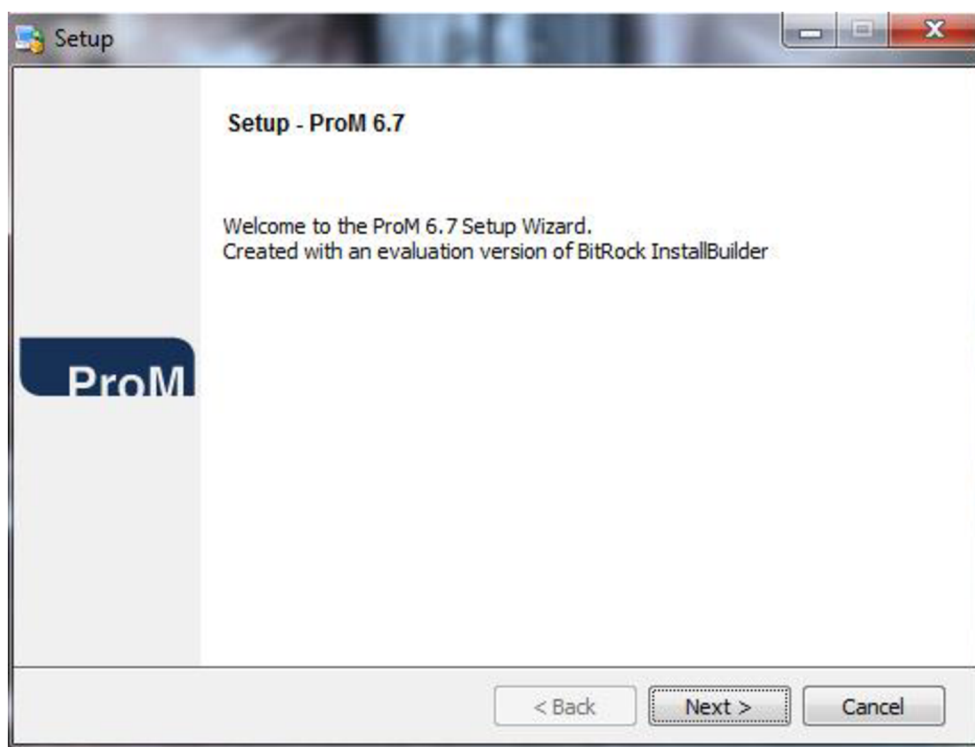


Obrázek 8-1: Instalační soubor ProM

Zdroj: vlastní zpracování

Pokud si nejste jisti, jestli máte 32bitový operační systém nebo 64bitový, stáhněte raději *ProM 6.7 with 32-bit JRE7*, popřípadě to můžete zjistit ve Windows 7 i Windows 10 v *Ovládací panel* -> *Systém a zabezpečení* -> *Systém* kolonka „typ systému“. My tedy zvolíme verzi *ProM 6.7 with 64-bit JRE7* a stáhneme instalační *.exe* soubor, který dvojklikem otevřeme.

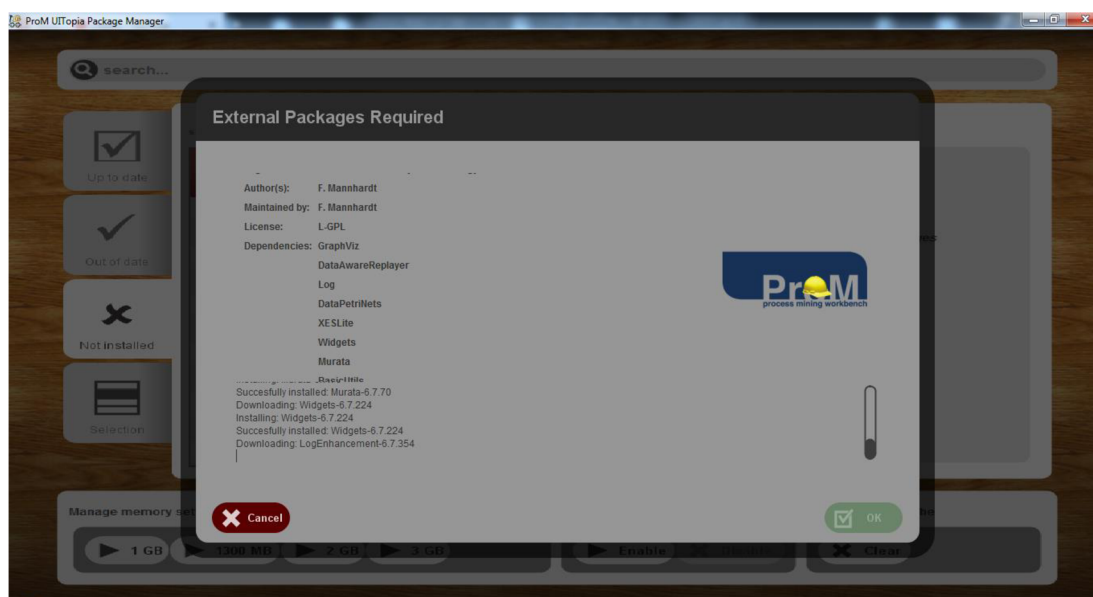
²¹ ProM. *ProM documentation* [online]. 2017 [cit. 2017-11-22]. Dostupné z: <http://www.promtools.org/doku.php>



Obrázek 8-2: Instalace ProM 6.7

Zdroj: vlastní zpracování

Objeví se klasické instalační okno, u kterého kliknete na políčko „Next“, poté znovu „Next“ – případně můžete změnit adresář, kde má být ProM nainstalován, a poté ještě jednou „Next“. V tuto chvíli započne instalace ProM. Ve chvíli, kdy instalace skončí, objeví se okno podobné tomu na obrázku 8-2 a však místo „Next“ klikneme na „Finish“.



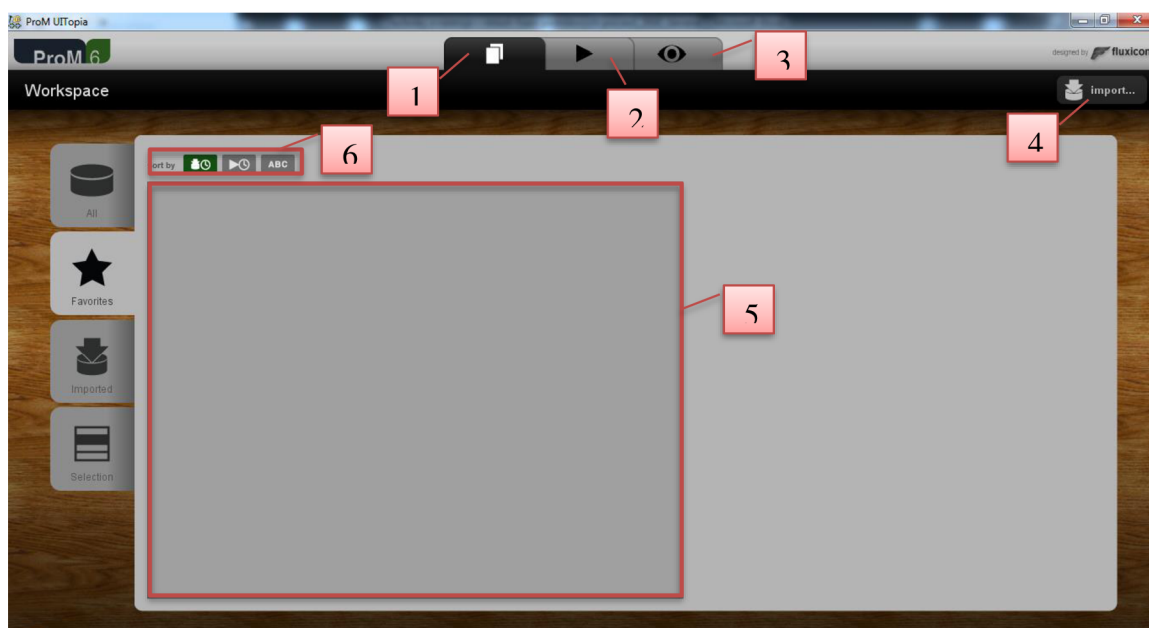
Obrázek 8-3: Prvotní instalace pluginů

Zdroj: vlastní zpracování

Ve chvíli, kdy ProM poprvé otevřeme, objeví se nám obrazovka jako na obrázku 8-3. To se do ProM začaly automaticky stahovat různé pluginy jako alfa-algorithmus, inductive miner, fuzzy miner a další, jistě si na ně z 6. kapitoly vzpomínáte. Vyčkejme tedy až se tento proces ukončí, v závislosti na výkonu Vašeho počítače a rychlosti internetu to může trvat i několik minut.

8.2 Analýza procesů v ProM

ProM máme nainstalován, pojďme se tedy podívat, jak s jeho pomocí můžeme analyzovat procesy. Úvodní pracovní plochu v ProM můžeme vidět na obrázku 8-4 níže.



Obrázek 8-4: Pracovní plocha ProM

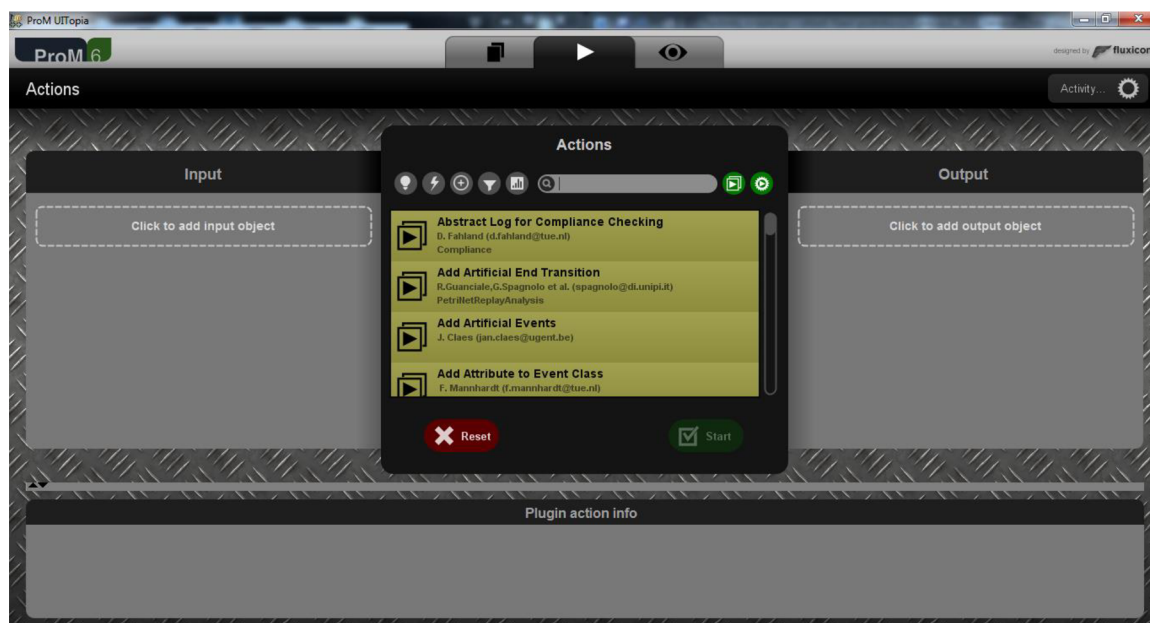
Zdroj: vlastní zpracování

Na obrázku 8-4 popisek 1 je označena pracovní plocha. Pracovní plocha zobrazuje všechny zdroje, které:

- byly do ProM importovány, popisek 4,
- nebo jsou výsledky prováděných akcí na jiných zdrojích.

Dále můžeme na obrazovce vidět čtyři různá zobrazení zdrojů označená „All“, „Favorites“, „Imported“ a „Selection“. V našem případě máme zvoleno zobrazení „Favorites“, nicméně jelikož jsme ještě žádná data neimportovali ani neprováděli žádné akce, je naše zobrazení zdrojů prázdné (obrázek 8-4 popisek 5). Zobrazení „All“ zobrazí všechny zdroje, zobrazení „Favorites“ zobrazí nejpoužívanější, zobrazení „Imported“ zobrazí importované zdroje. Zobrazení „Selection“ umožňuje zobrazit vybrané zdroje. Dále lze zdroje v da-

ném zobrazení členit s pomocí tři tlačítek, viz popisek 6. Tlačítko vlevo člení zdroje v závislosti na době vzniku, prostřední tlačítko dle toho, kdy byly naposledy užity a pravé tlačítko seřadí zdroje dle abecedy.



Obrázek 8-5: Akční plocha ProM

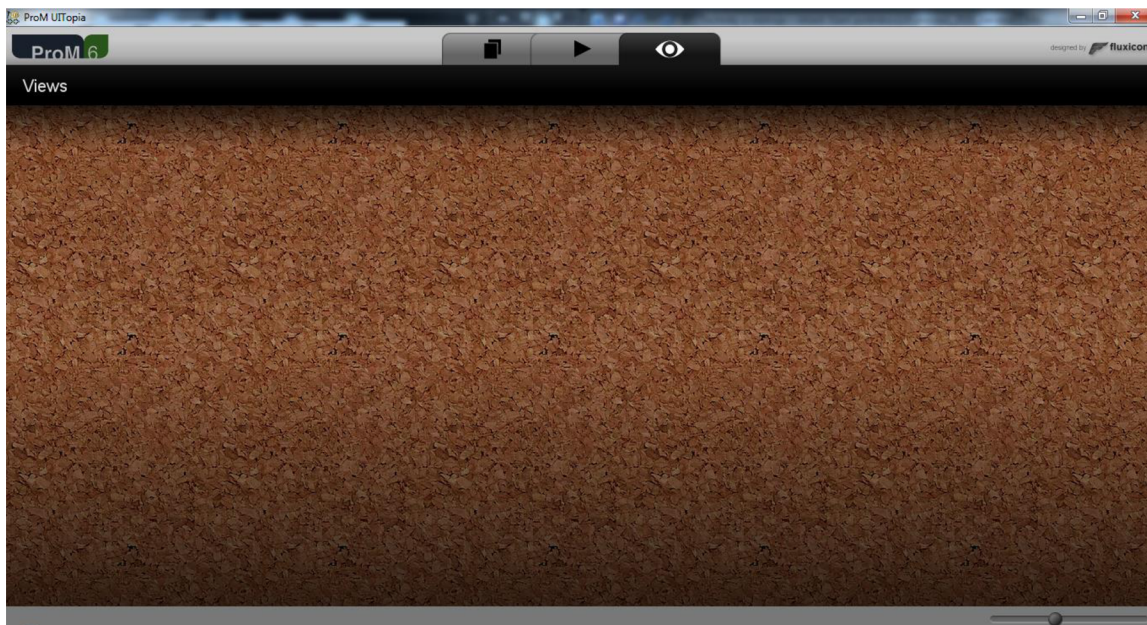
Zdroj: vlastní zpracování

Klikneme-li na tlačítko na obrázku 8-4 popisek 2, zobrazí se nám akční plocha, viz obrázek 8-5. Akční plocha zobrazí všechny akce („*Actions*“), které mohou být provedeny na zvolených vstupech („*Input*“), a které vyústí v jisté výstupy („*Output*“) v závislosti na zvolených kritériích.

Klikneme-li na tlačítko na obrázku 8-4 popisek 3, zobrazí se nám plocha zobrazení, viz obrázek 8-6. Na ploše zobrazení se zobrazí zdroje, či zdroje po provedení vybraných akcí atp. Jestli je Vám ProM stále cizí, nevádí, všechno si ukážeme na skutečných datech.

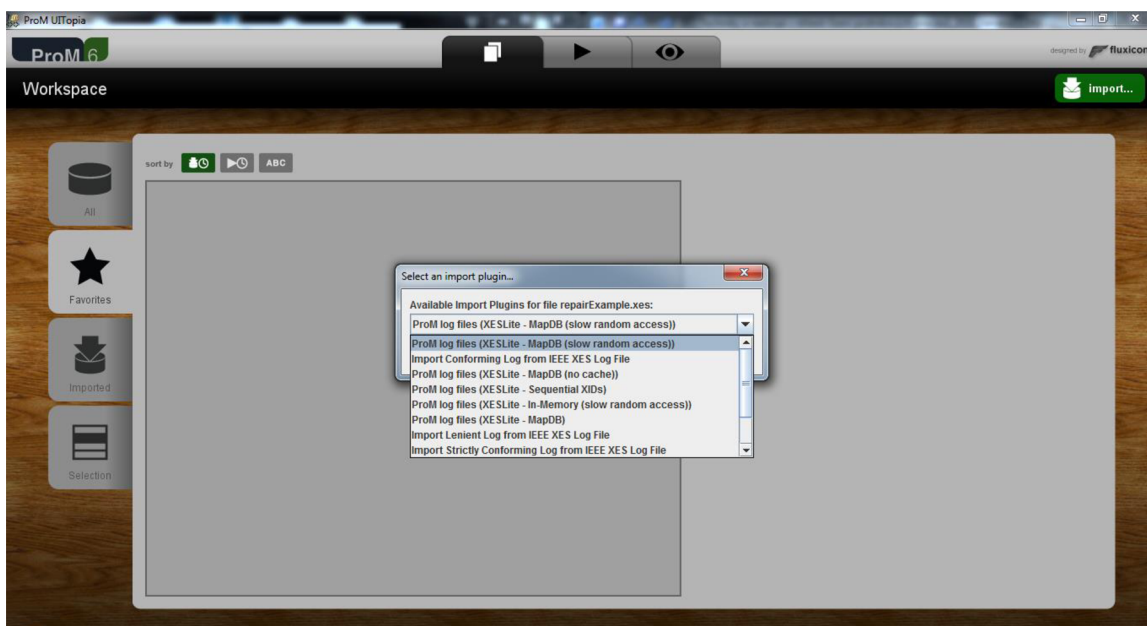
Proces, kterému se budeme věnovat, je proces opravy telefonů ve firmě, která se zabývá jejich prodejem a servisem. Firma opravuje tři typy telefonů (T1, T2 a T3). Proces začíná registrací opravovaného zařízení zasláného firmě zákazníkem. Po registraci je telefon k opravě zařazen do jedné z 10 kategorií dle závady na oddělení detekce problémů (DP). Poté je odeslán na oddělení oprav (R) a zákazníkovi je poslán email s informacemi o závadě. Oddělení oprav má dva týmy. První tým se stará o opravu jednodušších vad a druhý tým se stará o opravu komplexních vad. Nicméně některé kategorie mohou být opraveny oběma týmy. Jakmile jsou na oddělení oprav hotovi, je telefon odeslán na oddělení kvality (QA), kde je telefon analyzován, zdali byl telefon opravdu opraven. V případě, že zaměstnanec zjistí, že telefon nebyl řádně opraven, je opět zaslán na oddělení oprav (R). V případě, že je telefon opraven, záznam je archivován a opravený přístroj je zaslán zákazníkovi. Firma má zásadu, že v důsledku časové úspory, je každý telefon opravován jen do určitého

počtu pokusů. V případě, že se telefon opravit nepovede, je zákazníkovi zaslán zcela nový telefon.



Obrázek 8-6: Plocha zobrazení ProM

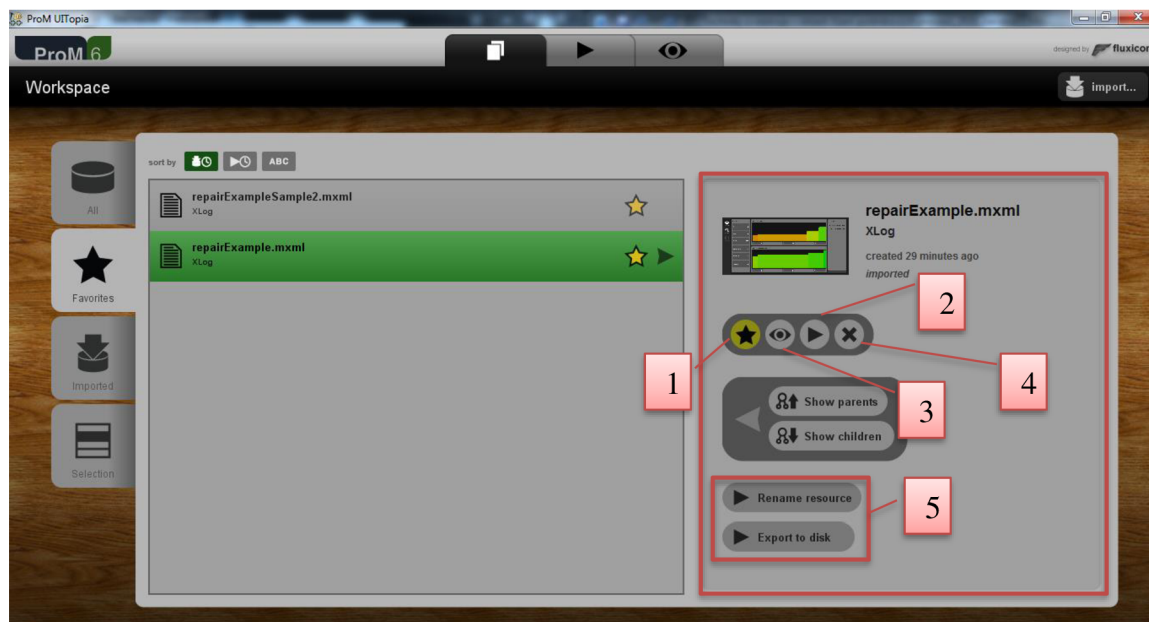
Zdroj: vlastní zpracování



Obrázek 8-7: Import dat

Zdroj: vlastní zpracování

Klikneme tedy na políčko „Import“ na pracovní ploše (obrázek 8-4 popisek 4) a najdeme zdroj našich dat, v našem případě se jedná o soubor typu *.xes* a zmáčkneme políčko „Open“. V seznamu (viz obrázek 8-7) vybereme hned první možnost pro volbu příslušného pluginu a volbu potvrdíme stisknutím políčka „Ok“.



Obrázek 8-8: Pracovní plocha s importovanými daty

Zdroj: vlastní zpracování

Do ProM jsme importovali soubory s názvy „*repairExample*“ a „*repairExampleSample2*“. Stisknutím políčka s popiskem 1 na obrázku 8-8 přidáme zdroj mezi oblíbené. Stisknutím políčka s popiskem 2 se dostaneme na akční plochu podobně jako na obrázku 8-4. Stisknutím políčka s popiskem 3 na obrázku 8-8 se dostaneme na plochu zobrazení opět podobně jako na obrázku 8-4. Stisknutím políčka s popiskem 4 lze daný datový soubor odstranit. Kliknutím na „*Rename resource*“ lze změnit název souboru a kliknutím na „*Export to disk*“ je možno data exportovat na disk, například po jejich úpravě (viz obrázek 8-8 popisek 5).

Po stisknutí políčka s popiskem 3 ať už se jedná o obrázek 8-4 nebo 8-8 si můžeme zobrazit statistiku o použitých datech. Jak můžeme vidět na obrázku 8-9, záznam „*repairExample*“ obsahuje 1 proces („*Process*“), 1104 případů („*Cases*“) a 11855 událostí („*Events*“). Těchto 11855 událostí je reprezentováno 12 různými třídami („*Event classes*“). Toto úzce souvisí s pojmem **klasifikátor**, kdy jednotlivé instance událostí mohou být klasifikovány jako natolik podobné, že je v tomto směru můžeme označit za stejné. Například v našem záznamu jsou jednotlivé události reprezentovány příslušnými aktivitami, kterých je v tomto případě 12. To, že jednotlivé aktivity se vyskytují v různých časech či mají různé zdroje napříč jednotlivými případy, nás bude zajímat až u časové a organizační perspektivy, nicméně pro samotnou tvorbu modelu, ať už se jedná o Petriho síť či procesní mapu, je tato klasifikace událostí užitečná, kdy vlastně upouštíme od zdrojů a

časových značek. Dále můžeme vidět, že v záznamu máme dva typy událostí a sice „start“ a „complete“ neboli začátek a ukončení události (viz obrázek 8-10). Toto úzce souvisí s dalším pojmem, který jsme v kapitole 6 nepředstavili, a sice **životní cyklus transakce** (*Transactional Life-Cycle*). Představme si, že jednotlivé události jako aktivity. V takovém případě často nejsou události reprezentovány jediným bodem v čase, nýbrž časovým intervalem, jelikož vykonání dané aktivity může nějaký čas trvat a u dané aktivity tak registrujeme dvě události, a sice její začátek a konec (viz obrázek 8-10).



Obrázek 8-9: Zobrazení statistiky dat

Zdroj: vlastní zpracování

The screenshot shows the 'Log Summary' view in ProM 6. It displays a table of event classes defined by the MXML Legacy Classifier. The table has three columns: Class, Occurrences (absolute), and Occurrences (relative). There are 12 classes listed. A 'Start events' section is visible at the bottom of the table.

Class	Occurrences (absolute)	Occurrences (relative)
Test Repair+complete	1508	12,72%
Test Repair+start	1508	12,72%
Register+complete	1104	9,313%
Analyze Defect+complete	1104	9,313%
Analyze Defect+start	1104	9,313%
Inform User+complete	1102	9,296%
Archive Repair+complete	1000	8,435%
Repair (Simple)+complete	785	6,622%
Repair (Simple)+start	785	6,622%
Repair (Complex)+start	725	6,116%
Repair (Complex)+complete	724	6,107%
Restart Repair+complete	406	3,425%

Obrázek 8-10: Typy událostí „start“ a „complete“

Zdroj: vlastní zpracování

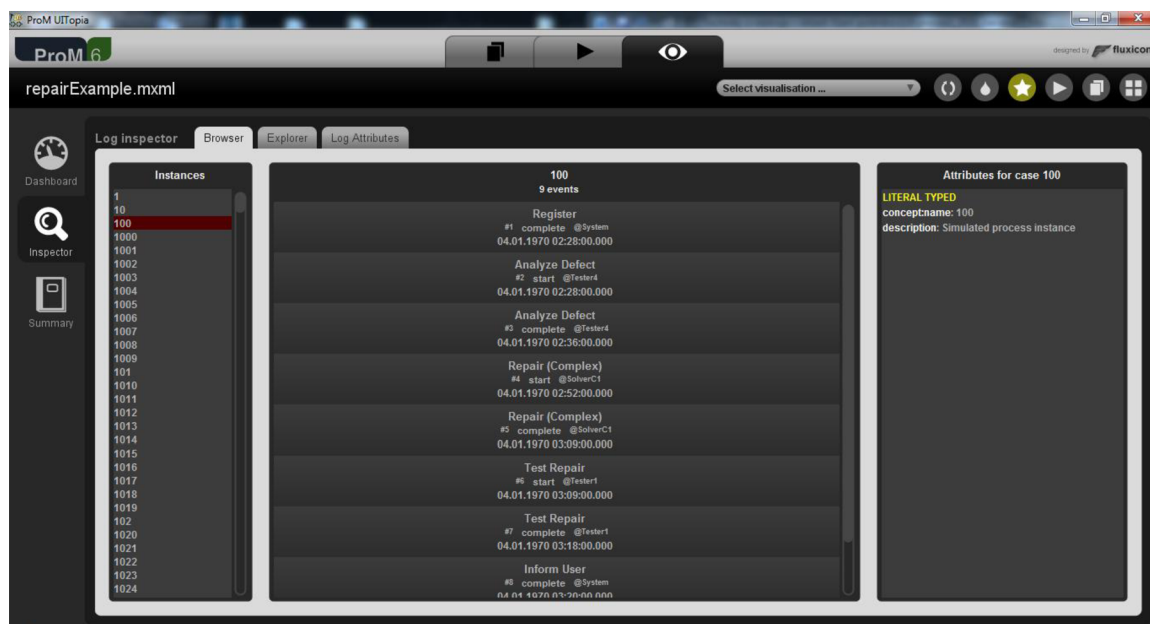
Dále můžeme na obrázku 8-9 můžeme vidět další statistiky jako minimální počet událostí na případ, což je rovno 4 („*Event per case*“), průměrný počet událostí na případ, což je 11 a maximální počet událostí na případ, což je 24. Pod diagramem počet událostí na případ vidíme diagram počtu tříd událostí na případ („*Event classes per case*“), kde minimum je 4, maximum je 12 a průměr je 9. Na základě obou průměrů a maxim můžeme v modelu očekávat smyčku.



ÚKOL K ZAMYŠLENÍ

Objasněte, proč nám průměry události na případ a třída událostí na případ mohou naznačovat vyskytující se smyčku v modelu. (Nápověda: soustřeďte se na to, jak vypadá stopa v modelu se smyčkou a podstatu toho, proč se oba průměry liší.)

Vysvětlete, proč nemůže být hodnota maximální počet tříd událostí na případ v tomto případě vyšší než 12.

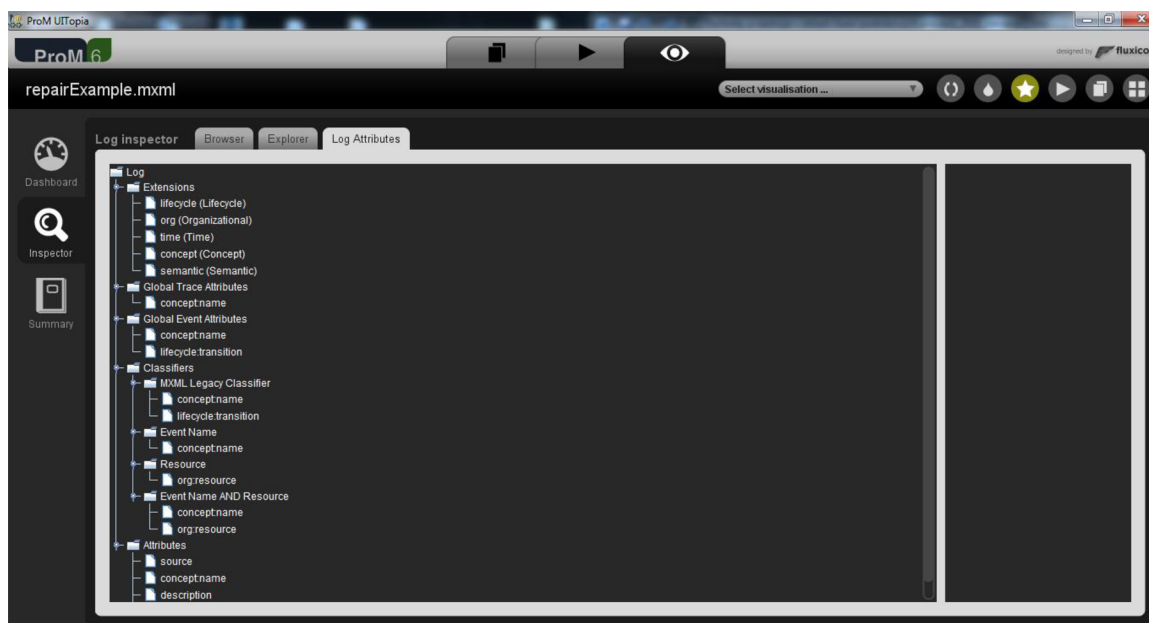


Obrázek 8-11: Plocha zobrazení položka „Inspector“

Zdroj: vlastní zpracování

Přejdeme-li z položky „*Dashboard*“ na položku „*Inspector*“ na ploše zobrazení, můžeme si detailně a přehledně zobrazit jednotlivé případy vyskytující se v záznamu prostřednictvím políčka „*Browser*“ (viz obrázek 8-11). Klikneme-li na vedlejší políčko „*Explorer*“, zobrazí se nám jednotlivé případy graficky. Klikneme-li na další políčko „*Log attributes*“,

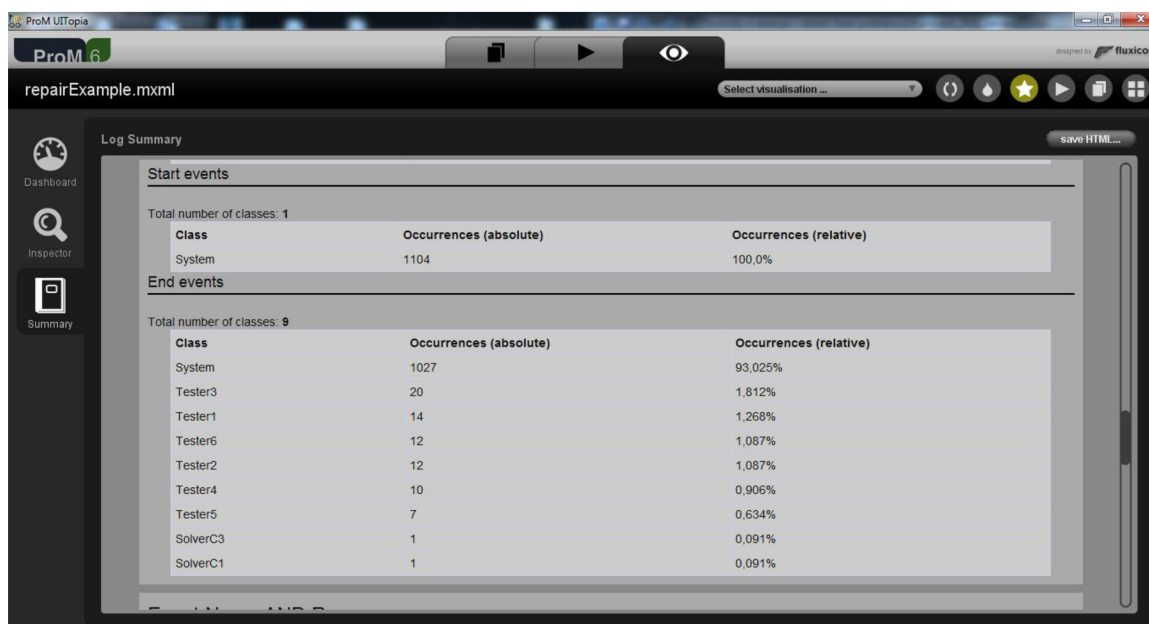
zobrazí se nám obdoba obrázku 7-1, tedy jakási struktura dat respektive záznamu (viz obrázek 8-12).



Obrázek 8-12: Struktura dat respektive záznamu

Zdroj: vlastní zpracování

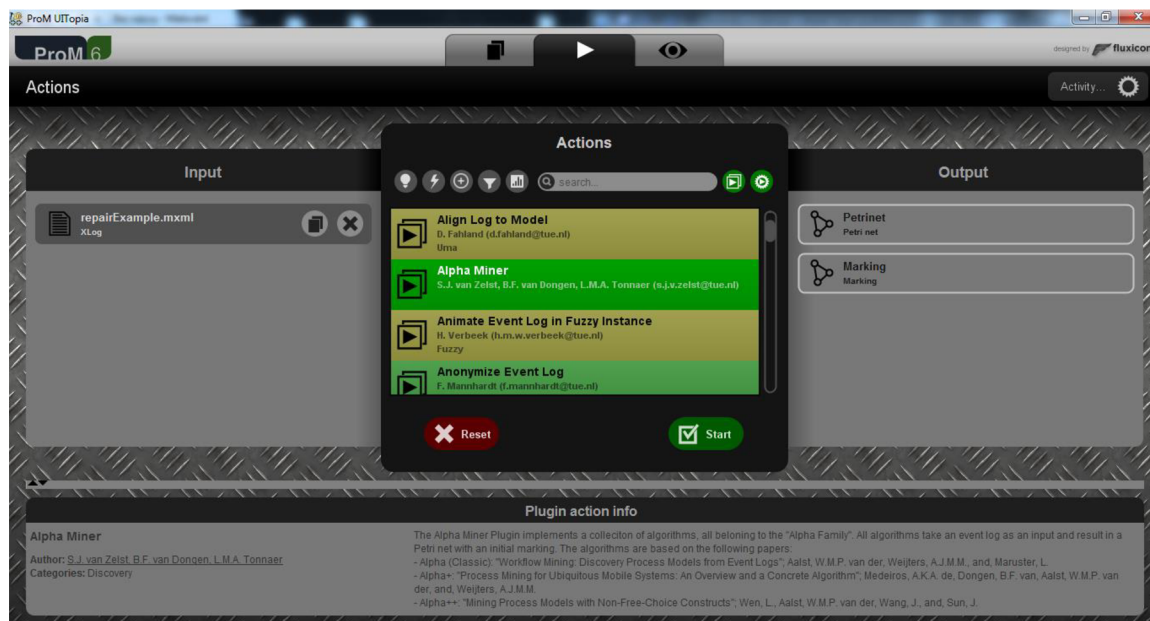
Poslední položka na ploše zobrazení je „Summary“, kde můžeme najít souhrnné informace o záznamu, jako například události vyskytující se na prvním či posledním místě ve stopách napříč záznamem, pravděpodobnost jejich výskytu atp. (viz obrázek 8-13).



Obrázek 8-13: Shrnující informace o záznamu

Zdroj: vlastní zpracování

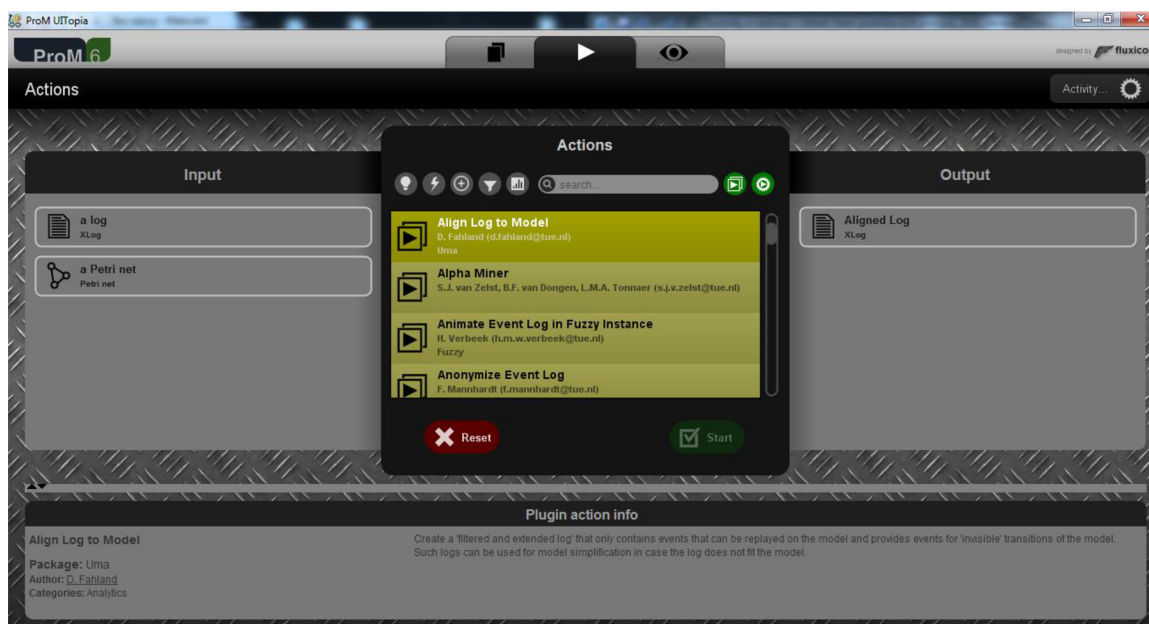
Podobně si můžeme předem analyzovat i záznam „*repairExampleSample2*“ či jiná data, která jsme si do ProM importovali. Nicméně nyní už pojďme k samotné analýze. Zobrazíme si tedy akční plochu (viz obrázek 8-4 nebo 8-8 popisek 2). Nacházíme-li se v akční ploše, klikneme na položku „*Click to add input object*“ v levém sloupci „*Input*“ a zvolíme soubor „*repairExample*“. V případě, že bychom zvolili postup z obrázku 8-8, již bychom nemuseli znovu volit data, jelikož zvolený datový soubor se nám automaticky vloží do sloupce „*Input*“.



Obrázek 8-14: Alfa miner v ProM

Zdroj: vlastní zpracování

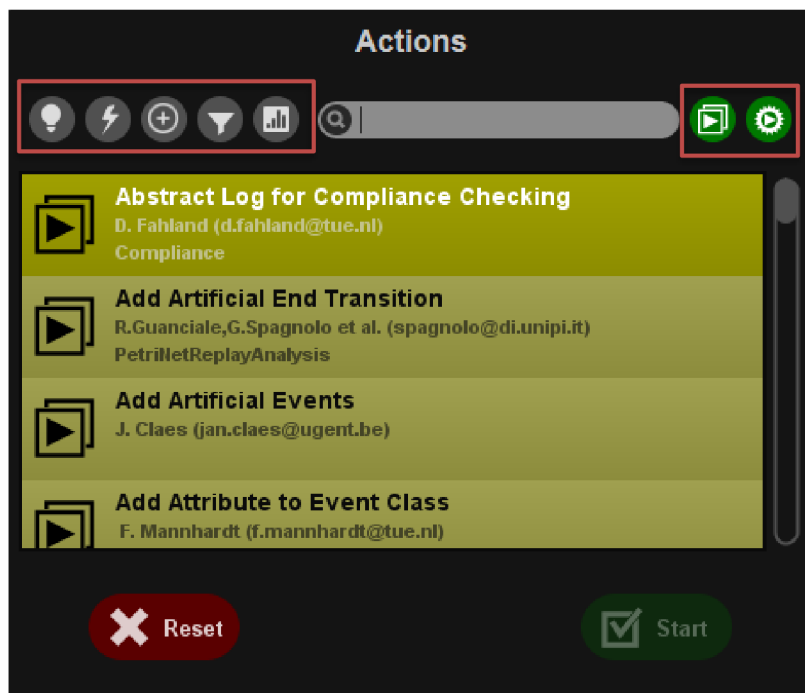
Ať již zvolíme postup dle obrázku 8-4 nebo 8-8, dostaneme se k akční ploše na obrázku 8-14 v podstatě stejně. Ve chvíli kdy jsme vybrali data („*Input*“), zvolíme na akční ploše v prostředním sloupci „*Actions*“, buď prostřednictvím vyhledávacího proužku („*Search*“) anebo posunem myši, najdeme plugin „*Alpha miner*“. Klikneme-li na „*Alpha miner*“ zobrazí se nám ve spodní liště informace o daném pluginu („*Plugin action info*“). V pravém sloupci vidíme výstup („*Output*“), který dostaneme po aplikaci dané akce. V našem případě Petriho síť („*Petri net*“) a značení („*Marking*“). V seznamu akcí, jsou zeleně podbarveny akce, které můžeme s daným vstupem provádět, žlutě jsou akce ke kterým nám chybí vstupní data. V případě, že se objevíme na akční ploše tak jako tomu bylo na obrázku 8-5, kdy ještě nemáme zvoleny vstupní data a nejsme si jisti, jaká data budeme pro danou akci potřebovat, stačí ji najít v prostředním sloupci. Poté co klikneme na příslušný plugin se v levém sloupci „*Input*“ vypíší data, kterou pro tuto akci potřebujeme, a zbytek zůstane stejný, ve spodní liště se objeví informace o daném pluginu a v pravém sloupci se objeví výstup dané akce. Toto můžeme vidět na obrázku 8-15, kde pro akci „*Align log to Model*“ je potřebný záznam a Petriho síť jako vstup, a výstupem je záznam, který je na dané Petriho síti přehratelný.



Obrázek 8-15: Zjištění požadavků na vstupní data pro jednotlivé akce

Zdroj: vlastní zpracování

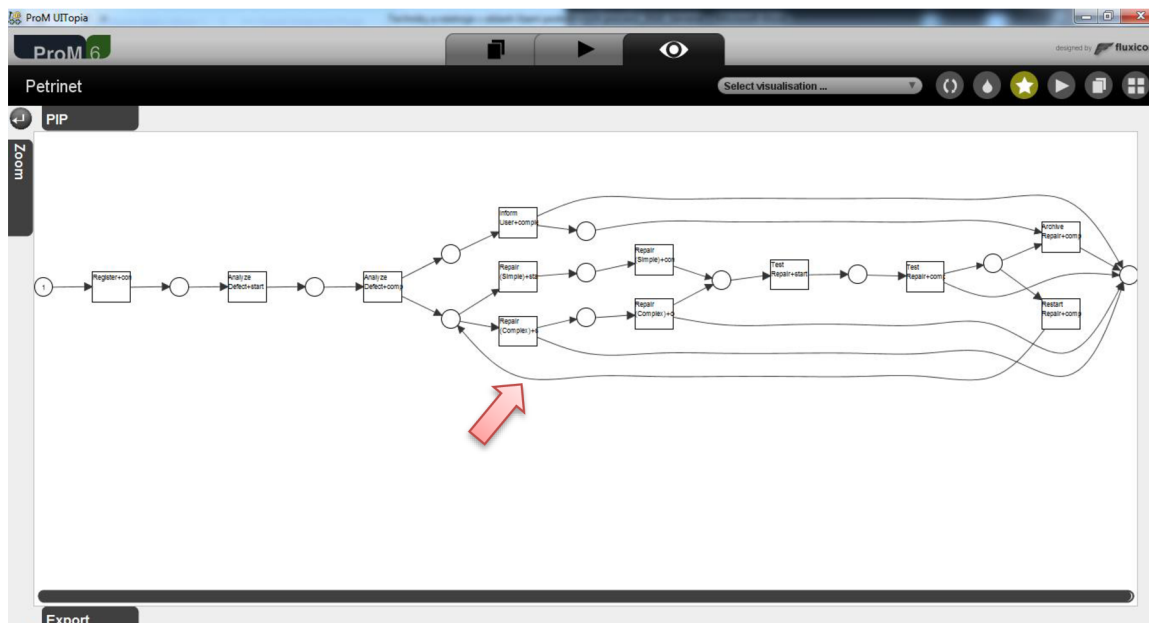
Jistě jste si po obou stranách vyhledávacího proužku všimli celkem sedmi ikon, pěti vlevo a dvou vpravo (viz obrázek 8-16 červené obdélníky). Nejprve se zaměříme na skupinku dvou ikon: zleva je filtr tzv. interaktivních akcí, což jsou akce, u nichž je potřeba jistého nastavení uživatelem, zatímco vpravo je filtr neinteraktivních akcí, u kterých již nemusí uživatel nic nastavovat. U druhé skupinky, tedy u skupinky pěti ikon se budeme věnovat jen prvním třem zleva: první ikonka je filtr objevovacích technik, druhá ikonka je filtr technik pro zjišťování shodnosti a třetí ikonka je filtr technik sloužících pro zlepšování procesů.



Obrázek 8-16: Filtry pro hledání vhodných akcí na akční ploše

Zdroj: vlastní zpracování

Nyní tedy již pojďme aplikovat „Alpha miner“ na náš „*reparExample*“ záznam. Vrátime se tedy na obrazovku 8-14 a klikneme na políčko „Start“. Na další obrazovce ponecháme „Event classifier“ na hodnotě „MXML legacy classifier“, a u políčka „Version“ zvolíme hodnotu „Alpha++“ a klikneme na políčko „Finish“.



Obrázek 8-17: Aplikace Alpha++ algoritmu v pluginu Alpha miner

Zdroj: vlastní zpracování

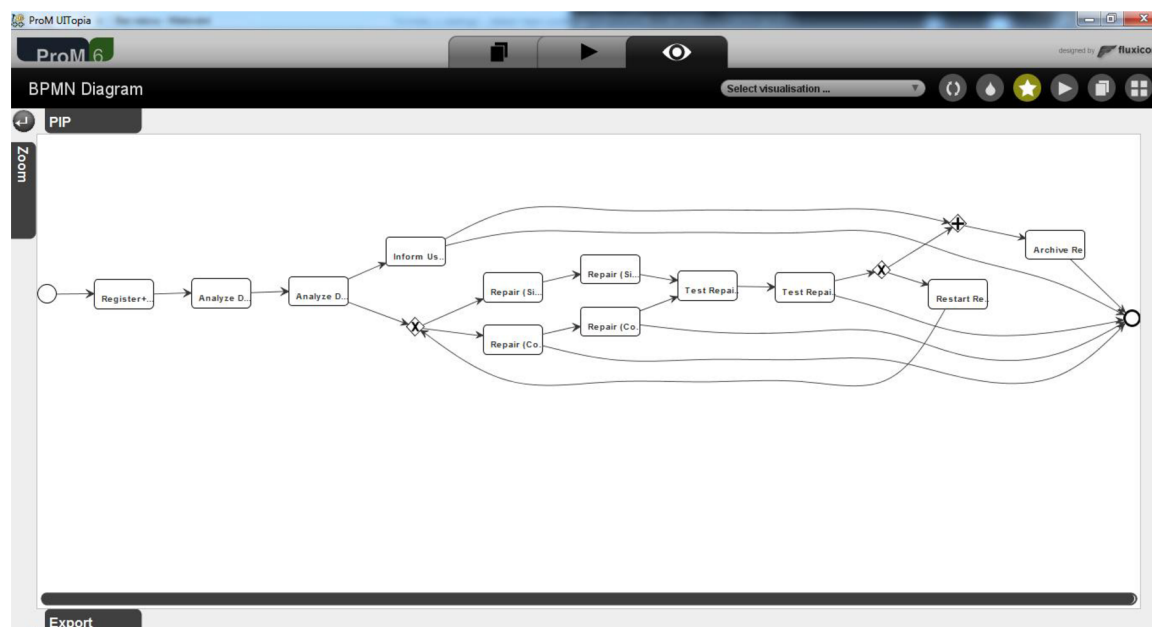
Alpha++ algoritmus objevil z dat, která jsou obsažena v souboru „repairExample“ model v podobě Petriho sítě, který lze vidět na obrázku 8-17 výše. Červenou šipkou je označena hrana smyčky, kterou jsme předpověděli na základě statistik z obrázku 8-9. Můžeme vidět, že objevený proces je dobře strukturovaný a potýkáme se tedy s lasagna procesem.

SAMOSTATNÝ ÚKOL



Jistě si z kapitoly 6.3.2 vzpomínáte na definici spolehlivé sítě. Ověřte, zdali je Petriho síť na obrázku 8-17 spolehlivá či nikoliv. (Nápověda: ověřte, že síť splňuje uvedené čtyři podmínky či nikoliv). Svě zjištění si můžete ověřit s pomocí akce „Analyze with Woflan“ (V případě, že uvidíte zprávu psanou červeně „The net is not a sound workflow net“ nejedná se o spolehlivou síť. Zbýlý text řeší diagnostiku respektive otázku proč, ten už nás ale v tuto chvíli nemusí zajímat.)

Ve třetí kapitole jsme se věnovali BPMN notaci. Pojdme si tedy s pomocí ProM převést objevenou Petriho síť na BPMN diagram. Což se provede jednoduše. Přejdeme na akční plochu a ve sloupci „Actions“ vyhledáme akci „Convert Petri net to BPMN diagram“. Jako vstup přidáme před chvílí vygenerovanou Petriho síť a stiskneme políčko „Start“. Výsledek konverze můžete vidět na obrázku 8-18.

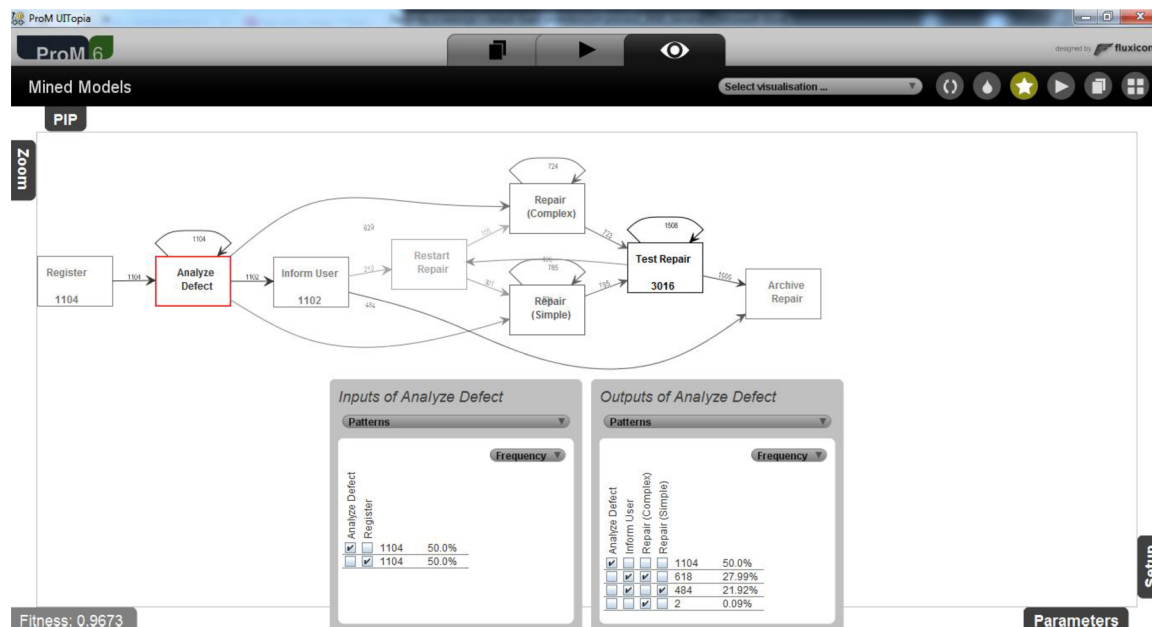


Obrázek 8-18: Převedení Petriho sítě z obrázku 8-17 na BPMN diagram

Zdroj: vlastní zpracování

Přestože jsme se v 6. kapitole intenzivně věnovali pouze alfa algoritmu, pojdme využít toho, že ProM nám nabízí široké možnosti a vyzkoušejme si i další objevovací techniky jako například heuristic miner či inductive miner. Opět se tedy přesuneme na akční plochu

a na náš soubor „repairExample“ aplikujeme heuristic miner, tedy konkrétně vyhledáme „Mine for a Heuristic Net using Heuristic miner“, v mezikroku opět ponecháme přednastavené hodnoty. Dostaneme „heuristic net“, která připomíná procesní mapu v Discu s příslušnými frekvencemi a klikneme-li na jednotlivé aktivity, můžeme si zobrazit statistiku o vstupních a výstupních aktivitách dané aktivity (na obrázku 8-19 máme zobrazenou statistiku pro aktivitu „Analyze defect“).

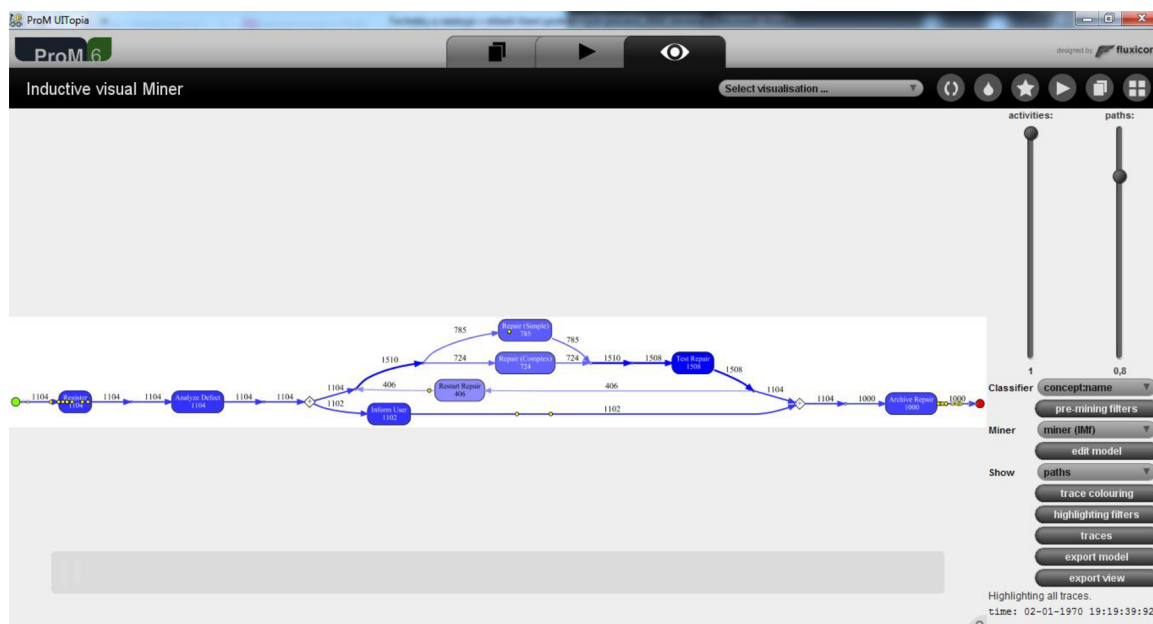


Obrázek 8-19: Heuristic net vygenerovaná prostřednictvím heuristic miner

Zdroj: vlastní zpracování

Už na první pohled je vidět, že model objevený prostřednictvím *heuristic mineru* se liší od toho produkovaného *alpha minerem*. Zájemci si mohou heuristic net konvertovat na Petriho síť a přesvědčit se sami prostřednictvím akce „Convert Heuristics net into Petri net“. Opět se tedy přesuňme na akční plochu a na náš soubor „repairExample“ aplikujeme ještě třetí techniku inductive miner, respektive vyhledáme „Mine process tree with inductive miner“ a v mezikroku opět ponecháme přednastavené hodnoty a obdobně i pro následující model.

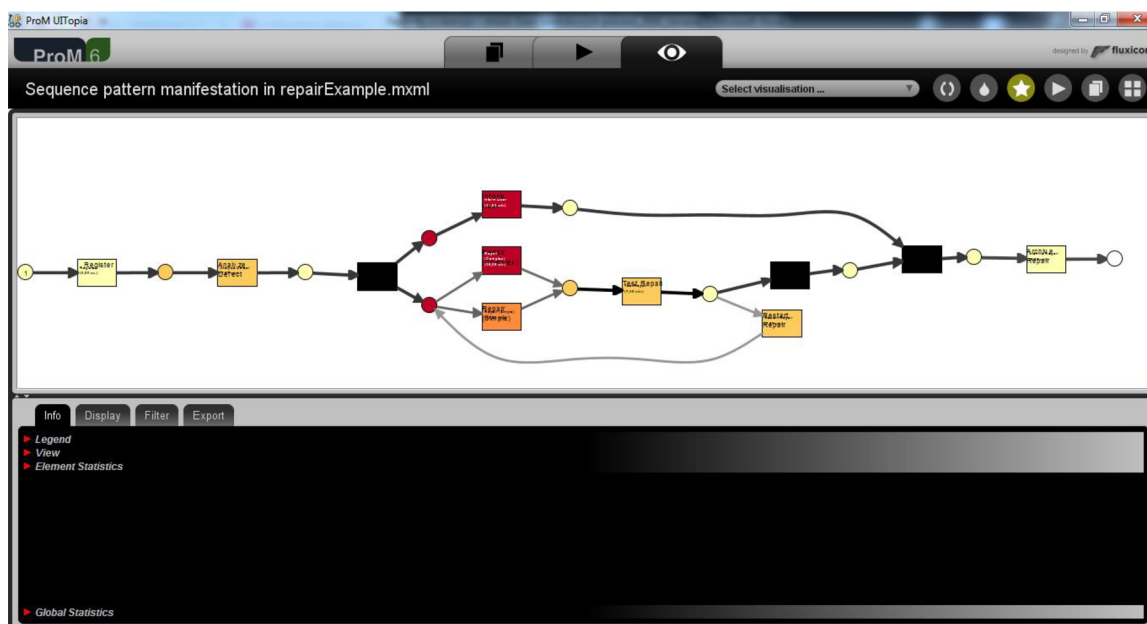
Opět můžeme vidět, že inductive miner objevil jiný model než alpha miner. Černým čtverečkům síti říkáme tzv. tiché přechody, jedná se o přechody, jejichž provedení není možné detekovat. Nicméně při přehrávání záznamu se s nimi pracuje jako s ostatními přechody. Alpha miner je v praktických aplikacích jen těžko použitelný, heuristic miner je na tom co aplikace na reálných datech daleko lépe, nicméně nejlépe je na tom právě inductive miner. Jednou z hlavních výhod je to, že vždy produkuje spolehlivou síť, což si opět můžete ověřit buď ručně anebo s pomocí akce „*Analyze with Woflan*“. V 7. kapitole jsme si u Disco ukázali možnost animace objeveného modelu, podobnou možnost nám poskytuje i akce „*Mine with inductive visual miner*“ (viz obrázek 8-22).



Obrázek 8-22: Vizualizace modelu procesu v ProM

Zdroj: vlastní zpracování

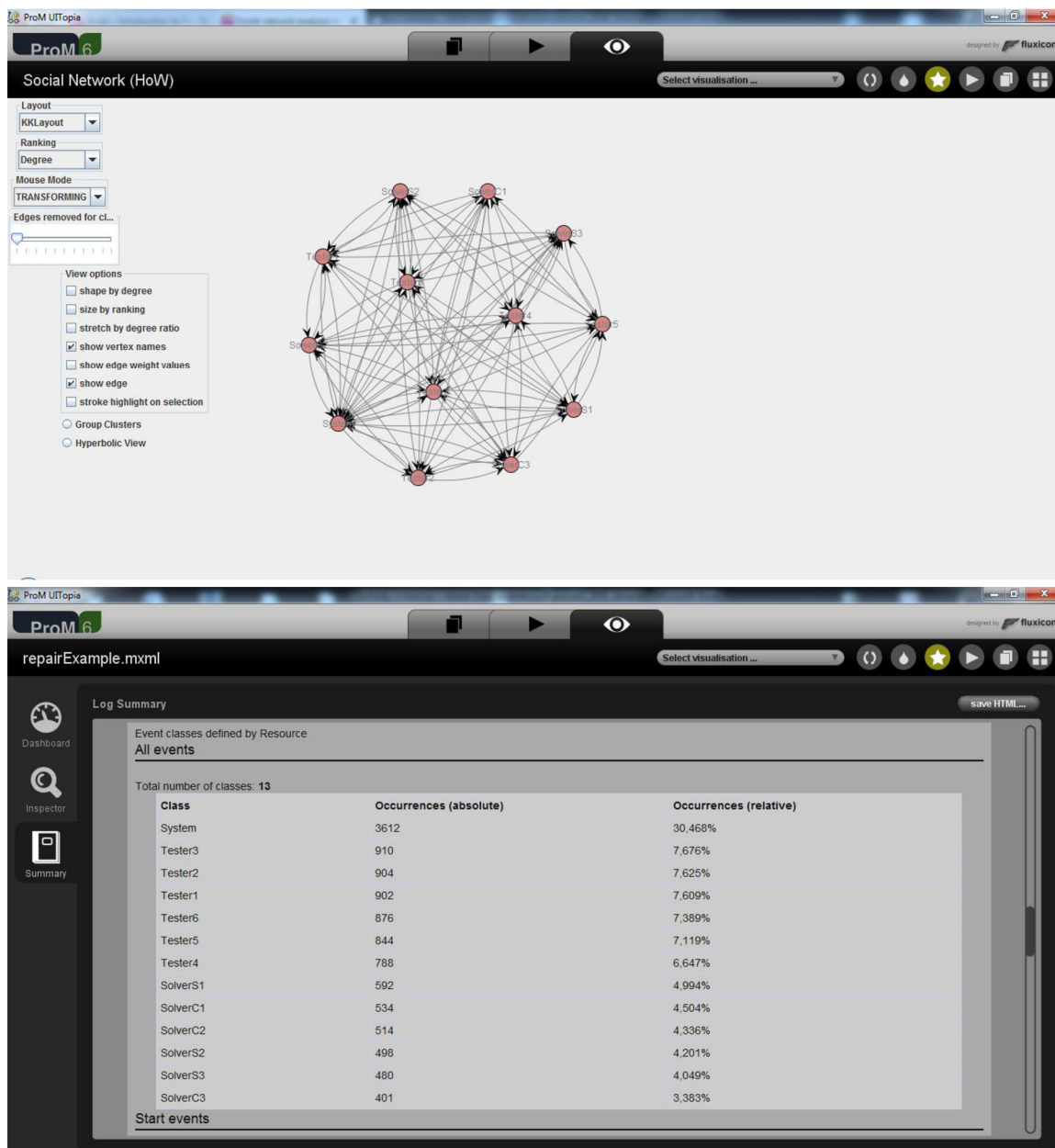
Podobně jako v Disco, i ProM umožňuje analyzovat výkonnostní složku daného modelu. Ať už s pomocí výše zmíněné akce „*Mine with inductive visual miner*“, ale lze využít i jiné akce jako například „*Replay a log on petri net for performance/conformance*“ se vstupními daty v podobě záznamu a Petriho sítě, kterých jsme si zde již vytvořili několik. Na obrázku 8-23 je použita Petriho síť objevená s využitím inductive mineru. Ve spodní liště na obrázku 8-23, je možné sledovat různé statistiky, nicméně hned při pohledu na daný model vidíme potencionální problémy u přechodů zbarvených do červena.



Obrázek 8-23: Analýza výkonnostní složky modelu

Zdroj: vlastní zpracování

Na konci 6. kapitole jsme si kromě časové perspektivy říkali také o perspektivě organizační, a konkrétně jsme si říkali o analýze sociálních sítí prostřednictvím atributu. Sociální síť můžeme vidět na obrázku 8-24. Tuto je možno zobrazit prostřednictvím akce „*Mine for handover-of-work social network*“, jejímž vstupem je analyzovaný záznam „*repairExample*“. ProM nabízí čtyři akce spojené s analýzou sociálních sítí, a jednou z nich výše zmíněné akce předávání práce („*handover-of-work*“), což znamená, že sledujeme předávání práce. Předávání práce je ve smyslu, měli-li bychom stopu $\langle a, b, c \rangle$ a věděli-li bychom, že aktivitu a vykonal *Petr*, aktivitu b *Tomáš* a aktivitu c *Adam*, pak by měli sociální síť se třemi uzly *Petr*, *Tomáš*, *Adam* orientovaná hrana by vedla od Petra k Tomášovi a od Tomáše k Adamovi.

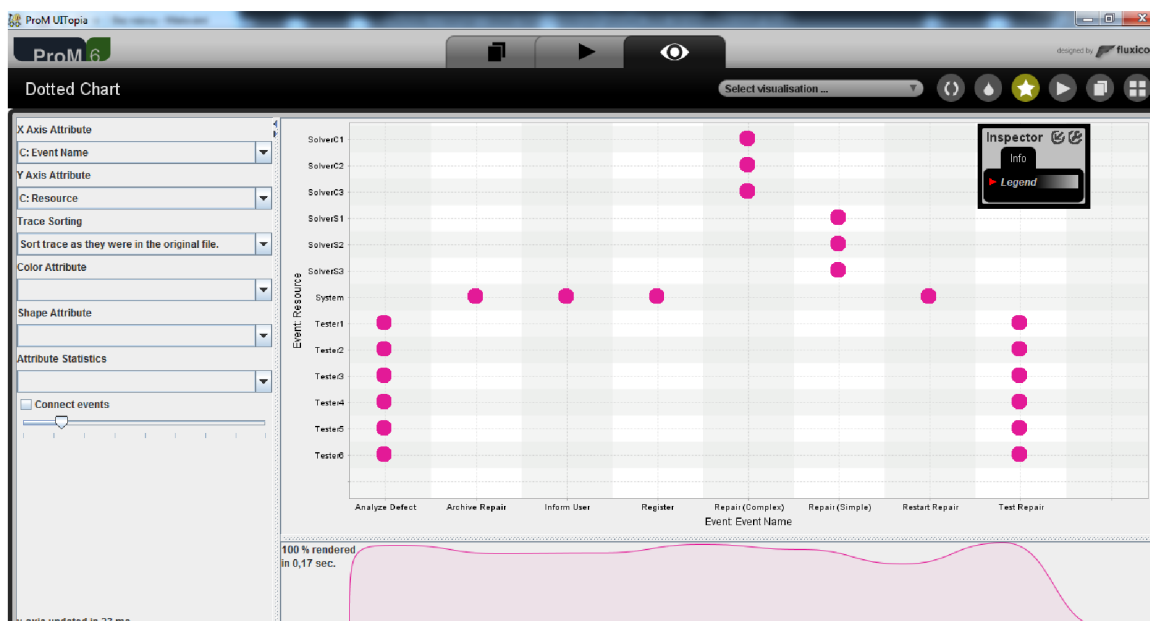


Obrázek 8-24: Analýza sociálních sítí doplněná o statistiku zdrojů vyskytujících se v záznamu „repairExample“

Zdroj: vlastní zpracování

Z obrázku 8-24 je vidět, že práci si předávají vzájemně všichni pracovníci mezi sebou a nenachází se zde žádný, na kterém by celý proces závisel, respektive tedy samozřejmě celý proces závisí na informačním systému, který se stará o třetinu (30 %) všech aktivit, a který daný proces podporuje. Ale co se opravářů týče, tak skupinka „SolverC1“, „SolverC2“ a „SolverC3“ starající se o opravy komplexních vad je v daném procesu zainteresovaná v přibližně stejném počtu událostí, jako skupinka „SolverS1“, „SolverS2“ a „SolverS3“ starající se o nekomplexní vady. Nicméně co je zajímavé, je, že podíváme-li se na obrázek 8-23, tak jedna z podezřelých aktivit je právě „Repair complex“ tedy oprava komplexních vad, což

dává smysl, jelikož opravy komplexních vad budou trvat daleko déle než vad méně komplexních.



Obrázek 8-25: Zobrazení událostí na ose x a příslušných zdrojů na ose y s využitím akce „Dotted chart“

Zdroj: vlastní zpracování

Komplexních vad je dokonce tolik, že „*SolverC1*“, „*SolverC2*“ a „*SolverC3*“, přestože by mohli se k opravám nekomplexních vad ani nedostanou. Z tohoto pohledu by tedy mohlo stát za zvážení přeškolení jednoho či dvou ze skupinky „*SolverS1*“, „*SolverS2*“ a „*SolverS3*“ do skupinky opravářů komplexních vad. Ale to už záleží na zvážení managementu firmy, jestli je toto úzké místo natolik vážné, aby stálo za to investovat prostředky do zaškolování pracovníků. Tímto uzavřeme případovou studii, týkající se představení softwarového produktu ProM. Zájemce samozřejmě opět nabádáme, pro bližší zkoumání ProMu.

OTÁZKY



1. Jaký je rozdíl mezi špagetovým a lasagna procesem?
2. Jaké znáte další objevovací techniky kromě alfa-algoritmu?
3. S čím souvisí pojem životní cyklus transakce?



SHRNUTÍ KAPITOLY

V této kapitole jsme navázali na předcházející kapitoly, kdy jsme si ukázaly další možnosti aplikace process miningových technik a jejich aplikaci tentokrát ale v daleko mocnějším nástroji zvaném ProM, který nabízí daleko více přístupů, avšak na úkor příjemného grafického rozhraní a uživatelského komfortu. Tentokrát jsme se věnovali reklamačnímu procesu firmy na rozdíl od nákupního procesu v kapitole 7. V rámci této studie jsme si také blíže představili prostředí ProM. Nicméně jsme opět nebyly schopni projít všechny možnosti, které ProM nabízí. Zájemce opět nabádáme k detailnějšímu prozkoumání tohoto softwaru.



ODPOVĚDI

1. Podkapitola 8.2, str. 177.
 2. Podkapitola 8.2, od str. 178.
 3. Podkapitola 8.2, str. 171.
-

LITERATURA

- [1] AALST, Will M.P. van der. *Process mining: Discovery, Conformance and Enhancement of Business Processes*. Heidelberg: Springer, 2011. ISBN 978-3642434952.
- [2] AALST, Will M.P. van der. *Process mining: Data science in action*. Heidelberg: Springer, 2016. ISBN 978-3662498507.
- [3] AALST, Willy M.P. van der, Ana K.A. de MEDEIROS a Ton A.J.M.M WEIJTERS. Genetic Process Mining. Applications and Theory of Petri Nets 2005, Lecture Notes in Computer Science, 2005, Berlin: Springer, 48-69.
- [4] AALST, Willy M.P. van der, V. RUBIN, Eric H.M.W. VERBEEK, Boudewijn F. van DONGEN, E. KINDLER a C.W. GÜNTHER. *Process mining: a two-step approach to balance between underfitting and overfitting*. Journal of Software and Systems Modeling, 2008, 9(1), 87–111
- [5] AALST, Will M.P. van der, Ton A.J.M.M WEIJTERS a Laura MARUSTER. *Workflow mining: discovering process models from event logs*. IEEE Transactions on Knowledge and Data Engineering. 2004, 16, 1128-1142.
- [6] AALST, Wil van der a Kees van HEE. *Workflow management: models, methods and systems*. Paperback ed. Cambridge: MIT Press, 2004. ISBN 978-0262720465.
- [7] BASL, Josef a Roman BLAŽÍČEK. *Podnikové informační systémy: podnik v informační společnosti*. 3., aktualiz. a dopl. vyd. Praha: Grada, 2012. Management v informační společnosti. ISBN 978-80-247-4307-3.
- [8] BERGENTHUM, R., J. DESEL, R LORENZ a R. MAUSER. *Process Mining Based on Regions of Languages*. Business Process Management, Lecture Notes in Computer Science, 2007, Berlin: Springer, 375-383.
- [9] DUMAS, Marlon. *Fundamentals of business process management*. Berlin: Springer, 2013. ISBN 978-364-2331-435.
- [10] GÁLA, Libor, Jan POUR a Zuzana ŠEDIVÁ. *Podniková informatika*. 2., přeprac. a aktualiz. vyd. Praha: Grada, 2009. Expert (Grada). ISBN 978-80-247-2615-1.
- [11] GOLFARELLI, M., J. LECHTENBÖRGER, S. RIZZI a G. VOSSSEN. Schema Versioning in Data Warehouses: Enabling Cross-Version Querying via Schema Augmentation. *Data and Knowledge Engineering*, 59(2), 435-459.
- [12] KINDLER, Evžen a Ivan KŘIVÝ. *Simulace a modelování: učební text*. Ostrava: Ostravská univerzita v Ostravě, 2001. Dostupné z: <http://vendulka.zcu.cz/Download/Free/SkriptaKindlerMS.pdf>

- [13] KISLINGEROVÁ, Eva. *Oceňování podniku*. Praha: C.H. Beck, 1999. C.H. Beck pro praxi. ISBN 80-717-9227-6.
- [14] Kolektiv autorů. *Dohoda o chápání pojmu simulace systémů*. Automatizace, 1986, **29**(12), 299–300.
- [15] LEEMANS, S.J.J., D. FAHLAND a Willy M.P. van der AALST. *Discovering Block-Structured Process Models from Event Logs - A Constructive Approach*. Application and Theory of Petri Nets and Concurrency, Lecture Notes in Computer Science, 2013, Berlin: Springer, 311-329.
- [16] LEEMANS, S.J.J., D. FAHLAND a Willy M.P. van der AALST. *Discovering Block-Structured Process Models from Event Logs Containing Infrequent Behaviour*. Business Process Management Workshops, Lecture Notes in Business Information Processing, 2013, Berlin: Springer, 66-78.
- [17] LEEMANS, S.J.J., D. FAHLAND a Willy M.P. van der AALST. *Discovering Block-Structured Process Models from Incomplete Event Logs*. Application and Theory of Petri Nets and Concurrency, Lecture Notes in Computer Science, 2014, Berlin: Springer, 91-110.
- [18] MEDEIROS, Ana K.A. de, Willy M.P. van der AALST a Ton A.J.M.M. WEIJTERS. Workflow Mining: Current Status and Future Directions. On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE, Lecture Notes in Computer Science, 2003, Berlin: Springer, 389-406.
- [19] MEDEIROS, Ana K.A. de, Boudewijn F. van DONGEN, Willy M.P. van der AALST a Ton A.J.M.M. WEIJTERS. Process Mining: Extending the α -algorithm to Mine Short Loops. 2005.
- [20] MEDEIROS, Ana K.A. de, Boudewijn F. van DONGEN, Willy M.P. van der AALST a Ton A.J.M.M. WEIJTERS. Process Mining for Ubiquitous Mobile Systems: An Overview and a Concrete Algorithm. Proceedings of the Second CAiSE Conference on Ubiquitous Mobile Information and Collaboration Systems, 2004, Berlin: Springer, 151-165.
- [21] MEDEIROS, Ana K.A. de, Ton A.J.M.M. WEIJTERS a Willy M.P. van der AALST. Genetic Process Mining: A Basic Approach and Its Challenges. Business Process Management Workshops, Lecture Notes in Computer Science, 2005, Berlin: Springer, 203-215.
- [22] ŘEPA, Václav. *Podnikové procesy: procesní řízení a modelování*. Praha: Grada, 2006. Management v informační společnosti. ISBN 80-247-1281-4.
- [23] WEIJTERS, Ton A.J.M.M., a Joel T.S. RIBEIRO. Flexible Heuristics Miner (FHM). 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), 2011, 310-317.

- [24] WEIJTERS, Ton A.J.M.M a Willy M.P. van der AALST. Rediscovering Workflow Models from Event-Based Data using Little Thumb. *Integrated Computer-Aided Engineering*, 2003, 10(2), 151-162.
- [25] WESKE, Mathias. *Business process management: concepts, languages, architectures*. New York: Springer, c2007. ISBN 978-3540735212.

SHRnutí STUDIjNÍ OPORY

Studijní opora „Techniky a nástroje v oblasti řízení podnikových procesů“ se vás pokusila seznámit se základními pojmy, metodami a přístupy k zlepšování podnikových procesů. Výchozím pojmem je nepochybně procesní model, který se používá na zachycení aktuálního stavu podnikového procesu – as-is procesní model a na návrh redesignu procesu – to-be procesní model. Smyslem řízení podnikových procesů je pomoci procesnímu řízení v podnicích, firmách nebo organizacích a důsledným sledováním životního cyklu BPM je možné podnikové procesy zlepšit, zjednodušit, případně automatizovat. Implementace BPM do řízení podniku je v současnosti nespornou konkurenční výhodou.

Studijní opora se pokusila přiblížit čtenáři techniky, nástroje a praktické postupy ke splnění výše uvedeného cíle.

PŘEHLED DOSTUPNÝCH IKON



Čas potřebný ke studiu



Klíčová slova



Průvodce studiem



Rychlý náhled



Tutoriály



K zapamatování



Řešená úloha



Kontrolní otázka



Odpovědi



Samostatný úkol



Pro zájemce



Cíle kapitoly



Nezapomeňte na odpočinek



Průvodce textem



Shrnutí



Definice



Případová studie



Věta



Korespondenční úkol



Otázky



Další zdroje



Úkol k zamyšlení

Název: **Techniky a nástroje v oblasti řízení podnikových procesů**

Autor: **doc. RNDr. Ing. Roman Šperka, Ph.D.**
Ing. et Ing. Michal Halaška
Ing. Dalibor Šimek

Vydavatel: Slezská univerzita v Opavě
Obchodně podnikatelská fakulta v Karviné

Určeno: studentům SU OPF Karviná

Počet stran: 188

Recenzenti: prof. Ing. Miroslav Hučka, CSc.
doc. Mgr. Petr Suchánek, Ph.D.

Tiskárna: X-MEDIA servis s.r.o.

Náklad: 50 ks

ISBN 978-80-7510-312-3

Tato publikace neprošla jazykovou úpravou.