

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Bakalářská práce

Srovnání nástrojů OntoUML a BORM z hlediska použitelnosti

Ondřej Tichý

© 2019 ČZU v Praze

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Ondřej Tichý

Informatika

Název práce

Srovnání nástrojů OntoUML a BORM z hlediska použitelnosti

Název anglicky

Comparison of OntoUML and BORM tools from a usability point of view

Cíle práce

Cílem práce je porovnat přístupy OntoUML a BORM z hlediska použitelnosti. Jednotlivé modely mají různé notace a je možné pomocí nich zobrazit procesní toky. Je nutné tyto přístupy prostudovat, vytvořit vhodné scénáře pro testy použitelnosti a ty následně provést.

Metodika

- Prostudování nástrojů OntoUML a BORM
- Navržení vhodných testovacích scénářů pro test použitelnosti obou nástrojů
- Provedení usability testů s vybranou skupinou participantů v laboratoři HUBRU
- Vytvoření záznamu, vyhodnocení a popis testu

Doporučený rozsah práce

43

Klíčová slova

OntoUML, Metoda BORM, Objektové modelování, Konceptuální modelování

Doporučené zdroje informací

Alan Cooper and Robert Reimann: About Face 2.0: The Essentials of Interaction Design (Mar 17, 2003), ISBN-13: 978-0764526411

J. Nielsen, R. L. Mack "Usability Inspection Methods" John Wiley & Sons, New York, NY ISBN 0-471-01877-5, 1994

MERUNKA, V. – POLÁK, J. – CARDA, A. – ČESKÁ SPOLEČNOST PRO SYSTÉMOVOU INTEGRACI. *Umění systémového návrhu : objektově orientovaná tvorba informačních systémů pomocí původní metody BORM*. Praha: Grada, 2003. ISBN 80-247-0424-2.

Předběžný termín obhajoby

2018/19 LS – PEF

Vedoucí práce

Ing. Josef Pavlíček, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Srovnání nástrojů OntoUML a BORM z hlediska použitelnosti" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 13.3.2019

Poděkování

Rád bych touto cestou poděkoval své rodině za neutuchající podporu během mých studií. Dále bych také rád poděkoval vedoucímu mé práce Ing. Josefu Pavlíčkovi, Ph.D. za odborné vedení práce a cenné rady, které mi pomohly tuto práci zkompletovat.

Abstrakt

Cílem bakalářské práce „Srovnání nástrojů OntoUML a BORM z hlediska použitelnosti“ je komparace metod pro tvorbu konceptuálních, respektive procesních diagramů z pohledu jejich možného využití při tvorbě informačních systémů. Model OntoUML pro testování použitelnosti byl vytvořen v aplikaci OLED, zatímco procesní diagram metody BORM v open source nástroji OpenPonk. Oba modely byly následně testovány dvěma různými způsoby na vybraných skupinách participantů. Ten první se uskutečnil v laboratoři HUBRU na České Zemědělské univerzitě v Praze pod dohledem doktora Pavlíčka a druhý na domácích stanicích uživatelů. Na základě interview s participanty byly vyhodnoceny kladné a záporné stránky obou modelů a navrženo zlepšení stávajícího stavu.

Klíčová slova

OntoUML, Metoda BORM, Objektové modelování, Konceptuální modelování, Procesní modelování

Abstract

The aim of the bachelor thesis "Comparison of OntoUML and BORM tools in usability point of view" is to compare methods for the creation of conceptual, respectively process diagrams. The OntoUML model used for usability testing was created in the OLED application while the BORM process diagram in open source tool OpenPonk. Both models were subsequently tested in two ways on the participants. The first one took place at the HUBRU laboratory at the Czech University of Life Sciences Prague under the supervision of Ing. Josef Pavlíček Ph.D. and the second one at the home stations of selected users. Based on the interview with the participants, the positive and negative aspects of both models were evaluated and an improvement of the current state was proposed.

Keywords

OntoUML, BORM Method, Object modelling, Conceptual modelling, Process modelling

Obsah

1 Úvod.....	7
1 Modelování statických datových modelů	8
1.1 Princip tří architektur datového modelování.....	8
1.1.1 Konceptuální úroveň.....	8
1.1.2 Technologické úroveň.....	8
1.1.3 Implementační úroveň	9
1.2 Prvky statických datových modelů	10
1.2.1 Entity.....	10
1.2.2 Vazby	11
2 Modelování dynamických modelů	12
2.1 Modelování procesních diagramů	12
2.1.1 Procesy.....	13
2.1.2 Typy procesů.....	15
2.1.2.1 Hlavní	15
2.1.2.2 Vedlejší.....	15
2.1.2.3 Řídící	15
2.1.2.4 Podpůrné.....	15
2.1.3 Workflow	16
2.1.4 Přístupy k procesnímu modelování.....	16
3 Modelovací jazyky a metody	17
3.1 Petriho síť.....	17
3.2 Metoda Aris.....	18
3.2.1 Diagramy ARIS	19
3.3 BPMN (Business process modeling notation)	19
3.3.1 Stručná historie	19
3.3.2 Diagramy BPMN	20
3.4 UML (Unified Modeling Language).....	22
3.4.1 Využití	22
3.4.2 UML diagramy	23
3.4.3 Třídní diagram	23
3.4.3.1 Objektový diagram	24
3.4.3.2 Stavový diagram.....	24
3.4.3.3 Diagram aktivit.....	25
3.4.3.4 Diagram komponent	25
3.5 Metoda BORM.....	26

3.5.1	Vznik a vývoj.....	26
3.5.2	Charakteristika.....	26
3.5.3	Procesní diagram BORM.....	27
3.5.4	Konceptuální diagram BORM.....	28
3.5.5	Grafické elementy notace BORM.....	29
3.5.6	Fáze životního cyklu.....	30
3.5.7	CASE nástroje.....	32
3.6	OntoUML.....	32
3.6.1	Ontologie.....	32
3.6.1.1	Filozofické hledisko.....	32
3.6.1.2	Ontologie z hlediska počítačových věd.....	33
3.6.2	Historie metody OntoUML.....	34
3.6.3	UFO.....	35
3.6.4	CASE nástroje.....	36
3.6.4.1	OLED.....	36
3.6.4.2	Menthor.....	36
3.6.5	Stereotypy OntoUML.....	36
3.6.5.1	Třídní stereotypy.....	37
3.6.5.2	Vztahové stereotypy.....	38
3.6.6	Překážky většího uplatnění nástroje.....	38
4	Test použitelnosti.....	39
4.1	Cíl studie.....	39
4.2	Metodika výzkumu.....	39
4.2.1	Úvod.....	39
4.2.2	HUBRU.....	39
4.2.3	Zúčastnění participantů.....	39
4.3	Test.....	40
4.3.1	BORM procesní diagram vydání knihy.....	41
5	Výsledky a diskuze.....	45
5.1	Další doporučení.....	46
5.2	Limitace studie.....	46
6	Závěr.....	47
7	Seznam obrázků.....	49
8	Seznam zdrojů použitých obrázků.....	50
9	Seznam použitých zdrojů.....	51

1 Úvod

Informační systémy v různé podobě jsou dnes nedílnou součástí každého podniku. Dny, kdy k řízení provozu a podpoře rozhodování na úrovních managementu postačovaly tak jednoduché nástroje jako jsou tužka a papír, případně tabulka vytvořená v Excelu jsou pryč. Dnešní obchody mají pochopitelně zájem na využití stále se vyvíjejících informačních technologií, které mohou v rámci podniku leccos usnadnit.

Implementace sofistikovanější řešení však vyžaduje velké množství práce jakou může být například nasbírání co největšího množství informací o daném podniku, o procesech odehrávajících se jak uvnitř, tak vně, o jeho struktuře, o problémech, které by implementovaný systém měl řešit, dále návrh řešení konkrétního systému, vývoj, testování a nakonec jeho samotná údržba.

A právě pro analýzu systémových potřeb a následné usnadnění komunikace mezi jednotlivými účastníky procesu implementace systému se v praxi využívají univerzální nástroje pro modelování různých typů diagramů popsané a zhodnocené v této bakalářské práci. Nemusí ale sloužit jen ke komunikaci při zavádění nového systému, své uplatnění naleznou i ve fázi provozu systému, kdy se může objevit situace, která vyžaduje zásah do stávajícího řešení a změnu aktuálních funkcionalit.

1 Modelování statických datových modelů

Jedná se o jednu z disciplín softwarového inženýrství. Modelování datových modelů je činnost, při které se analyzují a specifikují požadavky na datové struktury a uspořádání dat vyvíjeného softwarového řešení. To, co dělá z modelů modely statické je to, že nezohledňují časové hledisko, tím se zásadně liší od modelů dynamických zmiňovaných v další kapitole práce. Jedná se o pohled na realitu a její popis pomocí k tomu určených technik a nástrojů.

1.1 Princip tří architektur datového modelování

Princip tří architektur definuje abstrakci vývoje softwaru po jednotlivých úrovních. Tyto úrovně se zaměřují na 3 aspekty vyvíjeného systému – obsah, technologii a implementační specifika. Návrh nového systému probíhá v těchto třech fázích:

1.1.1 Konceptuální úroveň

Cílem konceptuálního modelu je identifikovat objekty, klasifikovat je do tříd, identifikovat vazby (relations) mezi jednotlivými objekty a identifikovat jejich vlastnosti, tedy zjednodušeně udává, co bude součástí systému. Jedná se o první fázi modelování, která následně slouží jako vstup pro implementaci konkrétního databázového řešení. Při návrhu konceptuálního modelu je důležité nezohledňovat budoucí technologické řešení vyvíjené databáze.¹

1.1.2 Technologické úroveň

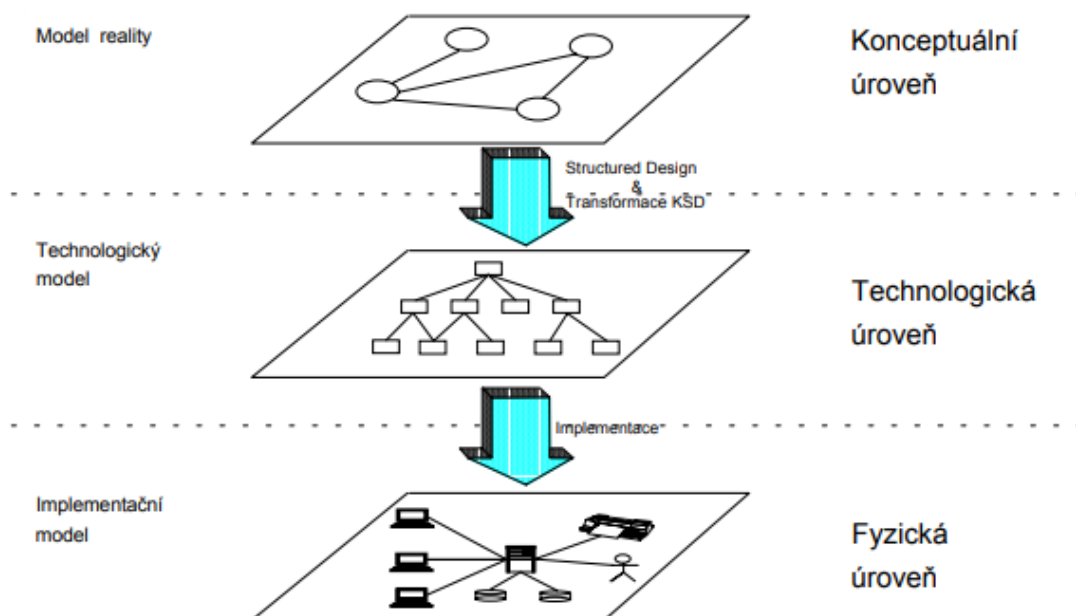
Často také nazývána logickou úrovní. V této fázi se vytváří model, který zohledňuje dostupné technologie pro koncepci řešení navrhovaného systému. Ani tento model nesmí být zatížen konkrétnímu implementačními specifiky řešení. Obsahové řešení bylo modelováno na konceptuální úrovni, na technologické se již neřeší. Technologický návrh říká, jak bude systém technologicky realizován.²

¹ Molhavec, Martin, 2006. [online]. Dostupné z: <http://prog-story.technicalmuseum.cz/images/dokumenty/Programovani-TSW-1975-2014/2006/2006-17.pdf>

² Řepa, Václav. VÝVOJOVÉ TRENDY METODIK VÝVOJEINFORMAČNÍCH SYSTÉMŮ - VÝZVA BPR. [online]. Dostupné z: <https://nb.vse.cz/~repa/veda/EurOpen99%20Paper.pdf>

1.1.3 Implementační úroveň

Určuje, čím bude technologické řešení systému realizováno. Probíhá zde implementace datové struktury v konkrétním programovacím jazyce, která byla navržena v předchozí fázi modelování.²



Obrázek 1 - Princip tří architektur datového modelování

Každá z těchto tří architektur disponuje rozlišným zaměřením a logikou. Každá z úrovní má v souvislosti s metodami tvorby informačních systémů svůj specifický dorozumívací způsob a metodiku návrhu.⁴⁷

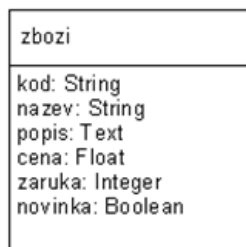
⁴⁷ ŘEPA, Václav. Podnikové procesy. 2. vyd. Praha: Grada Publishing, 2007. 281 s. ISBN 978-80-247-2252-8

² Řepa, Václav. VÝVOJOVÉ TRENDY METODIK VÝVOJE INFORMAČNÍCH SYSTÉMŮ - VÝZVA BPR. [online]. Dostupné z: <https://nb.vse.cz/~repa/veda/EurOpen99%20Paper.pdf>

1.2 Prvky statických datových modelů

1.2.1 Entity

Entita reprezentuje skupinu objektů sledované reality, kterým říkáme instance entity. Vyznačují se tím, že sdílí stejnou množinou atributů a mají společnou logickou souvislost. Každému atributu takové entity je přiřazen datový typ, tedy obor hodnot, kterých smí konkrétní atribut nabývat. Všechny instance entity se nazývají populací. Příkladem takové entity může být například „Zaměstnanec“ a instancemi této entity potom strojař „Pavel Hapal“ nebo řidič „Martin Suchařípa“. V modelech se entity zobrazují pomocí obdélníků, kde je v horní části zobrazen název entity a pod ním kolekce atributů s datovými typy (viz obrázek 2).



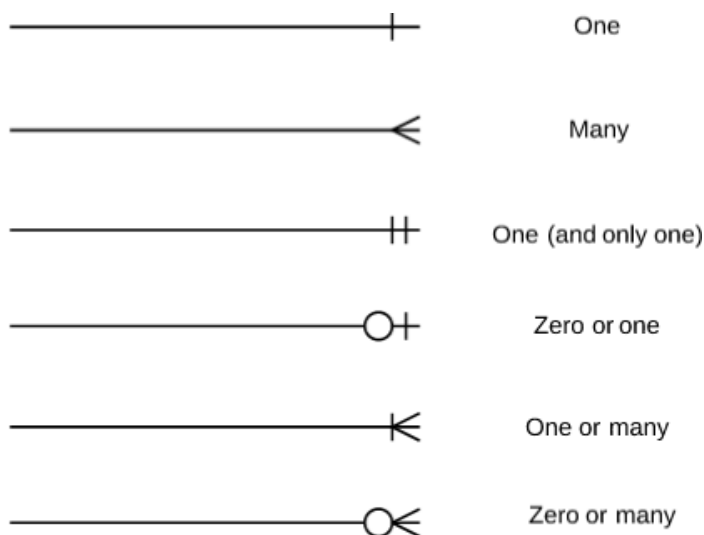
Obrázek 2 - Grafické znázornění entity

1.2.2 Vazby

Vazby představují logické vztahy mezi jednotlivými entitami systému. Jejich význam odpovídá slovesům použitým při popisu toho, co jednotlivé entity dělají, například entita „Spisovatel“ píše (vazba) entitu „Kniha“. Kardinalita těchto vztahů poté určuje spojitost mezi entitami na obou koncích binární vazby. Existují 4 základní typy vztahů:

- Žádná spojitost mezi objekty není
- 1:1 – jednomu záznamu z jedné entity náleží právě jeden záznam z entity druhé
- 1:N – jednomu záznamu z jedné entity může náležet více záznamů z entity druhé
- M:N – tento typ nastává ve chvíli, kdy jakémukoliv záznamu z první entity může být přiřazen jakýkoliv záznam z tabulky na druhém konci vazby. Tento vztah se při tvorbě databáze často řeší vytvořením vazební tabulky, která poté tvoří vazby 1:N a M:1 s původními tabulkami.⁴⁶

Při tvorbě statického datového modelu se modelují pouze vazby mezi dvěma entitami, tedy binární vazby. Na obrázku 3 jsou graficky zobrazeny jednotlivé druhy vazeb společně s jejich kardinalitami.



Obrázek 3 - Vazby

⁴⁶ Zedníček, Jan. Kardinalita vztahu, 2017. [online]. Dostupné z: <https://biportal.cz/BI-slovník/kardinalita/>

2 Modelování dynamických modelů

2.1 Modelování procesních diagramů

Na počátku procesního modelování stojí procesní analytik, který má za úkol popis elementárních prvků procesů. Čerpat při tom může z mnoha dostupných pramenů, jako jsou již existující procesní mapy, normy, směrnice, rozhovory s jednotlivými účastníky procesů, pozorování chodu podniku nebo organizační schéma.

Zmapování procesů je základním předpokladem pro tvorbu modelu procesů, může jít o prostý slovní popis nebo zápis ve strukturované tabulce, avšak ani jeden z těchto zápisů není zcela ideální. Další způsob, který je možné použít je grafické znázornění procesů na úrovni sledované organizace. Takové znázornění je mnohem přehlednější než slovní zápis. Procesy je možné znázornit na papír formou vývojového diagramu nebo k tomu použít specializované IDE pro tvorbu procesních diagramů. Na základě takového grafického modelu jsou společnosti schopny podnikovým procesům lépe porozumět, analyzovat a následně navrhnout jejich efektivnější řešení, umožňují také predikci a simulaci změn aplikovaných na procesy, či tvořit procesy nové. Takovéto modely velmi často nachází uplatnění při tvorbě interních informačních systémů a to jak při zavádění nového systému do podniku, tak při úpravách toho stávajícího.³

K tvorbě procesního modelu je vhodné využít předdefinované grafické objekty některé z modelovacích technik. Každá technika se skládá ze dvou provázaných částí – metodu modelování a modelovací jazyk. U jazyka dále odlišujeme syntaxi, tedy pravidla pro používání a sémantiku, význam používaných prvků.

³ ADJEI, Daniela. Standardy a metodiky modelování podnikových procesů [online]. Praha, 2009 [cit. 2019-03-10]. Dostupné z: <<https://theses.cz/id/q0wets/>>. Bachelor's thesis. University of Economics, Prague. Vedoucí práce Tomáš Vilím.

2.1.1 Procesy

Přibližme si, co takový proces v praxi znamená. Literatura definuje podnikový proces (business process) různě. Podle Václava Řepy se jedná o souhrn činností přeměňující za pomoci lidí a dalších nástrojů vstupy na výstupy, které mají hodnotu pro další procesy nebo přímo pro externí zákazníky (Řepa, 2006). Davenport jej definuje takto: „Proces je jednoduše strukturovaný, měřitelný soubor činností navržených za účelem vytvoření specifického produktu pro konkrétního zákazníka (Davenport, 1993).“ Hammer, jeden ze zakladatelů manažerské teorie BPR (Business process reengineering) zase říká: „Proces je soubor činností, který vyžaduje jeden nebo více druhů vstupů a tvoří výstup, který má pro zákazníka hodnotu (Hammer, 2000).“ Takový vstupem můžeme chápat například jako přijatou fakturu za odběr zboží, která prochází procesem zaplacení a konečným výstupem tohoto procesu je uhrazená faktura od dodavatele.

Obecně proces popisuje Oxford Advanced dictionary, jako posloupnost úkolů, které postupně směřují k jednomu předem určenému cíli. Jednotlivé procesy mezi sebou často interagují, tato interakce není omezena jen na procesy uvnitř podniku, ale často dochází i k interakci mezi procesy různých podniků (business to business processes).⁷

Každý jeden podnikový proces by měl mít svého jedinečného vlastníka. Vlastník procesu zodpovídá za nastavení procesu, tedy jeho způsob provádění a dodržování předepsaných postupů.

⁴ Řepa, V.: Podnikové procesy: Procesní řízení a modelování. Grada Publishing, a.s., Praha, 2006. 268 s. ISBN 80-247-1281-4., s. 13

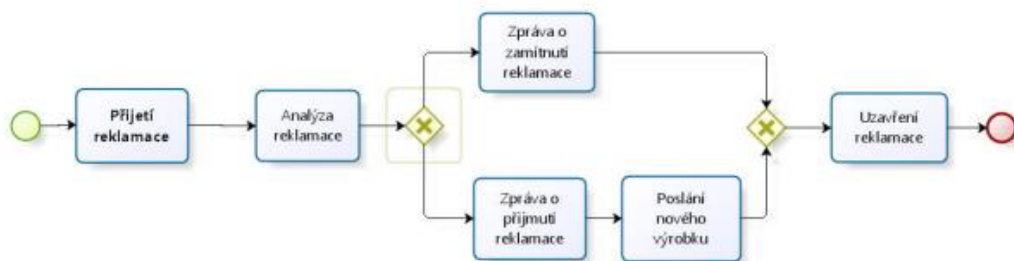
⁵ DAVENPORT, T. H. Process Innovation: reengineering Work through Information Technology. Harvard Business School Press, 1993, ISBN 087-58436-62.

⁶ HAMMER, M., CHAMPY, J. Reengineering - radikální proměna firmy. Manifest revoluce v podnikání. Praha: Management Press, 2000, 212 s. ISBN 80-7261-028-7.

⁷ HORNBY, A. S., TURNBULL, J., LEA, D., PARKINSON, D., PHILLIPS, P., & ASHBY, M. (2010). Oxford advanced learner's dictionary of current English.

Podnikové procesy představují koncept, který napomáhá vyššímu managementu a vlastníkům podniku pochopit, jakým způsobem společnost funguje. Přesná definice vnitropodnikových procesů pak značně usnadňuje integraci a implementaci informačních systémů.

Další výhodou procesního pohledu může být zavedení automatizace řízení v rámci podniku. Zavedení automatizace je opět závislé na přesné definici procesů. Tradičně veškeré řízení v rámci podniku probíhá tak, že řídicí pracovníci se zkušenostmi z praxe dohlíží na správné dodržování postupů, avšak korektní popis těchto procesů může toto řízení delegovat přímo na informační systém, který na základě procesního modelu dodržuje posloupnost jednotlivých činností a dohlíží na jejich dodržování, čímž se eliminuje možnost toho, že dojde během provádění procesu k chybě lidského faktoru.



Obrázek 4 - Vývojový diagram procesu reklamace

2.1.2 Typy procesů

2.1.2.1 Hlavní

Hlavní procesy představují klíčové činnosti podniku (tvorba výrobků, poskytování služeb), které mu přináší přidanou hodnotu, zpravidla se mapují jako první, jsou snadno identifikovatelné managementem společnosti, na venek podniku viditelné a často komplikované. Hlavní procesy lze na základě vizí a poslání společnosti dále dělit na klíčové procesy, které v rámci podniku tvoří jednotky, maximálně desítky procesů.

2.1.2.2 Vedlejší

Jsou obdobné těm hlavním, avšak na rozdíl od hlavních nejsou z hlediska vizí a strategie tak důležité, aby se podílely na hlavní činnosti podniku. Jejich výstupy jsou zpravidla určeny externím zákazníkům, pro vysvětlení, zákazníkem procesu je příjemce výstupu procesu, externím zákazníkem potom příjemce výstupu vně společnosti. Jedná se o procesy, které jsou hlavními aspiranty k tomu, aby byly outsourcovány mimo podnik.⁹

2.1.2.3 Řídící

Procesy, které zajišťují plynulý chod společnosti, jako například plánování, či tvorba strategie. Obvykle začínají na nejvyšší úrovni managementu při definování strategických cílů a postupně podél linie řízení tak, aby realizovali zadané cíle. Mapují se až jako poslední z toho důvodu, že samy o sobě nevytváří žádný zisk.

2.1.2.4 Podpůrné

Jak název napovídá, především podporují hlavní procesy podniky (školení, rozpočtování). Stejně jako řídicí procesy neprodukují zisk a mapují se jako druhé. Příkladem podpůrného procesu mohou být Human Resources nebo služební cesta.^{8,10}

⁸ Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z:

<http://www1.osu.cz/~zacek/mopop/mopop.pdf>

⁹ Workflow. In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 11.09.2018 [cit. 10.03.2019].

Dostupné z: <https://managementmania.com/cs/workflow>

¹⁰ Lucie, Pekárková, 2007. Techniky modelování a optimalizace podnikových procesů. Bakalářská práce. Masarykova univerzita.

2.1.3 Workflow

Workflow rozumíme automatizaci celého nebo části podnikového procesu, kdy jsou předávány dokumenty mezi jednotlivými zákazníky procesů dle procedurálních pravidel. Odehrávat se může v podobně předávání tištěných dokumentů nebo tokem digitálních informací, který je podpořen k tomu uzpůsobeným softwarem, například ERP (Enterprise Resource Planning) nebo DMS (Document Management System). Workflow napomáhá ke zrychlení, zefektivnění a zprůhlednění podnikových procesů. Nasazení takového workflow vyžaduje přesnou a jednoznačnou specifikaci procesu.

Aplikuje se především tam, kde dochází ke složitým aktivitám nebo transakcím, které je nutno koordinovat tak, aby dodržovaly předepsanou posloupnost a nedocházelo k selháním lidského faktoru. Takováto workflow mohou znamenat citlivý právní úkon, proto je důležité, aby byla zdokumentována a schvalování těchto transakcí mohly provádět jen k tomu oprávněné osoby v rámci organizační struktury společnosti.⁹

2.1.4 Přístupy k procesnímu modelování

K modelování podnikových procesů existují 3 základní přístupy, které vychází ze tří základních typů abstrakce, funkční přístup, strukturální a specifikace chování. Funkční přístup je nejobecnějším z nich. Objektem zájmu tohoto přístupu jsou vstupy a výstupy procesů a jejich funkce. Dal by se specifikovat tím, že odpovídá na otázku CO? Strukturální přístup zobrazuje entity účastníci se procesu a odpovídá na otázku KÝM a ČÍM? Přístup specifikace chování zobrazuje dynamiku a posloupnost činností směřujících k naplnění určitého cíle společně se spouštěči těchto činností. Tento přístup odpovídá na otázku JAK?⁴⁸

⁹ Workflow. In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 11.09.2018 [cit. 10.03.2019].

Dostupné z: <https://managementmania.com/cs/workflow>

⁴⁸ Koudelková, Anna, 2011. Procesní model organizace.[online]. Diplomová práce. Univerzita Pardubice.

Dostupné z: <https://docplayer.cz/13662439-Procesni-model-organizace.html>

3 Modelovací jazyky a metody

Pro tvorbu jak statických, tak dynamických modelů existuje nepřehledné množství metod. V následujících kapitolách přiblížím několik z nich a to především metody BORM (Business Object Relationship Modelling) a OntoUML.

3.1 Petriho síť

Petriho síť (Petri nets) označují řadu matematických reprezentací, které umožňují popisovat řídicí toky a informační závislosti uvnitř modelovaných systémů. Jejich vznik se datuje rokem 1962, kdy německý matematik Carl Adam Petri ve své dizertační práci „Kommunikation mit Automaten“ popsal nový koncept vzájemných závislostí mezi podmínkami a událostmi systému. Obsahují tři základní komponenty, místa, která obsahují stavovou informaci ve formě značek a přechody, které znázorňují možné změny stavů (viz grafické elementy na obrázku 5).⁹



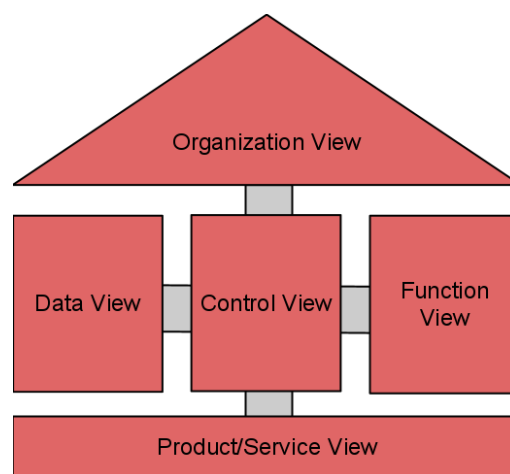
Obrázek 5 – Grafické elementy Petriho sítě

Petriho síť vychází z modelu konečných automatů, který rozšiřují o nové možnosti. Hlavním přínosem je možnost grafického vyjádření a možnost simulace dynamických chování modelu. Pokud je v modelu konečných automatů stav procesu dán souhrnem dílčích stavů, jsme nuceni každý takový stav vyjádřit samostatným prvkem. Při komplikovanějším problému tak narůstá objem takového modelu. Tento problém odpadá v případě užití Petriho sítě.

⁹ Workflow. In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 11.09.2018 [cit. 10.03.2019]. Dostupné z: <https://managementmania.com/cs/workflow>

3.2 Metoda Aris

Aris je metodika a nástroj pro analýzu procesů podniku. Původně se jednalo pouze o grafickou metodiku pro analýzu a návrh IS, později však našla uplatnění v procesním řízení a modelování. Metoda umožňuje tvorbu provázaných modelů, které věrně zachycují strukturu produktů, procesů, aplikací, produktů a služeb podniku. Neklade si za cíl vytvoření přesného postupu, jak postupovat při reengineeringu procesů, poskytuje spíše řadu různých pohledů pomocí nichž lze různé situace modelovat. Pohledy se znázorňují pomocí „Domu ARISu“ (viz obrázek 6).



Obrázek 6 - Základní pohledy metody ARIS

Aris (Architecture of Integrated Information Systems, česky pak Architektura integrovaných informačních systémů) byla vyvinuta profesorem Augustem Wilhelmem Sheerem v devadesátých letech minulého století na univerzitě Saarbrueckenu v Německu. Firma nesoucí jeho jméno, IDS Sheer, je současným dodavatelem notace nástroje. Společnost je mimo jiné také známá jako dodavatel informačního systému SAP. Právě spojení nástroje ARIS se systémem SAP je jednou z jeho předností.^{8, 48}

⁸ Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z: <http://www1.osu.cz/~zacek/mopop/mopop.pdf>

⁴⁸ Koudelková, Anna, 2011. Procesní model organizace.[online]. Diplomová práce. Univerzita Pardubice. Dostupné z: <https://docplayer.cz/13662439-Procesni-model-organizace.html>

Toto propojení umožňuje uživateli za předpokladu, že je nadefinován a popsán proces v nástroji ARIS, přímo spouštět funkce informačního systému, případně zjišťovat, kde se v procesu právě nachází nebo jaký bude následovat krok. Zmiňované propojení je současně i nevýhodou nástroje, jelikož neexistuje univerzální rozhraní, které by umožňovalo práci s jinými informačními systémy. V současnosti tento nástroj podporuje i BPMN zmíněné níže.

3.2.1 Diagramy ARIS

Metoda definuje několik diagramů. Pro popis hierarchie organizace existují takzvané Organigramy, které zachycují statickou strukturu a vztahy uvnitř sledované organizace. Pro definici procesů, podprocesů a jejich návazností slouží funkční strom. Tento diagram je vždy reprezentován stejným prvkem, a to čtyřúhelníkem se zaoblenými rohy. Jednotlivé procesy jsou pak zachyceny modelem přiřazených funkcí.⁸

3.3 BPMN (Business process modeling notation)

3.3.1 Stručná historie

Jedná se o grafickou notaci vhodnou k modelování obchodních procesů, za jejímž vznikem stojí iniciativa BPMI (Business Process Management Initiative), která se v roce 2004 sloučila s organizací OMG (Object Management Group) a na vývoji notace se následně podílely společně. BPMN je celosvětově nejrozšířenějším standardem a notací pro modelování firemních procesů.¹²

⁸ Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z: <http://www1.osu.cz/~zacek/mopop/mopop.pdf>

¹² HARMON, Paul a Celia WOLF. The State of Business Process Management

V roce 2005 byla vypracována první verze notace (BPMN 1.0) a v únoru roku 2006 byla přijata organizací OMG jako standard. Cílem této iniciativy byla tvorba notace, která bude čitelná všem entitám, které se účastní životního cyklu procesu, počínaje analytiky na jejím počátku a konče vývojáři a konečnými uživateli a manažery monitorujícími dané procesy a zároveň bude schopna umožnit zachycení všech složitostí procesů.¹³

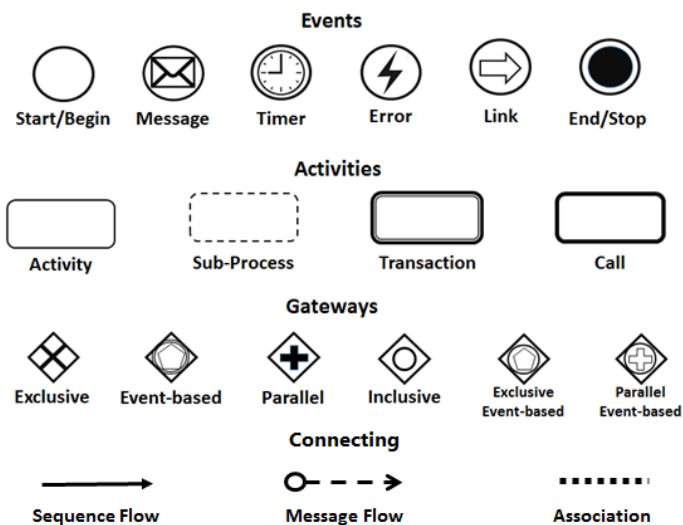
BPMN 2.0 byla vyvíjena v roce 2010 a představena v roce 2013. Vychází z předchozí verze, avšak nejedná se jen o notaci, v mnoha oblastech rozšiřuje původní standard a mění proto název na Business Process Model and Notation.

3.3.2 Diagramy BPMN

Procesní diagram (BPD) vychází z flowchart technologie, která je velmi podobná diagramu aktivit v Unified modeling language a je upraven pro vytváření grafických modelů operací byznys procesů. Jednotlivé elementy, ze kterých je BPD složen se nijak výrazně neliší od zažitých notací, které užívá například jazyk UML. Tvary elementů jsou pevně stanoveny a jediné, co na nich uživatel může měnit je jejich barva. Může také vytvářet prvky nové, nesmí být však shodné s prvky pevně stanovenými v BPMN. Čtyři základní druhy grafických elementů jsou rozepsány na další straně.

¹³ Vašíček, Petr. Úvod do BPMN. Dostupné z: <http://bpm-sme.blogspot.com/2008/03/3-uvod-do-bpmn.html>

- Propojovací objekty (connecting objects)
- Sekvenční tok (sequence flow)
 - Tok zpráv (message flow)
 - Asociace (association)
- Dráhy (swim lanes)
 - Bazén (pool)
 - Dráha (lane)
 - Plovoucí objekty (flow objects)
 - Událost (event)
 - Aktivita (activity)
 - Brána (gateway)
 - Artefakty (artifacts)
 - Datový objekt (data object)
 - Skupina (group)
 - Poznámka (annotation) ^{8, 14, 15}



Obrázek 7 - Grafické symboly BPMN

¹⁴ Modelování podnikových procesů. Dostupné z: <https://www.altaxo.cz/provoz-firmy/management/rizeni-podniku/modelovani-podnikovych-procesu>

⁸ Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z: <http://www1.osu.cz/~zacek/mopop/mopop.pdf>

¹⁵ White, Stephen. 1Process Modeling Notations and Workflow Patterns Dostupné z: http://www.bpmn.org/Documents/Notations_and_Workflow_Patterns.pdf

3.4 UML (Unified Modeling Language)

Unified modelling language je univerzální grafický modelovací jazyk pro tvorbu diagramů. Za pomoci tohoto jazyka jsme schopni modelovat schémata procesu vývoje různých softwarových řešení. Za vznikem jazyka stojí společnost Rational Software a to především tři inženýři Grady Booch, James Rumbaugh a Ivar Jacobsen, kteří v roce 1996 přišli s prvním návrhem UML vyvinutým na základě metody OMT (Object Modelling Technique). Společnost Rational v roce 2003 koupila IBM. Jejich cílem bylo vytvoření unifikované metodiky softwarového inženýrství, pro podporu vývoje objektově orientovaných programů jelikož do té doby existovalo množství metodik lišících se v prezentačních technikách (notacích), metodických postupech (procesech) a dostupných modelovacích nástrojích.

Za další vývoj standardu UML v současné době zodpovídá standardizační organizace Object Management Group, která v roce 1997 převzala tuto metodiku a začala ji doporučovat jako standard, v té době označený jako UML 1.0.¹⁷

3.4.1 Využití

Díky jeho zabudovaným rozšiřovacím mechanismům je vhodným nástrojem pro popis jakéhokoliv systému, jeho chování, struktury. Pro přiblížení níže popisují několik typů UML diagramů, které se ve světě informačních technologií nejčastěji používají. Z důvodu velkého množství různých notací popisují jen ty nejpoužívanější.⁵⁵

¹⁷ UML (Unified Modeling Language). In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 09.08.2016 [cit. 10.03.2019]. Dostupné z: <https://managementmania.com/cs/unified-modeling-language>

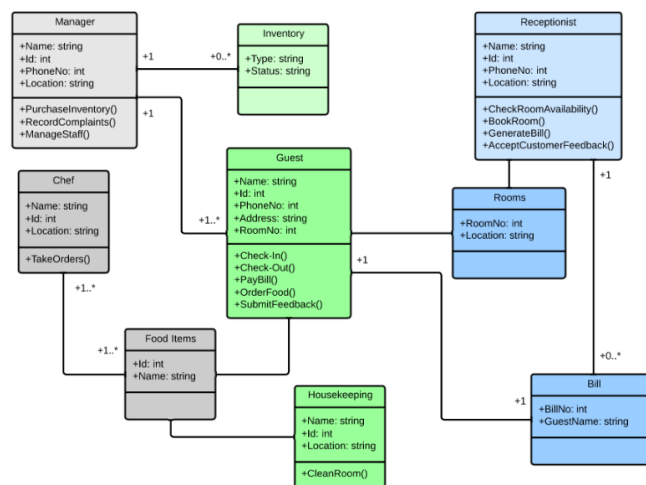
⁵⁵ Richta, Karel, 2003. Unifikovaný modelovací jazyk UML. [online]. Dostupné z: http://www.kiv.zcu.cz/~mautner/Pt/UML_richta.pdf

3.4.2 UML diagramy

UML dnes využívá 13 základních diagramů rozdělených do dvou skupin - Diagramy struktury (structure diagrams) a diagramy chování (behavior diagrams), přičemž diagramy struktury jsou využívanější. Liší se tím, že na rozdíl od diagramů chování jsou statické a nezohledňují časové hledisko.⁶⁰

3.4.3 Třídní diagram

V softwarovém inženýrství je takto nazývaná statická datová struktura popisující třídní uspořádání určitého systému. Hojně využíván je především pro tvorbu ERD, neboli Entity-relation diagramů a při návrhu implementace, tedy druhé fázi vývoje softwaru, kdy se za pomoci těchto diagramů tvoří abstrakce zdrojového kódu, která později slouží programátorům jako zdrojový dokument ve fázi samotné implementace softwaru. Silnou stránkou těchto diagramů je grafické znázornění jednotlivých entit systému, jejich vlastností (atributů) a vztahů.¹⁸



Obrázek 8 - UML Třídní diagram hotelového managementu

⁶⁰ ARIS tool and methodology for Enterprise and Solutions Architecture, 2011. [online]. Dostupné z: <https://viewpointsonitarchitecture.wordpress.com/2011/08/22/aris-tool-and-methodology-for-enterprise-and-solutions-architecture/>

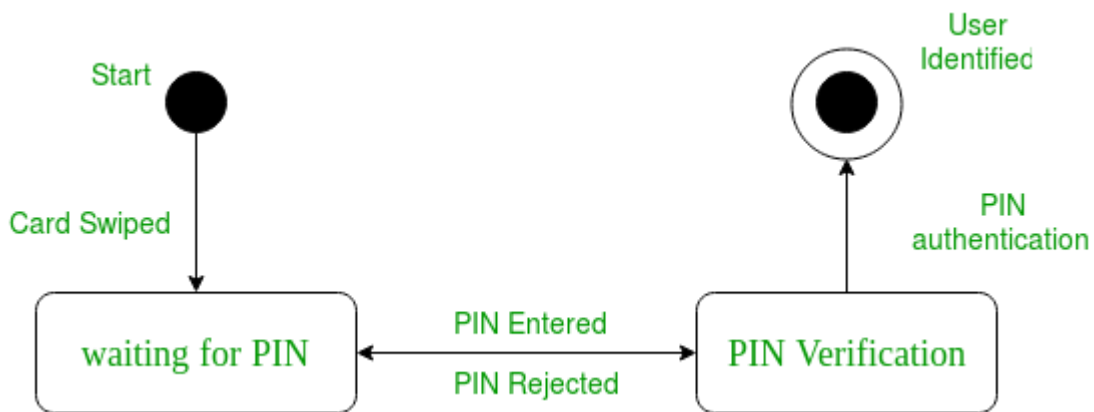
¹⁸ Salma, 2017. UML Class Diagrams Tutorial, Step by Step. [online]. Dostupné z: https://medium.com/@smagid_allThings/uml-class-diagrams-tutorial-step-by-step-520fd83b300b

3.4.3.1 Objektový diagram

Vychází z třídního diagramu. Jeho základem je objekt, tedy konkrétní instance třídy. Jeho úkolem je popis vztahů (asociací) mezi objekty systému v určitém časovém okamžiku. Často bývá také nazýván instančním diagramem. Používá se pro zobrazení konfigurace objektů ve chvíli, kdy třídní, či sekvenční diagram nepostačuje.

3.4.3.2 Stavový diagram

Tento diagram zobrazuje stavy určitého objektu, přechody mezi těmito stavy, inicializátory a podmínky přechodů a akce, které s přechody souvisí. Popisuje reakce objektu na nastalé události (events) a jejich posloupnost v čase. Jedná se o deterministický diagram, takže každý stav v diagramu musí mít jasně definované podmínky, které musí být splněny pro přechod do stavu jiného. Základními prvky takového stavového automatu jsou:



Obrázek 9 – UML Stavový diagram zadání PIN kódu

“

”

„PIN

Verification“)

- Události (viz „Card Swiped“, „PIN Entered“, etc)
- Přechody (viz „Card Swiped“, „PIN Entered/Rejected“, „PIN authentication“)

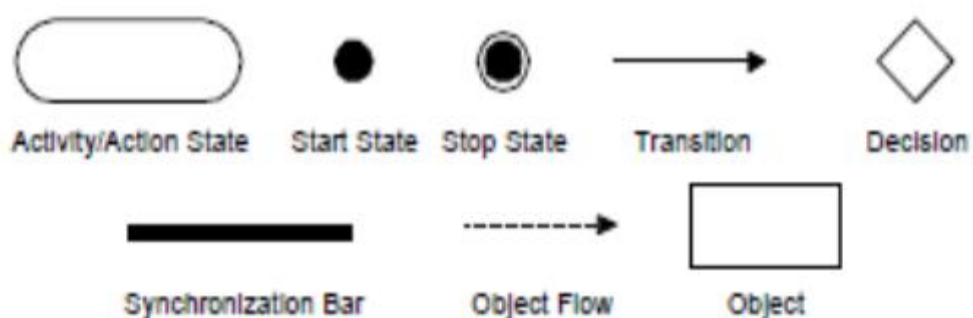
51

⁵¹ Suchan, Jan, 2007. Stavové diagramy – tutoriál. [online]. Dostupné z:

<http://www.minmax.cz/sites/default/files/fel-cvut/semestr04/x36sin/stavovy-diagram-tutorial.pdf>

3.4.3.3 Diagram aktivit

Jedná se o flow chart. Využívá stejné notace jako stavový diagram, i proto se mohou často plést. Slouží ke grafickému zaznamenání procesu, workflow, kdy je každý jednotlivý proces v diagramu reprezentován posloupností kroků. Uplatnění našel při tvorbě minispifikace, tedy analýze algoritmů elementárních funkcí. Diagram aktivit je možné rozšiřovat o nové prvky, jako jsou například takzvané swimlanes, což jsou prvky, které mohou specifikovat datové toky nebo datové objekty. Před uvedením UML 2.0 byl chápán jako specifický případ stavového diagramu, od verze 2.0 získal novou sémantiku, která je založena na grafickém modelovacím jazyce Petriho sítí.^{8,19}



Obrázek 10 - Grafické symboly UML diagramu aktivit

3.4.3.4 Diagram komponent

Vhodně ilustruje závislosti mezi komponenty softwarových řešení, jejich strukturu a vytváří ucelený fyzický náhled na systém. UML komponentami rozumíme entity, které zapouzdřují svůj obsah a jejich chování navenek lze nahradit. Což znamená, že pokud určitou komponentu nahradíme jinou, která poskytuje a vyžaduje stejné chování systému, je vše v pořádku. Interakce se systémem se realizuje buď za pomoci definovaného rozhraní nebo napřímo.²¹

⁸ Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z: <http://www1.osu.cz/~zacek/mopop/mopop.pdf>

¹⁹ ARLOW, Jim; NEUSTADT, Ila. UML 2 a unifikovaný proces vývoje aplikací : Objektově orientovaná analýza a návrh prakticky. Pěel. Bogdan Kizska. 1. vyd. Brno: Computer Press, 2008. 567 s. ISBN 978-80-251-1503-9.

²¹ Komponentové diagramy. [online]. Dostupné z: <http://ocup.ocup.cz/2010/07/komponentove-diagramy-component.html>

3.5 Metoda BORM

3.5.1 Vznik a vývoj

Business Object Relation Modeling neboli zkráceně BORM vznikla původně jako metoda pro tvorbu objektově orientovaných softwarových systémů založených na čistě objektově orientovaných jazycích, jakým je například Smalltalk, a objektových databázích. Později se ale ukázalo, že své uplatnění nalezne i při modelování business procesů. Její vývoj započal v roce 1993 v rámci grantu Vappiens, který je součástí mezinárodního programu „Know how fund of Czech Academic Link Programme“ Britské vlády. Metoda je to tedy o 12 let starší než níže popsané OntoUML.

Od roku 1996 je další vývoj podporován českou pobočkou společnosti Deloitte & Touche ve které je BORM také prakticky využíván a to především pro obchodní řešení. Metoda, především možnost jejího využití během analýzy v počátečních fázích vývoje systému byla v předchozích letech využita v mnoha projektech, počínaje identifikací procesů za účelem analýzy nákladů v pražských nemocnicích a konče správou českých telekomunikačních sítí.

Metoda byla též využita během případové studie informačního systému agrární komorou České republiky, s cílem zoptimalizovat produkci komodit poptávce zákazníků a tím zvýšit zisk celého agrárního sektoru.

Aktuálně za dalším rozvojem stojí především výzkumná skupina CCMi působící na Fakultě informačních technologií ČVUT.²²

3.5.2 Charakteristika

Metoda BORM byla navržena tak, aby pokrývala všechny fáze vývoje software od počáteční business analýzy až po realizaci UML diagramu založeném na principu MDA (Model Driven Approach), avšak největší pozornost klade prvotním fázím vývoje. V tom se liší od jiných metod, které povětšinou pouze utváří objekty a jejich operace na základě podstatných jmen, respektive sloves.

²² Merunka, Vojtěch: BORM - overview of the methodology and case study of the agrarian information systém. Agricultural Economics. Zemědělská ekonomika, 49, 2003, 9, 397-406 ISSN: 0139-570X

Výhodou metody BORM je používání pouze omezené sady pojmů pro každou fázi vývojového cyklu, předpokládá totiž, že v průběhu projektu dochází k přeměnám jednotlivých pojmů na jiné. To má za následek větší srozumitelnost modelu i uživatelům, kteří nejsou tak znalí business analýzy.

Další výhodnou vlastností této metody je i to, že nevyžaduje dělení statických a dynamických pohledů na systém, tím pádem není třeba tvořit různé relační diagramy při různých pohledech na systém. Ke znázorňování konceptuálních a softwarových pojmů používá většinu symbolů shodně s UML, dovoluje však vyjádřit konzistentním způsobem některé detaily softwarové konstrukce a zobrazit tedy více informací. BORM se stal nepoužívanější metodou založenou na jazyce UML.^{44, 24}

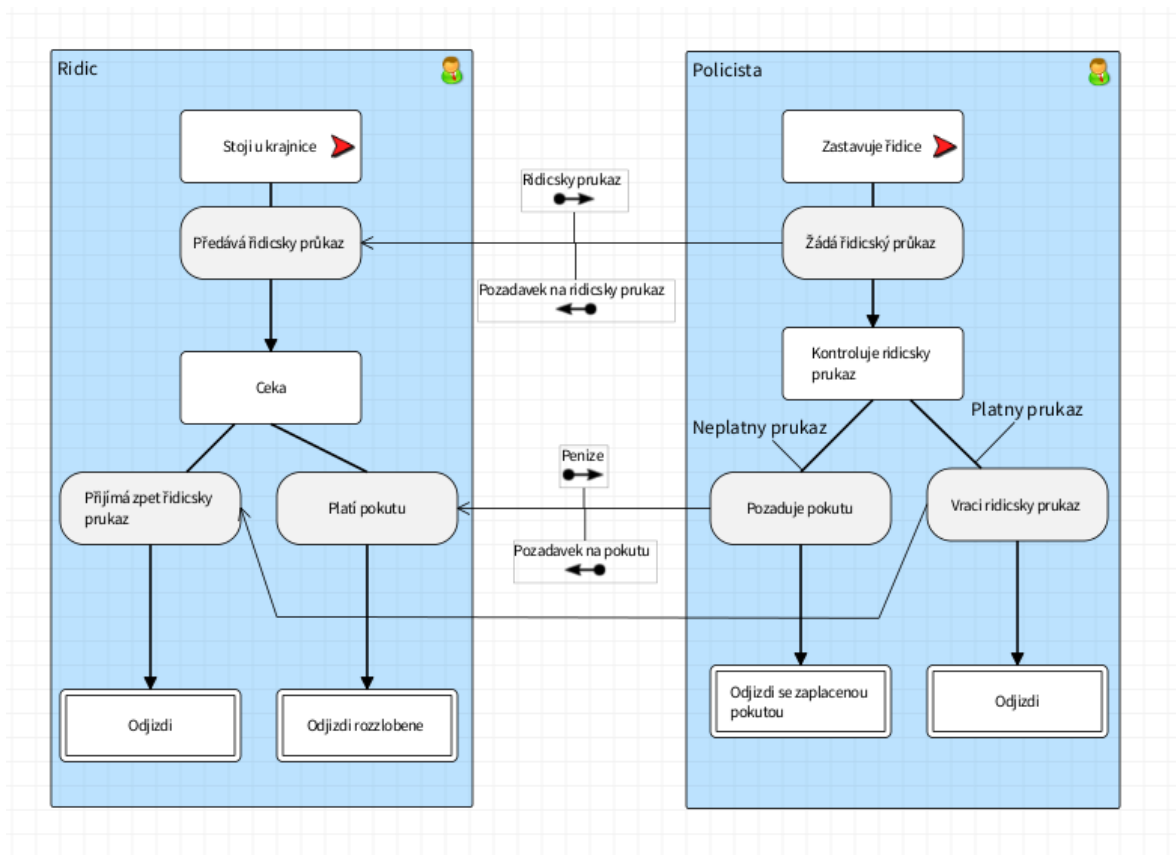
3.5.3 Procesní diagram BORM

Procesní diagram se skládá z množiny participantů, kde každý participant je dále množinou stavů a aktivit. Participant je vyobrazen jako modrý obdélník a reprezentuje entitu účastnící se popisovaného procesu, ať už je to osoba, organizace nebo služba.

⁴⁴ UNIFIED MODELING LANGUAGE SPECIFICATION VERSION 2.5.1, 2017. [online]. Dostupné z: <https://www.omg.org/spec/UML/About-UML>

²⁴ Merunka, Vojtěch, 1999. VYUŽITÍ METODY BORM PRO BUSINESS PROCESS REENGINEERING. [online]. Dostupné z: <http://www.agris.cz/clanek/104163>

Stavy participanta jsou znázorněny obdélníky a vzájemně propojeny pomocí přechodů, které jsou popsány šipkami. Participant mění své stavy na základě aktivit, které jsou zobrazeny pomocí oválů. Dvě aktivity, z nichž každá se nachází v jiném participantovi mohou být navzájem propojeny pomocí komunikace po které mohou v obou směrech nebo jednosměrně proudit datové toky. Procesní diagramy jsou velmi důležitým výstupem procesní analýzy systému a současně důležitým vstupem metody BORM. ²⁵



Obrázek 11 - Procesní diagram BORM



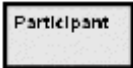





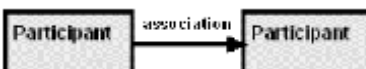

3.5.4 Konceptuální diagram BORM

Konceptuální model vychází z obchodního modelu, kdy se vybírají objekty, které reprezentují utvářený informační systém. Metoda BORM využívá pro tvorbu těchto diagramů mírně modifikovaný modelovací jazyk UML. Modifikace spočívá v bodech zmíněných na další stránce.

²⁵ Maxa, Oskar, 2015. Implementace procesní analýzy OR diagramů v metodě BORM.. Bakalářská práce. Univerzita Karlova v Praze. Vedoucí práce RNDR. Filip Zavoral, Ph.D.

1. Na rozdíl od klasického UML zohledňuje polymorfismus, dědění, hierarchii typů a hierarchii „AS A“.
2. Používá samostatný grafický symbol pro metodu.
3. Graficky rozlišuje pojmy třída objektů a množina objektů.²⁶

3.5.5 Grafické elementy notace BORM

Element	Graphic symbol	Description
Beginning of the role		Beginning of the action flow of a role.
End of the role		End of the action flow of a role.
Participant = WHO performs the role		Participant has some activities in the process
Activity = WHAT is done in the role		Every action is done by somebody in BORM. Activity is an active or passive (invoked by another participant) action.
State = WHEN something happens		Point in time where the process waits or something is done.
Communication		Control flow between activities. Crossed symbol indicates conditional communication.
Data flow		Exchange of information, data, money, etc.
Transition between states		Linkup between states in time. Crossed symbol means conditional transition.
Association = RELATION between participants		Connection or relation between participants (eg. ownership, dependency, ...).
Participant hierarchy = „IS-A“ taxonomy		When it is necessary to show that a participant is a special type of another participant.

Obrázek 12 – Grafická notace metody BORM

²⁶ Pícka, Marek, 2010. Vývoj objektově orientovaných informačních systémů pomocí metody postupných transformací. [online]. Disertační práce. Vedoucí práce Prof. Ing. Ivan Vrana DrSc. Dostupné z: <https://docplayer.cz/97950991-Vyvoj-objektove-orientovanych-informacnich-systemu-pomoci-metody-postupnych-transformaci.html>

3.5.6 Fáze životního cyklu

BORM stejně jako jiné objektově orientované metody je založen na spirálním modelu vývoje životního cyklu. Jednotlivými fázemi tohoto cyklu jsou strategická analýza, počáteční analýza, pokročilá analýza, úvodní návrh, implementace, fáze testování a provozu (viz obrázek 12).⁵⁵

Prvním třem fázím se také říká fáze rozvoje nebo expanze, účastní se jí především zadavatelé, tedy zákazníci, manažeři a zaměstnanci. Jde primárně o nasbírání co nejvíce informací o daném problému, definování rozhraní systému a zmapování všech požadovaných procesů. Rozvoj končí s dokončením detailního konceptuálního modelu za použití prostředků objektově orientovaného programování, který detailně popisuje řešení požadovaného problému. Tento model je jakýmsi dorozumivacím prostředkem mezi zadavateli a vývojovým týmem. Zadavatele nezatěžuje detaily počítačové implementace a vývojáři naopak nepodstatnými fakty kolem reálného řešeného problému.²⁷

Fáze úvodního návrhu je prvním momentem při kterém dochází ke snaze sestavit systém tak, aby byla možná jeho softwarová implementace.

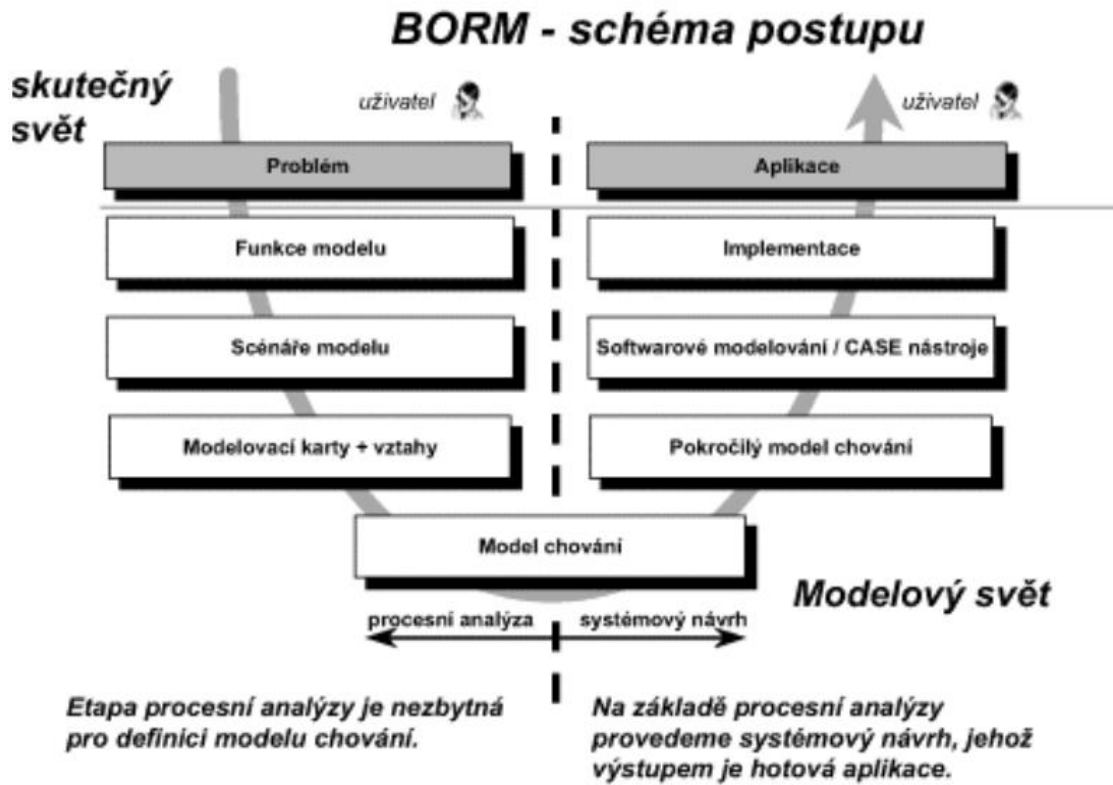
Další fází je fáze implementace, často také nazývána fází konsolidace, či ustálení. Tedy období, kdy se přetváří již dříve vytvořený konceptuální model a myšlenky do podoby softwarového řešení. Této fázi se účastní především vývojáři. Nezřídka se stává, že se během této fáze musí z časových nebo jiných důvodů některé nápady a myšlenky vypustit a do řešení neimplementovat.²⁸

⁵⁵ Yourdon, Edward. (2019). *Mainstream Objects : An Analysis and Design Approach for Business*

²⁷ POLÁK, J, V MERUNKA a A CARDA. *Umění systémového návrhu*. 2003.

²⁸ Merunka, Vojtěch, (2010). OBJECT-ORIENTED PROCESS MODELING AND SIMULATION. BORM EXPERIENCE. *Trakia Journal of Sciences*. Dostupné z: <http://www.uni-sz.bg>

Tím ale životní cyklus systému nekončí. Ve chvíli, kdy je systém implementován nastává období ostrého provozu a údržby. Jak poznamenává doc. Merunka, pracnost, časová a finanční náročnost této fáze se rovná 100 % až 400 % součtu pracnosti fází předešlých.²⁷



Obrázek 13 - Fáze BORM

²⁷ POLÁK, J, V MERUNKA a A CARDA. Umění systémového návrhu. 2003.

3.5.7 CASE nástroje

Již při vývoji metodologie BORMu byl její součástí výběr vhodného CASE nástroje. Do roku 2005 byl hlavním nástrojem pro tvorbu modelů pomocí metody BORM CASE nástroj finské společnosti MetaCase Ltd. CASE MetaEdit+ sice neobsahoval vestavěnou podporu pro BORM, ale umožňoval snadnou implementaci nových metod pomocí definování jejího metamodelu. Jeho součástí byl i skriptovací jazyk ve kterém jsou předdefinovány výstupy ve formátech .txt, .gif, .rtf, .html nebo .pct. Aplikace byla vyvinuta v jazyce SmallTalk v prostředí VisualWorks. Od roku 2005 je to komerční software vyvinutý v České republice firmou e-Fractal Craft.CASE na zakázku společnosti Deloitte&Touche, což je modelovací nástroj šitý na míru potřebám metody BORM, který není nijak náročný na prostředky počítače. Stejně jako MetaEdit+ byl Craft.Case napsán v jazyce SmallTalk. Pro účely této práce byl využit CASE nástroj OpenPonk.^{31, 24, 25}

3.6 OntoUML

3.6.1 Ontologie

Slovo ontologie bylo podle Filozofického slovníku poprvé užito v 17. století německým filozofem Rudolfem Gockelem (1547 - 1628). Pojem jako takový můžeme obecně chápat ze dvou různých úhlů pohledu, filozofického a infromatického.

3.6.1.1 Filozofické hledisko

Z toho filozofického se jedná o vědní disciplínu zabývající se bytím, jsoucnem v jeho nejobecnějším určení. Ve Slovníku filozofie se můžeme dozvědět, že podle Christiana Wolffa (1679-1754), německého osvícenského filozofa, vychází ontologie z metafyziky, která se takto skládá z psychologie, ontologie, teologie a racionální kosmologie a za jejíhož zakladatele se považuje řecký filozof Aristoteles. Liší se od metafyziky v tom, že zkoumání jsoucná a bytí entit rozšiřuje o zkoumání jejich vztahů.

²⁴ Merunka, Vojtěch, 1999. VYUŽITÍ METODY BORM PRO BUSINESS PROCESS REENGINEERING. [online]. Dostupné z: <http://www.agris.cz/clanek/104163>

²⁵ Maxa, Oskar, 2015. Implementace procesní analýzy OR diagramů v metodě BORM.. Bakalářská práce. Univerzita Karlova v Praze. Vedoucí práce RNDR. Filip Zavoral, Ph.D.

³¹ Merunka, Vojtěch. Metoda BORM. [online]. Dostupné z: <https://docplayer.cz/2267840-Vojtech-merunka-metoda-borm.html>

Například podle Bordase se jedná o jméno vytvořené samotným Wolffem jako ekvivalent k obecnější metafyzice. Ontologie se obecně snaží odpovědět na základní filozofické otázky, jako jak vznikl svět? Nebo co je lidská existence.^{32, 43}

3.6.1.2 Ontologie z hlediska počítačových věd

V 70. letech minulého století se objevila myšlenka využití ontologie v oblasti vývoje databází a to konkrétně na úrovni konceptuálního datového modelování. Důvodem bylo nahrazení ER modelu, který byl pro konceptuální model do té doby používán avšak popisoval pouze entity a vztahy mezi nimi, nezohledňoval datové operace, které budou probíhat. Ontologie z hlediska informačních technologií je slovníkem pro přenos poznatků týkajících se určité problematiky.

⁴³

Zahrnuje v sobě taxonomii a soubor odvozovacích pravidel. Dle panů Svátka a Vojtěcha (Svátek, Vojtěch, 2002) jde především o korektní definici tříd, podtříd objektů a jejich vztahů a to nezávisle na lidském úsudku o světě takovém, jaký je. Cílem ontologie jako infromatické disciplíny je definice znovupoužitelného modelu, který umožní lepší porozumění problému a podpoří pochopení fungování obsáhlejších systémů. Ontologie je dnes využívána v mnoha oborech jako třeba v umělé inteligenci, při tvorbě rozsáhlých webových vyhledávačů, v softwarovém inženýrství nebo v systémovém inženýrství pro tvorbu datových modelů, jako tomu bude v této práci.³³

³² Marek, Jan, 2017. Model klienta veřejné správy z pohledu Unified Foundational Ontology. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Prof. Ing. Václav Řepa, CSc.

³³ L. Benyovszky, Náhlost. Myšlení bytí z času. Praha: Oikumené 2006, 1. Filozofický slovník, Praha 1985, s. 21, Filozofický slovník, Praha 1986, s. 343., 3. G. Legrand: Vocabulaire Bordas de la philosophie, Paříž 1986. s. 242.

⁴³ Guskova, Natalia, 2018. Konceptualizace vybraných částí modelu bezpečnosti STAMP.

3.6.2 Historie metody OntoUML

Jak bylo zmíněno výše, metoda OntoUML je podstatně mladší než BORM. Její základy můžeme hledat v disertační práci Giancarla Guizzardiho zvané „*Ontological foundations for structural conceptual models*“, kterou s nejvyššími poctami obhájil na univerzitě v Twente v roce 2005. Modelovací jazyk OntoUML je postaven na ontologii UFO o které se čtenář dočte v kapitole níže. Jedná se o rozšíření jazyka UML, které je syntakticky postaveno nad notací třídních diagramů, kterou dále rozšiřuje o nové stereotypy. V jazyce OntoUML jsou třídy reprezentovány pomocí čtverců, kdy každý čtverec obsahuje název třídy a stereotyp (koncept ontologie UFO).^{32, 43}

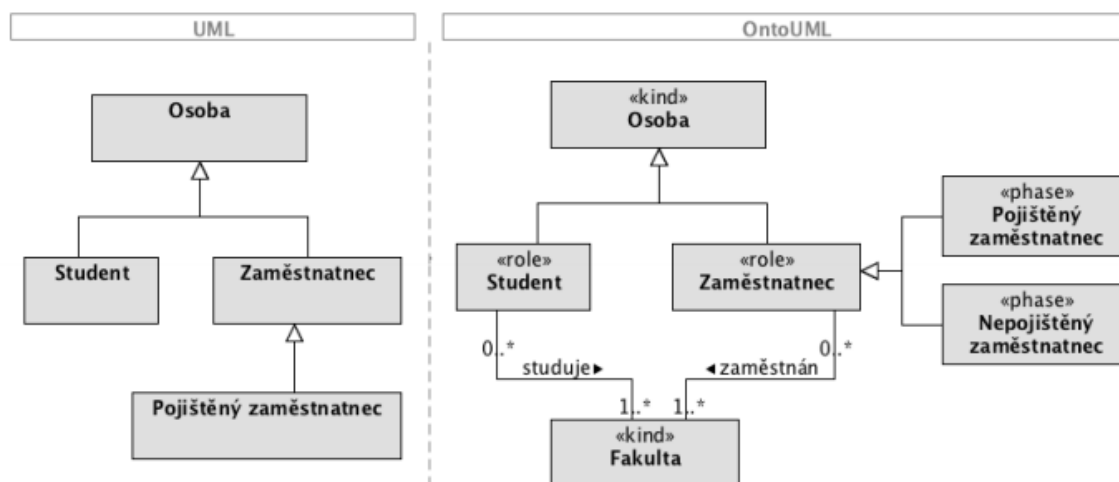
Metoda si klade za cíl spojení ontologické analýzy a konceptuálního modelování, maximalizaci výraznosti, pravdivosti a konceptuální jednoznačnosti.³⁵

³² Marek, Jan, 2017. Model klienta veřejné správy z pohledu Unified Foundational Ontology. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Prof. Ing. Václav Řepa, CSc.

⁴³ Guskova, Natalia, 2018. Konceptualizace vybraných částí modelu bezpečnosti STAMP.

³⁵ Pergl, Podloucký, 2014. OntoUMLa UFO-A Formalizace strukturálních biomedicínských znalostí. [online]. Dostupné z: http://www.euromise.net/wp-content/uploads/2014/05/Podlouck%C3%BD_Pergl.pdf

3.6.3 UFO



Obrázek 14 - Srovnání UML a OntoUML

I základy Unified Foundational Ontology můžeme hledat v dizertační práci doktora Giancarla Guizzardiho. Jedná se o ontologii vyššího stupně, ve které jsou nadefinované koncepty použitelné v jazycích založených na ontologii. UFO v současné době zahrnuje několik základních ontologií (Foundational Ontologies), UFO-A, UFO-B, UFO-C a UFO-S z nichž jen jedna, UFO-A, se považuje za dokončenou a otestovanou na mnoha případech v praxi, její další vývoj se tedy neočekává. Naopak metody UFO-B a UFO-C jsou v současné době předmětem intenzivního vývoje.

Všechny tyto základní ontologie fungují jako referenční model, který předepisuje koncepty, které by měly být konceptuálním jazykem dodržovány a utváří sémantiku pro jazyk popisující tyto koncepty. Dva základní koncepty UFO jsou enduranty a perduranty. V prvním případě se jedná o popis objektů a ve druhém o koncept událostí a procesů.

Za rozvojem UFO dnes stojí společnost NEMO, jejíž součástí je i Giancarlo Guizzardi.^{32, 43}

³² Marek, Jan, 2017. Model klienta veřejné správy z pohledu Unified Foundational Ontology. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Prof. Ing. Václav Řepa, CSc.

⁴³ Guskova, Natalia, 2018. Konceptualizace vybraných částí modelu bezpečnosti STAMP.

3.6.4 CASE nástroje

Pro tvorbu OntoUML diagramů existuje několik modelovacích aplikací, ve kterých je uživatel schopen namodelovat ontologický model, zkontrolovat jeho syntaxi a zvalidovat.

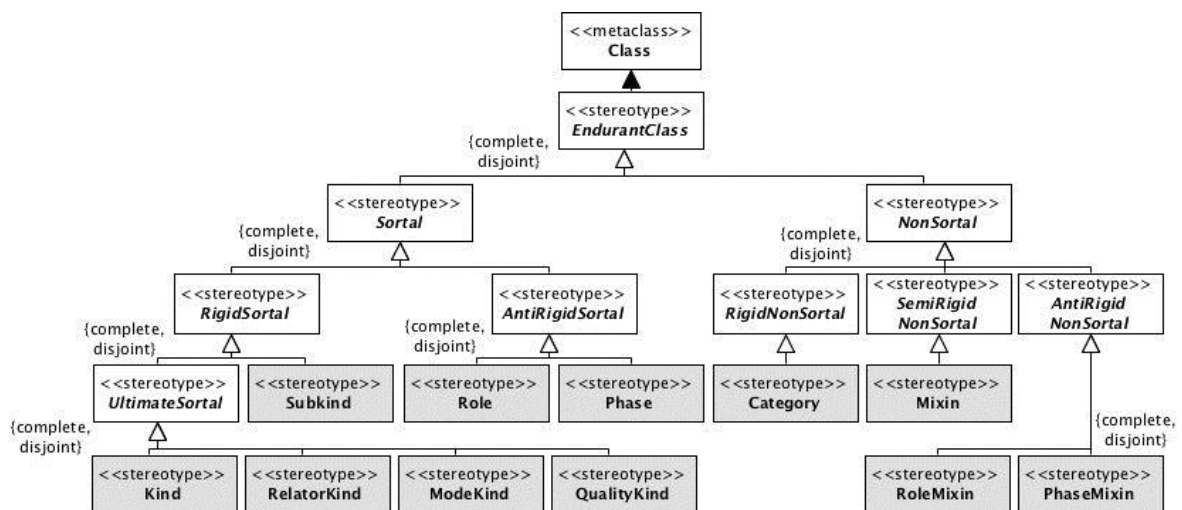
3.6.4.1 OLED

OLED neboli OntoUML lightweight editor je jednoduché vývojové prostředí výhradně pro tvorbu OntoUML konceptuálních diagramů. Naprogramován byl v jazyce Java. V tomto nástroji byly sestaveny i modely zmiňované v praktické části této práce.

3.6.4.2 Menthor

V roce 2016 studenti ze skupiny NEMO založili startup zvaný Menthor a následně vytvořila produkt Menthor Editor, což je open source nástroj napsaný v jazyce Java pro vytváření OntoUML modelů. Nástroj je dostupný pouze v anglickém jazyce na platformách Windows, Linux a OS X.

3.6.5 Stereotypy OntoUML



Obrázek 15 - OntoUML metamodel stereotypů

3.6.5.1 Třídní stereotypy

- Kind
Poskytuje identitu, jedná se o základní a nejpoužívanější prvek. Nemůže dědit od jiného stereotypu typu «Kind». Například - Osoba.
- Subkind
Konstrukt používaný pro specializaci stereotypů poskytujících identitu («Kind», «Relator», «Mode», «Collective», «Quantity»). Vždy musí mít svůj nadtyp. Například – Žena.
- Relator
Konstrukt, který musí existovat pro dva prvky propojené materiální vazbou. Reprezentuje skutečnost za tímto vztahem. Například – Manželství, Autorství.
- Role
Specializuje stereotypy, které poskytují identitu («Kind», «Collective», «Quantity», «Relator», «Mode»). Například – Student.
- Phase
Stejně jako «Role» specializuje stereotypy poskytující identitu, ale jen ty, které v čase mění své vlastnosti, jako věk u osoby nebo barva určitého objektu. Například – Rozbitý.

53, 54

⁵³ Suchánek, Marek. OntoUML Specification. [online]. Dostupné z: <https://ontouml.readthedocs.io/en/latest/classes/sortals/role/index.html>

⁵⁴ Rostislav Programming. OntoUML. [online]. Dostupné z: <http://www.rosprog.4fan.cz/navrh-software/ontouml/>

3.6.5.2 Vztahové stereotypy

- **Material**
Reprezentuje materiální vztah mezi dvě entitami, jenž je zprostředkován stereotypem «Relator».
- **Mediation**
Vztah mezi entitami, které jsou provázané pomocí stereotypu «Relator» a právě konstruktem «Relator».
- **Derivation**
Nepovinný vztah značící z jaké materiální vazby byl odvozen «Relator».

53, 54

Z důvodu velkého množství stereotypů využívaných v OntoUML zmiňuji jen ty, které se objevují v testovaných diagramech v praktické části práce.

3.6.6 Překážky většího uplatnění nástroje

Navzdory tomu, že OntoUML našlo uplatnění jak v akademickém, tak komerčním a státním sektoru, materiálů ze kterých lze čerpat poznatky o této metodě je velmi málo. Hlavním důvodem je to, že většina těchto materiálů je komerčního rázu, tudíž obsahuje informace, které nejsou veřejně publikovatelné. Dalším důvodem je to, že veřejně dostupná literatura je ve většině případů psána pod záštitou akademických institucí v cizím jazyce, což nikterak nenapomáhá jejímu šíření. Jazyku OntoUML se v současné době věnuje jen malý počet výzkumných zařízení na světě, především UFES v Brazílii, University of Twente v Nizozemsku, University of Trento a české VŠE a ČVUT.³²

³² Marek, Jan, 2017. Model klienta veřejné správy z pohledu Unified Foundational Onto-logy. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Prof. Ing. Václav Řepa, CSc.

⁵³ Suchánek, Marek. OntoUML Specification. [online]. Dostupné z: <https://ontouml.readthedocs.io/en/latest/classes/sortals/role/index.html>

⁵⁴ Rostislav Programming. OntoUML. [online]. Dostupné z: <http://www.rosprog.4fan.cz/navrh-software/ontouml/>

4 Test použitelnosti

4.1 Cíl studie

Cílem studie je na základě prostudování vhodné literatury a jiných zdrojů namodelovat diagramy odpovídající grafickým notacím nástrojů popsanych v teoretické části této práce. Následné provedení testů použitelnosti na vybrané skupině participantů a poté na základě rozhovoru s participanty po skončení testů použitelnosti zjistit, která ze dvou sledovaných metod je pro uživatele přívětivější a snazší k pochopení a vyhodnocení jejich návrhů na zlepšení as-is notací.

4.2 Metodika výzkumu

4.2.1 Úvod

K výzkumu byly zvoleny kvalitativní testy použitelnosti. Testy použitelnosti jsou definovány standardem ISO 9241 pokrývajícím ergonomii interakce mezi člověkem a počítačem. Jedná se o velmi důležitý nástroj pro zjištění toho, jak uživatelsky přívětivé je pro uživatele testované grafické rozhraní.

4.2.2 HUBRU

Testování použitelnosti bylo provedeno v kolaborativní laboratoři pro studium lidského chování HUBRU na půdě ČZU v Praze.

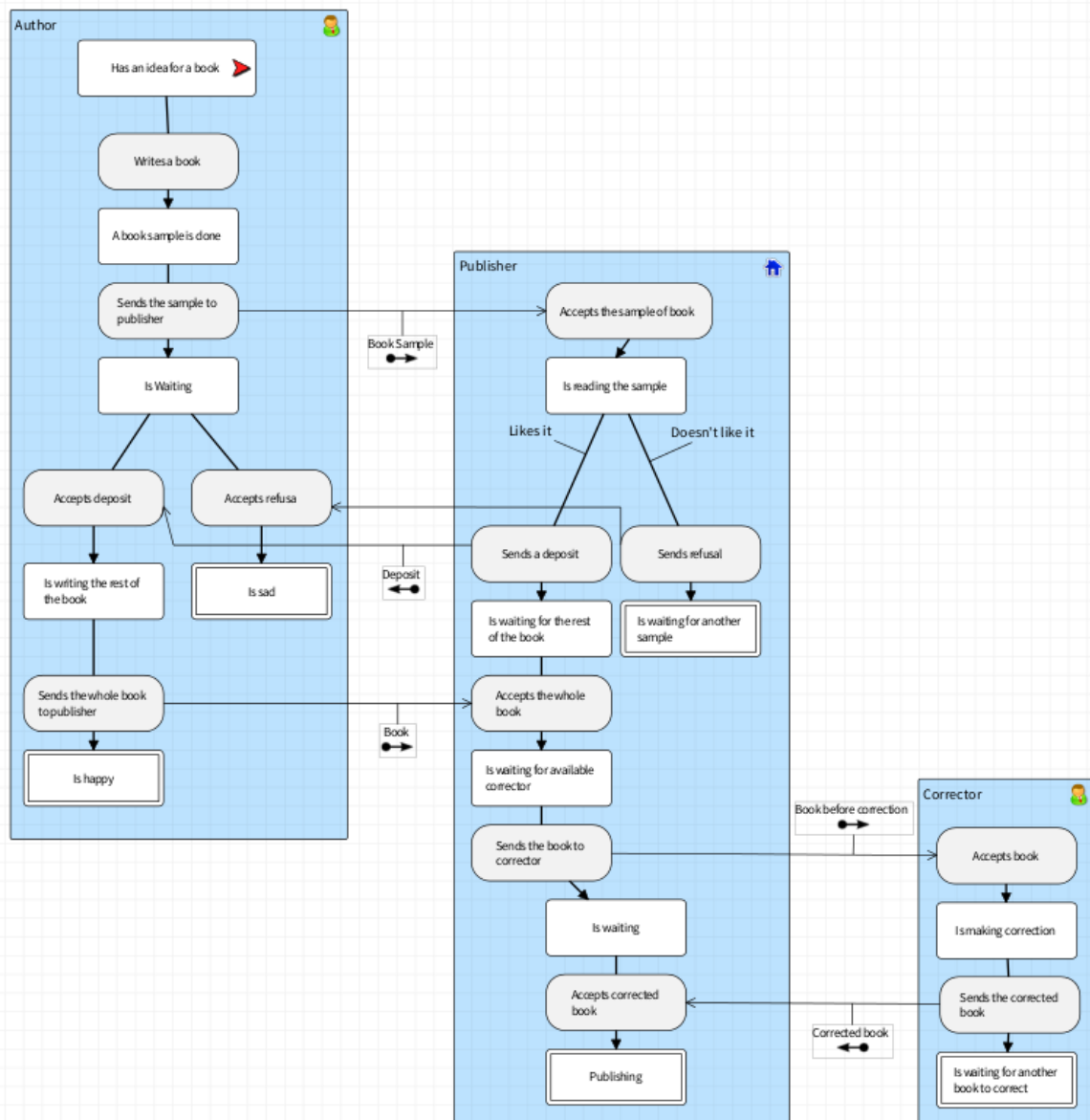
4.2.3 Zúčastnění participantů

Testu v laboratoři HUBRU se zúčastnili 4 lidé, z nichž 2 měli základní povědomí o různých modifikacích dynamických a statických diagramech, zbylí dva participantů se s podobnými diagramy nikdy neseťkali. Dále bylo otestováno 10 participantů na dálku s pomocí nainstalovaného pluginu v prohlížeči participantů, díky kterému bylo možné sledovat pohyb kurzoru po obrazovce počítače participanta.

4.3 Test

Před samotným testem byli všichni participanti seznámeni s tím, co je během testu čeká a uvědomění o tom, že testovány jsou grafické notace jednotlivých nástrojů, nikoliv oni. K testování jim poté byly předloženy diagramy na obrázcích níže s podotázkami ke každému z nich. Po skončení testování proběhlo skupinové interview v případě testování v HUBRU, individuální interview v případě, že se test odehrál na soukromé počítačové stanici participanta, kde se zjišťovalo, co dělalo uživatelům problém, co se jim líbilo (tzv. Likes) a nelíbilo (tzv. Dislikes).

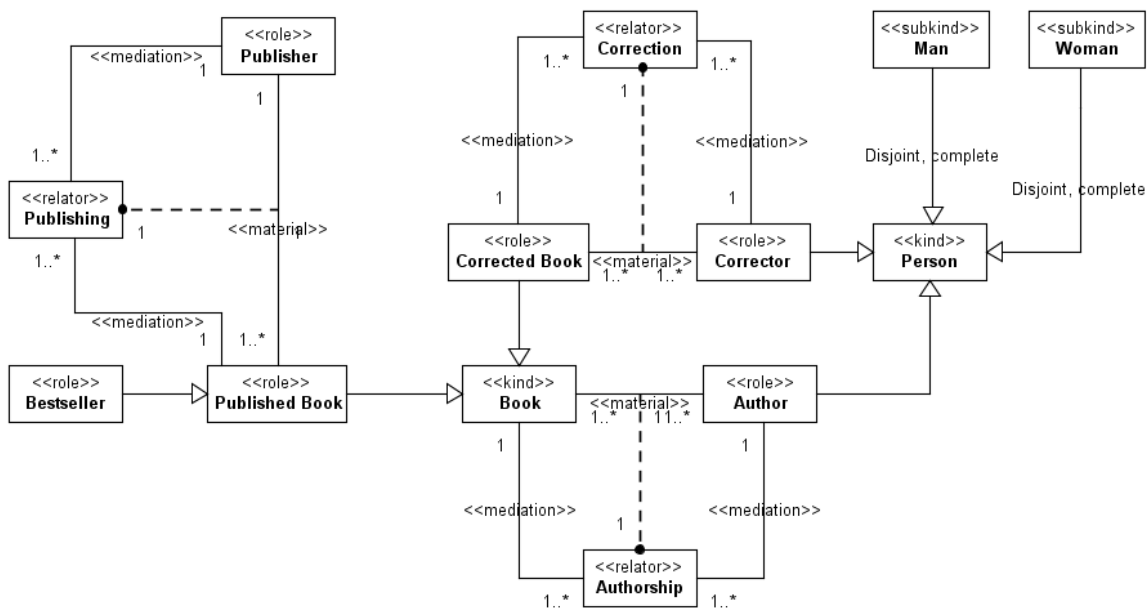
4.3.1 BORM procesní diagram vydání knihy



Obrázek 16 - Procesní diagram BORM - vydání knihy

Otázky zněly takto:

1. Zda si myslí, že porozuměli tomu, co diagram zobrazuje.
2. Co je inicializačním (počátečním) stavem diagramu.
3. V případě, že rozpoznali počáteční stav, na základě čeho se tak stalo.
4. Jestli se v diagramu objevil grafický prvek, kterému nejsou schopni porozumět, případně který.
5. Kolik a případně jací participantů se v diagramu objevují.
6. Kolik a případně jaké konečné stavy mohou nastat.
7. Jaké datové toky a jakým směrem diagramem proudí.

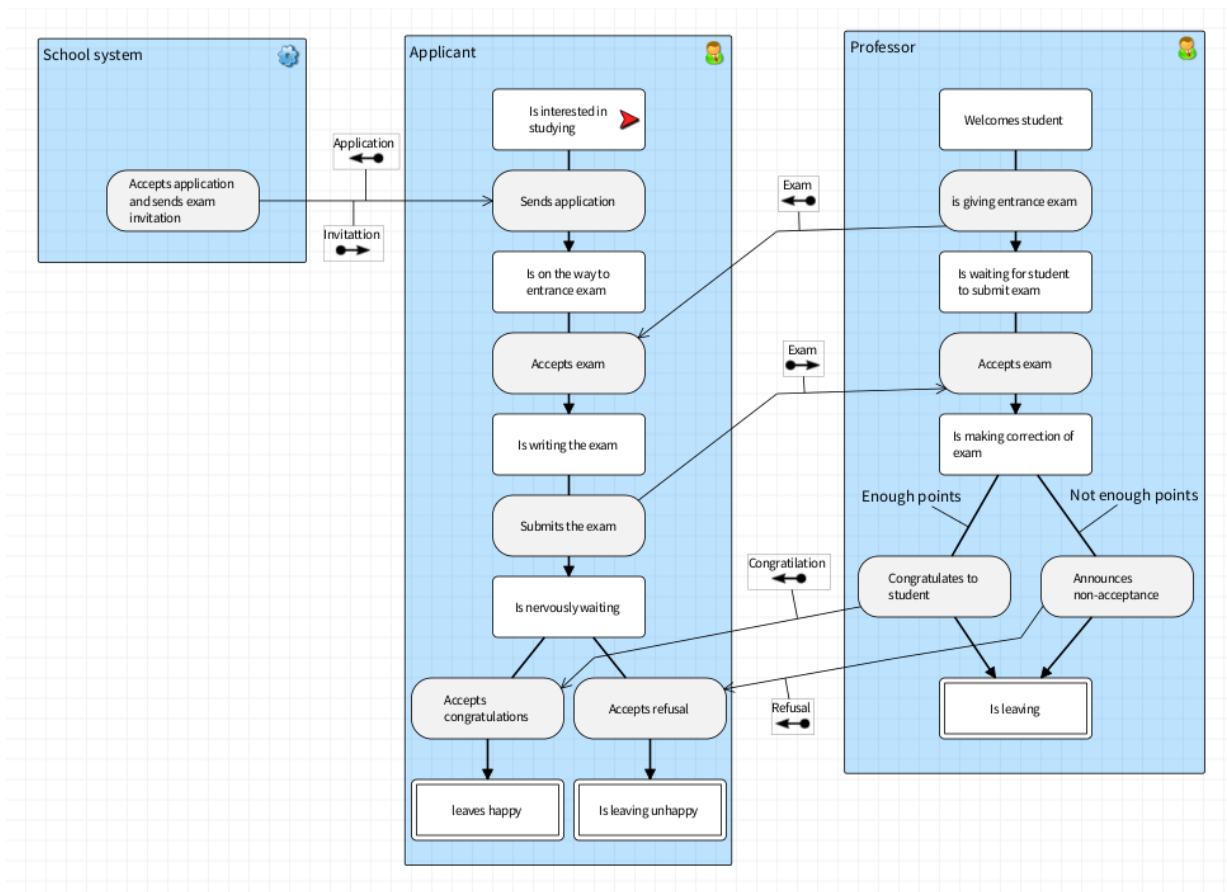


Obrázek 17 - Konceptuální diagram OntoUML - vydání knihy

U konceptuálního modelu OntoUML – vydání knihy namodelovaném v nástroji OLED bylo úkolem participantů porozumět struktuře a vztahům v diagramu.

Kladené otázky zněly takto:

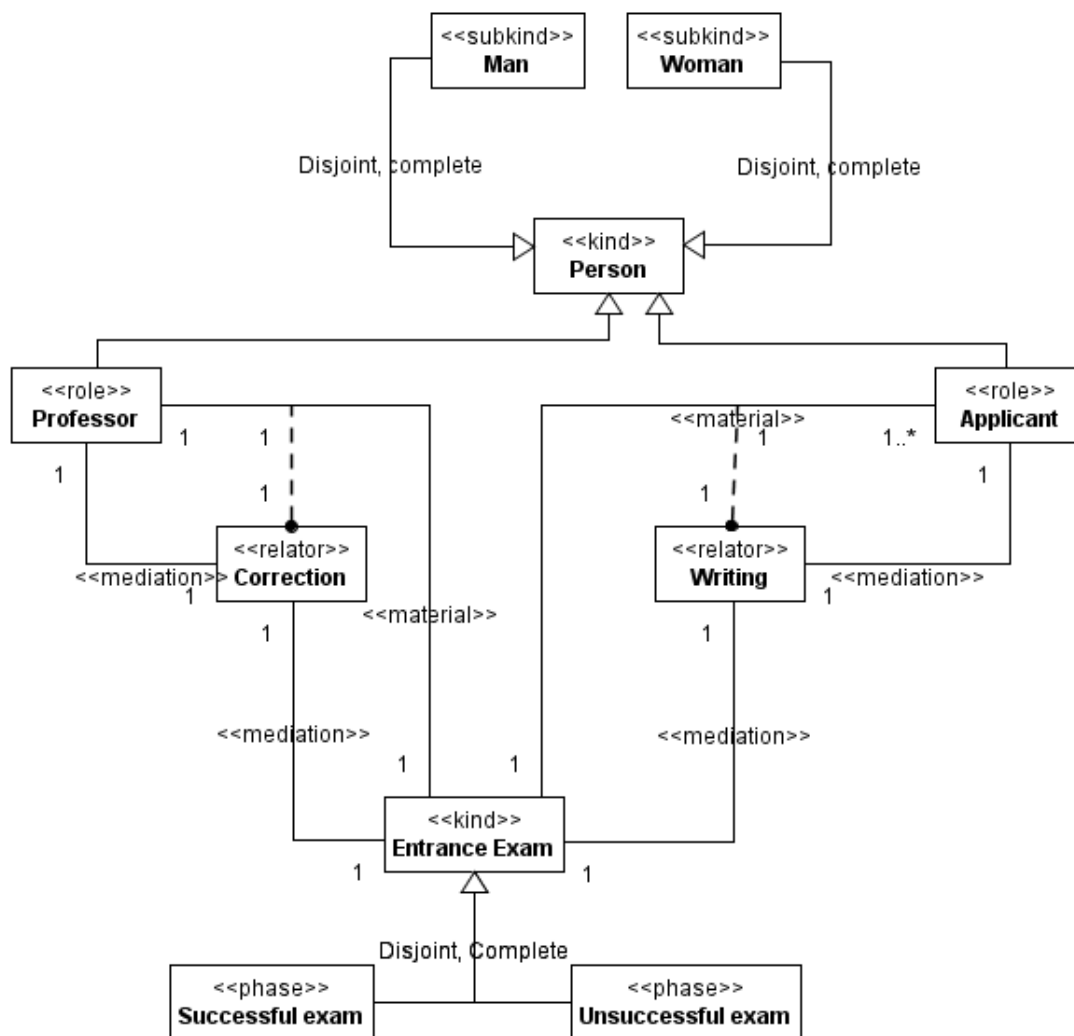
1. Myslíte si, že jste porozuměli namodelovanému diagramu.
2. Jednou větou popsat, co diagram zobrazuje.
3. Pokud se v modelu vyskytl grafický element, kterému špatně rozumíte, který to je.
4. Jaké objekty se na diagramu objevují.
5. Z jakého objektu vychází „Corrector“.
6. Z jakého objektu vychází „Author“.
7. Kolik korektorů může mít podle diagramu jedna kniha.
8. Kolik autorů může mít podle diagramu jedna kniha.



Obrázek 18 - Procesní diagram BORM – přijímací řízení

U procesního diagramu metody BORM – zápis kurzu měli za úkol zodpovědět:

1. Kolik, případně jací participanti se objevují v diagramu.
2. Jaké datové toky můžeme v diagramu vidět.
3. Co je počátečním (inicializačním) stavem diagramu.
4. V případě, že počáteční stav rozpoznali, na základě čeho se tak stalo.
5. Jaké konečné stavy mohou nastat.
6. Jaké aktivity mohou nastat po stavu „Oprava text“ a na základě čeho se tak může stát.
7. Jaké datové toky proudí diagramem.



Obrázek 19 - Konceptuální OntoUML diagram - přijímací řízení

K diagramu na obrázku 18 se vztahovaly otázky podobné těm k obrázku 16:

1. Myslíte si, že jste porozuměli namodelovanému diagramu.
2. Jednou větou popsat, co diagram zobrazuje.
3. Pokud se v modelu vyskytl grafický element, kterému špatně rozumíte, který to je.
4. Jaké objekty se na diagramu objevují.
5. Z jakého objektu vychází „Applicant“.
6. Z jakého objektu vychází „Professor“.
7. Kolik profesorů může podle diagramu opravovat jeden test.

5 Výsledky a diskuze

Na základě rozhovoru s participanty po skončení testování bylo zjištěno toto:

BORM

- Diagramy označili participanti za velmi přehledné a to i ti, kteří s žádnými podobnými diagramy nikdy do styku nepřišli.
- Bylo jasně zřejmé, kam daný proces může směřovat a co je jeho inicializačním a konečným stavem.
- Uživatelé by ocenili větší barevné rozlišení.
- Směrování a obsah datových toků byl jasně srozumitelný

OntoUML

- Participantům dělalo problém pochopit kardinalitu vztahů mezi objekty
- Participanti, kteří se v minulosti s nějakými statickými diagramy setkali, kardinalitu ve většině případů pochopili.
- Vztah derivace vyznačen čárkovanou čarou společně se vztahy «Mediation» a «Material» nebyl participanty pochopen.
- Stejně jako u BORM procesního diagramu by uživatelé ocenili barevné rozlišení jednotlivých objektů za účelem zpřehlednění celého modelu.
- U OntoUML modelů povětšinou pochopili, že svou strukturou navazují na procesní diagramy BORM
- To z jaké třídy objekty vychází dělalo participantům problém

Participanti pochopili, že diagramy mají společné prvky, ale každý zobrazuje něco jiného, OntoUML statickou datovou strukturu systému, zatímco BORM konkrétní proces v rámci systému.

5.1 Další doporučení

Především mladší z obou nástrojů OntoUML by zasloužil další zkoumání použitelnosti. Možností využití pro tento nástroj přistupující k modelování ontologicky je mnoho, avšak praxe podle toho dnes nevypadá a přednost se dává jiným metodám, jako je například UML. Podrobnější a rozsáhlejší studie použitelnosti OntoUML by mohla napomoci transformaci jeho grafické notace k přívětivější podobě.

Zvláště pak zajímavé by bylo srovnání nástrojů OntoUML a UML, jehož notaci metoda BORM používá k tvorbě diagramu při konceptuální fázi vývoje informačního systému, z pohledu UX.

5.2 Limitace studie

1. Jednou z limitací této studie je to, že většina prací týkajících se jak metody BORM, tak OntoUML jsou psány vždy pouze menší skupinou autorů. V případě metody BORM jde především o jejího zakladatele docenta Merunku a v případě OntoUML o skupinu akademických pracovníků z Brazílské univerzity Federal Espirito do Santo ve městě Victória seskupujících se kolem autora OntoUML Giancarla Guizzardiho a skupinu akademiků z Českého vysokého učení technického v Praze.
2. Velké množství materiálů týkajících se metody OntoUML a jejího použití je psáno v Portugalské, což značně omezuje její další rozšíření mezi širší veřejnost.
3. Stejný problém nastává u metody BORM, kdy je množství materiálů psáno v českém jazyce.

6 Závěr

Z provedených testů vyplývá, že oba grafické zápisy jsou většinou lidí alespoň do jisté míry pochopitelné. V případě procesního diagramu BORM trvalo uživatelům kratší dobu zorientovat se v rozložení jeho elementů a jejich vzájemné provázanosti, čemuž přispívalo především jeho rozložení, kdy posloupnost aktivit a stavů směřuje od shora dolů, což uživatelé považovali za vysoce intuitivní.

Konceptuální OntoUML model je mátl rozložením jednotlivých tříd a křížením vazeb. Nerozuměli tomu, proč jsou vazby v některých případech opatřeny na jednom konci šipkou a v jiném nikoli. Obecně byly tyto diagramy čitelnější především pro uživatele, kteří pracují, či studují informatiku.

Jak je zmiňováno výše, bylo by vhodné použitelnost těchto metod podrobit dalšímu zkoumání. Při tomto testování například nebyly z důvodu přehlednosti zohledněny atributy s jejich datovými typy.

Vzhledem k tomu, že každá z notací slouží jinému účelu, není možné s jistotou říci, která z metod je lepší nebo horší, práce však může posloužit jako zdroj informací další, komplexnější studii, která bude i na základě této práce schopna vyhodnotit všechny klady a zápory obou metod.

Seznam použitých zkratk

OMT	Technika objektového modelování (Object Modelling Technique)
UFO	Základní ontologie (Unified Foundational Ontology)
ERD	Objektové diagramy (Entity-relationship diagrams)
UML	Univerzální modelovací jazyk (Unified Modelling Language)
IBM	Technologická společnost
BPD	Diagram obchodního procesu
BPMN	Business process model and notation
BPMI	Nezisková organizace Business Process Management Initiative
OMG	Organizace Object Management Group
SAP	Společnost vyvíjející stejnojmenný informační systém
ERP	Podnikový informační systém (Enterprise Resource Planning)
DMS	Úložiště dokumentů (Document Management System)
IDE	Vývojové prostředí (Integrated Development Environment)
BPR	Optimalizace obchodních procesů (Business Process Reengineering)
UFES	Brazilská univerzita (Universidade Federal do Espírito Santo)
NEMO	Vědecká skupina pracující na výzkumu ontologie a konceptuálního modelování
ČVUT	České vysoké učení technické
CCMi	Výzkumná skupina z ČVUT
UX	User eXperience (uživatelský zážitek)
AS IS	Stávající stav

7 Seznam obrázků

Obrázek 1 - Princip tří architektur datového modelování.....	9
Obrázek 2 - Grafické znázornění entity	10
Obrázek 3 - Vazby	11
Obrázek 4 - Vývojový diagram procesu reklamace.....	14
Obrázek 5 – Grafické elementy Petriho sítě.....	17
Obrázek 6 - Základní pohledy metody ARIS	18
Obrázek 7 - Grafické symboly BPMN	21
Obrázek 8 - UML Třídni diagram hotelového managementu	23
Obrázek 9 – UML Stavový diagram zadání PIN kódu.....	24
Obrázek 10 - Grafické symboly UML diagramu aktivit.....	25
Obrázek 11 - Procesní diagram BORM.....	28
Obrázek 12 – Grafická notace metody BORM.....	29
Obrázek 13 - Fáze BORM	31
Obrázek 14 - Srovnání UML a OntoUML.....	35
Obrázek 15 - OntoUML metamodel stereotypů	36
Obrázek 16 - Procesní diagram BORM - vydání knihy.....	41
Obrázek 17 - Konceptuální diagram OntoUML - vydání knihy.....	42
Obrázek 18 - Procesní diagram BORM – přijímací řízení	43
Obrázek 19 - Konceptuální OntoUML diagram - přijímací řízení	44

8 Seznam zdrojů použitých obrázků

Obrázek 1 – Hanák, Lukáš, 2014. INFORMAČNÍ STRATEGIE FIRMY. [online]. Dostupné z: <https://core.ac.uk/download/pdf/30303128.pdf>

Obrázek 3 - <https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning?a=0>

Obrázek 4 - <https://www.fit.vutbr.cz/study/courses/TJD/public/0910TJD-Macel.pdf>

Obrázek 5 - Lucie, Pekárková, 2007. Techniky modelování a optimalizace podnikových procesů. Bakalářská práce. Masarykova univerzita., <https://managementmania.com/cs/workflow>

Obrázek 6 - <https://viewpointsonitarchitecture.wordpress.com/2011/08/22/aris-tool-and-methodology-for-enterprise-and-solutions-architecture/>

Obrázek 7 - <https://study.com/academy/lesson/business-process-model-and-notation-process-examples.html>

Obrázek 8 - <https://www.lucidchart.com/pages/uml-class-diagram?a=0>

Obrázek 9 - <https://www.geeksforgeeks.org/unified-modeling-language-uml-state-diagrams/>

Obrázek 10 - <https://pdfs.semanticscholar.org/884e/374af5c4e8cc1a68aa42a5c228191af48385.pdf>

Obrázek 13 - https://www.researchgate.net/profile/Vojtech_Merunka/publication/220920645/figure/fig1/AS:350482548707328@1460572959028/BORM-diagram-symbols.png

Obrázek 14 - http://www.euromise.net/wp-content/uploads/2014/05/Podlouck%C3%BD_Pergl.pdf

Obrázek 15 - https://www.researchgate.net/figure/OntoUML-profile_fig4_325995419

9 Seznam použitých zdrojů

- Molhavec, Martin, 2006. [online]. Dostupné z: <http://prog-story.technicalmuseum.cz/images/dokumenty/Programovani-TSW-1975-2014/2006/2006-17.pdf>
- Řepa, Václav. VÝVOJOVÉ TRENDY METODIK VÝVOJEINFORMAČNÍCH SYSTÉMŮ - VÝZVA BPR. [online]. Dostupné z: <https://nb.vse.cz/~repa/veda/EurOpen99%20Paper.pdf>
- ŘEPA, Václav. Podnikové procesy. 2. vyd. Praha: Grada Publishing, 2007. 281 s.ISBN 978-80-247-2252-8
- Zedníček, Jan. Kardinalita vztahu, 2017. [online]. Dostupné z: <https://biportal.cz/BI-slovník/kardinalita/>
- ADJEI, Daniela. Standardy a metodiky modelování podnikových procesů [online]. Praha, 2009 [cit. 2019-03-10]. Dostupné z: <<https://theses.cz/id/q0wets/>>. Bachelor's thesis. University of Economics, Prague. Vedoucí práce Tomáš Vilím.
- Řepa, V.: Podnikové procesy: Procesní řízení a modelování. Grada Publishing, a.s., Praha, 2006. 268 s. ISBN 80-247-1281-4., s. 13
- DAVENPORT, T. H. Process Innovation: reengineering Work through Information Technology. Harvard Bussiness Scool Press, 1993, ISBN 087-58436-62.
- HAMMER, M., CHAMPY, J. Reengineering - radikální proměna firmy. Manifest revoluce v podnikání. Praha: Management Press, 2000, 212 s. ISBN 80-7261-028-7.
- HORNBY, A. S., TURNBULL, J., LEA, D., PARKINSON, D., PHILLIPS, P., & ASHBY, M. (2010). Oxford advanced learner's dictionary of current English.
- Klimeš, Cyril, 2014. MODELOVÁNÍ PODNIKOVÝCH PROCESŮ. [online]. Dostupné z: <http://www1.osu.cz/~zacek/mopop/mopop.pdf>

Workflow. In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 11.09.2018 [cit. 10.03.2019]. Dostupné z: <https://managementmania.com/cs/workflow>.

Lucie, Pekárková, 2007. Techniky modelování a optimalizace podnikových procesů. Bakalářská práce. Masarykova univerzita.

Koudelková, Anna, 2011. Procesní model organizace.[online]. Diplomová práce. Univerzita Pardubice. Dostupné z: <https://docplayer.cz/13662439-Procesni-model-organizace.html>

HARMON, Paul a Celia WOLF. The State of Business Process Management

Vašíček, Petr. Úvod do BPMN. Dostupné z: <http://bpm-sme.blogspot.com/2008/03/3-uvod-do-bpmn.html>

Modelování podnikových procesů. Dostupné z: <https://www.altaxo.cz/provoz-firmy/management/rizeni-podniku/modelovani-podnikovych-procesu>

White, Stephen. 1Process Modeling Notations and Workflow Patterns Dostupné z: http://www.bpmn.org/Documents/Notations_and_Workflow_Patterns.pdf

UML (Unified Modeling Language). In: ManagementMania.com [online]. Wilmington (DE) 2011-2019, 09.08.2016 [cit. 10.03.2019]. Dostupné z: <https://managementmania.com/cs/unified-modeling-language>

Richta, Karel, 2003. Unifikovaný modelovací jazyk UML. [online]. Dostupné z: http://www.kiv.zcu.cz/~mautner/Pt/UML_richta.pdf

ARIS tool and methodology for Enterprise and Solutions Architecture, 2011. [online]. Dostupné z: <https://viewpointsonitarchitecture.wordpress.com/2011/08/22/aris-tool-and-methodology-for-enterprise-and-solutions-architecture/>

Salma, 2017. UML Class Diagrams Tutorial, Step by Step. [online]. Dostupné z: https://medium.com/@smagid_allThings/uml-class-diagrams-tutorial-step-by-step-520fd83b300b

Suchan, Jan, 2007. Stavové diagramy – tutoriál. [online]. Dostupné z: <http://www.minmax.cz/sites/default/files/fel-cvut/semestr04/x36sin/stavovy-diagram-tutorial.pdf>

ARLOW, Jim; NEUSTADT, Ila. UML 2 a unifikovaný proces vývoje aplikací : Objektově orientovaná analýza a návrh prakticky. Pěel. Bogdan Kizska. 1.vyd. Brno: Computer Press, 2008. 567 s. ISBN 978-80-251- 1503-9.

Komponentové diagramy. [online]. Dostupné z: <http://ocup.ocup.cz/2010/07/komponentove-diagramy-component.html>

Merunka, Vojtěch: BORM - overview of the methodology and case study of the agrarian information systém. Agricultural Economics. Zemědělská ekonomika, 49, 2003, 9, 397-406 ISSN: 0139-570X

UNIFIED MODELING LANGUAGE SPECIFICATION VERSION 2.5.1, 2017. [online]. Dostupné z: <https://www.omg.org/spec/UML/About-UML>

Merunka, Vojtěch, 1999. VYUŽITÍ METODY BORM PRO BUSINESS PROCESS REENGINEERING. [online]. Dostupné z: <http://www.agris.cz/clanek/104163>

Maxa, Oskar, 2015. Implementace procesní analýzy OR diagramů v metodě BORM.. Bakalářská práce. Univerzita Karlova v Praze. Vedoucí práce RNDR. Filip Zavoral, Ph.D.

Pícka, Marek, 2010. Vývoj objektově orientovaných informačních systémů pomocí metody postupných transformací. [online]. Disertační práce. Vedoucí práce Prof. Ing. Ivan Vrana DrSc. Dostupné z: <https://docplayer.cz/97950991-Vyvoj-objektove-orientovanych-informacnich-systemu-pomoci-metody-postupnych-transformaci.html>

Yourdon, Edward. (2019). Mainstream Objects : An Analysis and Design Approach for Business

POLÁK, J, V MERUNKA a A CARDA. Umění systémového návrhu. 2003.

Merunka, Vojtěch, (2010). OBJECT-ORIENTED PROCESS MODELING AND SIMULATION. BORM EXPERIENCE. Trakia Journal of Sciences. Dostupné z: <http://www.uni-sz.bg>

Merunka, Vojtěch. Metoda BORM. [online]. Dostupné z: <https://docplayer.cz/2267840-Vojtech-merunka-metoda-borm.html>

Marek, Jan, 2017. Model klienta veřejné správy z pohledu Unified Foundational Ontology. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Prof. Ing. Václav Řepa, CSc.

L. Benyovszky, Náhlost. Myšlení bytí z času. Praha: Oikumené 2006, 1. Filozofický slovník, Praha 1985, s. 21, Filozofický slovník, Praha 1986, s. 343., 3. G. Legrand: Vocabulaire Bordas de la philosophie, Paříž 1986. s. 242.

Guskova, Natalia, 2018. Konceptualizace vybraných částí modelu bezpečnosti STAMP.

Pergl, Podloucký, 2014. OntoUMLa UFO-A Formalizace strukturálních biomedicínských znalostí. [online]. Dostupné z: http://www.euromise.net/wp-content/uploads/2014/05/Podlouck%C3%BD_Pergl.pdf

Suchánek, Marek. OntoUML Specification. [online]. Dostupné z: <https://ontouml.readthedocs.io/en/latest/classes/sortals/role/index.html>

Rostislav Programming. OntoUML. [online]. Dostupné z: <http://www.rosprog.4fan.cz/navrh-software/ontouml/>