

# Kapitola 2: Techniky modelování podnikových procesů

Bc. Štěpán Heller

31. října 2015

## 1 Procesní model a důvody pro jeho tvorbu

Ať už člověk vytváří jakýkoliv model, jeho cílem je zachytit nějaký jev, který je potřeba kvůli své komplexnosti zobrazit zjednodušenou vizuální formou, která bude pochopitelná i pro jiné lidi než je sám tvůrce modelu. Umět jev zachytit ve formě modelu je jedním z prvních kroků na cestě k tomu tento jev upravovat.

Přeneseno do světa podnikových procesů je to velmi podobné. Jedním z hlavních důvodů, proč organizace přistupují k práci s BPM je potřeba procesy upravovat a zejména optimalizovat. Aby to bylo možné, je potřeba nejdříve stanovit metriky a tyto metriky být pak schopen měřit. Základem pro všechny tyto kroky je ale korektní procesní model, který proces věrně popisuje.

### 1.1 Definice procesního modelu

Základní definice procesního modelu podle

Procesní model je konceptualizací podnikového procesu v organizaci.  
<sup>1</sup>

Čtenářsky přístupnější definici pak nabízí [17]

Procesní model popisuje, většinou grafickou formou aktivity, události, jejich pořadí a propojení, které utváří podnikový proces.  
<sup>2</sup>

## 2 Základní techniky

V této sekci si popíšeme populární techniky pro tvorbu procesních modelů.

---

<sup>1</sup>Process model is a conceptualization of the (business) process in an enterprise. [8]

<sup>2</sup>Process model describe, typically in a graphical way, the activities, events and control flow logic that constitutes a business process.

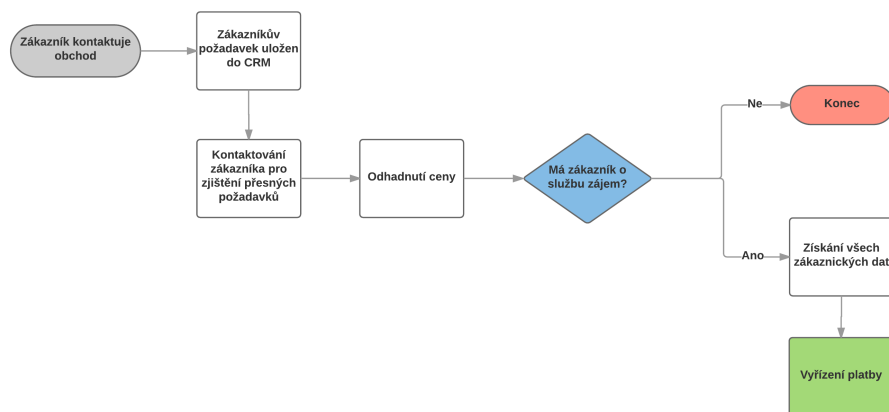
## 2.1 Vývojový diagram (flowchart)

Vývojový diagram je pravděpodobně nejpopulárnější technikou pro modelování podnikových procesů. Vděčí za to zejména své jednoduchosti, dostupnosti mnoha nástrojů, které tuto techniku podporují a také její velké srozumitelnosti, která jí činá velmi snadno uchopitelnou i pro uživatele v organizaci, kteří nejsou příliš obeznámeni s problematikou modelování podnikových procesů.

### 2.1.1 Základní pravidla

Vývojové diagramy se skládají z několika málo základních symbolů. Tyto symboly se nazývají: [7]

- Startovací a ukončovací symboly – používají se pro vyznačení začátku a konce procesu
- Šipky – zobrazují tzv. „řídící tok“, tedy přechod v čase mezi jednotlivými symboly
- Dílčí kroky procesu – jsou reprezentovány obdelníkem.
- Podprogramy – zobrazeny obdelníkem se svislými čarami po stranách. Používají se pro zobrazení skupiny kroků procesu pomocí jediného symbolu.
- Vstupy a výstupy – zobrazují tok informací směrem dovnitř i vně procesu, Pro jejich reprezentaci se používají lichoběžníky respektive rovnoběžníky.
- Podmíněný cyklus – zobrazuje událost opakující dokud je splněna jasně definovaná podmínka. Zobrazuje se pomocí šestiúhelníku.
- Podmíněný výraz – kosočtvercem je symbolizováno rozhodnutí a určuje tedy místo, kde dochází k větvení procesu.
- Spojovací symbol – inverzním symbolem ke kosočtverci je ve vývojovém diagramu kruh, který se používá ke spojení více toků do jednoho.



Obrázek 1: Nákupní proces pomocí vývojového diagramu

### 2.1.2 Výhody a nevýhody

Nespornou výhodou vývojových diagramů je právě jejich přístupnost pro uživatele a velmi strmá křivka učení, což dělá z této techniky první volbu pro případy, kdy je potřeba velmi rychle vymodelovat nějaký proces a organizace nemá zavedeny sofistikovanější metody BPM. Vývojové diagramy umožňují efektivnější komunikaci o problému v rámci týmu.

Největší přednost vývojových diagramů je zároveň jejich největší slabinou. Právě přílišná jednoduchost této techniky dělá z modelování komplexnějších procesů poměrně komplikovanou a nepřehlednou záležitostí. Ve vývojových diagramech je také složitější modelovat některé jevy, jako například tzv. „unhappy paths“ a další nestandardní události, která však v životě procesů nastávají poměrně běžně. U vývojových diagramů je také obtížné dělat změny, protože to často vyžaduje kompletní překreslení celého diagramu.

### 2.1.3 Použití

Vývojové diagramy mají mnoho využití. Hodí se například pro komunikaci mezi organizací a jejími externími zákazníky, protože se dá předpokládat, že se s vývojovými diagramy už v minulosti setkali a budou jim tedy rozumět. Vhodné je také použít vývojový diagram v dokumentaci k softwaru nebo jinému systému, kterou budou číst různorodé skupiny uživatelů.

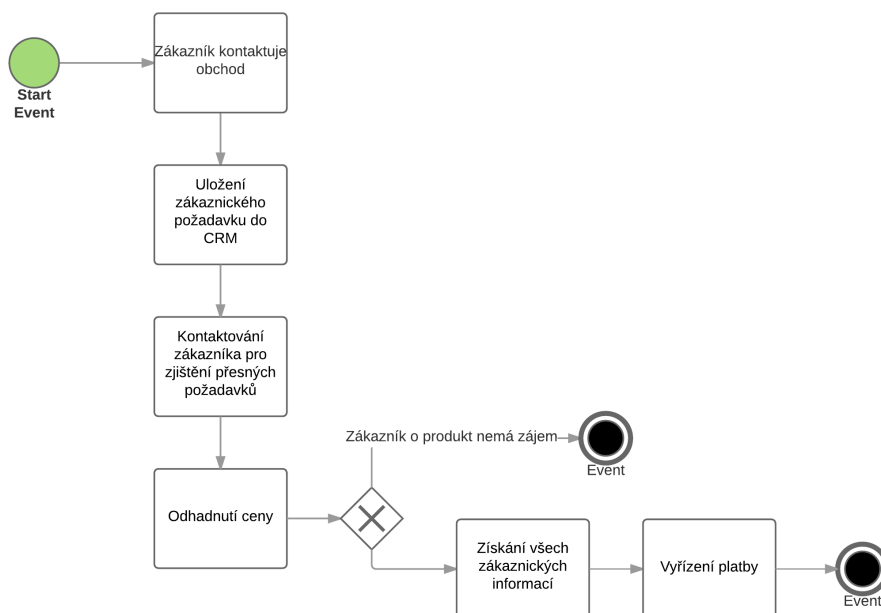
## 2.2 BPMN

BPMN nebo rozepsaně Business Process Modelling Notation je v současnosti de facto standardem na poli modelování podnikových procesů. Jeho využití je široké od IT přes obchod až například po komplexní dopravní systémy. Za svou popularitu vděčí zejména kombinaci dvou věcí, z nichž jedna je stále poměrně jednoduchá notace, která neobsahuje přehršel symbolů.

S trochou nadsázky by se dalo říct, že BPMN je vlastně rozšířením vývojového diagramu. Je určitě pravdou, že se BPMN touto jednoduchou technikou v mnohém inspirovalo a na jejích základech postavilo notaci, která umožňuje poměrně jednoduše modelovat i komplexní podnikové procesy a zároveň si stále uchovává dobrou srozumitelnost pro uživatele.

### 2.2.1 Základní pravidla

Jelikož se BPMN budeme podrobně věnovat v kapitole 3 nemá smysl na tomto místě zacházet do přílišných detailů. Zatím si vystačíme s tím, že v BPMN jsou zásadními objekty aktivity, události, brány, počáteční a koncové symboly a „šipky“ neboli symboly řízení toku procesu. Vzhledově se příliš neliší od korespondujících symbolů ve vývojovém diagramu.



Obrázek 2: Nákupní proces pomocí BPMN

### 2.2.2 Výhody a nevýhody

Mezi hlavní výhody BPMN určitě můžeme zařadit fakt, že BPMN je standard, tedy je přesně definované jak a kdy, které symboly používat. O BPMN se stará organizace OMG (Object Management Group) a kontinuálně pracuje na jeho rozvoji. Díky širokému rozšíření BPMN existuje na trhu velké množství placených i neplacených nástrojů, které umožňují modelování podnikových procesů pomocí této notace.

Další neoddiskutovatelnou předností je srozumitelnost notace, která je vysoká právě díky své podobnosti k vývojovým diagramům. Bez nutnosti zdlouhavého studia dokumentace je BPMN modelu schopen porozumět člověk z managementu společnosti a stejně tak i softwarový inženýr nebo vývojář. Právě pro ty ukrývá BPMN další výhodu a tou je poměrně přímočará převoditelnost BPMN modelů do strojově čitelných formátů, jako je jazyk XML nebo na něm založený jazyk BPEL.

Ačkoliv je používání notace BPMN definované v dokumentaci, reálné procesy v organizacích mohou být obtížně modelovatelné bez hluboké znalosti BPMN a může tedy docházet k vytváření nekorektních modelů nebo více různých modelů toho samého jevu. Dalším problémem dle [16] je, že někteří výrobci softwaru pro modelování v BPMN si tento standard ohýbají podle sebe nebo ho „obohacují“ o vlastní „vylepšení“, která pak dělají takto vytvořené modely obtížně přenositelnými.

### 2.2.3 Použití

Organizace OMG, která BPMN nyní spravuje, uvádí jako hlavní poslání BPMN přenositelnost procesních modelů vytvořených v této notaci nezávisle na tvůrčích konkrétního modelovacího nástroje. BPMN je vhodné pro modelování podnikových procesů v celé jejich šíři.

## 2.3 BPEL

BPEL neboli Business Process Execution Language (a správněji WS-BPEL Web Service Business Process Execution Language) je v našem výčtu jedinou technikou pro modelování podnikových procesů, který nemá grafickou reprezentaci. Je to totiž jazyk. Své využití nachází zejména při automatizaci podnikových procesů. BPEL je založen na XML a v podstatě standardizuje definici podnikových procesů právě pomocí XML. [1]

## 2.4 UML

UML neboli Unified Modeling Language je velmi populární grafický jazyk, zvláště v oblasti IT a vývoje softwarových systémů. Jak píše [11] právě s tímhle cílem bylo UML původně také vytvořeno. Jenže jeho popularita se rychle rozšířila i do světa byznysu a UML přestalo dostačovat potřebám svých uživatelů. Proto bylo postupně rozšiřováno o další aspekty, které pokrývaly modelování podnikových procesů v celé jejich šíři.

UML obsahuje standardizovaný mechanismus jak jazyk rozšiřovat tak, aby jeho obecné principy mohly být doplněny o další vyhovující specifickému účelu, jako je například právě modelování podnikových procesů. Proto byl již v době vzniku UML vytvořen standardní profil pro modelování podnikového procesu [22]. Tento profil pracuje zejména s Diagramem tříd a s Diagramem Use-Case. Diagram Use-Case je v tomto profilu používán pro zobrazení podnikových procesů a jejich interakce s aktéry a zákazníky. Oproti tomu Diagram tříd se používá

spíše k zobrazení vnitřní struktury popisované organizace. Faktem je, že se standardní profil pro modelování podnikového procesu v praxi příliš neujal, snad kvůli své přílišné snaze podnikové procesy přiblížit k IT a informačním systémům [22].

UML se přesto pro modelování podnikových procesů používá a to buď v neformální podobě, kdy je organizacemi používán především Diagram aktivit, který je pro modelování podnikových procesů vhodný. Activity Diagram totiž umožňuje sekvenční i paralelní zobrazování aktivit a objekty na vstupu i výstupu procesu a také závislosti mezi jednotlivými aktivitami.

Pro větší přenositelnost a potřeby standardizace UML jazyka pro modelování podnikových procesů však vznikla celá řada rozšíření třetích stran. Mezi nejpoužívanější pak patří rozšíření podle H. Erikssona. [22, 10].

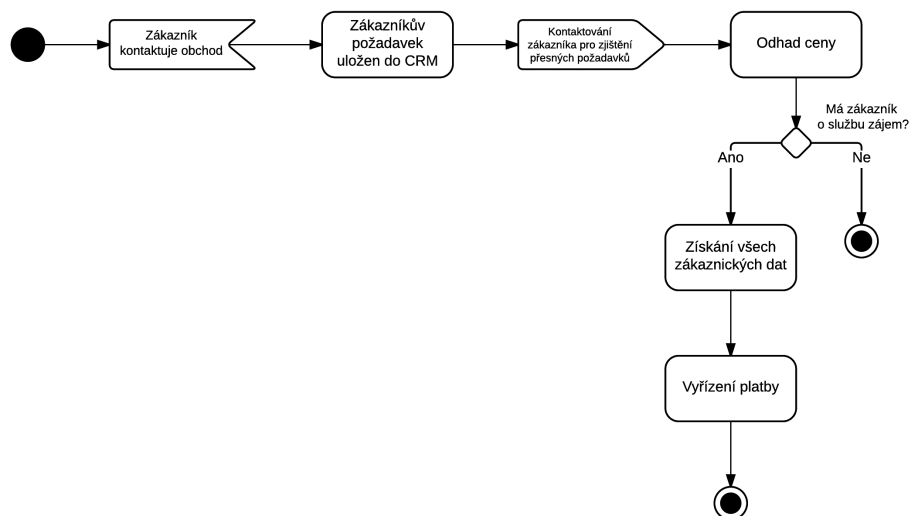
Erikssonův přístup je nejenom rozšířením UML, ale do značné míry plnohodnotnou metodou modelování procesů – určuje sadu modelů a diagramů, postavených vesměs na standardních diagramech UML. [22]

Erikssonovo rozšíření obsahuje více různých diagramů, přičemž pro potřeby samotného modelování podnikového procesu je nejzákladnější Diagram procesů, který je rozšířením právě již výše zmíněného Diagramu aktivit. Právě tomu se tedy budeme v této kapitole věnovat.

#### 2.4.1 Základní pravidla

Diagram aktivit obsahuje několik základních prvků: [4]

- Aktivita – proces, který modeluje
- Akce – jedná se o konkrétní činnost v rámci aktivity
- Zahájení a ukončení – Počáteční a koncový uzel
- Řídící tok – určuje pořadí vykonávání jednotlivých kroků aktivity, znázorněné pomocí šipek
- Rozhodnutí – rozvětvení procesu na základě určité podmínky
- Spojení – spojuje více toků do jednoho
- Rozdělovník a spojovník – pro provádění více akcí paralelně se používá rozdělovník a spojovník
- Příchozí událost – používá se, pokud je k provedení některých kroků potřeba akce zvenčí
- Spuštění události – používá se, pokud je potřeba znázornit akci, která spouští jiný proces nebo akci



Obrázek 3: Nákupní proces pomocí UML

#### 2.4.2 Výhody a nevýhody

Mezi výhody UML můžeme určitě zařadit velkou rozšířenost tohoto jazyka a s tím spojenou i velkou dostupnost nástrojů umožňujících modelování v UML, literatury i komunit, kde je možno hledat pomoc či radu.

Jako nevýhodu lze určitě uvést faktickou neexistenci standardu pro modelování podnikových procesů v UML. Standardní profil pro modelování podnikového procesu, který UML obsahuje, se příliš nepoužívá a od verze UML 2.0 už dokonce není součástí množiny standardních profilů. [22] Existují sice další rozšíření, ale ty nejsou standardem, což snižuje možnost jejich přenositelnosti i udržitelnosti do budoucna. UML je přeci jen jazyk stále velmi blízký vývoji softwaru a modelování podnikových procesů spíše umožňuje, než podporuje.

#### 2.4.3 Použití

Jak už bylo řečeno výše, UML nachází uplatnění v mnoha oborech lidské činnosti, ale prosadilo se zejména v IT a tvorbě softwaru. Některé nástroje, které umožňují modelování v UML dokonce nabízí možnost z UML modelu exportovat přímo zdrojový kód například pro tvorbu databáze.

Jelikož UML obsahuje různé druhy modelů, dostává tím celý jazyk další možnosti využití i mimo IT. Síla UML je v zachycení struktury jevu a již tolik nezáleží na tom, jestli se jedná o databázi nebo organizační strukturu podniku. Například Use-Case diagram najde využití nejen při modelování chování člověka při interakci se softwarem, ale například také obsluhy zákazníka v bance apod.

UML je tak mimo jiné využíváno v IT, bankovníctví, telekomunikacích, zdravotnictví nebo v obranném průmyslu.

## 2.5 DEMO

DEMO neboli Design & Engineering Methodology for Organizations je metodologie se silným teoretickým základem určená k modelování, analýze a grafickému zobrazení podnikových procesů. Za jejím vznikem stojí zejména Jan Dietz. DEMO obsahuje 4 modely, každý pro jiný účel a jiný pohled na proces.

Tímto způsobem jsme schopni rozmotat uzel, který dnešní organizace připomínají, odstranit chyby v jejich návrhu a přitom zajistit, aby vše fungovalo. Stejně jako by inženýr opravil most, letadlo nebo počítač. [2]<sup>3</sup>

Fundamentální vlastnosti DEMO modelu dle [8] jsou:

- ucelený,
- konzistentní,
- úplný,
- výstižný,
- obsahovat jen nezbytné množství informací.

Zjednodušeně řečeno, důvodem pro vytvoření této metodologie byla narůstající nevhodnost jiných metodologií, notací a jazyků obvykle používaných k modelování podnikových procesů. Modely vytvořené v těchto nástrojích jsou totiž většinou příliš podrobné a příliš technicky zaměřené, takže stěžují pohled na organizaci z vrchu, pohled jen na to důležité, co se v ní odehrává.

DEMO je naopak postaveno na modelování podnikových procesů pomocí ontologií, což znamená, že je zachyceno jen jádro problému, které většinou tvoří komunikace mezi lidmi. DEMO má silný teoretický základ v teorii PSI, o které si více řekneme v kapitole 4.

### 2.5.1 Základní pravidla

Detailnímu popisu modelování v DEMO se věnuje kapitola 4, takže na tomto místě zmíním jen základní věci. Kompletní model organizace (tzv. essential model) se skládá ze 4 různých modelů: [20]

1. Construction Model (CM)
2. Process Model (PM)
3. Action Model (AM)
4. State Model (SM)

---

<sup>3</sup>This way we can untangle the complex knot that organizations have become, fix the constructional mistakes, and put everything back together. Just like an engineer would repair a bridge, airplane or computer. [2]



Jedním z nejdůležitějších aspektů metodologie DEMO je tzv. „transaction pattern“, který dává přesnou strukturu tomu, jak probíhají transakce (například objednávka). V DEMO se takové transakce skládají vždy ze stejných kroků a tudíž mají všechny stejnou strukturu, což zanechává malý prostor pro více interpretací stejné transakce.

Nejdůležitější aspekty v DEMO jsou 2:

1. Ontologická transakce
2. Actor role

Podrobně budou tyto elementy rozebrány v kapitole 4.

### 2.5.2 Výhody a nevýhody

Mezi silné stránky metodologie DEMO určitě musíme zařadit jeho absolutní jednoznačnost. Zatímco u jiných nástrojů, jako je třeba vývojový diagram, ale i BPMN nebo UML, vzhledem k jejich poměrně vysoké úrovni detailu, může docházet k nejednoznačnostem, tj. že stejný jev je vymodelován různě. DEMO díky svým poměrně rigidním pravidlům a modelu postavenému na jasně strukturovaných transakcích dostáváme stejné modely pro stejné jevy, což je velmi pozitivní pro celkovou konzistenci BPM v organizaci a zároveň to usnadňuje analýzu a vylepšování procesů.

Jako hlavní přednost DEMO uvádí [6] schopnost modelů vytvořených pomocí této metodologie zachytit pouze základní podstatu každé organizace a schopnost modely abstrahovat od technických detailů, které jsou pro účely modelování organizací podružné.

Jako nevýhodu celé metodologie je určitě nutné označit poměrně dlouhou a pozvolnou křivku učení, která je v tomto případě dána 2 věcmi:

1. Široký teoretický základ, který sahá až do oblasti ontologie, filozofie a dalších oborů. I samotná metodologie má za sebou velmi robustní teorii, která není na první pohled zřejmá.
2. Narozdíl od vývojových diagramů nebo BPMN jsou modely vytvořené v DEMO pro člověka nezasvěceného do toho, jak DEMO funguje, prakticky nečitelné.

Když by chtěla organizace začít používat DEMO pro modelování svých procesů, je nutné investovat čas i finanční prostředky do zasvěcení odpovědných lidí do metodologie a jejího používání, přičemž vycvičení lidí na potřebnou úroveň může být poměrně zdoluhavé. Právě toto by mohlo být jednou z hlavních překážek, které brání většímu rozšíření metodologie DEMO mimo akademickou půdu.

### 2.5.3 Použití

Tato část je založena zejména na [20], kde jsou uvedeny nejméně 3 možné způsoby, jak využít DEMO.

1. **Návrh a optimalizace organizací** – Díky přednostem metodologie DEMO, které jsou popsány výše, je možné se na organizaci podívat z ontologického hlediska, což umožňuje snáze pochopit, jak organizace skutečně funguje a je možné fungování jednotlivých procesů vylepšit a zefektivnit.
2. **Softwarová podpora organizace** – Jelikož velká část podnikových procesů je dnes podporována IT systémy. DEMO rozděluje druhy softwaru do tří úrovní, aby korespondovaly se strukturou organizace tak, jak jí vidí DEMO (ontologická, infologická, datalogická). Pomáhá tak jasně určit strukturu používaného softwaru uvnitř organizace.
3. **Vývoj softwaru** – Ačkoliv DEMO abstrahuje při modelování procesů od implementačních detailů, tak může být užitečné při jeho vývoji. DEMO modely totiž mohou sloužit jako odrazový můstek na začátku vývoje. DEMO model totiž obsahuje základní informace o každém procesu: kdo ho iniciuje, kdo ho provádí a jaké aktivity obsahuje a v jakém pořadí. Dle [18] jsou DEMO modely velmi snadno převoditelné do Use Case modelů.

## 3 Srovnání technik

Tato sekce se soustředí na porovnání objektivních měřítek, podle kterých můžeme jednotlivé techniky porovnávat. Její ambicí rozhodně není rozhodnout, která z nich je „lepší“ nebo „horší“, jelikož to vždy závisí především na konkrétním účelu, ke kterému chceme danou techniku použít. Z porovnání je vyjmut BPEL, jelikož nepodporuje grafické znázornění procesu.

Tabulka 1: Porovnání vybraných technik k modelování podnikových procesů

	<b>Vývojový diagram</b>	<b>UML</b>	<b>BPMN</b>	<b>DEMO</b>
Použití				
Dostupnost nástrojů				
Dostupnost literatury				
Velikost komunity				
Výhody				
Nevýhody				

## Reference

- [1] BPEL – jazyk pro automatizaci procesů.
- [2] Enterprise Engineering and DEMO.

- [3] ISO 9000:2005.
- [4] UML 2 Activity Diagramming Guidelines.
- [5] Mike Bandor. Process and Procedure. 2007.
- [6] J. Barjis. Enterprise Modeling and Simulation Within Enterprise Engineering. *Journal of Enterprise Transformation*, 1(3):185–207, 2011.
- [7] Jiří Chytil and Zdeněk Lehocký. Vývojové diagramy - 1. díl, 2005.
- [8] Jan L G Dietz. *Enterprise ontology: Theory and methodology*. 2006.
- [9] Marlon Dumas, M La Rosa, Jan Mendling, and Hajo Reijers. *Fundamentals of Business Process Management*. Springer Berlin Heidelberg, 2013.
- [10] Hans-Erik Eriksson and Magnus Penker. *Business Modeling With UML: Business Patterns at Work*. 2000.
- [11] Hans-Erik Eriksson and Magnus (Open Training) Penker. Business Modeling with UML.
- [12] Paul Harmon. *Business Process Change: A Business Process Management Guide for Managers and Process Professionals*. Morgan Kaufmann, 2014.
- [13] Andreas Jedlitschka, Outi Salo, and Frank Bomarius. Process Management. *Journal of Software Maintenance and Evolution: Research and Practice*, (May):143–149, 2010.
- [14] Pavel Náplava. A7B16ISP Informační systémy a procesní řízení, 2015.
- [15] Theodore Panagacos. *The Ultimate Guide to Business Process Management: Everything you need to know and how to apply it to your organization*. CreateSpace Independent Publishing Platform, 2012.
- [16] Gregor Polancic. Managing business processes with BPMN – SWOT Analysis, 2014.
- [17] Jan C. Recker, Michael Rosemann, Marta Indulska, and Peter Green. Business process modeling: a comparative analysis. *Journal of the ...*, 10(4):333–363, 2009.
- [18] Boris Shishkov and Jan Dietz. Deriving Use Cases from Business Processes - The advantages of DEMO. pages 249–257, 2005.
- [19] Petr Vašíček. Úvod do bpmn, 2008.
- [20] Zuzana Vejražková. Design and Engineering Methodology for Organizations. 2012.
- [21] Mathias Weske. *Business Process Management*. Springer, 2007.

- [22] Václav Řepa. *Podnikové Procesy. Procesní řízení a modelování*. Grada Publishing, a.s., 2007.
- [23] Václav Řepa. Řízení procesů versus procesní řízení, 2008.