



ZADÁNÍ BAKALÁ SKÉ PRÁCE

Název:	Možnosti vzájemného p evodu popisu podnikových proces v notacích BPMN a S-BPM
Student:	Dominik Mazura
Vedoucí:	Ing. Pavel Náplava
Studijní program:	Informatika
Studijní obor:	Informa ní systémy a management
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2016/17

Pokyny pro vypracování

Porovnejte metody popisu podnikových proces pomocí notací BPMN a S-BPM. Analyzujte silné a slabé stránky notací, identifikujte potenciální místa ztráty zachycené informace. Navrhn te zp sob p evodu model z jedné notace do druhé tak, aby ke ztrát informací nedocházelo, respektive aby ztráta byla minimální.

Postupujte následovně :

- 1) Seznamte se s notací BPMN a S-BPM (v etn metodiky tvorby diagram).
- 2) Analyzujte rozdíly mezi notacemi (použité elementy, zp sob zachycení toku informací, identifikace ú astník procesu atd.).
- 3) Navrhn te metodiku p evodu model z jedné notace do druhé tak, že v prvním kroku bude identifikována možná ztráta obsažené informace (nap . v druhé notaci chyb ící element, jiný zp sob zachycení toku informací atd.). Ve druhém kroku bude navržen zp sob minimalizace ztráty informace (nap . kombinace n kolika element) a ve t etím kroku bude proveden p evod modelu do cílové notace.
- 4) Metodiku ov te a demonstруйте na vybraných modelech, které dodá vedoucí práce.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
d kan

V Praze dne 11. listopadu 2015

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



Bakalářská práce

Možnosti vzájemného převodu popisu podnikových procesů v notacích BPMN a S-BPM

Dominik Mazura

Vedoucí práce: Ing. Pavel Náplava

11. května 2016

Poděkování

Děkuji Ing. Pavlu Náplavovi za vstřícnost, ochotu a cenné rady, které mi pomohly při psaní bakalářské práce. Rád bych také poděkoval rodině, za celkovou podporu jak při tvorbě bakalářské práce, tak v průběhu celého studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, avšak pouze k nevýdělečným účelům. Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 11. května 2016

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2016 Dominik Mazura. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Mazura, Dominik. *Možnosti vzájemného převodu popisu podnikových procesů v notacích BPMN a S-BPM*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Cílem práce je analýza a porovnání notací BPMN a S-BPM určených pro popis byznys procesů. Součástí práce je zmapování a následná minimalizace možných ztrát informací, při převodu z jedné notace do druhé. Ztráta je způsobena rozdílnými notacemi a přístupy k vytváření modelů. BPMN je jedna z nejvyužívanějších notací pro popis procesů. Důvodem pro převod modelů do notace S-BPM je snaha o vytvoření uživatelsky příjemnějšího diagramu, jehož modelování je rychlejší. Výsledkem práce je popsána metodika pro obousměrný převod notací. S-BPM je vhodné pro mapování malých procesů, které potřebují rychle reagovat na změny, ale s rostoucí složitostí procesu roste nepřehlednost diagramů. Součástí práce je převod části modelu aplikace závěrečných prací provozované na FIT ČVUT.

Klíčová slova analýza, převod, notace, procesní řízení, byznys proces, přehlednost, BPMN, S-BPM

Abstract

The major goal of this thesis is analysis and comparison notations of the business processes BPMN and S-BPM. One of the part of the thesis is mapping

and minimization losses while transferring from one notation to another. Losses are made by differences between notations and their ways of modelling processes. BPMN is one of the most used notation for describing business processes. The reason for transferring to S-BPM is effort to make more user friendly diagrams and faster modelling of them. The result of the thesis is methodology for bidirectional transfer of notations. S-BPM is good for modelling small process which need respond quickly for changes but the more complicated the process is, the lesser the diagrams are readable. Part of the thesis is transferring part of the model of application diploma thesis which is used by FIT CTU.

Keywords analysis, transfer, notation, Business Process Management, business process, readability, BPMN, S-BPM

Obsah

Úvod	1
1 Úvod do procesního řízení	3
1.1 Proces	3
1.2 Procesní řízení	3
1.3 Zachycení procesů	4
2 Business Process Management (A-BPM)	7
2.1 Životní cyklus A-BPM	7
2.2 Způsoby vytváření BPMN modelů	8
2.3 A-BPM diagramy	10
3 Subject-Oriented Business Process Management (S-BPM)	13
3.1 Základní přestavení S-BPM	14
3.2 Způsoby vytváření S-BPM modelů	14
3.3 Zainteresované strany procesu	16
3.4 S-BPM diagramy	17
4 Notace BPMN a S-BPM	19
4.1 Business Process Management Notations (BPMN)	19
4.2 Subject-Oriented Business Process Management Notation (S-BPM)	25
5 Srovnání notací BPMN 2.0 a S-BPM	29
5.1 Porovnání notací	29
5.2 Stav S-BPM a jejich ekvivalenty v BPMN	29
5.3 Sekvenční toky a přechody	31
5.4 Subjekty a plavecké dráhy	32
5.5 Tok procesu diagramem	33
5.6 Shrnutí kapitoly	33

6	Metodika převodu notací	35
6.1	Plavecké dráhy jako subjekty	35
6.2	BPMN brány	36
6.3	Podproces	38
6.4	Ztráty informací při převodu	38
6.5	Výsledná metodika převodu notací	39
6.6	Tabulka převodu elementů	40
7	Ukázky převodu diagramů	43
7.1	Modelovací nástroje	43
7.2	BPMN → S-BPM	44
7.3	S-BPM → BPMN	51
7.4	Překlad procesu schválení zadání závěrečné práce	55
7.5	Shrnutí poznatků z ukázek převodů	58
	Závěr	61
	Literatura	63
	A Seznam použitých zkratk	67
	B Diagramy a přehled BPMN	69
	C Obsah příloženého CD	73

Seznam obrázků

2.1	Životní cyklus A-BPM	8
2.2	BPMN procesní diagram	10
2.3	BPMN kolaborační diagram	11
2.4	BPMN konverzační diagram	11
2.5	BPMN diagram choreografie	12
4.1	BPMN události	20
4.2	BPMN aktivity	21
4.3	BPMN gateways	22
4.4	BPMN spojovací objekty	23
4.5	BPMN swimlanes	23
4.6	BPMN data objects	23
4.7	Příklad BPMN diagramu	25
4.8	Příklad S-BPM SID	28
4.9	Příklad S-BPM SBD kuchař	28
4.10	Příklad S-BPM SBD poslíčka	28
5.1	Recive úlohy a recive stavy	31
5.2	Sekvenčních toky a přechody	31
7.1	BPMN diagram procesu objednání pizzy	44
7.2	SID procesu objednání pizzy	45
7.3	SBD zákazníka v procesu objednání pizzy	46
7.4	SBD obchodníka v procesu objednání pizzy	47
7.5	SBD kuchaře v procesu objednání pizzy	47
7.6	SBD poslíčka v procesu objednání pizzy	47
7.7	BPMN diagram cestovního plánu	48
7.8	SID procesu žádosti o cestovní plán	49
7.9	SBD cestovní kanceláře v procesu žádosti o cestovní plán	50
7.10	Převedený BPMN diagram procesu objednání pizzy	51
7.11	SID procesu cestovní žádosti	53

7.12	SBD vedoucího v procesu cestovní žádosti	53
7.13	SBD zaměstnance v procesu cestovní žádosti	54
7.14	BPMN diagram procesu cestovní žádosti	54
7.15	Ukázka BPMN diagramu aplikace závěrečných prací	56
7.16	Náhled SID procesu závěrečné práce	57
7.17	SBD subjektu Systém-přepřeracování zadání	58
B.1	SBD subjektu student pro ZP	70
B.2	SID procesu závěrečné práce	71
B.3	BPMN 2.0 přehled symbolů	72

Seznam tabulek

4.1	S-BPM entity pro SID	27
4.2	S-BPM entity pro SBD	27
6.1	Převod entit BPMN a S-BPM	40

Úvod

Mapování procesů by mělo být nedílnou součástí každé organizace. Zachycení know-how je bezpochyby jedním z důležitých požadavků pro úspěšný rozvoj organizace a její konkurenční výhodu.

Procesní řízení a byznys procesy neboli podnikové procesy jsou důležité pro pochopení a zmapování procesů v organizaci. Protože bez znalosti procesů organizace nemůžeme procesy zlepšovat a zrychlovat. Pro popis procesů existuje několik notací, které jsou ve většině případů odlišné, a každá je vhodná pro jiný přístup k modelování. Pokud existují dva podobné procesy zachycené v odlišných notacích, je velmi obtížné jejich porovnání. Z těchto důvodů jsem si zvolil jednu z nejčastěji používaných notací BPMN, která se zaměřuje na aktivity v procesu a rychle se rozvíjející S-BPM, která staví do popředí člověka.

Práce se zabývá porovnáním a vzájemným převodem těchto dvou notací a klade důraz na redukci ztrát informací při převodu. Součástí práce je ukázka převodu diagramů aplikace závěrečných prací na FIT ČVUT.

Cílem bakalářské práce je najít vhodný způsob obousměrného převodu notací BPMN a S-BPM. Práce se zaměřuje na porovnání notací a toku informací v byznys procesech. Dále také na porovnávání rozdílných způsobů vytváření modelů a přístupů k procesu. Jedním z cílů je zmapování řešení a vytvoření metodiky pro převod notací s ohledem na redukci ztrát při převodu. Při převodu z jedné notace do druhé je záměrem převést co nejvíce informací obsažených v diagramech, zachovat jejich logický význam. Záměrem je také snaha o vytvoření přehledných a snadno čitelných diagramů. Cílem práce je také pomocí metodiky S-BPM a její notace smazat pomyslnou komunikační bariéru při porozumění modelům mezi technickými a netechnickými účastníky procesu.

Práce je členěna do sedmi kapitol. V první kapitole jsou vysvětleny základní pojmy spojené s procesním řízením, jeho důležitost a účel v organizaci. V druhé a třetí kapitole jsou představeny jednotlivé přístupy pro tvorbu modelů. Obsahem těchto kapitol je popsání jednotlivých přístupů k vytvá-

ření modelů, životní cyklus tvorby modelů a zainteresované strany v procesu. Čtvrtá kapitola popisuje jednotlivé notace BPMN a S-BPM. Zaměřuje se na význam a kategorizaci jednotlivých symbolů a jejich použití pro popis procesů. V páté kapitole jsou notace metodik porovnány a popsány jejich společné i odlišné vlastnosti. V šesté kapitole jsou diskutovány možné řešení převodu. Tato kapitola také obsahuje vytvořenou metodiku pro převod notací. Obsahem poslední kapitoly je ukázka převodu diagramů z jedné notace do druhé a hledá se nejvhodnější způsob pro zachování přenášených informací a čitelnosti diagramů. V této kapitole jsou převáděny malé i rozsáhlejší procesy.

Úvod do procesního řízení

Každá organizace vykonává opakující se činnosti pro dosažení svých stanovených cílů. Definované posloupnosti činností, které mají začátek a konec nazýváme proces.

1.1 Proces

„Proces se skládá ze souboru činností, které jsou prováděny v souladu s organizačním a technickým prostředím. Aktivitý procesu společně dosahují daného cíle. Každý proces je prováděn pouze v rámci jedné organizace, ale může interagovat s procesem jiné organizace.“ [1]

Jednoduše můžeme říci, že proces je soubor činností, které přeměňují vstupy na výstupy. Vstupy mohou být svojí povahou informační, materiální, lidské nebo finanční. Výstupem je pak dosažení stanoveného cíle, který můžeme označit za produkt. Na proces se také můžeme dívat jako na návod, který určuje pořadí vykonávaných činností. Proces je základem pro procesní řízení.

Byznys proces, firemní proces nebo podnikový proces jsou možné názvy pro jeden a ten samý pojem. V práci se držím označení byznys proces nebo pouze proces, protože si myslím, že český překlad firemní proces by mohl evokovat domněnku, že procesy a s nimi spojené procesní řízení je využíváno pouze firmami. Avšak jakákoliv organizace, nebo jen část organizace, může používat procesní řízení a jejich hlavním cílem nemusí být generování zisku.

1.2 Procesní řízení

S pojmem byznys proces se úzce pojí procesní řízení neboli Business Process Management (BPM), které slouží k zachycení a plánování byznys procesů.

Definice procesního řízení je tolik, kolik je procesů [2].

„Procesní řízení zahrnuje koncepty, metody a způsoby pro podporu tvorby, administrace, průběhu a analýzy byznys procesů. Základem BPM je jednoznačná definice procesu s jeho aktivitami a omezeními mezi nimi při průběhu.“ [1]

Jednou z výhod procesního řízení je možnost zachycení know-how organizace, což je velmi důležité. Protože pokud má organizace zmapované své procesy, snižuje riziko ztráty znalostí při odchodu jejího člena. Úkolem procesního řízení je právě také zdokumentování procesů. Pokud organizace nezná své procesy, pak nedokáže určit, kde jsou slabiny a nedostatky v procesech a následně je velmi obtížné jejich odstranění nebo alespoň minimalizování. Procesní řízení může také sloužit k částečné nebo dokonce plné automatizaci procesů a tím snížit lidský faktor chybovosti. Protože díky exaktně definovaným procesům můžeme generovat spustitelný kód. Procesní řízení je také do jisté míry motivující pro zaměstnance, protože každý vidí, že je členem něčeho velkého a práce, kterou vykonává v rámci procesu, je užitečná a smysluplná.

1.3 Zachycení procesů

Procesy můžeme zachytit do diagramů pomocí notací a jejich grafických symbolů. V práci se zabývám notacemi BPMN a S-BPM, které si dále v práci představíme. BPM využívají i další notace jako je ARIS, IDEF nebo UML.

1.3.1 Model a diagram

Rád bych nejprve upřesnil, co znamenají pojmy diagram neboli schéma a co model, protože tyto dva pojmy nejsou totožné a jsou často zaměňovány. Diagram je podmnožinou modelu. Diagram je složen z geometrických symbolů znázorňující informaci, kterou reprezentuje model. Na rozdíl tomu model je vzor nebo plán pro zachycení hlavního předmětu systému nebo konceptu. Model se tedy skládá z několika diagramů a nabízí nám ucelený pohled. Model si můžeme představit jako 3-D model domu a diagramy jsou jednotlivé 2-D pohledy na dům, které mohou být vytaženy z modelu. Tedy model poskytuje celkový pohled a obsahuje všechny potřebné informace. V diagramu je obsažena pouze část modelu. Model členíme na diagramy z důvodu lepší přehlednosti procesů.

V práci se zabývám převodem jednotlivých diagramů, které jsou obsahem modelů. U jednodušších příkladů, kdy model obsahuje pouze jeden diagram, je jejich význam totožný, ale i v těchto případech se držím označení diagram.

Dle [3] má model oproti diagramu několik výhod:

Konzistence – Model obsahuje všechny pohledy a tedy nemůže nastat jev, kdy model obsahuje protichůdné informace.

Spolupráce – Všichni členové týmu pracují na jednom zdroji pravdy a přidávají do digramů informace a elementy zpřesňující model.

Komplexní abstrakce – Model může být velmi velký a obsáhlý, ale skládá se z jednotlivých diagramů, které mohou být malé a jednoduché. Modely mohou sloužit pro generování zdrojového kódu, a také mohou být validovány.

Procesní řízení není záležitostí pouze informačních technologií, ale zohledňuje také lidský aspekt a cíle organizace.

Důležité je si uvědomit, že člověk je stále klíčový element a sebelepší model ho nedokáže nahradit. Proto o procesním řízení mluvíme jako o nástroji, nikoliv jako o technologii. Procesní řízení neřeší problémy v organizaci, je pouze nástrojem na zachycení procesních modelů a jejich ztransparentnění. Zmapování vlastních procesů je velice důležité, protože bez jejich pochopení není možné najít „úzká hrdla“, zrychlit a zefektivnit procesy odehrávané v organizaci.

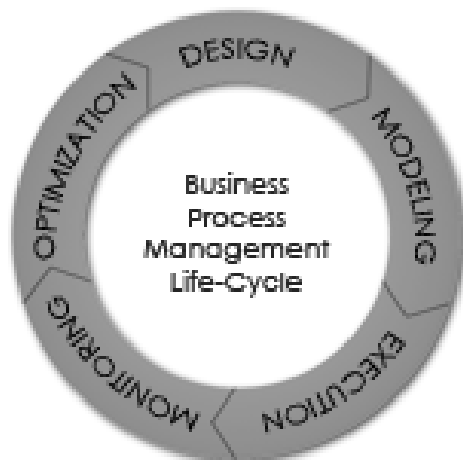
Business Process Management (A-BPM)

Pro odlišení dvou pojmů obecného procesního řízení (Business Process Management) a přístupu BPM, který využívá notaci BPMN, jsem za pomoci vedoucího práce zvolil označení Aktivita-Business Process Management neboli A-BPM. Předpona aktivity je zvolena z důvodu, že A-BPM se při tvorbě procesů zaměřuje na činnosti oproti S-BPM, které se zaměřuje na člověka čili subjekt, který činnosti provádí. Tímto vydefinováním pojmů jsem postavil A-BPM a S-BPM na stejnou úroveň, která je nezbytná pro jejich porovnávání.

A-BPM je přístup k tvorbě modelů, který vychází z procesního řízení (BPM). Úkolem A-BPM je pomocí notace BPMN zachytit procesy organizace jak interní tak externí takovým způsobem, aby byly srozumitelné a přehledné. A-BPM neslouží pouze pro vytváření diagramů, ale také podporuje jejich implementaci a následné provádění. Dohromady o těchto činnostech hovoříme jako o životním cyklu A-BPM. Jedná se o jeden z nejrozšířenějších přístupů pro práci s byznys procesy, který využívá například firma IBM.

2.1 Životní cyklus A-BPM

Průběh životního cyklu A-BPM můžeme rozdělit do následujících pěti částí. 2.1 První fází je zmapování a návrh, který slouží k popsání procesu tak, aby byl co nejjednodušší, ale zároveň dostatečně detailní aby nedocházelo k nejasnostem. V první fázi také popíšeme všechny faktory, které proces ovlivňují. Druhou fází je modelování, kde se proces pomocí modelovacího nástroje a některé z notací zanesou do diagramu, ve kterém se také určí posloupnost úloh. V této fázi je možné proces simulovat. Po úspěšné simulaci je proces schválen a je možné postoupit do další fáze, kterou je vykonání procesu. V této fázi je model převeden do spustitelného kódu v některém z programovacích jazyků.



Obrázek 2.1: Životní cyklus A-BPM [1]

Součástí této fáze je také pozorování procesu z důvodu vyhodnocení a analýzy v další fázi životního cyklu. Poslední fází životního cyklu A-BPM je optimalizace a pokud ve fázi monitorování došlo k nějakým problémům, v této fázi se opraví. [1]

Formálně a přesně definované procesy by měly být podmínkou pro dobře navržené verze procesních modelů. Proto po modelování procesu následuje jeho validace, kde se zjistí, jestli model odpovídá skutečnému procesu v organizaci. [4] Validace může probíhat tak, že všechny zainteresované osoby zkontrolují své aktivity, nebo se proces může nasimulovat krok po kroku. Jakmile je procesní model navržen a zvalidován, může začít jeho implementace. Jedním z důvodů procesního řízení je, aby všechny zainteresované strany chápaly danou problematiku stejně a nedocházelo k odlišnému výkladu procesu. Proto je velmi důležité, aby koncept byznys modelu byl jednoduchý, relevantní a intuitivně pochopitelný. [5]

V mojí práci se budu zabývat především druhou fází životního cyklu A-BPM, tedy modelováním.

2.2 Způsoby vytváření BPMN modelů

Jednotlivé činnosti vykonávané v rámci procesu označujeme v A-BPM za aktivity. Můžeme říci, že aktivita je „základním stavebním kamenem“ pro mapování procesů. Při vytváření modelů se soustředíme a stavíme do popředí strukturu procesu a interakci mezi účastníky. Technické záležitosti a implementace procesu pro nás nejsou tolik důležité, protože změna ve způsobu vykonávání aktivity nemá vliv na proces jako takový. A-BPM se soustředí na aktivity procesu a na vztahy mezi nimi. Tato skutečnost má zásadní vliv na

tvorbu modelů. Zdroje pracují, aby byly naplněny byznys cíle organizace. Modelování probíhá takzvaně *shora dolů*. To je způsob, kde tvorba modelu začíná obecným pohledem na proces a postupně jsou definovány dílčí části procesu.

Diagramy jsou vytvářeny manuálně, protože A-BPM nespecifikuje jak přesně by měl diagram vypadat. Důvodem je pravděpodobně fakt, jak uvádí Bruce Silver na svém blogu [6], že BPMN se snaží být abstraktní a detailní zároveň. Na druhou stranu osoba vytvářející model není vázána striktními pravidly při modelování a má spoustu prostoru pro kreativitu.

2.2.1 Granularita modelu

Definováním procesu s pomocí modelu je zapotřebí se dostat až do stavu, kdy granularita modelu je dostatečně velká na to, aby model reflektoval proces a nemohlo dojít k nejasnostem způsobeným nedostatečně detailním popisem procesu. [1] Takto obecné určení pro míru detailu modelu se zdá být jasné. Ale při jeho tvorbě záleží na modeláři, jak detailní model vytvoří, protože A-BPM blíže neuvádí, jak velkou granularitu musí model mít. V modelu musí být zachyceny všechny informace potřebné pro vykonání procesu.

2.2.2 Zainteresované strany procesu

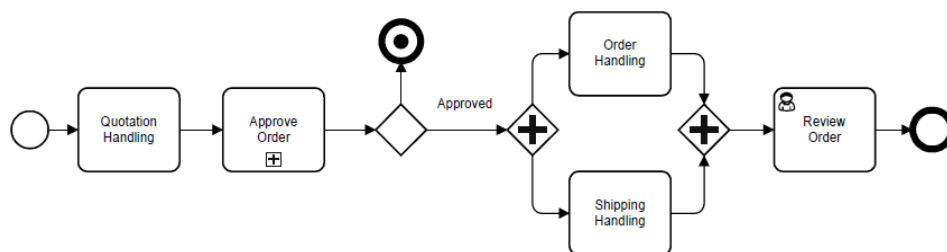
Každý proces musí mít osobu, která za něj nese zodpovědnost. Také v průběhu celého životního cyklu musí být předem specifikovány osoby, které mají zodpovědnost za určitou činnost, nebo se podílí na tvorbě procesu. Proto A-BPM definuje skupiny, takzvané role, které jsou rozděleny podle zájmů, míry vlivu na proces, druhu práce v rámci procesu nebo jeho vytváření, atd. A-BPM rozděluje zainteresované strany, které nazýváme stakeholdery, do několika rolí. Jednomu člověku mohou být přiděleny role několika těchto skupin. Všechny zainteresované strany by spolu při tvorbě procesu měli spolupracovat.

Jak uvádí [1], rozlišujeme několik následujících rolí:

CPO - Chief Process Officer – Zastává nejvyšší roli v organizaci, co se tvorby procesů týká a je zodpovědný za strategii organizace. Definuje pravidla, metodiky a politiku a také hlídá jejich dodržování.

Business Engineer – Určuje cíle organizace a procesů. Má zkušenosti i z ne-technických oborů a dokáže dosáhnout takového řešení, aby vyhovovalo všem zainteresovaným stranám.

Process Designer – Zodpovídá za správnou analýzu a tvorbu byznys procesů. Blíže komunikuje s ostatními účastníky, především s aktéry procesu, od kterých získává detailní informace o jednotlivých aktivitách, z kterých pak vytváří celkový pohled a zachycuje jej do modelu.



Obrázek 2.2: Procesní diagram [7]

Process Participant – Vykonává jednotlivé instance byznys procesu. Hraje důležitou roli při tvorbě procesních modelů, protože má velmi dobrou znalost o jednotlivých aktivitách organizace.

Knowledge Worker – Používá software organizace, s pomocí kterého samostatně vykonává jednotlivé aktivity byznys procesu.

Process Responsible – Každý byznys model má jednu zodpovědnou osobu, která je zodpovědná za správné vykonávání byznys procesů využívající model. Hledá v procesu neefektivní části a snaží se je zlepšit. Blíže spolupracuje s modeláři a účastníky.

System Architect – Nese zodpovědnost za vývoj byznys procesů.

Developer – Implementuje byznys procesy na základě jejich modelů.

2.3 A-BPM diagramy

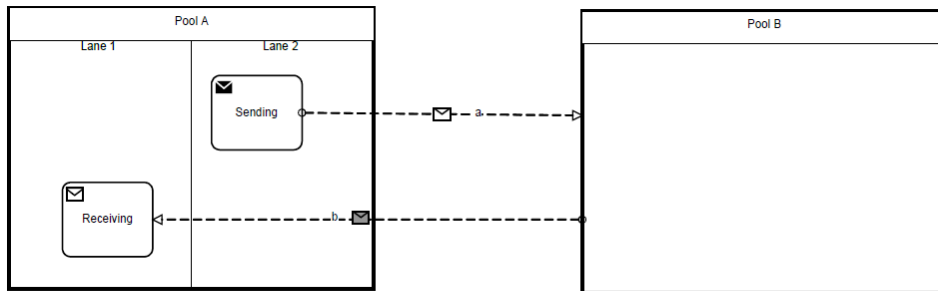
V A-BPM máme možnost zobrazit proces v různé míře detailu pomocí několika diagramů.

A-BPM definuje čtyři druhy diagramů [7].

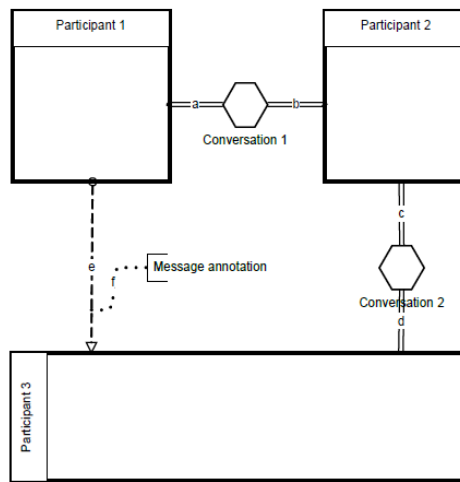
Procesní diagram – Zachycuje proces pouze jednoho účastníka 2.2. Znárodnuje posloupnost aktivit procesu, tak jak byl definován. Jedná se o popis procesu s dostatečnou mírou detailu pro vykonávání daného procesu.

Kolaborační diagram – Zobrazuje komunikaci mezi účastníky v procesu, kde procesy jednotlivých účastníků nemusí být definované, jak tomu je na obrázku 2.3. Soustředí se na rozlišení jednotlivých procesů pomocí účastníků, kteří procesy vykonávají.

Konverzační diagram – Poskytuje zjednodušený pohled na kolaborační diagram a vynechává procesní logiku jednotlivých účastníků 2.4.



Obrázek 2.3: Kolaborační diagram [7]

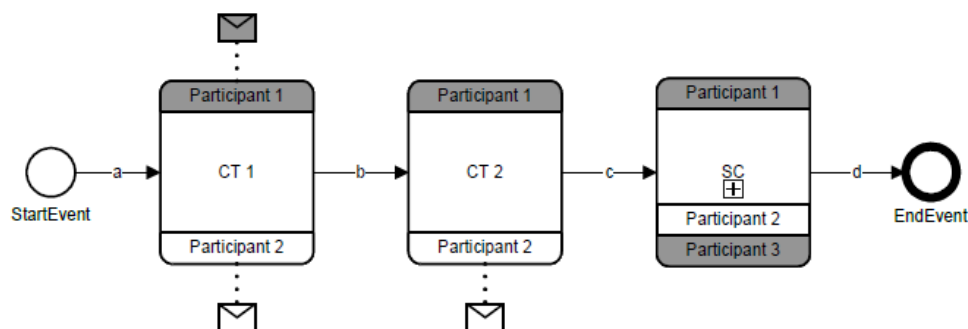


Obrázek 2.4: Konverzační diagram [7]

Diagram choreografie – Zaměřuje se na interakci mezi jednotlivými procesy a na informace, které si předávají. Používá se především pro znázornění interakce mezi dvěma a více organizacemi [2.5]. Každý krok v diagramu zahrnuje dva a více účastníků. Také zobrazuje interakci mezi dvěma účastníky pomocí zpráv, které si posílají.

V práci se zabývám obsahem vzniklým spojením procesního a kolaboračního diagramu. V literatuře se můžeme setkat s rozdělením pouze do tří druhů diagramů, kde právě procesní a kolaborační diagramy jsou sloučeny do jednoho kolaboračního diagramu [8]. V případě sloučení těchto dvou diagramů máme k dispozici kompletní procesní logiku celého procesu. Vidíme celý proces od startovacího symbolu až po koncový symbol procesu a se všemi možnostmi průchodu procesem.

Konverzační a kolaborační diagramy nepřidávají žádnou další informaci o procesech a pouze zobecňují diagramy pro jejich lepší přehlednost. Odlišně



Obrázek 2.5: Diagram choreografie [7]

úrovně detailu procesů jsou vhodné pro různé A-BPM role. Například osoba s rolí developer musí mít možnost zobrazení pohledu v co největším detailu a pro tuto roli je vhodný procesní diagram spolu s kolaboračním diagramem. Na druhou stranu osoba v roli CPO nemusí znát detailní popis jednotlivých procesů, a proto pro ni může mít mnohem větší hodnotu diagram choreografie nebo kolaborační diagram. Kolaborační diagram můžeme také využít v situaci, kdy chceme, aby určitý proces daného účastníka zůstal skrytý.

A-BPM umožňuje zachytit celý model do jednoho kolaboračního diagramu. Pokud by se velikost diagramu stala nepřehlednou, je možné „schovat“ logické celky částí procesu do takzvaných podprocesů. Podprocesy se pak dále definují v jiných diagramech, tím se sníží počet symbolů v diagramu původním. Množství zobrazených informací je stále zachováno, pouze je rozdělíme do více diagramů. Podprocesy vysvětlují dále v části notace 4.1.2.

Subject-Oriented Business Process Management (S-BPM)

Moderní organizace kladou důraz na rychlost a čas, který hraje velkou roli v úspěchu. Protože rychlé dodání produktu zákazníkovi představuje velkou konkurenční výhodu. Proto se urychluje i mapování procesů. S-BPM dovoluje dělat v jednotlivých iteracích rychlé změny a dokáže být oproti A-BPM více flexibilní [9].

„Zákazníci nejsou schopni zodpovědět na všechny otázky, a tak dochází k tomu, že si často vymýšlí, jak by to mohlo vypadat. Na druhou stranu dokáží velmi detailně popsat malou část problému.“

– Jiří Knesl

Tímto výrokem Jiřího Knesla jsem chtěl poukázat na to, že jedním z hlavních důvodů, proč dochází k častým změnám při analýze a tvorbě modelu, je zákazník nebo obecněji vlastník procesu. Častým jevem při analýze procesu je skutečnost, že vlastník procesu popisuje proces takový, jaký by chtěl aby byl a ne jeho reálný stav. Změny v průběhu tvorby modelu jsou přirozené a model tak prochází jakýmsi vývojem. Tyto změny není možné úplně eliminovat a proto je nutné se jim přizpůsobit.

Další fakt na, který citát poukazuje je dobrá schopnost popsání části procesu aktérem. Aktér dokáže velmi detailně popsat své úlohy v procesu a na této skutečnosti také S-BPM staví. Protože každý subjekt v S-BPM má svůj proces a nemusí znát procesy ostatních subjektů. Oproti tomu při modelování v A-BPM je nutné znát celý proces.

Životní cyklus A-BPM uvedený v kapitole 2.1 a životní cyklus S-BPM jsou velmi podobné a průběh tvorby procesu funguje na stejném principu pěti fází [10], proto ho již v této kapitole neuvádím.

3.1 Základní přestavení S-BPM

Autorem metodiky S-BPM je Albertem Fleischmannem, který ji poprvé představil na konferenci S-BPM One v roce 2009. Od té doby se každý rok tato konference pořádá. A. Fleischmann a jeho tým na této konferenci představují změny ve vývoji metodiky.

Metodika S-BPM vychází z BPM a zaměřuje se na člověka a tvorby modelů z přirozeného jazyka. Jedním z cílů metodiky je smazání pomyslné bariéry při porozumění modelů mezi technickými a netechnickými osobami, které jsou v procesu zainteresovány.

3.2 Způsoby vytváření S-BPM modelů

S-BPM staví do popředí subjekt, proto se také zaměřuje na komunikaci mezi nimi. Proces je znázorněn jako spolupráce několika subjektů, kteří spolu komunikují. S-BPM se nesnaží nahradit A-BPM, ale v některých případech je rychlejší a efektivnější, než robustní A-BPM. [11]

3.2.1 Modelování pomocí přirozeného jazyka

Podle [12] je možné generovat přirozený jazyk z S-BPM modelů. Také je možný obrácený postup a s pomocí podmnožiny přirozeného jazyka je možné generovat S-BPM modely. Tato vlastnost S-BPM má za následek značné urychlení mapování a pochopení procesů. Mohlo by se zdát, že model vytvořený touto cestou není exaktní. Herbert Kindermann ředitel společnosti Metasonic GmbH však uvádí, že S-BPM model je dostatečně přesný na to, aby mohl být chápán jako abstraktní stavový automat a tudíž je možné generovat spustitelný kód [13].

S-BPM se snaží vycházet z přirozeného jazyka a tím zachytit proces při běžné mluvě. Za pomoci této vlastnosti se snaží docílit snadnějšího pochopení výsledného modelu. Jak uvádí [9], z běžné věty můžeme identifikovat potřebné informace pro jejich snadné zachycení do diagramu. Podmět ve větě určuje subjekt (kdo), přísudek (co) určuje akci subjektu a jedná se o jeho vnitřní chování a předmět (s čím) určuje objekt, s kterým subjekt pracuje. Ve většině případů je objektem zpráva. Angličtina je pro modelování snadnější, protože má pevně daný slovosled.

Pro lepší pochopení uvádím příklad proces žádosti o služební cestu z prezentace [14] od autora S-BPM Alberta Fleischmanna.

Představme si situaci, kdy zaměstnanec, který potřebuje odjet na služební cestu, musí mít schválení od svého vedoucího. Tuto situaci reflektuje následující slovní popis procesu.

„Zaměstnanec vyplní cestovní žádost. Zaměstnanec vyplní začátek a konec cesty a její důvod. Vedoucí zkontroluje žádost a informuje zaměstnance o jeho rozhodnutí. Cestovní žádost může být schválena nebo zamítnuta. Pokud je žádost schválena, je vedoucím odeslána cestovnímu agentovi¹, který ji zpracuje.“

Poté co jsme definovali a slovně popsali byznys proces, nalezneme v textu informace pro jeho namodelování.

Identifikujeme **subjekty**, které jsou v textu zvýrazněny zeleně. Následně identifikujeme **zprávy a byznys objekty** označené modře, které si subjekty vyměňují a nakonec definujeme **vnitřní chování** subjektů značené červeně.

Nyní máme všechny potřebné informace pro vytvoření modelu tohoto procesu. Diagramy procesu je možné nalézt v části ukázek převodů diagramů 7.11. Touto cestou jsem chtěl zdůraznit a předvést vlastnost S-BPM, díky které jsme schopni celkem jednoduše z přirozeného jazyka získat potřebné informace pro namodelování procesu.

3.2.2 Granularita procesu

Na rozdíl od A-BPM, kde není jasně specifikováno jak detailní by měl model být, má modelování v S-BPM jasná pravidla. Proto také o S-BPM mluvíme jako o metodice a o A-BPM nikoli. Granularita modelu S-BPM by měla být tak podrobná, aby bylo jasné, kdo má za co zodpovědnost. Není možné, aby pro jednu akci existovalo více účastníků nebo dokonce akce bez účastníka. [11]

Subjekt definuje granularitu procesu. Z toho vyplývá, že modelář by měl začít definováním subjektů (účastníků) procesu a ne akcemi, jak tomu je u A-BPM. Při modelování postupujeme ze *zdola nahoru*, právě naopak, jak tomu je při modelování pomocí A-BPM, které modelujeme shora dolů.

3.2.3 Nezávislost subjektů

Dalším velkým přínosem S-BPM je fakt, že subjekty jsou na sobě nezávislé. Pro jednotlivé subjekty v procesu je dostatečnou informací znalost o subjektu nebo subjektech, s kterým komunikují. Pokud spolu subjekty neinteragují, znamená to, že si neposílají žádné informace, o ostatních subjektech ani neví, že existují. Z tohoto důvodu modeláři mají usnadněnou práci ve smyslu, že při vytváření diagramu chování subjektu nemusí znát celý proces, ale pouze činnosti, které daný subjekt zastává. Následná validace je také přehlednější a snadnější, protože máme možnost zvalidovat chování každého subjektu zvlášť. Z důvodu nezávislosti subjektů S-BPM oproti A-BPM, S-BPM rozděluje model vždy na diagramy „interakci subjektů“ a „interní chování subjektů“. V A-BPM je také možné v kolaboračním diagramu rozdělit diagramy pomocí podprocesů na účastníky, ale tato volba není povinná.

¹Výraz cestovní agent vznikl přeložením anglického výrazu *travel agent*. Jedná se o zaměstnance organizace, který má za úkol evidovat a zpracovávat služební cesty zaměstnanců.

3.3 Zainterесované strany procesu

Podobně jako A-BPM, také S-BPM rozděluje účastníky procesu, které nazýváme stakeholdery, do několika skupin rozdělených podle jejich pravomocí a zodpovědností v procesu. Albert Fleischmann v knize S-BPM [15] rozděluje účastníky do čtyř následujících kategorií:

Govenоři – Lidé, kteří se starají o daný proces, mají za něj zodpovědnost a řídí ho. Podobně jak tomu je u CPO v A-BPM, definují pravidla, metodiky a politiku a hlídají jejich dodržování

Aktéři – Aktivní účastníci v procesu, kteří reálně vykonávají jednotlivé aktivity

Experti – Specialisté v určitém oboru, kteří mají za úkol poskytovat řešení v jejich oboru všem ostatním účastníkům

Facilizátoři – Lidé zodpovědní za tvorbu procesu a jeho organizaci. Určují všechny zainterесované strany a zajišťují mezi nimi vnitřní i vnější síť komunikace jak formální, tak i neformální

Kategorií je o poznání méně než tomu je v A-BPM a jsou více obecnější, protože z definovaného procesu je snadněji čitelnější, kdo má za co zodpovědnost a v tom vidím výhodu, kterou S-BPM přináší.

Detailnější popis pravomocí a zodpovědností účastníků v průběhu životního cyklu procesu je možné dohledat v knize [15]. Práce je zaměřená na notaci, proto se účastníky dále nezabývám.

3.4 S-BPM diagramy

Jak už jsem uváděl, v S-BPM rozdělujeme model na dva druhy diagramů, které popisují proces z důvodu nezávislosti subjektů a lepší přehlednosti modelu. Oproti A-BPM, kde počet diagramů je libovolný, v S-BPM je vždy přesně dán jejich počet. Model vždy obsahuje jeden komunikační diagram a pro každý subjekt jeden diagram interního chování.

SID - Subject Interaction Diagram (komunikační diagram subjektů)

Zobrazuje celkový pohled na proces a zachycuje veškerou komunikaci mezi všemi subjekty, která je realizována pomocí zpráv. Pokud zpráva obsahuje strukturované informace, nazýváme ji byznys objektem. Pro proces existuje právě jeden SID.

SBD - Subject Behavior Diagram (diagram interního chování subjektu)

Zobrazuje chování jednotlivého subjektu, jako jsou informace o tom, jak subjekt reaguje na přijetí zprávy nebo její odeslání. Každý subjekt má svůj interní model chování, tedy počet subjektů se rovná počtu SBD.

Rozdělení diagramů do logických celků usnadňuje jejich následnou validaci a případnou modifikaci. Více o diagramech S-BPM je uvedeno v části notace 4.2.2.

Notace BPMN a S-BPM

Je známo, že obrázek vydá za tisíc slov. To je jeden z důvodů, proč procesy znázorňujeme graficky pomocí symbolů. Symboly, které vytváří diagramy nazýváme notací.

Notace jednotlivých metodik mají za úkol zachytit jednotlivé procesy, aby byly pochopitelné pro všechny zainteresované strany. Jedná se o standardizovanou cestu, jak propojit návrh byznys procesu a jeho implementaci. A-BPM i S-BPM mají svoji notaci pro zachycení informací, které v procesu probíhají. V práci se zabývám notacemi S-BPM a BPMN. Snažím se najít jejich podobné vlastnosti, ale i znaky, ve kterých se liší za účelem zmapování informací, které zjednoduší převod.

4.1 Business Process Management Notations (BPMN)

Jako první si představíme notaci BPMN. V této práci se zabývám pouze notací BPMN ve verzi 2.0, která byla zveřejněna roku 2011, dřívější verze obsahovaly nedostatky a už se nepoužívají, proto jsou pro mojí práci neúčinné.

4.1.1 Postup při modelování

Jak uvádím, diagramy jsou vytvářeny manuálně, protože A-BPM nspecifikuje, jak přesně by měl diagram vypadat. BPMN se snaží být abstraktní a detailní zároveň. Záleží na modeláři, který určí, pomocí kterých symbolů zanesou proces do modelu.

Elementy BPMN můžeme rozdělit do pěti základních kategorií: Flow objects, Connecting objects, Swimlanes, Data object a Artifacts. Plovoucí a spojovací objekty jsou elementy notace pro zachycení chování objektů. Všechny symboly notace BPMN 2.0 jsou obsaženy v příloze B.3.

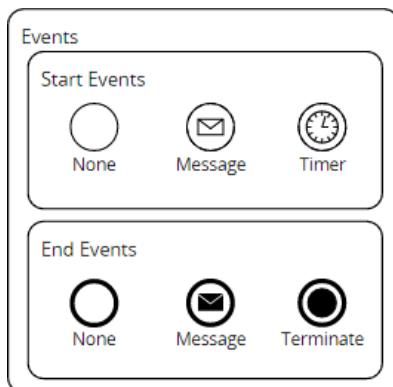
4.1.2 Flow objects

Flow objects (plovoucí objekty) rozdělujeme podle [16] [1] do tří skupin. Flow objects můžeme označit za stavební kameny byznys procesu, které spojujeme za účelem plynulého průchodu procesem.

Events – Události jsou obvykle příčinou nebo důsledkem nějaké aktivity. Události rozdělujeme do tří kategorií, podle jejich vlivu na průběh procesu, startovací, koncové a intermediate. Intermediate událostí můžeme označit cokoliv, co se stane v procesu a zároveň se nejedná o startovací ani koncovou událost. Takovéto události mohou přijímat nebo předávat informace. Příkladem intermediate eventů může být odeslání nebo přijetí zprávy uprostřed procesu.

BPMN 2.0 definuje několik druhů událostí. Zpráva slouží k přenosu informace z jedné aktivity do druhé. Timer označuje časový interval, kdy má začít následující aktivita a tato časová událost nemůže být koncová. Symbol události je znázorněn jako kruh a jejich příklady můžeme vidět na obrázku 4.1.

Dalším typem je podmíněná událost, která určuje za jaké podmínky se může přejít k další aktivitě a také nemůže být koncová. Dále také definujeme signální událost, která je podobná události zprávy s rozdílem, že signál nemá přesně určeného adresáta. Poslední typem události je vícenásobná událost, která nám může slučovat několik událostí dohromady, ale jak uvádí [1] pro lepší pochopení a přehlednost diagramu, je lepší se tomuto typu události, pokud možno, vyhnout.



Obrázek 4.1: Příklad BPMN událostí

Aktivita – Můžeme říci, že aktivita je jakákoliv vykonávaná činnost v rámci procesu. Reprezentují práci vykonávanou v průběhu procesu. Jejich grafické znázornění je na obrázku 4.2.

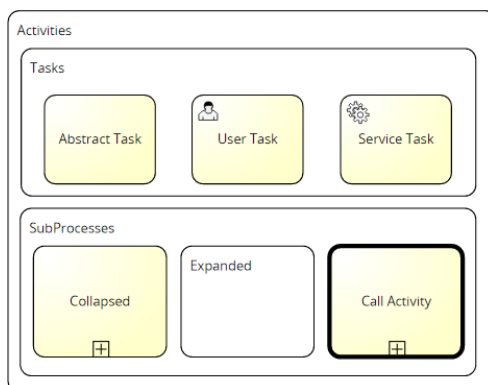
Aktivita se dále dle [17] [16] dělí do čtyř kategorií:

Task – Úloha je aktivita, která se už dále nemůže dělit a neobsahuje žádné další aktivity. BPMN dále rozděluje úlohy do několika dalších podtypů, které jsou označeny symboly pro snadnější a detailnější pochopení diagramu. Příkladem je manuální úloha, která znázorňuje, že úloha je vykonávána člověkem nebo úloha, která přijímá zprávy. Symbolu úlohy můžeme také přidat různé typy značek, které mají různý význam, jako je smyčka, vícenásobná instance nebo kompenzace. Tyto symboly můžeme libovolně kombinovat.

Podproces – Rozdíl úkolu a podprocesu je ten, že podproces je dále dělitelný a obsahuje nejméně dva úkoly. Můžeme říci, že podprocesy „schovávají“ úkoly pro lepší přehlednost diagramu. Pouze u podprocesu se může vyskytnout značka „tilda ~“, která vyjadřuje, že podproces je adhoc. Aktivity v adhoc podprocesu mohou být vykonávány v libovolném pořadí, mohou se opakovat, nebo mohou být přeskočeny.

Transakce – Speciální případ podprocesu, která obsahuje skupinu aktivit. Rozdíl mezi obecným podprocesem a transakcí je takový, že v transakci musejí všichni účastníci dokončit svoji část podprocesu, abychom mohli označit transakci za dokončenou. To znamená, že buď se provedou všechny aktivity najednou anebo ani jedna a v tom případě se vrátí do původního stavu.

Call aktivita – Globální proces, který umožňuje znovupoužití úlohy nebo podprocesu.



Obrázek 4.2: Přehled BPMN aktivit

Gateway – Brány zajišťují dělení, sloučení a řízení toků v cestě diagramu. Můžeme říci, že každá brána obsahuje podmínku, která určuje další směr průchodu diagramem. Brány jsou dále rozděleny podle chování do čtyř následujících skupin: Exkluzivní, inkluzivní, paralelní a typu event. Brány jsou zobrazené na obrázku 4.3.

Některé typy bran se rozhodují na základě logické hodnoty. Při průchodu exkluzivní branou (XOR) se vykoná právě jedna větev. Inkluzivní brána rozdělí průchod procesem do jednoho a více průchodů (OR). Pozn. k značení, brána s „X“ i bez mají stejný význam tedy XOR. Paralelní brána nemá žádnou podmínku, ale zachycuje aktivity, které nastávají souběžně. Tok procesu se v paralelních branách rozděluje. V bráně typu event tok procesu čeká na první událost, která nastane. Každá větev brány typu event musí začínat událostí.



Obrázek 4.3: Přehled BPMN gateways

4.1.3 Connecting objects

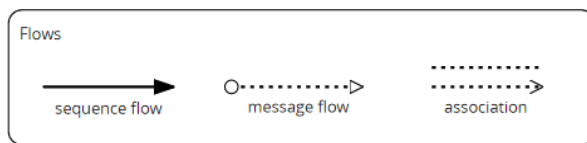
Spojovací objekty slouží pro definování toku procesu a spojují jednotlivé plovoucí objekty. Pokud toky plovoucích objektů všech větví v diagramu nekončí v koncovém stavu, diagram není validní.

Sequence flows – Znázorňují posloupnost aktivit, které jsou v rámci procesu vykonávány. Spojují jednotlivé plovoucí objekty v rámci jednoho bazénu.

Message flows – Slouží pro přenášení zpráv mezi účastníky. Zprávy znázorňují komunikaci mezi účastníky procesu. Message flows můžeme použít pouze pro spojení plovoucích objektů, které jsou v odlišných bazénech.

Associations – Spojuje informace (většinou textové) a artefakty s elementy procesu pro detailnější popis procesu.

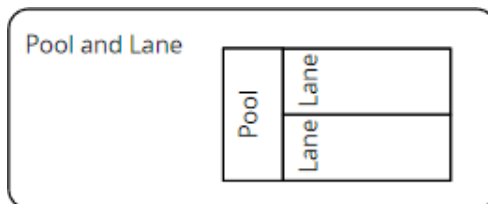
Symbolsy pro spojovací objekty jsou zobrazeny na obrázku 4.4.



Obrázek 4.4: Přehled BPMN spojovacích objektů

4.1.4 Pool a Swimlanes

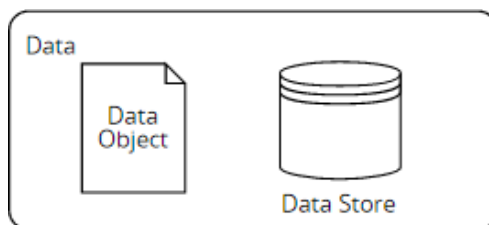
Plavecké dráhy mají za úkol zorganizovat elementy pro lepší přehlednost a rozdělení podle jejich příslušnosti k účastníkům projektu. Definujeme pool (bazén) a jeho lanes (dráhy) 4.5. Elementy jsou shlukovány do bazénů a plaveckých drah podle rolí a zodpovědnosti vůči jednotlivým aktivitám. Sekvenční toky nemohou překročit, na rozdíl od zpráv, hranice bazénu. Bazénem ve většině případů ohraničujeme jednotlivé organizace, v kterých probíhají jednotlivé procesy. Procesy jednotlivých bazénů můžeme navzájem ovlivňovat pouze zprávami.



Obrázek 4.5: BPMN bazén a plavecké dráhy

4.1.5 Data objects

Reprezentují vstupy a výstupy jednotlivých aktivit. Data objekty jsou používány aktivitami procesu, ve kterých mohou být vytvářeny nebo upravovány. Představují například papírovou nebo elektronickou verzi dokumentu. Také mohou znázorňovat datové úložiště 4.6.



Obrázek 4.6: Přehled BPMN data objektů

4.1.6 Artefacts

Artefakty jsou používány pro přidání další informace o procesu pomocí textové anotace. Artefakty jsou spojovány s grafickými elementy pomocí asociací.

Artefakty pouze přidávají doplňující vysvětlení k elementům BPMN a pokud je z modelu odstraníme, průběh procesu tím neovlivníme. S-BPM nemá elementy podobné artefaktům a proto je pro další použití nezohledňuji.

4.1.7 Využití BPMN elementů

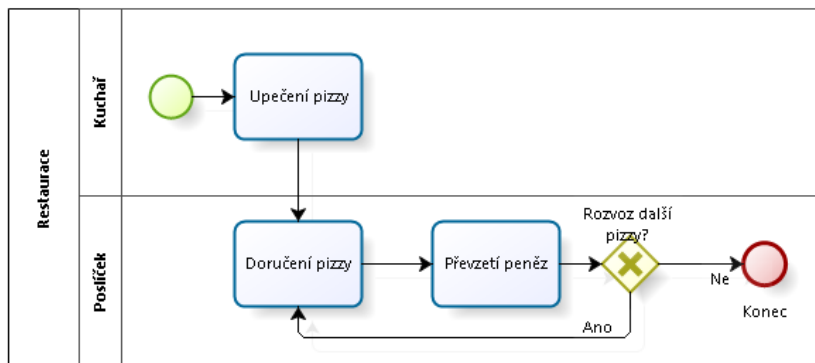
Rád bych v této části práce uvedl, že jsem popsal všechny existující BPMN elementy, ale realita je taková, že ne každý modelovací nástroj všechny tyto elementy využívá. BPMN symbolů je více jak 140 a často se setkáváme s tím, že modelovací nástroje využívají pouze určitou podmnožinu BPMN elementů, to však neznamená, že bychom v daném nástroji nedokázali popsat některý z problémů. Jak uvádím v úvodu kapitoly, BPMN není omezeno mnoha pravidly a jeden problém můžeme vyjádřit několika způsoby za pomoci odlišných elementů.

Důkazem je společnost StrICT Solutions, která vyvinula produkt InFlow [18]. Tento nástroj dokáže pomocí BPMN modelovat S-BPM procesy, ale využívá pouze 12 BPMN symbolů. Mým záměrem je však spíše převod již existujících modelů, které nejsou limitovány počtem symbolů.

4.1.8 Ukázka BPMN diagramu

Pro ukázkou použití notace je v diagramu 4.7 zobrazena část procesu pro objednání pizzy. V diagramu vidíme startovací událost a kuchař po startu procesu začne péct pizzu. Událost by později mohla být typu zprávy a kuchař by začal péct pizzu až v případě, že mu byla doručena objednávka. Bazén restaurace je rozdělen a dvě dráhy kuchaře a poslíčka, ve kterých vidíme, kdo provádí jakou aktivitu. Dráha poslíčka také obsahuje exkluzivní bránu, kde se tok procesu rozhoduje, jestli skončí na základě podmínky, jestli poslíček má rozvést ještě další pizzu nebo ne. V následující části práce je zobrazen stejný proces s použitím notace S-BPM.

4.2. Subject-Oriented Business Process Management Notation (S-BPM)



Obrázek 4.7: Příklad BPMN diagramu

4.2 Subject-Oriented Business Process Management Notation (S-BPM)

Při modelování pomocí S-BPM se upřednostňuje subjekt. Notace se snaží být co nejjednodušší, aby byla snadno pochopitelná i pro lidi z netechnických oborů. Proto S-BPM používá pouze pět grafických symbolů a snaží se přiblížit přirozenému jazyku.

4.2.1 Postup při modelování

A. Fleischmann v knize S-BPM One [4] definuje posloupnost jednotlivých kroků při modelování v S-BPM.

Postup při modelování:

1. identifikace byznys procesu a vytvoření specifické sítě procesu
2. identifikace subjektů v procesu
3. identifikace výměny zpráv mezi subjekty
4. identifikace metadat zpráv
5. určení vnitřního chování subjektů
6. vložit model do kontextu příslušného prostředí (většinou organizace)

Při modelování nám tedy vždy vzniká několik diagramů. Jeden diagram komunikace mezi subjekty (Subject Interaction diagram - SID) a poté modely vnitřního chování (Subject Behavior Diagram - SBD) pro každý subjekt. V případě že subjekt je externí, model vnitřního chování samozřejmě nemá.

I z definovaného postupu modelování vidíme jaký důraz S-BPM klade na interakci subjektů, body 2-4 se přímo týkají tvorby diagramu SID.

S-BPM používá pouze několik málo symbolů pro tvorbu modelu, které ještě dále rozdělujeme na symboly, které používáme v SID a na ty které v SBD.

Subjekt – Aktér v procesu má model vnitřního chování a komunikuje s ostatními subjekty. Subjekt může reprezentovat jak osobu, tak stroj s definovaným chováním. Provádí jednotlivé akce a interaguje s ostatními subjekty pomocí zpráv. Subjekt může být externí nebo interní. U interního subjektu musí vždy být definován model vnitřního chování. Externí subjekt se nachází v procesu, ale jeho vnitřní chování neznáme.

Stavy – Rozlišujeme dva druhy stavů, komunikační stavy pro interakci se ostatními subjekty, kterými jsou stavy typu „send“ a „recive“. V textu práce se držím anglického označení recive a send, protože do češtiny přeložené přijímací a odesílací stavy mi přijdou úsměvné.

Dalším druhem stavu je funkční stav, pomocí kterého aktéři provádějí jednotlivé úkoly. Stavy jsou využívány pouze v diagramech vnitřního chování subjektů. Jednotlivé stavy jsou spojeny přechody stejně jak tomu je v abstraktním stavovém automatu.

Z povahy stavů vyplývají následující pravidla:

Recive stav – Pro přijetí zprávy může mít jeden stav více odchozích přechodů v závislosti na obsahu zprávy definované v diagramu komunikace. Odesílatel nemusí být totožný a jedním stavem můžeme přijmout zprávy od odlišných subjektů. Odchozí přechod může vést do jakéhokoli typu stavu. Recive stav může mít libovolný počet příchozích přechodů.

Send stav – Má právě jeden odchozí přechod, který obsahuje zprávu a adresáta, kteří jsou definováni v diagramu komunikace. Může mít libovolný počet příchozích přechodů.

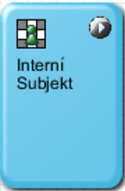
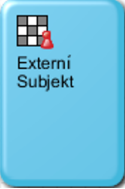
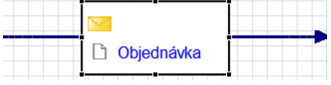
Funkční stav – Může mít libovolný počet odchozích i příchozích přechodů. V případě že má více příchozích respektive odchozích přechodů, slučuje respektive dělí tok procesu.

Dále pro modelování také definujeme:

Zpráva – Subjekty komunikují pomocí zpráv, jejichž komunikace je zachycena v diagramu interakce subjektů.

Byznys objekt – Může být poslán mezi subjekty společně se zprávou. Objekt je ucelený balík informací, který může být v průběhu procesu modifikován a využívám subjekty, příkladem může být jakýkoliv dokument. Objekty nemění chování procesu, pouze přidávají informační hodnotu.

4.2. Subject-Oriented Business Process Management Notation (S-BPM)

	Interní subjekt
	Externí subjekt
	Zpráva s byznys objektem

Tabulka 4.1: Symboly pro SID

	Funkční stav
	Send stav
	Recive stav
	Počáteční stav
	Koncový stav

Tabulka 4.2: Symboly pro SBD

4.2.2 Interakce subjektů

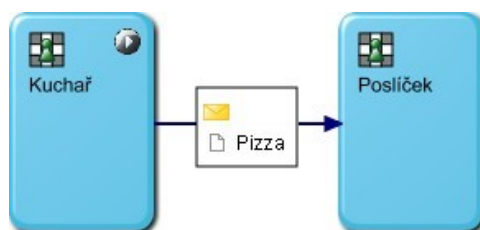
Každý proces obsahuje jeden diagram interakce mezi subjekty. Pro tvorbu diagramu interakce používáme pouze symboly pro subjekt a zprávu. Z pohledu na diagram by mělo být jasné, kdo se procesu účastní, jakou v něm má roli a kdo s kým interaguje.

4.2.3 Vnitřní chování subjektu

Každý diagram vnitřního chování subjektu je definovaný abstraktním konečným automatem, tedy musí mít počáteční a alespoň jeden koncový stav. Receive stav respektive send stav slouží pro přijímání respektive odesílání zpráv jiným subjektům procesu. Zprávy, jak jsme si už uvedli, mohou obsahovat byznys objekty. Funkční stav slouží pro zaznamenání jakékoliv vykonávané činnosti subjektem, kromě operací se zprávami. Objekty jsou zpracovávány právě v těchto funkčních stavech.

4.2.4 Ukázka S-BPM diagramů

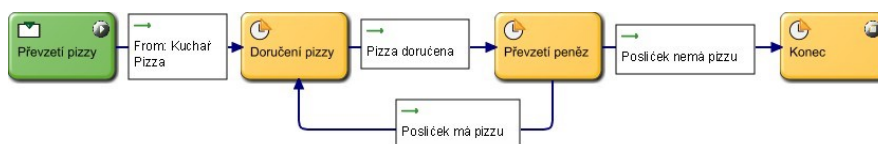
V části BPMN jsme si ukázali použití notace na příkladu části procesu pro objednání pizzy. Pro porovnání uvádím stejnou část proces v notaci S-BPM. Z jednoho diagramu BPMN vznikl jeden S-BPM diagram komunikace (SID) 4.8 a dva diagramy vnitřního chování subjektů (SBD) 4.9 a 4.10. Subjekty si mezi sebou posílají zprávu s byznys objektem „pizza“. V diagramech interního chování je vždy jeden počáteční stav a alespoň jeden stav koncový. V SBD poslíčka vidíme, že rozhodování zda proces ukončit je provedeno pomocí ohodnocených přechodů. A v případě, že poslíček už nemá žádnou pizzu, proces končí.



Obrázek 4.8: Příklad S-BPM SID



Obrázek 4.9: Příklad S-BPM SBD kuchaře



Obrázek 4.10: Příklad S-BPM SBD poslíčka

Srovnání notací BPMN 2.0 a S-BPM

Tato část práce vychází z článku Stephana Sneed [19], kde autor podle mého názoru zachytil nejdůležitější aspekty převodu notací, ale už detailněji nezkoumal další možné problémy s převodem spojené. Z tohoto důvodu se v práci snažím z tohoto článku vycházet a dále zkoumat a rozvíjet myšlenky a postupy v něm uvedené.

5.1 Porovnání notací

Na úvod této části bakalářské práce bych rád uvedl, že pokud začneme uvažovat o převodu musíme si uvědomit několik skutečností. Přístupy A-BPM a S-BPM jsou při vytváření modelů značně rozdílné. A-BPM nemá oproti metodice S-BPM mnoho pravidel. Také jejich notace se v mnohém odlišují, už jen z pohledu počtu symbolů. Proto nemůžeme uvažovat o převodu 1:1, kde bychom pro každý symbol definovali jeho ekvivalent v druhé notaci. Najdeme i výjimky, ale je jich málo.

Pokud chceme porovnávat jednotlivé notace, musíme začít od skutečnosti, že chceme převést stejné informace, ale do jiného grafického zápisu. Hledáme ekvivalentní zápis k jednotlivým entitám dané notace obsažené v diagramu, ale vždy s ohledem na zachování procesní logiky.

5.2 Stav S-BPM a jejich ekvivalenty v BPMN

Jako na první ze symbolů notace se zaměříme na stavy v S-BPM a aktivity v BPMN, protože to jsou symboly, které přenášejí informaci o vykonávaných činnostech v rámci procesu.

5.2.1 Funkční stav a úloha

Nositelem funkcionalit v S-BPM je vždy subjekt s pomoci stavů a interakcí mezi nimi. Proces je tvořen jedním a více subjekty, které spolu komunikují pomocí zpráv. Na rozdíl tomu funkcionalita v BPMN je reprezentována aktivitami. Z toho vyplývá, že aktivity v BPMN zastávají stejnou funkci jako stavy subjektů v S-BPM. Avšak některé aktivity se dále mohou dělit. Proto můžeme za ekvivalentní aktivity k stavům subjektů označit pouze úlohy.

Funkční stav můžeme dále specifikovat. Například přidáním informace, že v rámci stavu existují nějaké další pod-akce. Poté je tento stav považován za automatizovaný. V BPMN existuje automatizovaná úloha, která se nazývá service task, a proto jsou tyto dva elementy také ekvivalentní.

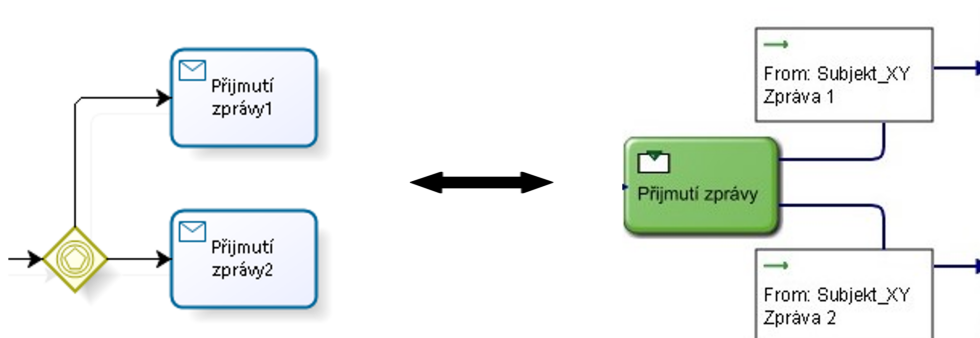
5.2.2 Komunikace mezi účastníky a subjekty

Komunikace mezi subjekty v S-BPM stejně jako mezi účastníky v BPMN probíhá pomocí zpráv. Výměnu zpráv v S-BPM umožňují dva typy stavů. Stav typu send a receive. Stav typu send nemůžeme přiřadit jiný element z BPMN než úlohu pro odeslání, protože oba elementy mají totožný význam pro odeslání zprávy. Také zpráva přenášená mezi subjekty v S-BPM je ekvivalentní k zprávě posílané mezi bazény v BPMN.

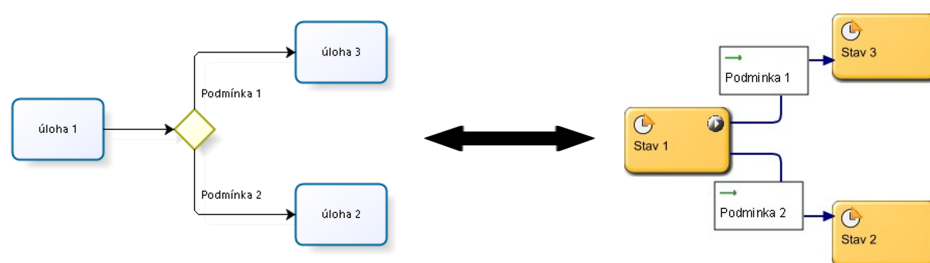
Mohlo by se zdát, že stav typu receive bude také ekvivalentní k receive úloze, opak je však pravdou a narážíme na následující problém. V S-BPM dokážeme jedním stavem typu receive přijmout více zpráv od více subjektů, avšak toto v BPMN není možné, protože každá receive úloha má právě jednu zprávu a jednoho odesílatele. Faktem je, že jeden receive stav může přijmout více zpráv, neznamená, že není deterministický, protože v S-BPM je každý odchozí přechod receive úlohy uskutečněn pouze tehdy, jestliže daná zpráva byla přijata. V BPMN nejsou zprávy spojeny s přechody, ale s receive úlohami. Pokud bychom chtěli v BPMN odesílat více zpráv, musíme před každou receive úlohu přidat bránu typu event. Situace je zobrazená na obrázku 5.1. Tato skutečnost je pro převod notací velice důležitá, protože v S-BPM se bez posílání a přijímání zpráv neobejdeme.

5.2.2.1 Externí subjekt

Když účastník BPMN komunikuje s externím systémem, u kterého neznáme jeho proces, je pro nás tento systém tzv. „black box“. V diagramu může být externí systém nebo účastník vyjádřen pomocí podprocesu nebo bazény, které nemají definovaný svůj proces. Do externích entit posíláme nějaký vstup a ten nám vrací nějaký výstup. Tuto situaci můžeme namapovat do S-BPM velice jednoduše pomocí externího subjektu, kterému pošleme zprávu a on nám jinou zprávu vrátí.



Obrázek 5.1: Recive úlohy a recive stavy



Obrázek 5.2: Sekvenčních toky a přechody

5.3 Sekvenční toky a přechody

Když jsme si porovnali symboly, které nesou funkcionalitu procesu, musíme také porovnat symboly, které určují jejich posloupnost v diagramu.

Dalším elementem, na který se v notacích zaměříme, je spojování uzlů v diagramu. Proces v obou notacích je v diagramu zachycen především pomocí zmíněných uzlů a hran. Uzly v S-BPM nazýváme stavy a v BPMN se jedná o úlohy, obecněji již všechny dříve definované plovoucí objekty. Spojujeme vždy právě dva uzly. V BPMN nazýváme hrany sekvenčními toky (sequence flows) a S-BPM přechody (transitions). V S-BPM jsou přechody vždy podmíněné a závisí vždy na stavech, ze kterých vychází a do kterého vchází.

Přechod se provede jen ve chvíli, kdy funkční stav vrátí nějakou hodnotu, která se rovná hodnotě přechodu, popřípadě je přijata nebo odeslána zpráva. V BPMN můžeme této vlastnosti docílit pomocí exkluzivních bran.

Zaměření se na podmínky větvení při překladu považují za velice důležité a zápis stejné funkcionality v jednotlivých notacích je znázorněn na obrázku 5.2.

5.4 Subjekty a plavecké dráhy

Pokud se podíváme na model jako celek, také najdeme mezi notacemi určité podobnosti. Dosud jsme se v rámci srovnání notací zaměřovali spíše na notaci, která je v S-BPM zanesena v diagramech interního chování subjektů. Nyní se blíže podíváme na srovnání diagramu komunikace subjektů a jeho podobnost v BPMN.

Pokud v BPMN modelu existuje více bazénů, komunikují spolu pomocí zpráv. Mohlo by nás to svádět k tomu, abychom bazén postavili ekvivalentní subjektu, který posílá zprávy jinému subjektu (bazénu). Ve skutečnosti jsou ale zprávy posílány mezi bazény, v diagramu přenášeny pomocí účastníků, kteří jsou v plaveckých drahách.

V BPMN jednotliví účastníci vykonávají svůj dílčí proces v rámci své plavecké dráhy a komunikace s ostatními účastníky procesu je znázorněna jako překročení hranice dráhy sekvenčním tokem. Zde vidíme podobnost s S-BPM, kde se na účastníky v plaveckých branách můžeme dívat jako na subjekty v S-BPM.

V S-BPM nedokážeme rozlišit, které subjekty patří do kterého bazénu a narážíme zde na úskalí, které se nám projeví při převodu.

5.4.1 BPMN události

Protože BPMN obsahuje oproti S-BPM mnoho typů událostí, jako jsou startovací, koncové a intermediate, které navíc můžeme kombinovat se zprávami nebo jim nastavovat časovou prodlevu v podobě timeru. V S-BPM neexistují žádné podobné elementy, ale dokážeme zachovat kontext těchto symbolů a informaci, kterou přenášejí. Jako příklad můžeme uvést `message-start-event`, který znamená, že účastník procesu v BPMN pasivně čeká na žádost od jiného účastníka. Tato situace může být v S-BPM vyřešena tím, že počáteční stav subjektu bude `receive` stav, kde subjekt také pasivně čeká na žádost od jiného účastníka.

Elementy nejsou ekvivalentní pokud se podíváme na přijímání zpráv ze strany BPMN. Protože proces čekající na zprávu můžeme v BPMN zachytit jako `message-start-event` nebo jen jako `start-event`, po kterém následuje `receive` úloha. Zdánlivě se obě situace mohou jevit jako totožné, ale jak uvádí [20] v prvním případě proces začíná pasivně a v druhém aktivně. Tento fakt pro potřeby práce s notací není podstatný, ale pokud bychom chtěli proces implementovat, mohli bychom narazit na nekonzistenci modelů.

5.5 Tok procesu diagramem

Poslední skupinou, kterou jsme neporovnali, jsou elementy, které rozhodují o průchodu procesem. V BPMN se jedná o brány a v S-BPM je rozhodování umožněno na základě ohodnocených přechodů. Největší rozdíl, na který zde narážíme, je fakt, že v A-BPM se může tok procesu rozdělit na několik větví. Aktivity v těchto větvích se následně vykonávají nezávisle na sobě a současně. Tuto vlastnost umožňují paralelní brány.

V S-BPM není možné rozdělit tok procesem, protože diagram je znázorněn pomocí abstraktního stavového automatu. Tedy v každém stavu se tok procesu musí rozhodnout, kterou větví se vydá na základě hodnoty přechodu. A není možné, aby přešel dvěma a více přechody najednou. Z tohoto důvodu také v S-BPM není možné, aby z jednoho stavu vedly dva stejně ohodnocené přechody.

Inkluzivní brána také může rozdělit tok procesu do více větví s rozdílem, že větve se provést nemusí.

Pouze exkluzivní brána v BPMN je ekvivalentní při rozhodování toku procesem jako přechody v S-BPM. Brána typu event větví proces, kde v jednotlivých větvích čeká na událost. Událost, která přijde jako první určuje kudy tok procesu bude pokračovat.

5.6 Shrnutí kapitoly

V této kapitole práce jsme si ukázali rozdílné přístupy k zachycení procesů pomocí dvou notací BPMN a S-BPM. Porovnali jsem nejdůležitější elementy a zjistili jsme, že v mnoha případech jsou podobné. Ukázali jsme si, že z jednoho modelu BPMN nám vždy vzniknou minimálně dva diagramy S-BPM, jeden SID a pak několik SBD podle počtu subjektů. Přístupy pro tvorbu procesu jsou odlišné. A-BPM klade důraz na činnosti v procesu a S-BPM se zaměřuje na jeho subjekty, což se projevuje v digramech, ale mají i mnoho podobného.

Metodika převodu notací

V této části práce jsou diskutovány možná řešení problémů při převodu notací, které jsme si identifikovali v kapitole o srovnání notací. Závěrem této kapitoly je vytvořená metodika pro převod notací.

Cílem při převodu z jednoho diagramu do druhého je zachycení stejného procesu v jiné notaci. Při převodu se soustředíme na snížení možné ztráty informací, která je způsobena rozdílnou notací a přístupem pro tvorbu modelů.

6.1 Plavecké dráhy jako subjekty

Začneme z pohledu S-BPM v diagramu SID, který nám dává ucelený pohled na proces a komunikaci mezi subjekty.

Plavecké dráhy ohraničují v BPMN jednotlivé činnosti daného účastníka. Účastníky můžeme do S-BPM převést jako subjekty, kteří si posílají zprávy pokaždé, když sekvenční tok BPMN účastníka opustí danou dráhu.

Jak jsme si uváděli v předchozí kapitole, převodem účastníků na subjekty přijdeme o informaci, z kterého bazénu účastník pochází.

Nevýhoda S-BPM v této situaci je zjevná, protože nedokáže rozlišit jednotlivé organizace.

6.1.1 Zachování informace o bazénech

Pokud bychom chtěli zachovat v S-BPM rozlišení plaveckých drah a bazénů, můžeme si pomoci názvem subjektu. Například plavecká dráha s označením vývojář v bazénu s názvem IBM bude do S-BPM převeden jak subjekt s názvem „IBM_vývojář“. Tím docílíme toho, že při zpětném převodu do S-BPM bude možné zachovat počet bazénů a plaveckých drah. Pokud bychom převáděli z S-BPM model do BPMN a měli k dispozici pouze znalosti, které vyčteme z diagramů, máme dvě možnosti, jak postupovat. Můžeme každý proces subjektu umístit do jednotlivého bazénu, nebo všechny procesy subjektů umístit do jednoho bazénu s více plaveckými dráhami.

6.1.1.1 Možné řešení

Podle mého názoru je situace řešitelná s pomocí externích subjektů. Každá organizace má svůj diagram komunikace a s ostatními organizacemi komunikuje přes externí subjekt. Představme si dvě organizace *A* a *B*. Protože zprávy v S-BPM jsou definované typem zprávy a adresátem, stačí v každém diagramu SID jeden externí subjekt pro organizaci. Tedy SID diagram organizace *A* obsahuje externí subjekt organizace *B* a naopak. Poté externímu subjektu namapujeme odpovídající již existující proces druhé organizace.

Výsledkem jsou oddělené organizace a jejich diagramy. Jednotlivé subjekty z organizace *A* komunikují v rámci své organizace mezi sebou, jak bylo dříve definováno. A s organizací *B* komunikují přes externí subjekt a naopak. Reálně jsou zprávy přenášeny mezi subjekty bez ohledu na příslušnost k organizaci² a výsledek je podle mého mínění totožný.

Možná úskalí tohoto řešení vidím v nepřehlednosti a při větším počtu organizací ztrácím celkový pohled na proces.

V literatuře [4] [15] [11] jsou uvedeny pouze ukázky modelů, které nemají mnoho subjektů a nejsou pro mojí potřebu dostatečně rozsáhlé. A všechny subjekty jsou definovány pouze v jednom diagramu SID, proto jsem se této zavedené metodiky v průběhu překladač držel. Nemyslím si, že by neexistoval žádný větší model, ale pravděpodobně budou pouze v rámci komerční sféry, do kterých organizace zachycují své know-how, a proto nejsou volně dostupné. Jak dále uvádím, překládal jsem poměrně velký BPMN model aplikace závěrečných prací provozované na FIT ČVUT, kde jsem tento problém řešit nemusel, protože celý proces probíhá v rámci jedné organizace.

Tato problematika je možným předmětem zkoumání pro pokračování této bakalářské práce.

6.2 BPMN brány

BPMN brány přináší největší problém pro převod notací.

6.2.1 Paralelní brány

Paralelní brána rozděluje tok procesu do dvou a více větví. Při rozvětvení se aktivity vykonávají současně a nezávisle na sobě. Protože je S-BPM diagram vyjádřen pomocí abstraktního stavového automatu, tuto vlastnost neumožňuje. Z S-BPM stavu tok při průchodu procesem může využít pouze jeden přechod a vždy jsou v jeden moment vykonávány pouze činnosti v rámci jednoho stavu.

²V nástroji Metasonic Suite, který jsem pro modelování využíval, je nutné, pokud chceme posílat zprávy mimo proces diagramu SID, definovat zprávy globálně místo lokálně.

S. Sneed uvádí jako možné řešení přidání subjektu do procesu, který bude vykonávat paralelní větev procesu. Ale zároveň dodává, že toto řešení nemusí být ideální. Podle mého názoru a s ohledem na cíl práce by řešení s přidáváním subjektů v případě více větví paralelních bran znemožnilo čitelnost diagramu. Proto jsem zvolil odlišný způsob řešení.

Jako nejschůdnější řešení problému s paralelními bránami vidím v možnosti pomyslného rozdělení větví brány na podprocesy, které budeme v průběhu procesu vykonávat. Podprocesy je nutné umístit na vhodná místa v diagramu pro zachování logického smyslu procesu. V ukázkách převodu toto řešení bylo vždy použitelné s malou změnou procesu. Ale je možné, že pro některé procesy nebude možné toto řešení aplikovat. Tento způsob není však možné zautomatizovat a je zapotřebí jistá znalost daného procesu. Obecně tedy není možné převést paralelní brány z BPMN do S-BPM.

6.2.2 Inkluzivní brány

Inkluzivní brány reprezentují logickou hodnotu OR. Problém s převodem inkluzivních bran je podobný problému paralelních bran, protože v obou typech bran se tok procese rozděluje.

Větve následující po inkluzivní bráně se můžou, ale také nemusí provést. Vždy ale tok procesu musí projít alespoň jednou cestou. Toky větví se zase musí spojit v jeden tok a z toho vyplývá, že inkluzivní brány musí být v BPMN diagramu párové.

Problém s inkluzivními bránami jsem vyřešil podobně jako v předchozím případě s paralelními branami. Namísto větvení jsem dal jednotlivé stavy, které reprezentují BPMN aktivity, sekvenčně za sebou. Poté jsem z každého stavu, který byl obsažen v některé z větví brány, přidal přechod, díky kterému se stav nemusí provést. Avšak i toto řešení naráží na logický problém, a tím je fakt, že dovoluji, aby neproběhl ani jeden stav, což inkluzivní brána nedovoluje. Příklad řešení toho problému je znázorněn v diagramu vnitřního chování cestovní kanceláře na obrázku 7.9.

Výhodou převodu inkluzivních bran je fakt, že mnou navržené řešení je možné uskutečňovat strojově, protože díky párování bran dokážeme snadno určit, kdy se proces rozvětjuje a kdy spojuje. Také dokážeme snadno určit počet možných průchodů. Otázkou je, jestli v případě většího počtu větví by byl diagram stále přehledný a snadno čitelný.

6.2.3 Exkluzivní brány

Převést exkluzivní bránu do S-BPM je velmi jednoduché, protože v tomto typu brány se tok nerozvětjuje a průchod závisí na podmínce brány. Exkluzivní brána reflektuje přechody v S-BPM. Proto funkční stav ze kterého

vede více přechodů, převedeme na aktivitu, kterou následuje exkluzivní brána s podmínkou odpovídající hodnotám přechodů.

6.2.4 Brána typu event

Brány typu event také nerozdělují tok procesu. Ale na rozdíl od exkluzivní brány, tok procesu v bráně čeká na událost, podle které se rozhoduje, kterou větví bude pokračovat. Situaci si můžeme představit na příkladu, kdy proces čeká v bráně na zprávu a pokud nepřijde, po časovém intervalu se proces ukončí. Tuto situaci také není složité do S-BPM namapovat díky přechodům, protože přechod se uskuteční až po provedení dané činnosti.

6.3 Podproces

Společně s podprocesem narážíme na další problém při převodu diagramů. Dle [19] podproces můžeme vyjádřit jako externí subjekt, který následně do-
definujeme. V další části práce ukazují tento problém na příkladu a problém řeším převodem podprocesů na subjekty. Tento postup není ideální, ale podle mého názoru se jedná o nejvíce přijatelné řešení. Dalším možným řešením tohoto problému je „rozbalení“ podprocesů do diagramu, ale touto cestou můžeme získat velký a nepřehledný diagram a ochudit se tak o přehlednost pomocí SBD diagramů jednotlivých subjektů. Myslím si, že ideálním řešením pro tento problém je kombinace obou řešení, které navrhuji. Ale v tomto případě je zapotřebí jistá zkušenost modeláře, který určí, kdy je vhodné proces „rozbalit“ a kdy je vhodnější ho nahradit subjektem. Při zpětném převodu není možné určit, jestli byl daný subjekt podprocesem či nikoli. S. Sneed také na tento problém naráží a jak sám uvádí, toto téma je předmětem hlubšího zkoumání.

6.4 Ztráty informací při převodu

Stephan Sneed [19] popisuje entity, které není možné přeložit, a proto dochází k ztrátě informací o procesu. Sneed se především zabývá dopadem převodu diagramu na jeho implementaci, ale tímto aspektem převodu se v práci nezabývám a soustředím se pouze na notaci. Proto některé entity označuji jako za nepřevoditelné. Mým záměrem je však především převod notace z důvodu lepšího pochopení a zpřehlednění diagramu.

6.4.1 Ztráty při převodu z BPMN do S-BPM

Jednou z entit, kterou nedokážeme z BPMN převést, jsou paralelní brány, ve kterých se tok procesu rozděluje na více toků a aktivity se vykonávají souběžně, což není v S-BPM možné.

Podproces je dalším problémem při převodu. Podproces můžeme vyjádřit jako subjekt nebo externí subjekt, v závislosti jestli známe jeho proces nebo ne. Pokud pomocí názvu neoznačíme subjekt jako podproces, o informaci podprocesu přijdeme.

Další ztrátou informace jsou artefakty, které v S-BPM neexistují a není způsob, jak tuto informaci zachytit. Jak už jsem uváděl, artefakty nemění proces, pouze plní funkci slovního doplnění. Proto artefakty při převodu ignorujeme a spokojíme se s informacemi, které nám poskytuje samotný diagram.

Poslední skupinou elementů, které je obtížné přeložit, jsou časové události. Konkrétně se jedná o timer-start-událost a timer-end-událost. Tyto události je možné v S-BPM zachytit pomocí přidání stavu za počáteční stav, jehož přechodem bude časový interval. Timer-end-událost nemá v S-BPM ekvivalenci a není možné ji přeložit.

6.4.2 Ztráty při převodu z S-BPM do BPMN

Převod na druhou stranu je o poznání snazší, avšak také se zde objevují určité problémy.

V S-BPM je možné subjektům přiřadit role, což v A-BPM možné není. V S-BPM je vždy díky subjektu jasně určeno, kdo vykonává jakou úlohu. V BPMN je toho znázorněno pomocí plavečkových drah, ale ty mohou obsahovat aktivity, které vykonává někdo jiný. Tento fakt však nevidím jako ztrátu informace o procesu v pravém slova smyslu a opět má větší dopad na proces při jeho implementaci.

Při převodu z BPMN do S-BPM narážíme na mnohem více problémů. Za úvahu stojí možné řešení, kterým je odstranění zmíněných entit z diagramů ještě před jeho přeložením a jejich nahrazením za jiné dostupné entity. Informační hodnotu, kterou byznys proces přenáší, bychom nezměnili a převod by byl snazším. Otázkou je, jestli jsme schopni odstranit všechny „problémové entity“ a nahradit je jinými.

Navzdory tomu, že BPMN má mnohem více symbolů (přes 140) je zřejmé, že je možné vyjádřit většinu informací, stejně tak dobře pomocí S-BPM.

6.5 Výsledná metodika převodu notací

Shrnutím této kapitoly dostáváme výsledné metodiky pro obousměrný převod. Jak jsme si popsali, některé elementy není možné přeložit. A při jejich nahrazování je zapotřebí zkušenost modeláře pro zachování procesní logiky.

BPMN	S-BPM
úloha	stav
plavecká dráha	subjekt
service úloha	stav s pod-akcemi
podproces	(externí) subjekt
tok zprávy	zpráva
send úloha	send stav
recive úloha	recive stav
exkluzivní brána	stav s odchozími přechody

Tabulka 6.1: Obousměrný převod entit notací

6.6 Tabulka převodu elementů

V tabulce 6.1 jsou obsaženy pouze elementy, u kterých je možný obousměrný převod. Na levé straně je vždy element z BPMN a na straně pravé z S-BPM. Do tabulky jsem zanesl jen takové entity, u kterých je překlad vždy stejný a obousměrný. Vidíme, že tabulka je relativně malá, protože není možné ve všech případech překládat entity 1:1 a vždy záleží na kontextu diagramu.

6.6.1 Metodika pro převod z BPMN do S-BPM

Převod účastníků na subjekty – Prvním krokem pro převod je identifikace všech účastníků v procesu. Převodíme účastníky na subjekty a také nalezneme všechny zprávy. Získání zpráv z BPMN diagramu pro S-BPM diagram komunikace získáme následovně: Všechny message flows převedeme na S-BPM zprávy a za druhé, jak už jsem uváděl, zprávu získáme vždy, když sekvenční tok překročí hranici plavecké dráhy. Pokud je ke zprávě v BPMN připojen nějaký objekt, společně se zprávou posílanou mezi subjekty také posíláme byznys objekt. Když jsme získali všechny subjekty a zprávy, vytvoříme diagram komunikace.

Převod aktivit na stavy – Všechny aktivity, které nejsou typu recive nebo send, převedeme na funkční stavy. Aktivity typu send a recive jsme už převedli na zprávy v prvním kroku za pomocí message flows.

Převod gateways (bran) – Exkluzivní bránu převedeme na odchozí přechody ze stavu, který jsme převedli z aktivity, jejíž sekvenční tok vedl do brány. Pokud do brány vedlo více sekvenčních toků z více aktivit, přidáme stejné přechody stavům, které jsme získali převodem z aktivit.

Bránu typu event převedeme stejným způsobem a události za branou převedeme na stavy.

Větve inkluzivní brány seřadíme do jednoho toku a přidáme z každého stavu, který byl obsahem některé z větví, přechod. Přidaným přechodem dovolíme nevykonání činností v daném stavu.

Parelelní bránu není obecně možné převést.

6.6.2 Metodika pro převod z BPMN do S-BPMN

Převod subjektů na účastníky – Nejprve začneme převodem SID. Pokud z kontextu procesu nedokážeme určit příslušnost subjektů k organizaci, tak máme dvě možnosti. Všechny subjekty budou v jednom bazénu rozlišeny plaveckými dráhami, nebo každý subjekt bude mít svůj bazén. Externí subjekty převedeme na podproces do plavecké dráhy účastníka, s kterým komunikuje v rámci SID.

Převod zpráv – Komunikaci mezi subjekty převádíme v rámci SBD subjektů. Pokud jsou subjekty v odlišných bazénech, receive a send stavy převedeme na aktivity odesílání a přijímání zpráv. V případě, že jsme subjekty převedli do jednoho bazénu, receive a send stavy převedeme na manuální aktivity, které pojmenujeme předání a přijetí nebo podobným označením. Sekvenční tok těchto aktivit bude vždy překračovat hranici mezi dráhami. Jestliže jeden receive stav může přijímat více zpráv, do BPMN tuto situaci převedeme jako bránu typu event s větvemi, ve které budou aktivity pro přijetí zprávy. Situace je znázorněna na obrázku 5.1 v kapitole o porovnání notací. Poslední možností je fakt, že receive stav může být zároveň počátečním stavem. V této situaci převedeme stav do BPMN na úlohu typu receive, ale jen v případě, že komunikace probíhá mezi bazény. V opačném případě stav převedeme na aktivitu.

Převod funkčních stavů – Funkční stavy převedeme na aktivity. V případě že z některého stavu vede více přechodů, v BPMN vytvoříme exkluzivní bránu s podmínkami, které odráží dané přechody. Podobně jak tomu je na obrázku 5.2. Startovní BPMN událost získáme jako počáteční funkční stav z SBD počátečního subjektu. Pokud je zřejmé, že v tomto počátečním funkčním stavu probíhají nějaké činnosti, převedeme ho na aktivitu a před něj přidáme startovací událost. Koncovými stavy v SBD jsou vždy funkční stavy, které můžeme převést na koncové události pouze v případě, že proces končí. Jestliže celý proces v koncovém stavu nekončí, stav nepřevádíme.

Ukázky převodu diagramů

Práce je zaměřena především na převod mezi diagramy, proto v této kapitole uvádím převod několika menších diagramů jak z BPMN do S-BPM, tak i převod opačného směru. V druhé polovině kapitoly je také popsán převod části modelu aplikace závěrečných prací.

Příklady diagramů pro převod jsme vybrali společně s vedoucím práce. Pro převod jsme vybrali již existující modely, které byly zvalidovány a já tak nezanašels jejich tvorbou do následného převodu chyby. Dalším důvodem výběru již existujících řešení je snaha se oprostit od modelování BPMN diagramů pomocí S-BPM metodiky. To znamená, abych nevyužíval S-BPM postupy při modelování BPMN modelu a tak nezjednodušil jeho následný převod.

Jako příklady modelů jsme vybrali modely takové, aby obsahovaly co možná nejvíce problémových situací, které uvádím v kapitole s porovnáním notací 5. Výběr modelů jsme také vybírali z různých oborů, abych ukázal, že ne jen v technických oborech je možné jejich využití.

7.1 Modelovací nástroje

Pro zachycení procesů jsem využil volně dostupného modelovacího nástroje *Bizagi Modeler*, který slouží pro tvorbu BPMN diagramů. A pro tvorbu S-BPM modelů jsem použil nástroj *Metasonic Suite*.

Nástrojů pro tvorbu BPMN diagramů existuje nepřehledné množství a není těžké si vybrat nástroj podle preferencí. Oproti tomu nástrojů pro tvorbu S-BPM modelů je velmi málo a jako jediný vhodný se jevil již zmíněný nástroj od společnosti *Matasonic*, u kterého jsem navíc musel požádat o několika měsíční licenci pro studijní účely.

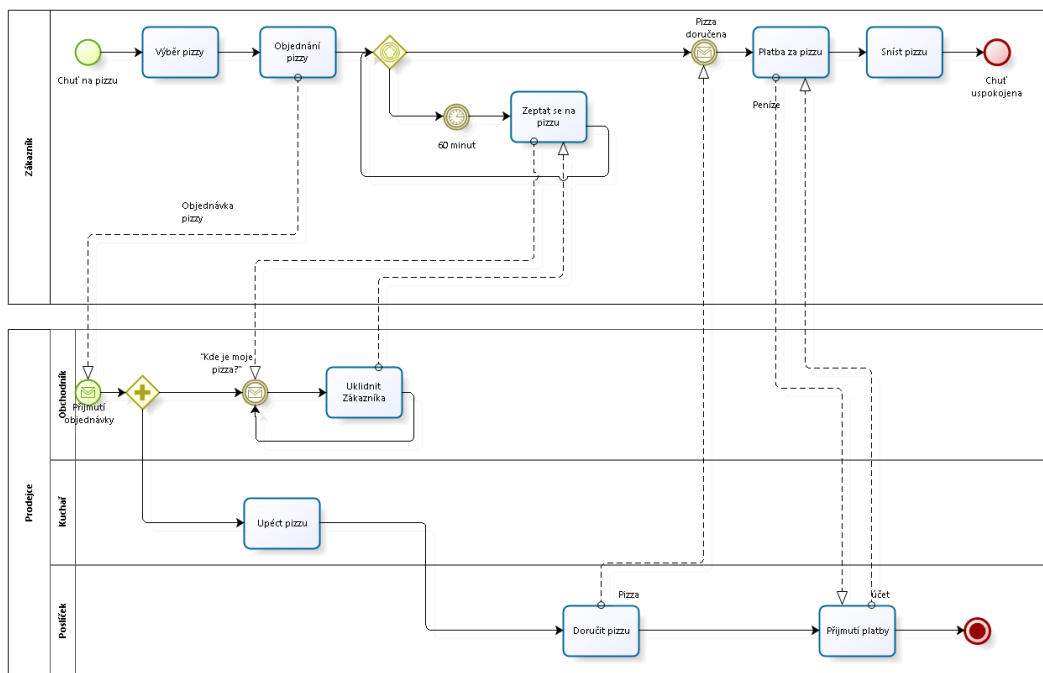
7.2 BPMN → S-BPM

V první části se podíváme na ukázky převodu náročnějšího směru a tím je převod z BPMN do S-BPM.

7.2.1 Proces objednání pizzy

Převod příkladu BPMN modelu objednání pizzy 7.1, který je uveden v dokumentu [7]. Dokument obsahuje příklady BPMN modelů a byl vytvořen společností Object Management Group, kterou pokládám za velmi důvěryhodný zdroj informací o A-BPM. Společnost mimo jiné definuje standart notace BPMN.

Model se může na první pohled zdát velmi jednoduchý, ale obsahuje několik situací, které je velmi obtížné převést do S-BPM.



Obrázek 7.1: Původní BPMN model objednání pizzy [7]

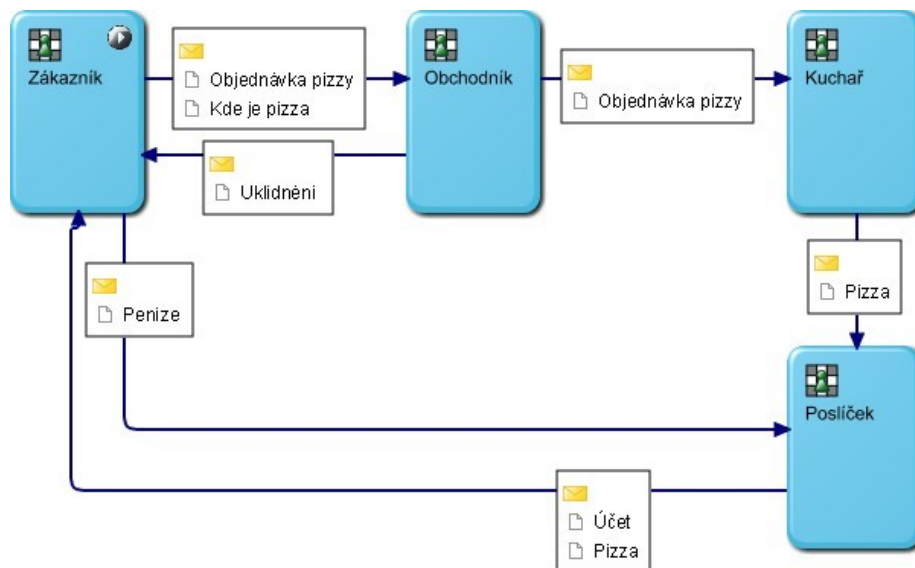
Diagram komunikace subjektů (SID)

Převod účastníků na subjekty a také nalezení všech zpráv není složité. Získání zpráv z BPMN modelu pro S-BPM diagram komunikace získáme následovně: Za prvé všechny message flows převedeme na S-BPM zprávy a za druhé, jak už jsem uváděl, zprávu získáme vždy, když sekvenční tok překročí hranici plavečkové dráhy. Společně se zprávami posílaných mezi subjekty také

posíláme byznys objekty. V tomto případě to může být objednávka, účet nebo dokonce pizza, což je naprosto korektní.

První ztrátou informace, o kterou jsme přišli z důvodu převodu, je fakt, že jsme nebyli schopni zachytit bazény a nyní z diagramu komunikace nepoznáme, který účastník patří do kterého bazénu. Bez znalostí procesu a jeho kontextu z SID nepoznáme, jestli původní model vůbec nějaké bazény obsahoval. S tím se pojí i další ztráta informace a tou je smazání rozdílu zpráv posílaných mezi bazény a zprávami posílanými v rámci bazénu mezi plaveckými dráhami.

Subjekty a definice zpráv, které obsahují informaci o typu zprávy, odesílateli a adresátovi máme definované. Tyto informace jsou dostatečné pro tvorbu diagramu komunikace (SID) 7.2.



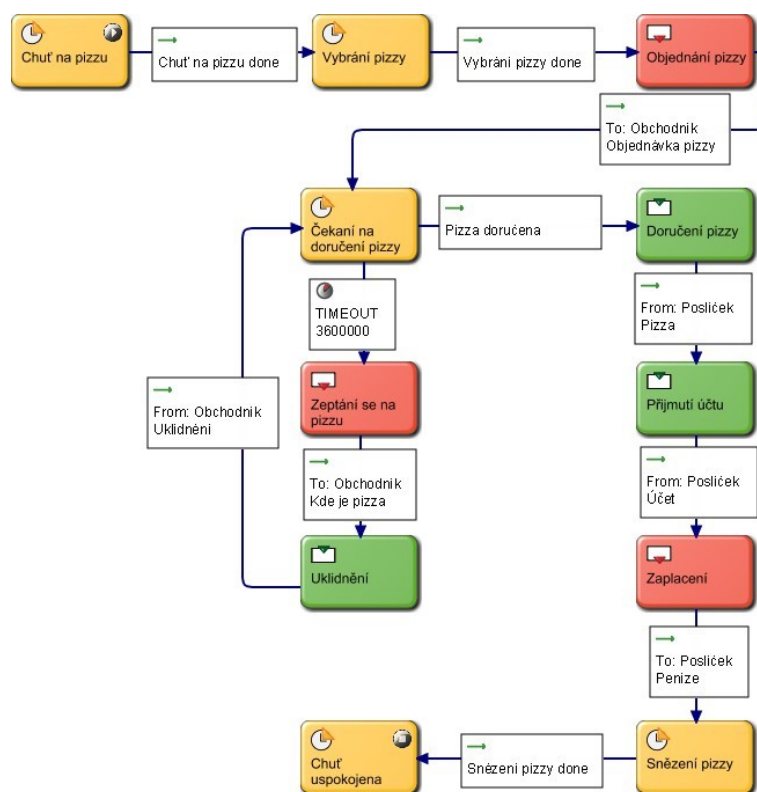
Obrázek 7.2: Diagram komunikace S-BPM procesu objednávky pizzy

SBD subjektů

Dalším krokem je definování diagramů vnitřního chování subjektů (SBD). SBD zákazníka 7.3 se od původní odpovídající části BPMN diagramu značně neliší. Brána typu event je převedena na stav, který má přechody ohodnoceny odpovídajícími BPMN událostmi. Diagram obsahuje více stavů z důvodu, že pro každé přijetí nebo odeslání zprávy je zapotřebí jeden stav.

Problém nastává u SBD obchodníka, který obsahuje paralelní bránu. Startovací událost je přeložena jako startovací stav. Protože S-BPM je založeno na abstraktním stavovém automatu, není možné, aby tok procesem byl rozdělen a vykonávám současně. Proto není v diagramu obchodníka možné zachytit

7. UKÁZKY PŘEVODU DIAGRAMŮ

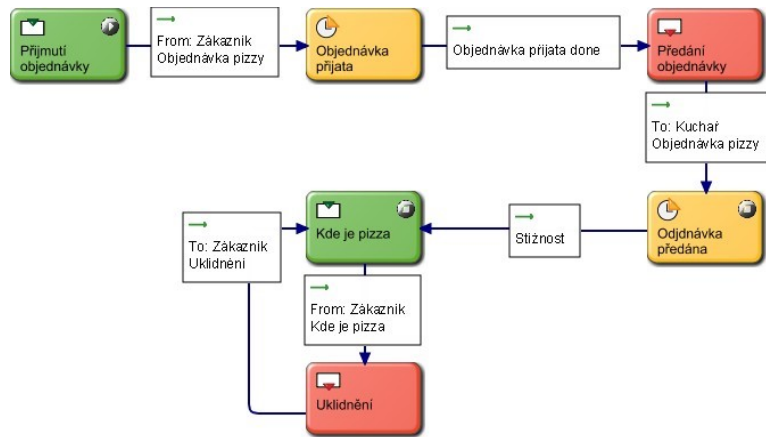


Obrázek 7.3: SBD subjektu zákazníka

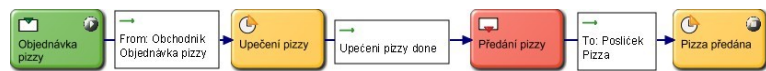
situaci, kdy zároveň předává objednávku kuchaři a zároveň pasivně čeká na možnou stížnost od zákazníka.

Proces z důvodu paralelní brány nebylo možné v současném stavu zachovat a musel být upraven. Diagram jsem několikrát modifikoval a hledal nejlepší řešení, výsledkem je SBD obchodníka 7.4. V SBD obchodníka řeším problém paralelní brány přidáním druhé větve až po skončení předání objednávky. Diagram obsahuje dva koncové stavy a tedy po předání objednávky kuchaři proces obchodníka končí v případě, že si zákazník nebude stěžovat. Logický rozdíl procesů je takovým, že v S-BPM si zákazník nemůže stěžovat do doby, než je objednávka předána. Uvědomuji si úskalí toho řešení, ale jevílo se mi jako nejvhodnější, protože je zachována smyčka pro více stížností.

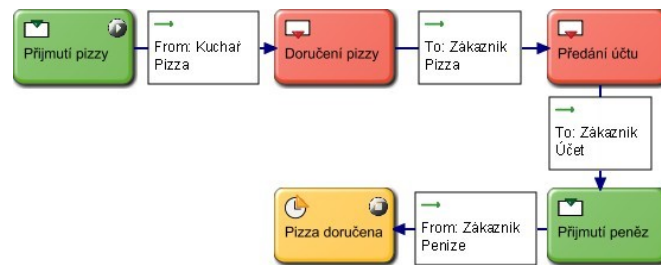
Dalším možným řešením, které jsem pro jeho neefektivnost zavrhl, je možnost přidání receive stavu a časovače do každého stavu, který následuje od přijetí objednávky. Tímto způsobem bych se po každém provedeném stavu „zeptal“, jestli od zákazníka nepřišla stížnost. Tím bych docílil stejné nezměněné funkcionality procesu, ale proces by trval déle a S-BPM diagram by se stal velmi nepřehledným. Důležité je si uvědomit, že zde neexistuje pouze



Obrázek 7.4: SBD subjektu obchodník



Obrázek 7.5: SBD subjektu kuchař



Obrázek 7.6: SBD subjektu poslíček

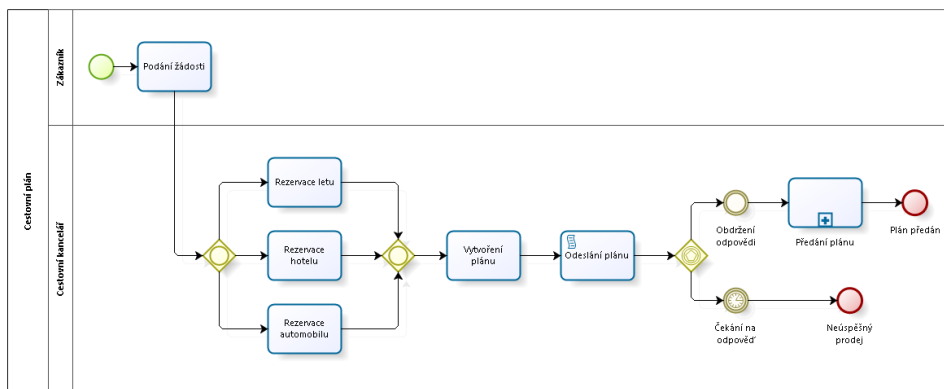
jedno správné řešení.

Získání diagramů vnitřního chování kuchaře 7.5 a poslíčka 7.6 nebylo obtížné, protože překlad spočíval pouze v převedení aktivit na subjekty a odesílání zpráv.

7.2.2 Proces pro vytvoření cestovního plánu

Dalším BPMN modelem, který jsem si pro převod vybral, je proces vytvoření cestovního plánu. Model uvádí jako příklad ve svém dokumentu s příklady [21] firma Bizagi, která také jako jedna z mnoha pomáhá výše zmíněné organizaci Object Management Group ve standartizaci BPMN notace.

Proces vyjadřuje zákazníka, který v cestovní kanceláři podá žádost s požadavky na cestu a ta žádost zpracuje a vytvoří cestovní plán.



Obrázek 7.7: BPMN diagram cestovního plánu

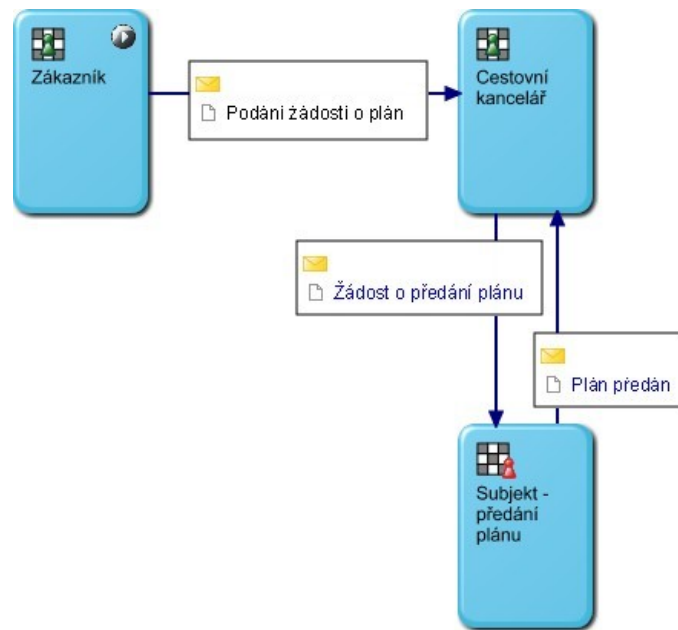
SID

V první části převodu se zase nejdříve zaměříme na vytvoření SID. BPMN diagram obsahuje dva účastníky, zákazníka a cestovní kancelář. Proto vytvoříme dva subjekty. Sekvenční tok překračující hranice plavečkových drah je v diagramu pouze jeden a proto mezi subjekty existuje pouze jedna zpráva. SID je znázorněn na obrázku 7.8. Diagram komunikace také obsahuje externí subjekt, jehož převod z BPMN diagramu je popsán níže.

SBD subjektů

BPMN diagram pro vytvoření cestovního plánu, oproti předchozímu modelu pro objednávku pizzy, neobsahuje žádné paralelní brány a mohlo by se zdát, že převod bude jednodušší. Diagram však v plavečkové dráze cestovní kanceláře obsahuje inkluzivní brány, které mají logickou hodnotu OR. Problém s inkluzivními bránami je podobný problému paralelních bran, protože v obou typech bran se tok procesu rozděluje.

Větvě následující po inkluzivní bráně se můžou, ale také nemusí provést. Vždy ale tok procesu musí projít alespoň jednou cestou. Tedy v případě našeho diagramu, který obsahuje tři větve, existuje sedm možností průchodu přes



Obrázek 7.8: S-BPM SID žádosti o cestovní plán

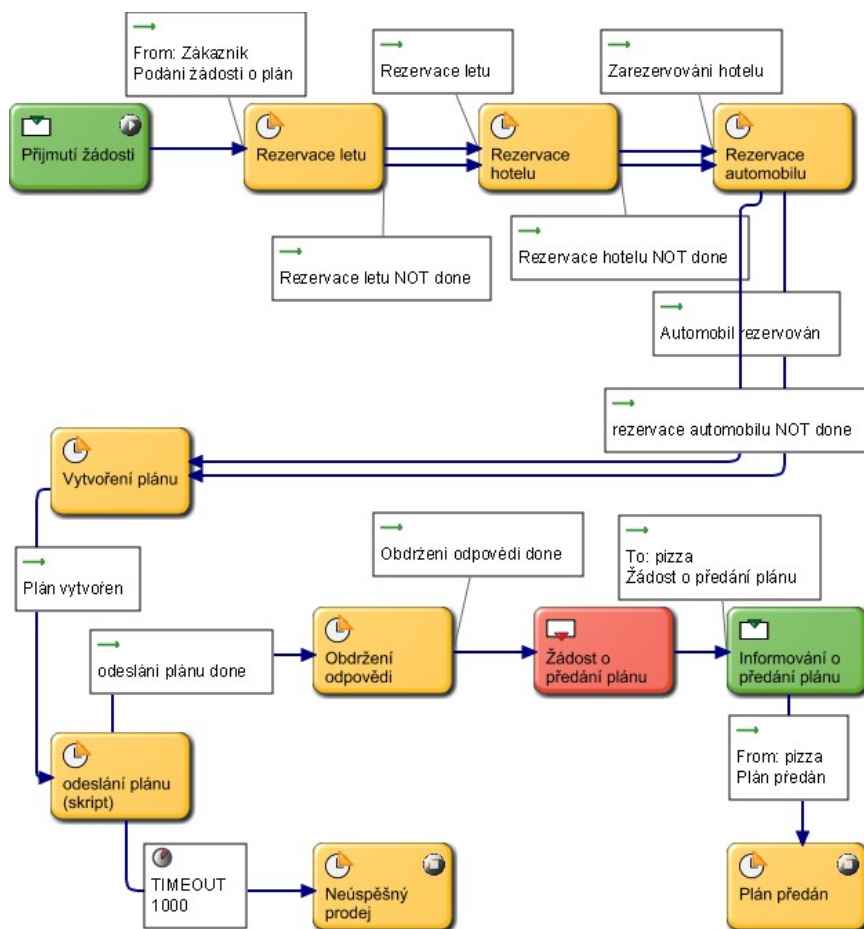
inkluzivní bránu. Toky větví se zase musí spojit v jeden tok a z toho vyplývá, že inkluzivní brány musí být v BPMN diagramu párové.

Problém s inkluzivními bránami jsem již vyřešil v části metodiky a jeho řešení je podobné jako v předchozím případě s paralelními bránami. Diagram vnitřního chování cestovní kanceláře je znázorněn na obrázku 7.9. Namísto větvení jsem dal jednotlivé stavy, které reprezentují BPMN aktivity, sekvenčně za sebou. Poté jsme z každého stavu, který byl obsažen v některé z větví brány, přidal přechod díky kterému se stav nemusí provést. Avšak tímto řešením dovoluji, aby neproběhl ani jeden stav, což inkluzivní brána neumožňuje. Ale i tento nedostatek by se mohl v tomto procesu odstranit. Například validací žádosti, ve které by musela být vyplněna alespoň jedna položka, ale tím už upravujeme proces. Na druhou stranu průchod BPMN diagramem v případě prázdné žádosti by nikdy neskončil, takže toto řešení je podle mého názoru korektní.

Převod podprocesu

BPMN diagram využívá podproces s názvem „Předání plánu“, ve kterém je zákazníkovi předán cestovní plán a jedná se o samostatný proces, který by byl v jiném diagramu. V tomto procesu však není definovaný a jedná se takzvaně o „black box“, tedy nám není známa jeho vnitřní chování a pouze mu zadáváme vstup a očekáváme nějaký výstup.

7. UKÁZKY PŘEVODU DIAGRAMŮ



Obrázek 7.9: S-BPM SBD subjektu cestovní kanceláře

Podproces přeložíme do S-BPM jednoduše jako subjekt. A protože se jedná o podproces, který nemá definovaný proces, přeložíme jej jako externí subjekt. Tím se vracím k diagramu komunikace 7.8, do kterého tento externí subjekt přidáme. Externí subjekt *předání plánu* komunikuje se subjektem *cestovní kancelář* a vyměňují si dvě zprávy, které jsem přeložil jako vstup a výstup z podprocesu. Ostatní entity už přeložíme obdobně jako v prvním příkladu.

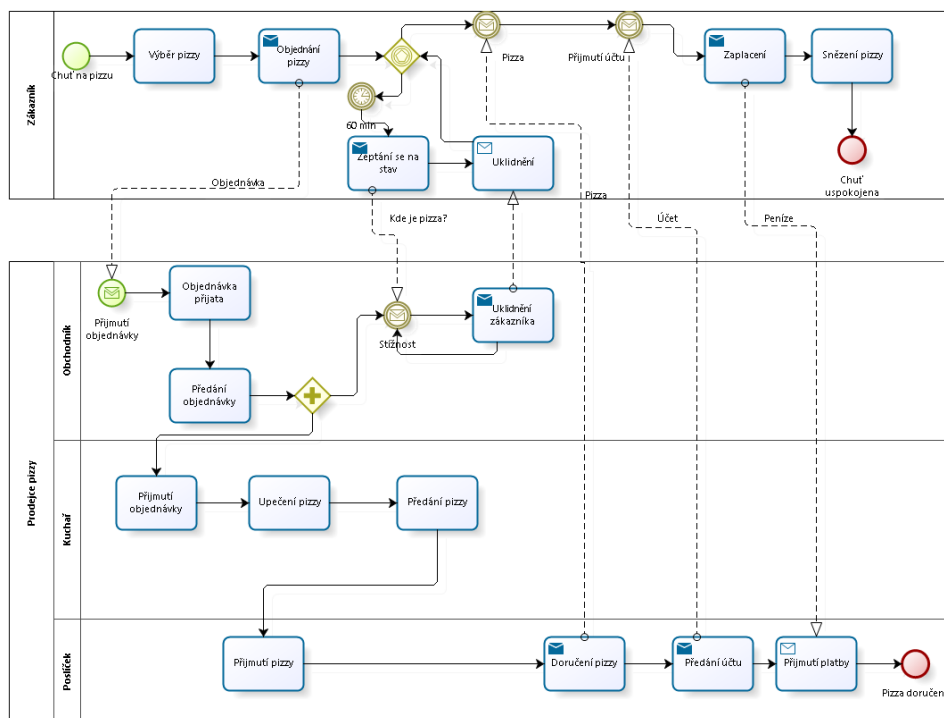
Záměrně v práci neuvádím S-BPM diagram vnitřního chování subjektu zákazníka, který v procesu pouze vyplní žádost a odešle ji, a proto není pro převod zajímavý.

7.3 S-BPM → BPMN

V této části práce se dostáváme k opačnému převodu notací, tedy z S-BPM do BPMN.

7.3.1 Převod procesu objednání pizzy zpět do BPMN diagramu

Model pro objednání pizzy jsem za pomoci vytvořené metodiky přeložil zpátky z S-BPM do BPMN 7.10. Tento model jsem přeložil až s odstupem několika týdnů, abych nebyl ovlivněn znalostí původního BPMN diagramu 7.1. Při porovnání dvou BPMN diagramů popisující totožný byznys proces můžeme vidět několik odlišností. V druhém BPMN diagramu se nachází více aktivit, především typu odeslání a přijetí, což je způsobeno S-BPM modelem, který klade důraz na komunikaci. A také ze skutečnosti, že pro každé překročení plavecké dráhy je v S-BPM zapotřebí zpráva.



Obrázek 7.10: Převedený BPMN model objednání pizzy z S-BPM modelu

V S-BPM rozhodování toku procesu při větvení závisí na podmíněných přechodech, které určují, do kterého stavu bude proces dále pokračovat. V tomto případě nemůžeme použít exkluzivní bránu, protože jeden z přechodů je ohodnocen časovým intervalem. Časový interval musíme do BPMN převést jako

událost. Proto výsledný diagram obsahuje bránu typu event.

Převod SBD diagramu obchodníka byl o poznání snazší než při převodu z BPMN do S-BPM. Z důvodu dvou koncových stavů, které se v diagramu objevují, jsem použil paralelní bránu, abych v BPMN docílil možného pokračování procesu po ukončení dané větve. Na tomto příkladu je vidět, že moje řešení s paralelními bránami má vliv i na převod zpět z S-BPM do BPMN. Pokud bychom se chtěli vyhnout paralelním drahám existuje další řešení. Druhou možností, jak umožnit převod koncového stavu, ve kterém proces nemusí končit, je přerušující událost typu cancel připojená k aktivitě. Tento typ události však není obsažen v nástroji Bizagi, proto jsem zvolil první řešení.

Zachování bazénů je způsobeno kontextem a jednoduchostí modelu, ze kterého je patrné, do kterého bazénu jednotlivý účastníci patří. Pokud bychom neměli informace o příslušnosti subjektu k organizaci, museli bychom umístit všechny účastníky do jednoho bazénu.

I na tak malém procesu vidíme, že překlad není jednoduchý. Závěrem z převodu procesu objednávky pizzy je fakt, že největším problémem při převodu jsou paralelní brány obsažené v BPMN diagramu.

7.3.2 Proces žádosti o služební cestu

Jako příklad S-BPM diagramu jsem si vybral proces žádosti o služební cestu z [14] od autora S-BPM Alberta Fleischmanna. Proces popisuje situaci, kdy zaměstnanec, který potřebuje odjet na služební cestu, musí mít schválení od svého vedoucího.

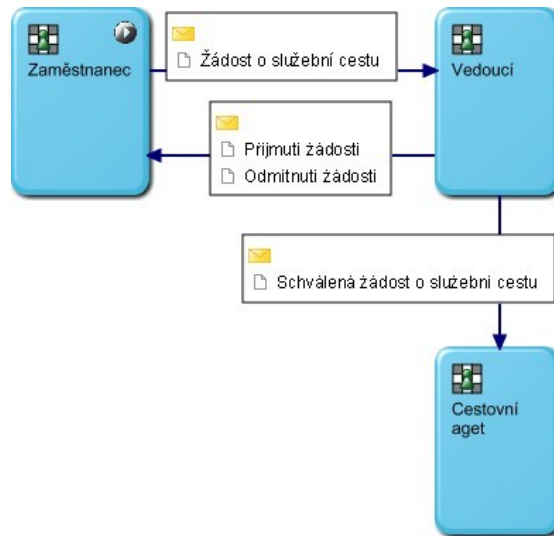
Proces jsme si již popsali v kapitole o S-BPM 3.2.1, kde jsme si ukázali možné identifikování subjektů, zpráv a vnitřního chování z přirozeného jazyka.

Slovní popsání procesu uvádím i v této části práce, protože je potřebný pro namodelování S-BPM diagramu, který budeme následně převádět. S-BPM model byl zvalidován A. Fleischmannem, ale pro názornost uvádím i postup jeho vytvoření.

„Zaměstnanec vyplní cestovní žádost. Zaměstnanec vyplní začátek a konec cesty a její důvod. Vedoucí zkontroluje žádost a informuje zaměstnance o jeho rozhodnutí. Cestovní žádost může být schválena nebo zamítnuta. Pokud je žádost schválena, je vedoucím odeslána cestovnímu agentovi, který žádost zpracuje.“

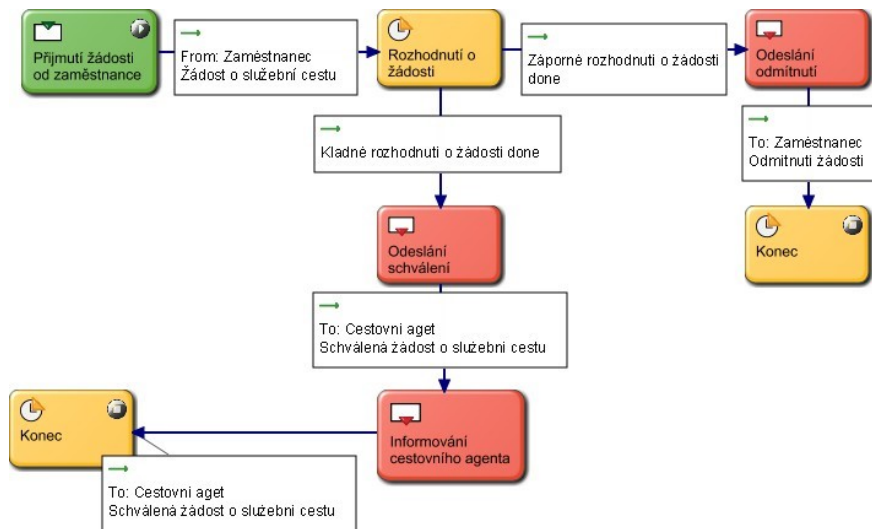
Z textu jsou jasně patrné subjekty a zprávy které si předávají. Jedná se o zaměstnance, vedoucího a cestovního agenta. Když jsme si identifikovali subjekty a zprávy, nic nám nebrání ve vytvoření SID 7.11.

Dále zaneseme do diagramů vnitřní chování subjektů. V SBD zaměstnanec 7.13 si můžeme povšimnout dříve zmiňované vlastnosti a tou je více přechodů z receive stavu „Přijmutí odpovědi“, který obsahuje dva typy zpráv. Proces se v tomto stavu větví v závislosti na typu zprávy.



Obrázek 7.11: SID cestovní žádosti

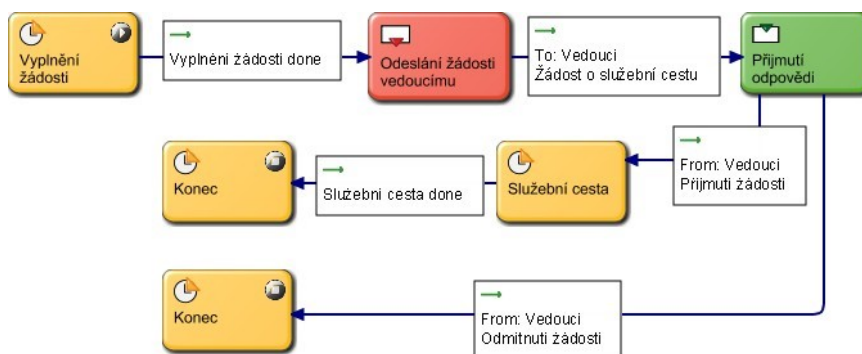
Podobná situace nastává v SBD Vedoucího 7.12, který o žádosti rozhoduje. Ze stavu „Rozhodnutí o žádosti“ vedou dva přechody, které jsou ohodnocené v závislosti na rozhodnutí o žádosti.



Obrázek 7.12: S-BPM SBD Vedoucího

Jediný problém, na který jsem narazil, bylo zdánlivé přesvědčení, že celý proces je jasně definovaný. Nejasnost na kterou tu narážím, je fakt, že z do-

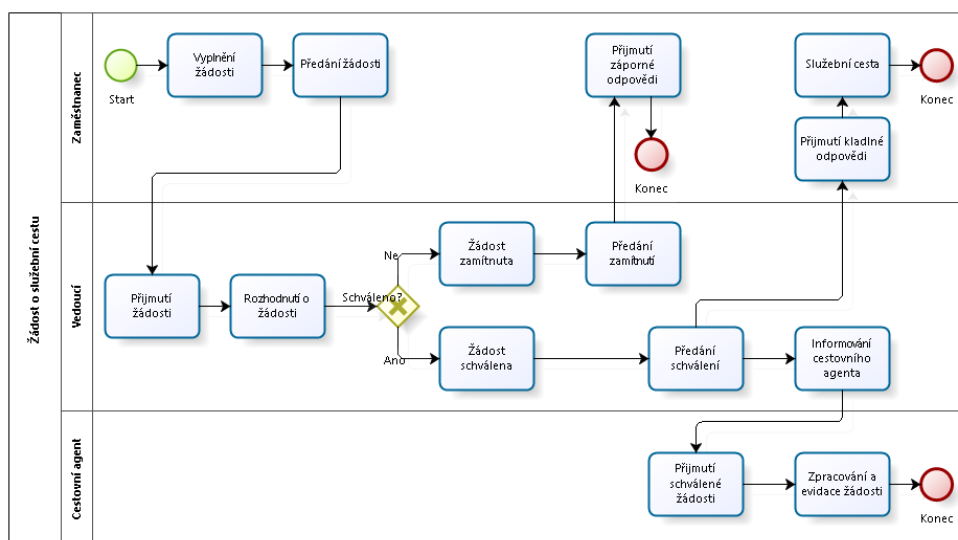
7. UKÁZKY PŘEVODU DIAGRAMŮ



Obrázek 7.13: S-BPM SBD Zaměstnanec

stupných informací nedokážeme určit, zda cestovní agent je členem organizace nebo není. Diagram jsem nechal ve stavu v jakém ho uvádí A. Fleischmann. Tedy cestovního agenta chápeme jako zaměstnance organizace, který má za úkol evidovat a zpracovávat služební cesty zaměstnanců. S jistotou můžeme subjekt označit za externí, pouze v případě, pokud nemá definované vnitřní chování.

Když jsme si namodelovali proces v S-BPM notaci, můžeme ho převést do BPMN diagramu.



Obrázek 7.14: BPMN diagram procesu cestovní žádosti

Subjekty jsme převedli na plavecké dráhy a umístili do jednoho bazénu. Pro zjednodušení převodu si nejprve převedeme SBD vedoucího. Jeho proces

tedy začíná přijmutím žádosti od zaměstnance. Nenechme se zmást, protože účastníci jsou v jednom bazénu, tak si reálně žádné zprávy posílat nemohou a komunikace probíhá přes sekvenční toky. Otázkou optimalizace by mohlo být odstranění těchto BPMN aktivit, které byly převedeny ze stavů typu send a receive. Protože aktivity přijmutí a předání jsou v jednom bazénu a některé ztrácí smysl.

Stav „Rozhodnutí o žádosti“ rozvětňuje proces a jedná se o funkční stav, který do BPMN snadno převedeme jako exkluzivní bránu. Proces vedoucího končí předáním odpovědi pomocí aktivit.

Při převodu SBD zaměstnance by se mohlo zdát, že je zapotřebí řešit situaci, kdy receive stav přijímá více zpráv. Ale v případě tohoto procesu se řešení s bránou typu event a více aktivitami vyhneme. To na první pohled nemusí být zřejmé, ale v SBD vedoucího se proces rozvětvil v závislosti na rozhodnutí o žádosti a jednotlivá rozhodnutí přijímáme až v těchto větvích.

Cestovní agent pouze přijme a zpracuje žádost. Výsledný BPMN diagram je znázorněn na obrázku 7.14.

Převod z S-BPM do BPMN je o poznání jednodušší, protože neobsahuje velké množství různých druhů elementů a vystačíme si pouze s několika symboly BPMN.

Počet stavů a entit nemusí odpovídat ani v tomto směru převodu. Příkladem jsou koncové stavy. Protože každý diagram vnitřního chování má nejméně jeden koncový stav, ale do diagramu BPMN jsou převedeny jen ty koncové stavy, ve kterých končí celý proces.

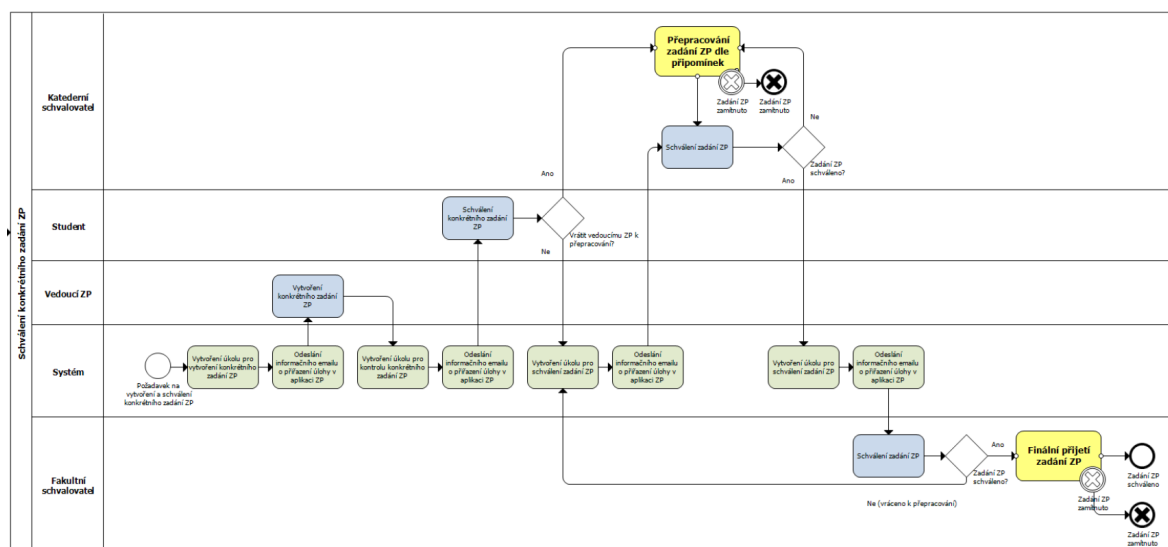
7.4 Překlad procesu schválení zadání závěrečné práce

Abych ukázal, že překlad notací můžeme použít v reálných situacích, přeložil jsem část modelu aplikace závěrečných prací, která se používá na FIT ČVUT a je provozována Centrem znalostního managementu. Pro překlad jsem si vybral část „Vytvoření rezervace a schválení zadání závěrečné práce“. Tuto část procesu shledávám jako nejvíce obtížnou na pochopení a participuje zde nejvíce účastníků. Celý procesní BPMN diagram aplikace je možno zobrazit na webových stránkách [22]. Ukázka části převáděného diagramu je na obrázku 7.15

7.4.1 Převod podprocesů

Převod nebyl jednoduchý a narazil jsem na několik problémů. Ihned na začátku překladu jsem se musel rozhodnout, jak vyřeším problém podprocesů. Vycházel jsem z dvou uváděných možností. Převést podproces na subjekt

7. UKÁZKY PŘEVODU DIAGRAMŮ



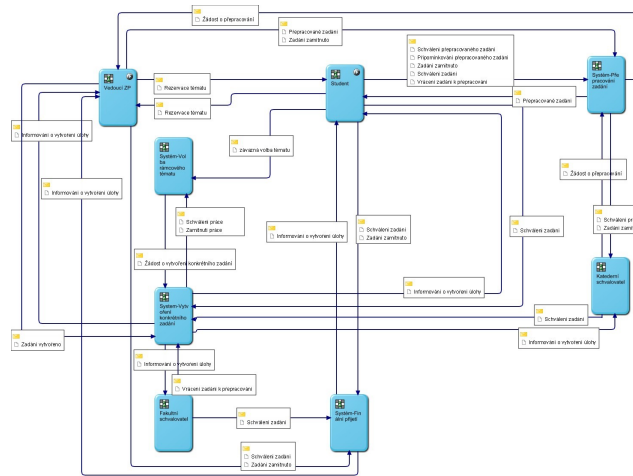
Obrázek 7.15: Ukázka BPMN diagramu aplikace závěrečných prací [22]

nebo podproces před překladem vložit přímo do diagramu namísto jeho symbolu. Po zvážení jsem došel k závěru, že „rozbalovat“ podprocesy by nebylo moudré. Rozbalením podprocesů bych docílil velké nepřehlednosti, protože některé podprocesy jsou dokonce do sebe vnořené. Tímto způsobem bych převodem vytvořil velké diagramy vnitřního chování subjektů a orientace v nich by byla velice náročná. Zároveň bych ochudil S-BPM model o jednu z jeho klíčových vlastností, kterou je snadná úprava procesů z důvodu rozdělení do logických celků podle subjektů.

Proto jsem zvolil odlišný způsob převodu, a tím je způsob, při kterém se na každý podproces díváme jako na samostatný subjekt, i když fyzicky se jedná pouze o jeden. Pro každý sekvenční tok BPMN diagramu, který spojuje aktivitu s podprocesem, vytvoříme v komunikačním diagramu S-BPM zprávu. V rámci co nejlepší přehlednosti procesu jsem při rozdělování subjektů zvolil notaci „*Jméno subjektu-Jméno podprocesu*“. Podle mého názoru je po dodržení této notace z diagramu zřejmé, kolik fyzických subjektů se procesu účastní. Touto metodou sice narostl diagram komunikace, který je možné najít v příloze B.2, ale velikosti diagramů vnitřního chování byly v poměru k diagramu BPMN při převodu zachovány. Pro názornost uvádím náhled SID i v této části na obrázku 7.16.

Všechny podprocesy se týkají subjektu systému, takže pouze jeden fyzický subjekt byl rozdělen na čtyři logické celky. Protože se jedná o systém, tedy automatizovanou činnost, rozdělení tohoto subjektu nebrání v snadném pochopení diagramu. Myslím si, že jsem zvolil správný postup překladu a SID splňuje svoji funkci. Každá zainteresovaná strana v procesu při pohledu na diagram snadno vidí jaké zprávy komu odesílá a jaké od koho přijímá a přitom

7.4. Překlad procesu schválení zadání závěrečné práce



Obrázek 7.16: Náhled SID Výběr a schválení tématu závěrečné práce

nemusí znát procesy ostatních. Je tomu stejně jako v realitě, kdy student nebo vedoucí práce nemusejí znát vnitřní chování systému, ale stačí vědět, kdy je od nich jaká zpráva požadována.

7.4.2 Převod účastníků

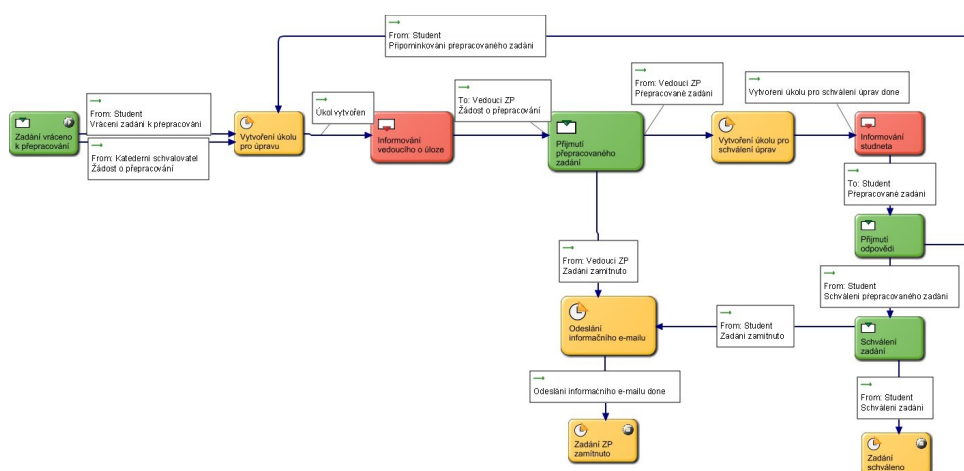
Při převodu se mi osvědčila již předem uvedená metodika. Nejdříve jsem se zaměřil na identifikování všech účastníků a zpráv, které si předávají. A bez hlubší znalosti chování subjektů jsem mohl vytvořit SID, z kterého je dostatečně patrný průběh procesu.

BPMN diagramy neobsahovaly žádné paralelní brány a při překladu jsem nenarazil na žádné další problémy, které jsem neřešil už dříve. Důvodem snadnějšího převodu je omezené použití BPMN symbolů. Asi nejzajímavější diagram vnitřního chování nalezneme u studenta, který je možné najít v příloze B.1. Diagram se zdá být velký, ale polovina jeho obsahu se týká přepracování zadání, které u většiny případů není potřeba a proces končí v koncovém stavu „Schvalování zadání“. V průběhu překladu tohoto procesu jsem postupoval podle vytvořené metodiky. A i přes velikost diagramu a složitost procesní logiky nebyl překlad složitý.

Zajímavým SBD diagramem je také proces subjektu „Systém-přepracování zadání“, znázorněném na obrázku 7.17. Proces začíná receive stavem přijmutí žádosti na přepracování zadání od studenta nebo katederního schvalovatele. Tedy předem není jasné, kdo změnu bude iniciovat. V BPMN diagramu je tato situace znázorněna dvěma sekvenčními toky, které vedou do podprocesu. Čitelnost obou notací v tomto konkrétním případě považuji za rovnocennou. V procesu subjektu jsou přijímány zprávy od téměř všech zainteresovaných

7. UKÁZKY PŘEVODU DIAGRAMŮ

osob, studenta, vedoucího práce i katederního schvalovatele. Zprávy jsou připojené k přechodům, které vedou ze stavů sloužících pro odeslání nebo přijetí zprávy. A právě v tomto SBD je možné vidět výhodu notace S-BPM, kterou shledávám jako obsah popisu zpráv. Popis zpráv v diagramu obsahuje informaci o tom, od koho je zpráva přijímána, nebo komu je posílána. Zároveň obsahují informaci o tom, co je typem neboli obsahem zprávy. A i z vnitřního chování subjektu vidíme informaci o tom, s kým subjekt komunikuje. V BPMN si musíme dávat pozor na pořadí plavečkových drah a také uspořádání sekvenčních toků, abychom zachovali dobrou přehlednost komunikace.



Obrázek 7.17: SBD subjektu Systém-přepracování zadání

Z poznatků při převodu jsem došel k závěru, že v současné fázi vývoje metodiky S-BPM je notace S-BPM spíše vhodnější pro menší typy procesů. Avšak je možné i modelování rozsáhlých procesů. Podle mého názoru S-BPM cílí na modelování menších procesů, kde je zapotřebí rychle reagovat na změny při tvorbě procesu. Avšak proces typu závěrečných prací není nutné často měnit a upravovat, protože před jeho tvorbou byla provedena důkladná analýza.

7.5 Shrnutí poznatků z ukázek převodů

Stejně tak jako je potřeba lidského faktoru pro tvorbu modelů, tak je zapotřebí při překladu notací. Obecně není možný strojový překlad diagramů, avšak kdybychom omezili notace, především tedy BPMN, na předem definovanou podmnožinu elementů, jsem přesvědčen o tom, že by automatizovaný překlad možný byl. Ale cílem mé práce byl spíše převod již existujících modelů, které nejsou omezeny. Avšak není pochyb o tom, že překlad z S-BPM do BPMN a obráceně je možné provádět bez velké ztráty informací.

Jak jsme si ukázali, největší ztrátou při převodu jsou paralelní brány, které v BPMN dovolují rozdělit tok procesu. Další znatelnou ztrátou při převodu je smazání rozdílu mezi bazény a plaveckými dráhami. Tyto ztráty se týkají převodu z BPMN do S-BPM. Pokud převádíme z S-BPM do BPMN v rámci notace jsme schopni převést všechny dostupné informace.

Každý z přístupů je vhodný k jinému účelu. Pomocí A-BPM jsme schopni popsat ty největší informační systémy a s S-BPM dokážeme rychle analyzovat a měnit malé až střední procesy.

Jeden ze scénářů, kde vidím smysluplné použití S-BPM je takový, že s pomocí této metodiky dokáží rychle namodelovat část procesu a případně ji rychle měnit v závislosti na změně dostupných informací. Například při diskusi se zákazníkem jsme schopni modelovat proces takzvaně „on the fly“, tedy v reálném čase. A jelikož notace není obtížná, nebude mít ani netechnický zákazník problémy s okamžitým pochopením diagramu a okamžitě bude schopný popsat jeho nedostatky a změny reflektující skutečnost. Následně jsme schopni diagram popisující danou část procesu převést pomocí vytvořené metodiky do BPMN, kde už jej můžeme dále dle potřeby upravovat.

Závěr

V této bakalářské práci bylo úkolem zmapovat možné řešení převodů notací BPMN a S-BPM. Ukázali jsme si rozdílné přístupy pro vytváření modelů obou notací. Zanalyzovali jsme a vzájemně porovnali notace jednotlivých přístupů. Nalezli jsme metodiku pro převod diagramů, kterou jsme dokázali použitelnost převodu v reálných projektech.

Převodli jsme několik diagramů od malých až po ty velké a zjistili jsme, že velikost diagramu nezávisí na složitosti jeho převodu. Došli jsme k závěru, že převod z BPMN do S-BPM je mnohem náročnější než převod obráceným směrem. Tato skutečnost je zapříčiněna především možností paralelního průchodu procesem v BPMN modelech. Identifikovali jsem elementy BPMN, které jsou obtížné na převod a našli jsme jim možné ekvivalenty, které nemají vliv na změnu procesu.

Za hlavní přínos práce pokládám porovnání notací a především dokázání možnosti jejich převoditelnosti. Další přínos shledávám ve faktu, že úspěšné zmapování byznys procesů je nezávislé na těchto dvou notacích. Na základě výsledků práce je S-BPM vhodnější pro mapování menších procesů, nebo částí velkých, především z důvodu dobré čitelnosti diagramů. Podle mého názoru, práce přinesla nový pohled na možnosti mapování procesů s důrazem na jednoduchost a rychlost modelování.

Na tuto práci je možné navázat zkoumáním možností převodu modelů s dopadem na implementaci procesů. Dalším pokračováním práce může být automatizace převodu diagramů s pomocí omezené množiny BPMN symbolů. V této práci je také možné pokračovat odlišným přístupem k problému, kterým je úprava BPMN diagramů před jejich převodem.

Stejně tak jako je potřeba lidského faktoru pro tvorbu modelů, tak je zapotřebí při převodu notací. Obecně není možné zautomatizovat překlad diagramů. Ale jsem přesvědčen o tom, že automatizovaný překlad je možný s omezenou notací BPMN. Avšak není pochyb o tom, že překlad z S-BPM do BPMN a obráceně je možné provádět bez velké ztráty informací.

Literatura

- [1] Weske, M.: *Business Process Management : Concepts, Languages, Architectures*. Berlin New York: Springer, první vydání, 2007, ISBN 3540735216, 368 s.
- [2] Jeston, J.; Nelis, J.: *Business process management*. Taylor and Francis, třetí vydání, 2014, ISBN 9780203081327, 689 s.
- [3] Braga, M.: A diagram is not a model: The huge difference between them. 2014. Dostupné z: https://www.ibm.com/developerworks/community/blogs/invisiblethread/entry/a_diagram_is_not_a_model_the_huge_difference_between_them?lang=en
- [4] Oppl, S.; Fleischmann, A. (editoři): *S-BPM ONE - Education and Industrial Developments, Communications in Computer and Information Science*, ročník 284. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, ISBN 978-3-642-29293-4, doi:10.1007/978-3-642-29294-1. Dostupné z: <http://www.springerlink.com/index/10.1007/978-3-642-29294-1>
- [5] Osterwalder, A.; Pigneur, Y.: *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons, 2013, ISBN 978-0470-87641-1, 288 s.
- [6] Silver Bruce: More on Executable BPMN 2.0. 2011. Dostupné z: <http://brsilver.com/more-on-executable-bpmn-2-0/>
- [7] Object Management Group: BPMN 2.0 by Example. 2010. Dostupné z: <http://www.omg.org/spec/BPMN/20100601/>
- [8] Weske Mathias, Weidlich Matthias, M. J.: *Business Process Modeling Notation*. Berlin: Springer, 2010, ISBN 978-3-642-16297-8, 128 s.
- [9] Fleischmann, A.: *S-BPM ONE – Setting the Stage for Subject-Oriented Business Process Management: First International Workshop, Karlsruhe*,

- Germany, October 22, 2009. Revised Selected Papers*, kapitola What Is S-BPM. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, ISBN 978-3-642-15915-2, s. 85–106, doi:10.1007/978-3-642-15915-2_7. Dostupné z: http://dx.doi.org/10.1007/978-3-642-15915-2_7
- [10] Fleischmann, A.; Schmidt, W.: Cognitive Support for S-BPM User Interfaces Intertwining Modeling and Execution. In *Proceedings of the 2014 European Conference on Cognitive Ergonomics - ECCE '14*, New York, New York, USA: ACM Press, sep 2014, ISBN 9781450328746, s. 1–4, doi:10.1145/2637248.2637278. Dostupné z: <http://80.dl.acm.org/dialog.cvut.cz/citation.cfm?id=2637248.2637278>
- [11] Fleischmann, A.; Raß, S.; Singer, R.: *S-BPM Illustrated*. Berlin, Heidelberg: Springer Berlin Heidelberg, první vydání, 2013, ISBN 978-3-642-36903-2, 143 s., doi:10.1007/978-3-642-36904-9. Dostupné z: <http://link.springer.com/10.1007/978-3-642-36904-9>
- [12] Sneed, S. H.: *Subject-Oriented Business Process Management: Second International Conference, S-BPM ONE 2010, Karlsruhe, Germany, October 14, 2010. Selected Papers*, kapitola Exporting. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ISBN 978-3-642-23135-3, s. 163–179, doi:10.1007/978-3-642-23135-3_9. Dostupné z: http://dx.doi.org/10.1007/978-3-642-23135-3_9
- [13] Kindermann, H.: Why S-BPM? 2014. Dostupné z: <https://www.metasonic.de/en/why-s-bpm>
- [14] Albert Fleischmann, I. U.; Werner Schmidt, T. H. I.: S-BPM – A NEW IMPETUS IN BUSINESS PROCESS MANAGEMENT. 2014. Dostupné z: <http://www.i2pm.net/publications/2014>
- [15] Fleischmann, A.; Schmidt, W.; Stary, C.; aj.: Subject-Oriented Business Process Management. nov 2012: str. 379. Dostupné z: <http://dl.acm.org/citation.cfm?id=2412065>
- [16] Camuda: BPMN Modeling Reference. 2014. Dostupné z: <https://camunda.org/bpmn/reference/#events-basic-concepts>
- [17] Lucid Software: Business Process Modeling and Notation. 2015. Dostupné z: <https://www.lucidchart.com/pages/bpmn>
- [18] Structured Information and Communication Technology: InFlow. 2015. Dostupné z: <http://www.strict-solutions.com/en/s-bpm-in-bpmn/>
- [19] Sneed, S.: Mapping Possibilities of S-BPM and BPMN 2.0. In *S-BPM ONE - Education and Industrial Developments*, editace A. Fleischmann;

- S. Oppl, Springer Berlin Heidelberg, 2012, ISBN 978-3-642-29294-1, s. 91–105. Dostupné z: http://link.springer.com/chapter/10.1007/978-3-642-29294-1_7
- [20] Object Management Group: Business Process Model and Notation. 2013. Dostupné z: <http://www.omg.org/spec/BPMN/2.0.2/>
- [21] Bizagi: BPMN by Example. 2014. Dostupné z: <http://www.bizagi.com/en/community/process-xchange>
- [22] Centrum znalostního managementu: Vytvoření, rezervace a schvalování zadání závěrečných prací. 2014. Dostupné z: <https://www.fel.cvut.cz/procesy/element.jsf?qprid=413295872&model=16713>
- [23] Object Management Group: BPMN 2.0 Poster. Dostupné z: <http://www.bpmn.org/#tabs-docs>

Seznam použitých zkratk

S-BPM Subject-Oriented Business Process Management

BPM Business Process Management

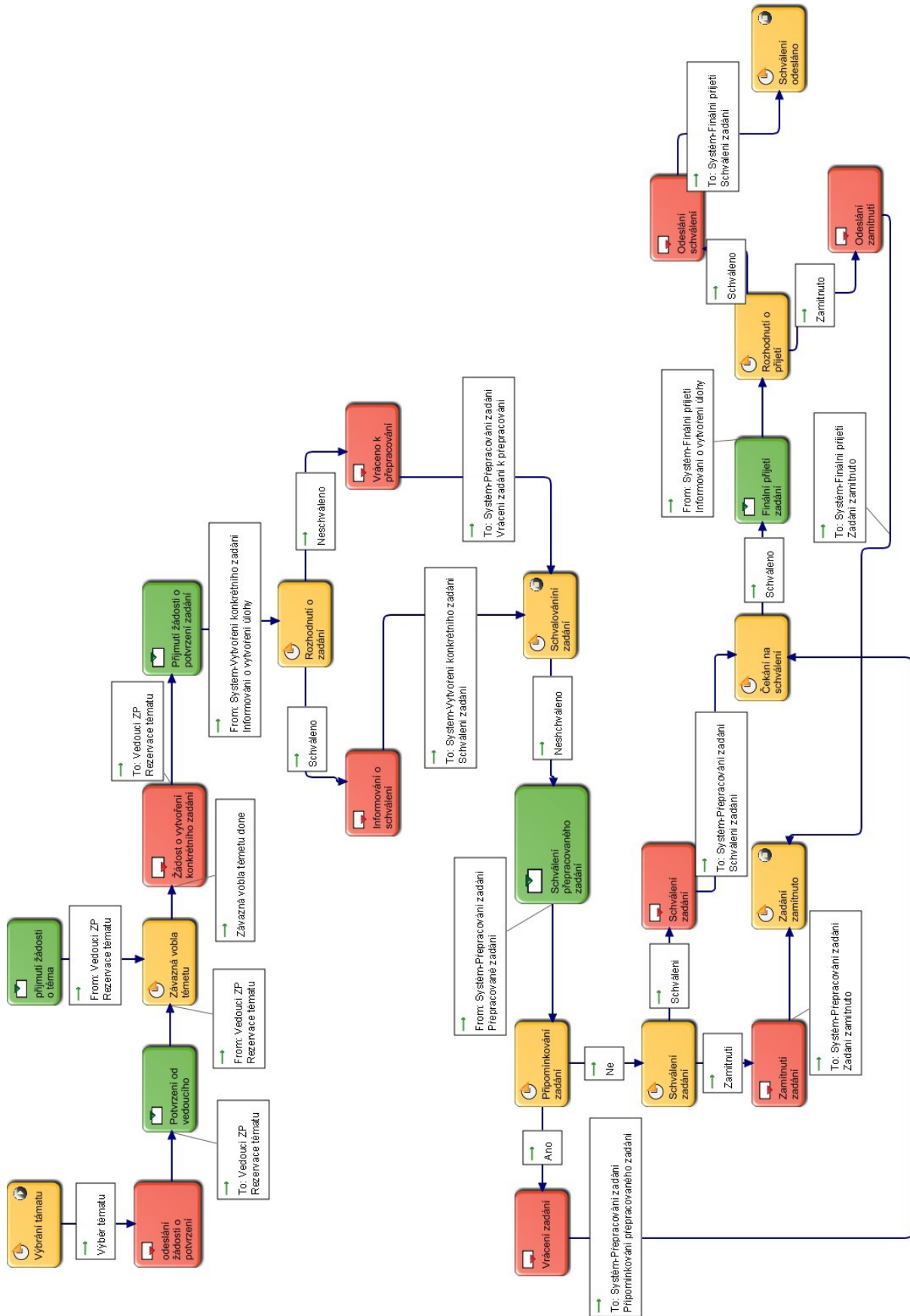
A-BPM Aktivita-Business Process Management

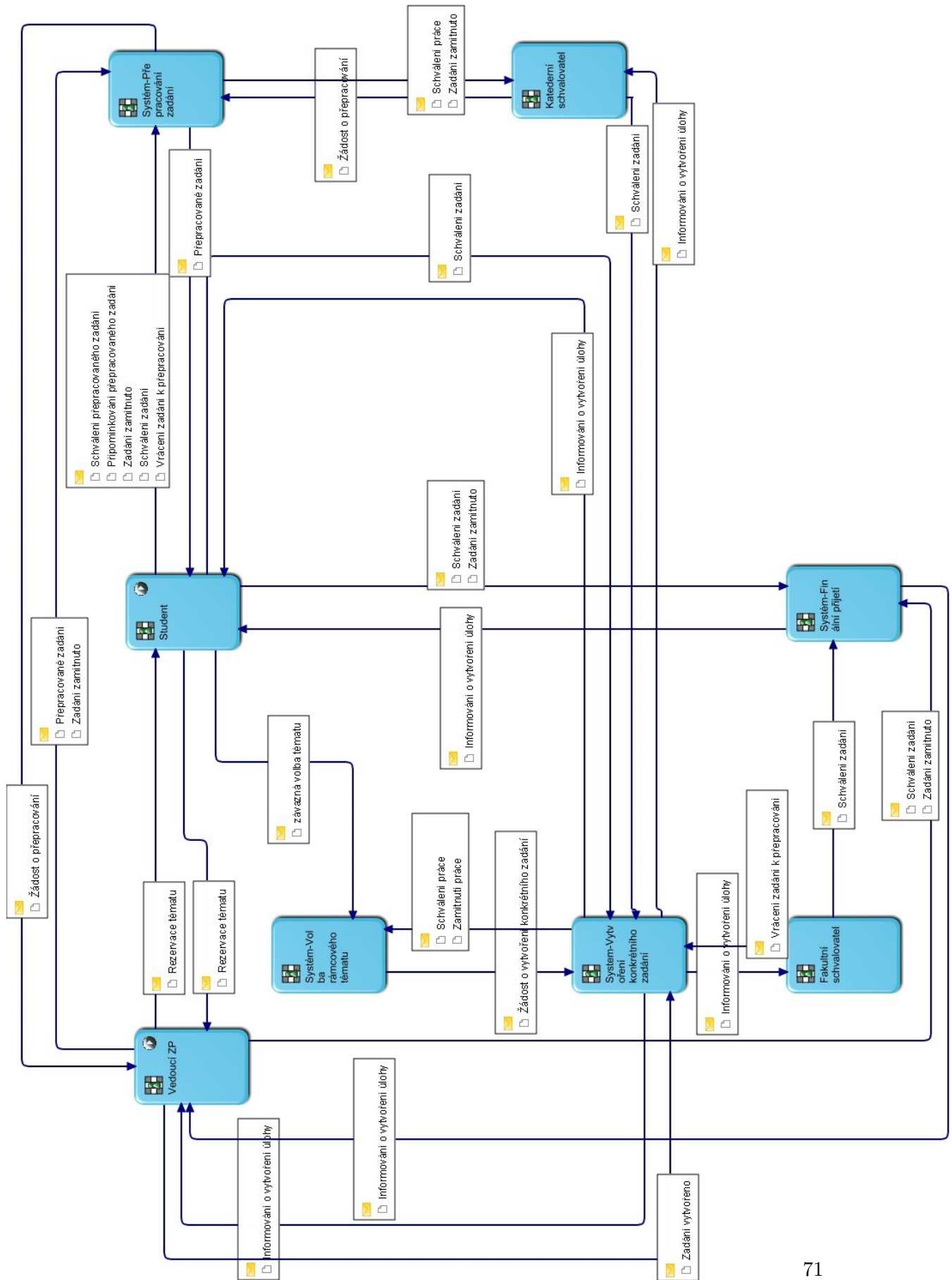
BPMN Business Process Management Notation

SID Subject Interaction Diagram

SBD Subject Behavior Diagram

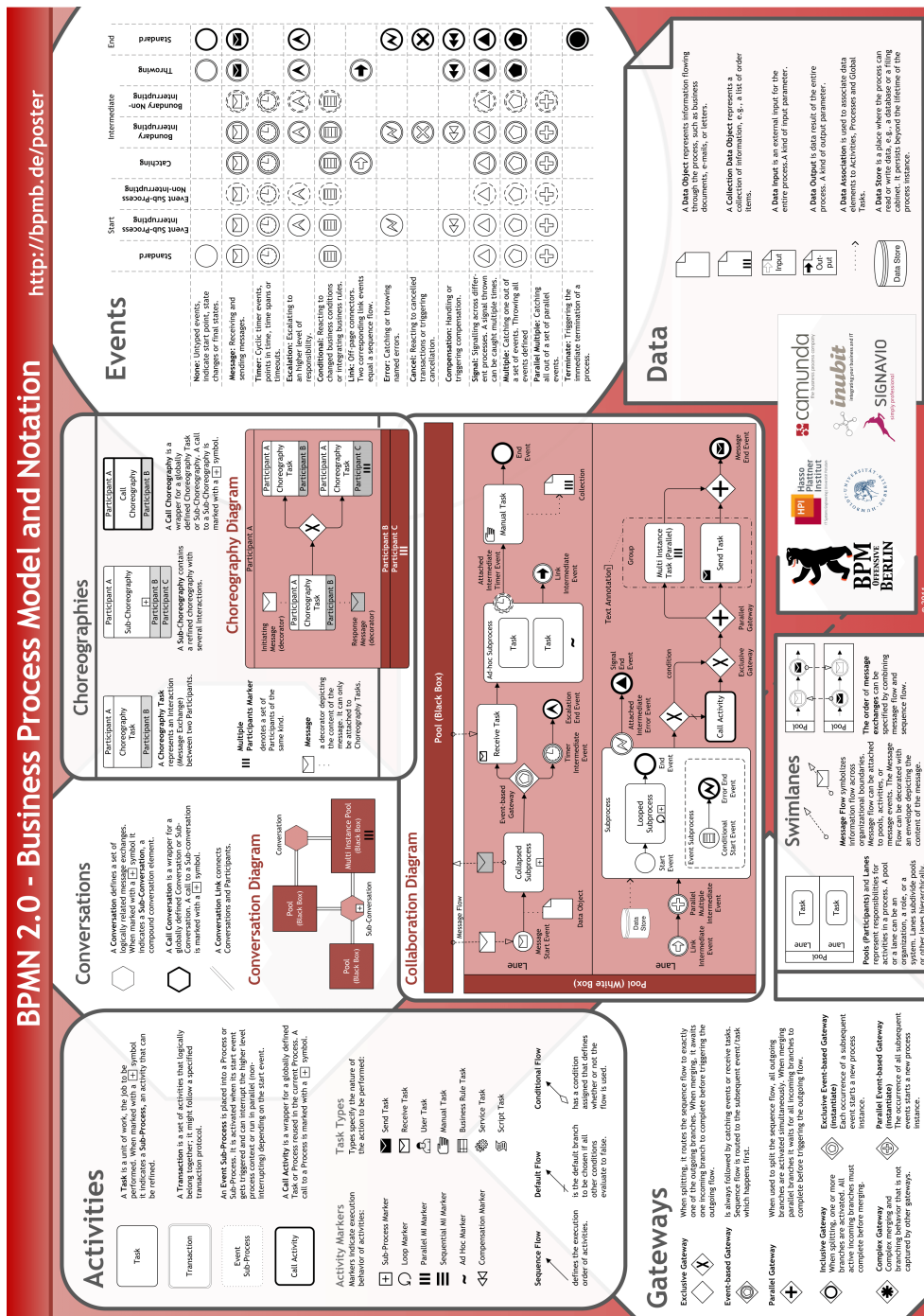
Diagramy a přehled BPMN





Obrázek B.2: S-BPM SID Výběr a schválení tématu závěrečné práce

B. DIAGRAMY A PŘEHLED BPMN



Obrázek B.3: Přehled symbolů BPMN [23]

Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
src	
├── BP_Mazura_Dominik_2016.tex....	zdrojová forma bakalářské práce ve formátu L ^A T _E X
├── mazurdom_bib.bib	soubor s bibliografickými informacemi
├── diagramy	
│ ├── BPMN	zdrojové soubory BPMN diagramů ve formátu BPM
│ ├── S-BPM ..	zdrojové soubory S-BPM diagramů ve formátu pro nástroj Metasonic Suite
│ └── Diagram_images	obrázky diagramů ve formátu png
text	text práce
├── BP_Mazura_Dominik_2016.pdf	text práce ve formátu PDF
└── BP_Mazura_Dominik_2016_Zadani.pdf .	text zadání práce ve formátu PDF