Univerzita Pardubice
Fakulta ekonomicko-správní
Ústav systémového inženýrství a informatiky

Nestrukturovaná data v organizaci

Diplomová práce

2021 Jiří Jemelka

Univerzita Pardubice
Fakulta ekonomicko-správní
Akademický rok: 2020/2021

ZADÁNÍ DIPLOMOVÉ PRÁCE
(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Bc. Jiří Jemelka**
Osobní číslo:	**E18485**
Studijní program:	**N6209 Systémové inženýrství a informatika**
Studijní obor:	**Informatika ve veřejné správě**
Téma práce:	**Nestrukturovaná data v organizaci.**
Zadávající katedra:	**Ústav systémového inženýrství a informatiky**

Zásady pro vypracování

Cílem práce je charakterizovat typy dat v organizaci, porovnat možnosti využití dat strukturovaných a nestrukturovaných, navrhnout postup pro sběr požadavků a tvorbu návazných modelů v rámci využití Big data v organizaci.

Osnova:
- Základní pojmy související se zpracovávanou problematikou.
- Tok dokumentů organizací.
- Uplatnění nestrukturovaných dat.

Rozsah pracovní zprávy: **cca 50 stran**
Rozsah grafických prací:
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

BIG DATA FRAMEWORK. *Enterprise Big Data Professional*. Bonn, Germany: Big Data Framework, 2018. 121 s. ISBN 978-90-828958-0-3.
GÁLA, Libor, POUR, Jan a Zuzana ŠEDIVÁ. *Podniková informatika. Počítačové aplikace v podnikové a mezipodnikové praxi*. Praha: Grada Publishing, 2015. 240 s. ISBN 978-80-247-5457-4.
HOLUBOVÁ, Irena, KOSEK, Jiří, MINAŘÍK, Karel a David NOVÁK. *Big Data a NoSQL databáze*. Praha: Grada Publishing, 2015. 288 s. ISBN 978-80-247-5466-6.
LEE, James, WEI, Tao a Suresh Kumar MUKHIYA. *Hand-on Big Data Modeling*. Birmingham, UK: Packt Publishing, 2018. 306 s. ISBN 978-1-78862-090-1.
MAYER-SCHÖNBERGER, Viktor a Cukier KENNETH. *BIG DATA: Revoluce, která změní způsob, jak žijeme, pracujeme a myslíme*. Brno: Computer Press, 2014. 256 s. ISBN 978-80-251-4119-9.

Vedoucí diplomové práce: **doc. Ing. Stanislava Šimonová, Ph.D.**
Ústav systémového inženýrství a informatiky

Datum zadání diplomové práce: **1. září 2020**
Termín odevzdání diplomové práce: **30. dubna 2021**

L.S.

prof. Ing. Jan Stejskal, Ph.D.
děkan

RNDr. Ing. Oldřich Horák, Ph.D.
vedoucí ústavu

V Pardubicích dne 1. září 2020

Prohlašuji:

Práci s názvem Nestrukturovaná data v organizaci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 7/2019 Pravidla pro odevzdávání, zveřejňování a formální úpravu závěrečných prací, ve znění pozdějších dodatků, bude práce zveřejněna prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 22.04.2021.

Jiří Jemelka v.r.

PODĚKOVÁNÍ

Rád bych poděkoval, vedoucí mé diplomové práce doc. Ing. Stanislavě Šimonové, Ph.D. za věcné připomínky, cenné rady a její podporu, kterou mě věnovala při zpracovávání mé diplomové práce. Dále mé poděkování patří mé rodině a mým přátelům za podporu během studia.

ANOTACE

Diplomová práce se zabývá charakteristikou typů dat v organizaci s možností využití strukturovaných, semistrukturovaných a nestrukturovaných dat. S návrhem postupů pro sběr požadavků s tvorbou návazných modelů pro využití Big data v organizaci. Určit způsoby pro modelování a vizualizaci Big dat v organizaci se zaměřením na nestrukturovaná data.

KLÍČOVÁ SLOVA

Big data, strukturovaná data, semistrukturovaná data, nestrukturovaná data, datové proudy, strojové učení, umělá inteligence, analýzy dat v reálném čase, analýza sentimentu,

TITLE

Unstructured data in the organization

ANNOTATION

The diploma thesis deals with the characteristics of data types in the organization with the possibility of using structured, semi-structured and unstructured data. With the design of procedures for the collection of requirements with the creation of follow-up models for the use of Big Data in the organization. Identify ways to model and visualize Big Data in an organization with a focus on unstructured data.

KEYWORDS

Big Data, structured data, semistructured data, unstructured data, streaming data, machine learning, artificial intelligence, real time data analysis, sentiment analysis.

OBSAH

SEZNAM ILUSTRACÍ

SEZNAM TABULEK

SEZNAM ZKRATEK A ZNAČEK

AAM	Active Appearance Model (Aktivní vzhledový model)
ACID	Atomicity, Consistency, Isolation, Durability (Atomicita, Konzistence, Izolovanost, Trvalost)
AI	Artificial Intelligence (Umělá inteligence)
AP	Availability, Partition tolerance (Dostupnost, Odolnost vůči rozpadu sítě)
ASR	Automatic Speech Recognition (Automatické rozpoznávání řeči)
ASU	Automatic Speech Understanding (Automatické pochopení řeči)
BASE	Basicalli, Available, Soft sate, Eventual consistency (Převážná dostupnost, Volný stav, Občasná konzistence)
BI	Business Intelligence (Obchodní inteligence)
CA	Consistency, Availability (Konzistence, Dostupnost)
CAD	Computer Aided Design (Počítačem podporovaný design)
CAP	Consistency, Availability, Partition tolerance (Konzistence, Dostupnost, Odolnost vůči rozpadu sitě)
CIS	Customer Information System (Zákaznický informační systém)
CMS	Content Management System (Systém pro správu obsahu)
CP	Consistency, Partition tolerance (Konzistence, Odolnost vůči rozpadu sítě)
DBMS	Data Base Management System (Systém pro správu databází)
DHW	Data Warehouse (Datový sklad)
DIKW	Data, Information, Knowledge, Wisdom (Data, Informace, Znalosti, Moudrost)
DMS	Document Management System (Systém správy dokumentů)
ECM	Enterprise Content Management (Správa podnikového obsahu)
EDI	Electronic Data Interchange (Elektronická výměna dat)
EIS	Executive Information System (Informační systém pro vrcholný management organizace)

ERP	Enterprise Resource Planning (Plánování podnikových zdrojů)
GIS	Geographical Information System (Geografický informační systém)
GPS	Global Position System (Globální poziční systém)
HDFS	Hadoop Distributed File System (Systém distribuovaných souborů Hadoop)
HTML	Hyper Text Markup Language (Hyper-textový značkovací jazyk)
HW	Hardware (technické vybavení v oblasti IT)
ICR	Intelligent Character Recognition (Inteligentní rozpoznávání znaků)
IoT	Internet of Things (Internet věcí)
IS	Information Systém (Informační systém)
IT	Information Technology (Informační technologie)
JSON	Java Script Object Notation (Datový formát pro objektový zápis)
MES	Manufacturing Excusion System (Systém provádění výroby)
MIS	Management Information System (Manažerský informační systém)
MLP	Multi Layer Perception (Vícevrstvé neuronové sítě)
NAS	Nettwork Attached Storage (Úložiště připojené k síti)
NBDRA	NIST Big Data Reference Architecture (Referenční architektura velkých dat NIST (Národního institutu pro standardy a technologie))
NIST	National Institute of Standards and Technology (Národní institut pro standardy a technologie)
NLP	Natural Language Processing (Zpracování přirozeného jazyka)
OCR	Optical Character Recognition (Optické rozpoznávání znaků)
OIS	Office Information System (Kancelářský informační systém)
OLAP	Online analytical processing (Online analytické zpracování)
OLTP	Online transaction processing (Online zpracování transakcí)
PDCA	Plan-Do-Check-Act (Plánujte, kontrolujte a konejte)

PDF	Portable Document Format (Přenosný formát dokumentu)
POS	Part of Speech Tagging (Označování části řeči)
RIS	Reservation Information System (Rezervační informační systém)
RDBMS	Relation Data Base Management System (Systém pro správu relačních databází)
RMS	Record Management System (Systém správy záznamů)
SAN	Storage Area Network (Síťové úložiště)
NoSQL	Not only SQL (využívá i jiný jazyk než strukturovaný dotazovaný jazyk)
SQL	Structured Query Language (Strukturovaný dotazovaný jazyk)
SVR	Support Vector Recognition (podpora vektorového rozpoznávání)
SW	Software (programové (aplikace atd.) vybavení v oblasti IT)
TPS	Transaction Processing System (Systém zpracování transakcí)
UML	Unified Modeling Language (Unifikovaný modelovací jazyk)
XML	eXtensible Markup Language (Rozšiřitelný značkovací jazyk)
ZB	Zettabyte (Zetabajt, 10^{21} bajtů)

ÚVOD

Mezi nejdůležitější komodity dnešní doby patří informace a znalosti. Jejich efektivní využití při rozhodovacích procesech může vést k získání konkurenčních výhod, které mohou mít pozitivní vliv na podnikání. Efektivní využití informací a znalostí ve veřejné správě vede nejen k zlepšení veřejných služeb, ale i k jejich optimalizaci a zefektivnění. Zdrojem informací a znalostí jsou data, která vznikají stálým rozvojem a využíváním informačních a komunikačních technologií v organizacích, ve veřejné správě a v běžném životě. Jsme svědky, že objem, charakter, dostupnost, rychlost generování a tvorby dat má neustále vzrůstající trend. Proto je nutné měnit požadavky na jejich sběr, zpracování, uchování a následné využití. Pro data o velkém objemu a pro větší množství různorodých dat je v současnosti využíván zavedený pojem Big Data, který zastřešuje data strukturovaná, semistrukturovaná a nestrukturovaná.

Diplomová práce je zaměřena na charakterizaci typů dat využívaných v oficiální organizaci, která slouží jako prostředek k dosažení určitých cílů. Zabývá se porovnáním možností využití strukturovaných a nestrukturovaných Big Data. Zaměřuje se na návrhy postupů pro sběr požadavků s tvorbou návazných modelů pro využití nestrukturovaných Big Data v organizaci.

Cílem práce je zaměření se na uplatnění nestrukturovaných dat v organizaci s využitím ECM (Enterprise Content Management), který pokrývá téměř veškeré nestrukturované informace v organizaci (dokumenty, webové stránky, obrázky, výkresy, reporty, e-maily atd.). Data v dnešních organizacích mnohdy splňují charakteristiky pro Big Data, proto se v další části této diplomové práce zaměřuji na Big Data s popisem jejich charakteristik a obecného rámce Big Data Framework pro smysluplné využití Big Data v organizaci. Následně se zaměřuji na popis zpracování nestrukturovaných dat, na technologie jejich uskladnění, jejich datovou analýzu a vizualizaci. Následuje porovnání mezi strukturovanými, semistrukturovanými a nestrukturovanými typy dat, s příklady jejich uplatnění pro Big Data s popisem rizik při jejich využívání. V další části této práce se věnuji modelování dat, se zaměřením na Big Data, a to jak na strukturovaná, tak i na nestrukturovaná data. Pro názorné porovnání jsem vytvořil modelový příklad s popisem sběru dat s následným modelováním procesu pro plánované objednání materiálu v libovolné organizaci. Jak z pohledu relačních technik pro tvorbu datových modelů, tak i z pohledu NoSQL technik tvorby datových modelů.

V závěru diplomové práce jsem popsal techniky modelování pro strukturovaná a nestrukturovaná data Na základě využití modelového příkladu pro plánované objednání materiálu v libovolné organizaci.

1. CHARAKTERISTIKA DAT V ORGANIZACI

Mnoho organizací je přesvědčeno, že jejich data jsou důležitým aktivem. Protože data a informace mohou být využita v procesech při dosahování strategických cílů dané organizace. Přes toto přesvědčení bohužel jen málo organizací spravuje data jako aktiva, z kterých je možno získávat průběžnou hodnotu strategických cílů.

Data Management neboli správa dat se zabývá vývojem, prováděním a dohledem nad plány, politikami, programy a postupy, které dodávají, kontrolují, zabezpečují a zvyšují hodnotu dat a informací po celé období jejich životního cyklu. Činnosti správy dat jsou velmi rozsáhlé a zahrnují veškeré činnosti od schopnosti činit důsledná rozhodnutí o tom jak získat strategickou hodnotu z dat, až po specifikaci technických a softwarových prostředků pro dané řešení. Proto je nutné zajistit sdílenou odpovědnost v této oblasti tak aby byly zajištěny vysoce kvalitní data, která splňují strategické potřeby organizace. [9]

V této části práce se zaměřím na oblast Data Managementu pro charakterizaci a využití dat v oficiální organizaci. Pojmem oficiální organizace definujeme organizaci, která slouží jako prostředek k dosažení určitých cílů jako je např. podnik nebo organizace veřejné správy. V další části této práce se zaměřím na problematiku, která je specifikována pojmem Big Data, s porovnáním využití strukturovaných a nestrukturovaných Big Dat v organizaci s následným zaměřením na nestrukturovaná Big Data (pojem Big Data bude popsán v části 2. BIG DATA).

1.1 Data, informace, znalosti, moudrost

Data a informace nelze chápat pouze jako aktiva, do kterých organizace investují za účelem získání budoucí hodnoty. Protože data a informace jsou velmi zásadní pro každodenní chod většiny organizací, a proto je nutné jim věnovat náležitou pozornost. Z pohledu informačních technologií jsou za data považovány i informace, které byly uloženy v digitálním formátu, a proto za data je možné považovat i elektronické verze videí, obrázků, zvukových nahrávek a různé typy dokumentů. [9]

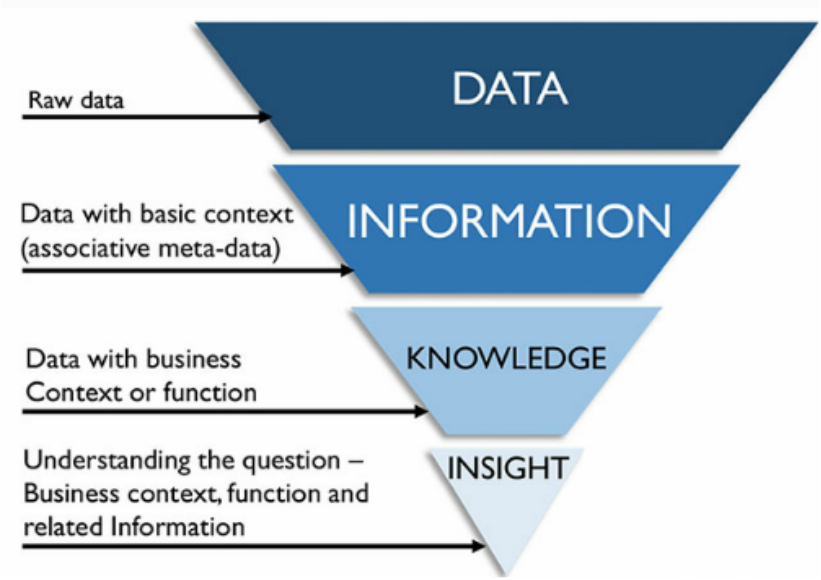
Data jsou prostředkem reprezentace faktů z reálného světa, která bez kontextu nemají žádný smysl. Definice dat [14, s. 14]:
„Data jsou formalizovaný záznam lidského poznání pomocí symbolů (znaků), který je schopný přenosu, uchování, interpretace či zpracování.“

Kontext lze považovat za reprezentační systém dat, kdy takový systém zahrnuje společný slovník a sadu vztahů mezi komponentami. Pokud známe konvence takového systému, je

možné v něm interpretovat data. Tyto konvence jsou často dokumentovány v konkrétním druhu dat označovaných jako metadata. A tak nám vznikne informace. Mezi informacemi a daty je propletený vztah, ale data a informace jsou na sobě nezávislé. Přesto je nutné pro data i informace nastavit společnou správu. Důvodem je praktické využití dat a informací, kdy z dat, které představují zdroj informací, je možné vytvořit několik odlišných relevantních informací dle požadovaného zadání o stejném datovém základu. [9] [15]

Informace, které jsou doplněny o další kontext např. obchodního charakteru nebo o další význam, jsou interpretovány jako znalosti. Znalosti mají to, co informace postrádají, jde o síť vzájemných vztahů, kdy jedna část znalosti může vysvětlit druhou. Aby se informace stala znalostí, musí být smysluplná a relevantní. Znalost je použitelná k realizaci rozhodovacích procesů v organizaci. Moudrost je výsledkem nejlepšího výběru způsobu, jak dosáhnout na základě znalostí požadovaného cíle. Jde o zkušenost, která může být založena i na předchozích pokusech pro dosažení požadovaného cíle. Znalosti a moudrost mohou být následně využity v procesech učící se organizace a znalostní organizace. [4] [9] [15]

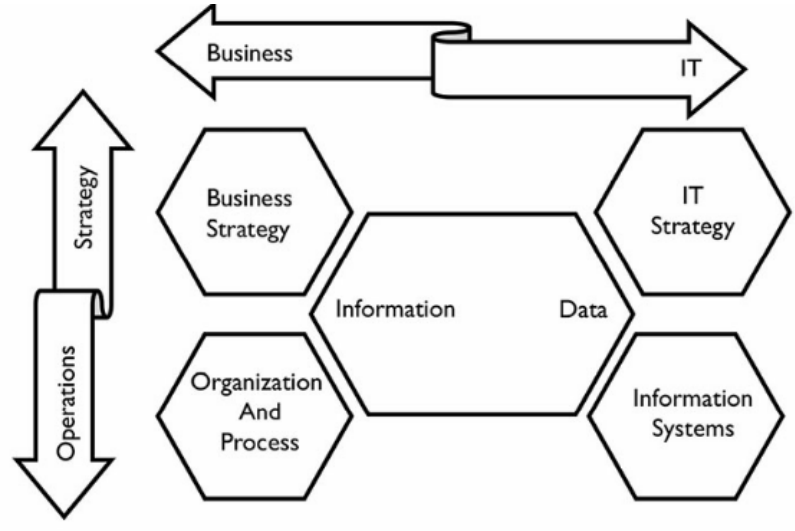
Vztah mezi daty, informacemi, znalostmi a moudrostí nám názorně zobrazuje DIKW pyramida (Data = data, Information = informace, Knowledge = znalosti, Wisdom / Insight = moudrost), viz. Obrázek č.1.



Obrázek 1: DIKW pyramida

Zdroj : [9]

Model vztahu dat a informací k obchodním strategiím a provoznímu využití v organizaci, kdy data a informace jsou spojena s informační technologií a procesy organizace, viz. Obrázek č.2.



Obrázek 2: Strategický model vztahu dat a informací k organizaci.

Zdroj : [9]

1.2 Typy dat v organizaci

Data a informace v organizacích jsou prezentována, zpracovávána a ukládána nebo také přenášená na jiná místa k dalšímu prezentování, zpracování a uložení pomocí aplikací. Jde o aplikace informačních technologií, které mohou být softwarem, hardwarem nebo kombinací obojího. Procesy aplikace pro prezentování, zpracování a ukládání dat a informací jsou označovány jako logiky aplikace. Prezentování = Prezentační logika aplikace, Zpracování = Aplikační logika aplikace, Ukládání = Datová logika aplikace. [14]

Prezentační logika aplikace je založena na charakteru dané aplikace, kdy data jsou uživateli prezentována tak, aby je byl schopen akceptovat svými smysly. Obvykle se jedná o obrazová (text, grafika, formuláře, obrázky, videa) a zvuková data (multimediální formáty dat aj.) [14].

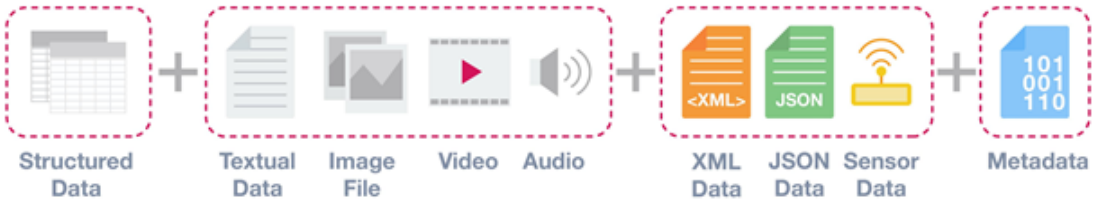
Aplikační logika je zodpovědná za vlastní zpracování dat aplikace dle algoritmu a realizuje procesy nebo funkcionality aplikace, pro které byla vytvořena. Problematika aplikační logiky je založena na schopnosti porozumění aplikace datům. Tedy zda lze zavést v aplikaci co nejvyšší míru automatizovaného zpracování dat, kterého je možné docílit pomocí popisu organizace dat (datových struktur) nebo zaměřením se na zvýšení úrovně pokročilosti algoritmů aplikace. Z pohledu popisu datové struktury jsou rozlišovány dvě mezní charakteristiky dat, a to data strukturovaná a data nestrukturovaná. [14]

Datová logika aplikace spolupracuje s datovými úložišti a zajišťuje komunikaci s jinými aplikacemi. Tyto procesy jsou obousměrné. Část aplikace připravuje data do vhodné struktury pro uložení ve zvoleném typu úložiště tak i pro případný přenos do jiné aplikace, a další část aplikace připravuje data z datového úložiště tak i z přenesených dat z jiné aplikace do vhodné struktury k následnému zpracování v dané aplikaci podle aplikační logiky. Obvykle využíváme pro ukládání dat souborový a databázový přístup. [14]

Typy dat lze specifikovat pomocí různých faktorů. Pro potřeby této práce se zaměřím na datovou strukturu, způsob a místo vzniku dat.

1.2.1 Typy dat podle datové struktury

Datové struktury v organizaci obecně členíme na strukturovaná, semistrukturovaná, nestrukturovaná data a meta data.



Obrázek 3: Typy datových struktur v organizaci.

Zdroj : [5]

Strukturovaná data

Strukturovaná data jsou logicky uspořádaná data podle předefinovaných datových modelů daného systému, a proto je možné snadno zadávat, ukládat a provádět na nich přímou analýzu. Datové modely nám specifikují typy dat (text, číslo, datum atd.) a formáty dat (stanovení maximální délky, počty desetinných míst atd.). Vlastní zpracování strukturovaných dat je proto jednodušší. [14]

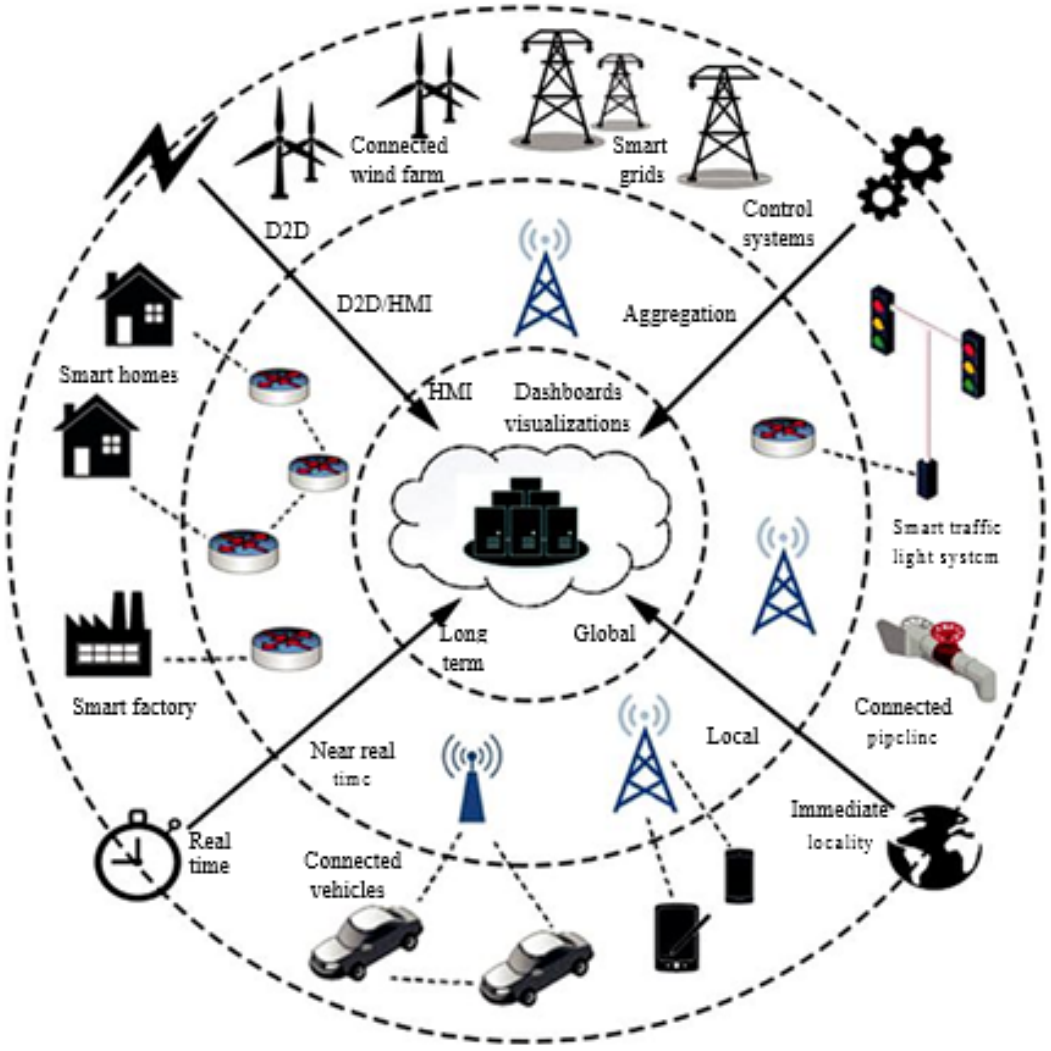
Představitelem strukturovaných dat jsou relační databáze a tabulky. Jde o data v tabulkovém formátu se vztahem mezi různými řádky a sloupci v dané tabulce. Jako příklad je možné uvést soubory MS Excel nebo databáze SQL. Oba případy obsahují strukturované řádky a sloupce, které lze třídit.

Semistrukturovaná data

Pro semistrukturovaná data jsou v literatuře také použity termíny jako částečně strukturovaná data, polostrukturovaná data. Semistrukturovaná data jsou formou strukturovaných dat, která

neodpovídají formální struktuře datových modelů asociovaných, například s relačními databázemi nebo s některými formami tabulek, ale obsahují tagy nebo jiné značky (metadata), které oddělují sémantické prvky a umožní tak vytvořit hierarchické uspořádání. Vlastní zpracování semistrukturovaných dat je komplikovanější, protože jsou zde vyšší nároky na úroveň pokročilost algoritmů aplikace, která s daty pracuje. [14]

Představitelem semistrukturovaných dat jsou data typu JSON, XML, EDI atd. Jedná se o data z různých senzorů (tzv. IoT = internet věcí) průmyslových zařízení, ale i ze zařízení, které nás dnes doprovází v běžném životě (jako jsou např. senzory mobilního telefonu, domácích spotřebičů atd.) nebo může jít o zprávu elektronické pošty, u níž je část, která má charakteristiku strukturovaných dat (odesílatel, příjemce, předmět zprávy) a část která má charakteristiku nestrukturovaných dat (vlastní textový obsah zprávy) atd.



Obrázek 4: Příklad oblastí vzniku semistrukturovaných dat z senzorů IoT.

Zdroj : [22]

Nestrukturovaná data

Nestrukturovaná data nemají předdefinovaný datový model nebo nejsou organizovány předem definovaným způsobem. Neobsahují žádná metadata, která by popisovala jejich obsah. Proto pro přímé zpracování nestrukturovaných dat nelze použít analytické nástroje užité pro relační databáze. Pro zpracování nestrukturovaných dat se využívají nerelační databáze, NoSQL a aplikace, které mají vysokou náročnost na úroveň pokročilosti algoritmů aplikace, která s daty pracuje. Nalezení informací, které jsou uloženy v nestrukturovaných datech, může být velmi komplikované, zdlouhavé a finančně náročné. Náklady na zpracování nestrukturovaných dat jsou vyšší, protože tato data je nutné nejdříve sofistikovaně zpracovat na strukturovaná nebo semistrukturovaná data. Důvodem je následné efektivnější uložení dat do úložišť s využitím velkého množství již existujících analytických nástrojů (aplikací) pro následné získání potřebných informací z takto zpracovaných dat. [14]

Nestrukturovaná data lze rozdělit na textová data (textové soubory, e-maily, texty ze sociálních sítí jako např. Facebook či Twitter, webové stránky, různé články atd.), logovací soubory různých systémových událostí, geolokační údaje, netextová data (multimediální data: digitální fotografie, video, audio atd.), atd.

Metadata

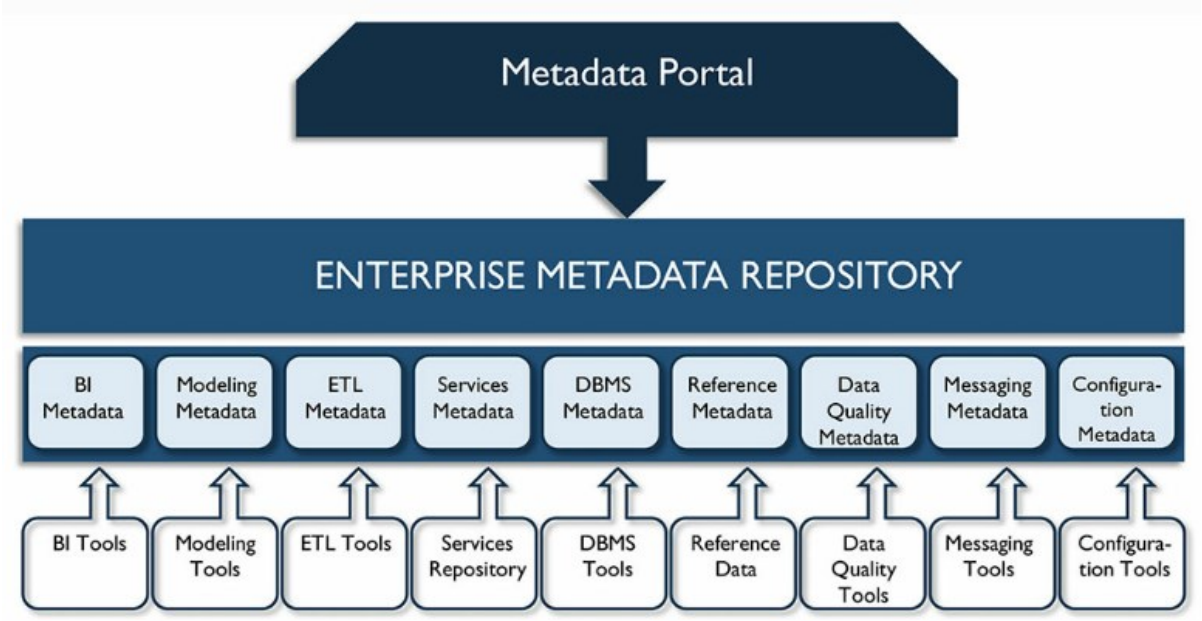
Metadata nejsou technicky samostatnou datovou strukturou, poskytují informace o konkrétní sadě dat. Metadata jsou „data o datech“, bohužel tato definice je příliš zjednodušená. Protože metadata zahrnují informace o technických a obchodních procesech, o pravidlech a omezeních dat, o logických a fyzických datových strukturách. Popisují samotná data (např. databáze, datové prvky, datové modely …), koncepty, která data představují (např. obchodní procesy, aplikace, technologickou infrastrukturu …) a spojení neboli vztahy mezi daty a koncepty. Metadata jsou nezbytná pro správu dat i jejich využití. Metadata pomáhají organizaci pochopit její data, její procesy a pracovní postupy. Umožňují hodnocení kvality dat a jsou nedílnou součástí správy databází a dalších aplikací. Přispívají ke schopnosti zpracovávat, udržovat, integrovat, zabezpečovat, kontrolovat a spravovat další data. [9]

V rámci organizace mají jednotlivci různé úrovně znalostí o datech, ale žádný jednotlivec neví o datech vše. Tyto informace musí být zdokumentovány, jinak může organizace ztratit cenné znalosti. Metadata poskytují primární prostředek k zachycení a správě organizačních znalostí o datech. Správa metadat však není jen správou znalostí, jde zde také o řízení rizik. Metadata jsou nezbytná k zajištění toho, aby organizace mohla identifikovat důležitá nebo citlivá data

a aby mohla spravovat jejich životní cyklus při dosahování svých strategických cílů s minimalizací rizik. Bez spolehlivých metadat organizace neví, jaká data má, co data představují, odkud pocházejí, jak se pohybují organizací, kdo k nim má přístup. Správa metadat má zajistit, aby data byla vysoce kvalitní. Bez metadat nemůže organizace spravovat svá data jako aktivum. Bez metadat by organizace nemusela být vůbec schopna spravovat svá data. [9]

S vývojem technologií se zvýšila rychlost generování dat. Proto se technická metadata stala součástí správy v IT systémech organizací. Norma ISO Metadata Registry Standard, ISO / IEC 11179, má umožnit metadatovou výměnu dat v heterogenním prostředí na základě přesných definic dat při zachování jejich vlastnictví, bezpečnostních požadavků. [9]

Stejně jako ostatní data, metadata vyžadují správu. Jak se zvyšuje úložná kapacita organizací při shromažďování a ukládání dat, tím roste význam metadat ve správě dat. Aby mohly být procesy v organizaci řízeny daty, musí být řízena i metadata (Metadata Management). Viz. ukázka hybridní metadatové struktury v organizaci řízené metadatovým portálem – Obrázek č.5. [9]



Obrázek 5: Hybridní architektura metadat.

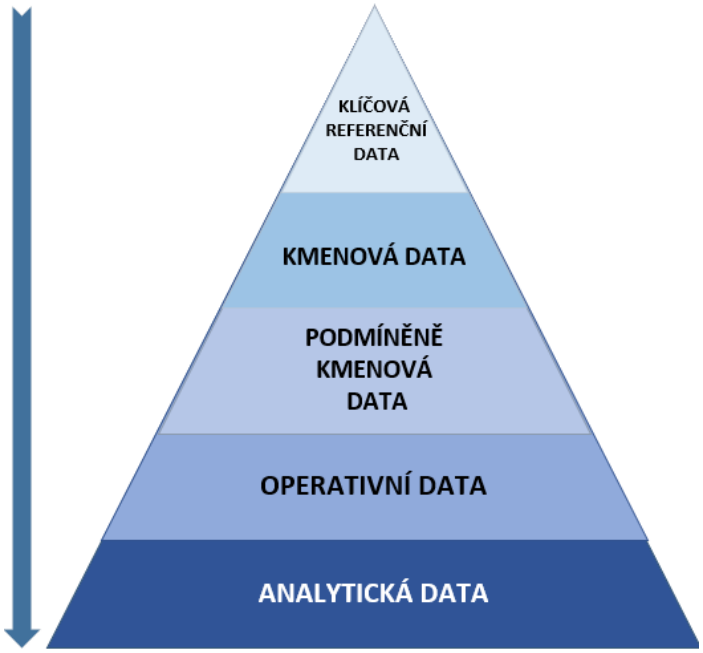
Zdroj : [9]

1.2.2 Typy dat podle místa vzniku

Podle místa vzniku rozdělujeme data na interní a externí. Externí data vznikají primárně mimo organizaci. Jsou to data o společenských podmínkách v okolí organizace, případně o trhu, na kterém organizace působí, zahrnují i otevřená data a data, která organizace externě zakoupí (např. data z výzkumů, anket, průzkumů atd.). Interní data jsou data, která vznikla procesy uvnitř organizace a mají velký potenciál pro využití při různých analýzách predikcích. Dále se budu zabývat podrobnějším popisem jen pro interní data organizace.

Interní data organizace

Z pohledu využívání interních dat organizací je možné rozčlenit do základních kategorií a to na klíčová referenční data, kmenová data (kmenová data, podmíněně kmenová data) transakční data, operativní data, analytická data. Pro dané kategorie dat zobrazím hierarchický vztah.



Obrázek 6: Hierarchie kategorií interních dat organizace.

Zdroj : Vlastní tvorba podle [9] [14]

Klíčová referenční data

Klíčová referenční data obsahují údaje o zdrojích a schopnostech organizace, kdy tyto údaje definují organizaci a její prvky včetně omezujících vlastností. Jde o například o lidské a technické zdroje, charakterizaci a lokalizaci činností, které jsou spojeny s transformačními

procesy organizace (např. oddělení, útvary, divize, atd.) anebo geografické údaje o logistice, atd. [9] [14]

Kmenová data

Kmenová data (master data) vycházejí z klíčových a referenčních dat, která jsou spojena s transformačním procesem organizace. Kmenová data vytvářejí konzistentní a jednotný soubor identifikátorů a dalších atributů, které nám popisují hlavní subjekty organizace a jsou sdíleny mezi procesy organizace. Pokud organizace dokáže správně specifikovat správná kmenová data, mohou být využita pro zlepšení byznys procesů a rozhodování.

Podmíněně kmenová data

Tato kategorie nám specifikuje pravidla a omezení, které jsou aplikována v různých situacích a umožní nám efektivnější využití procesů organizace s minimalizací rizik. Např. je možné předejít nesprávné komunikaci se zákazníkem, chybám v dodávkách zboží a služeb, které jsou způsobeny nesprávnými údaji, nepřesný reporting atd.

Kmenová a podmíněně kmenová data jsou důležitou součástí pro procesy při získávání a udržení kvality správných dat. Prvním krokem v tomto procesu je definice správných dat včetně jejich popisu a kvalifikace. Pro tyto činnosti využijeme metadata. Tato činnost je rozčleněna na dvě části. Na procesní a IT systémový pohled. Procesní pohled nám definuje tyto data v návaznosti na procesy, a to jak na již existující, tak na zamýšlené budoucí procesy. IT systémový pohled vychází z požadavků IT systémů na data. Pro správné řešení je nutné najít vyvážené využití obou pohledů. Vlastní vytvořená data jsou uložena do úložišť a pro práci s nimi je využit tzv. repozitář (úložiště pod odborným vedením), který je rozčleněn do dvou oblastí. Jde o oblast pracovní a produkční. V pracovní oblasti pověření uživatelé organizace provádějí správu kmenových dat, kdy jsou na závěr každé změny dat spuštěny schvalovací workflow, které uvolní kmenová data k standardnímu využití v produkční oblasti. Dané procesy musí probíhat kontinuálně protože, kvalita kmenových dat není v čase stálá a musí se přizpůsobovat v čase potřebám v dané organizaci. [9] [14]

Transakční data

Transakční data slouží k uchovávání informací o jednotlivých aktivitách v procesech organizace a využívají kmenová data. Transakční data primárně vznikají v provozních systémech a aplikacích, a obvykle nejsou sdílena s ostatními provozními systémy a aplikacemi. Vlastní transakce, která specifikuje proces organizace má vlastnosti ACID: atomicity (vždy

proběhne jako celek nebo neproběhne vůbec), trvalosti (neboli jakmile je potvrzena již nemůže být zrušena), konzistence (proběhne za přesně definovaných pravidel) a izolovanosti (nezasahuje do ostatních transakcí a ony do ní). [9] [14]

Operativní data

Operativní data přiřazují transakčním datům vazby k subjektům (zákazníci, dodavatelé, zaměstnanci atd.). Operativní data jsou sdílena mezi systémy a aplikacemi organizace.

Analytická data

Analytická data jsou vytvářena z operativních, transakčních dat a ze všech kategorií analytických dat na nižších úrovních. Analytická data jsou využívána k vytváření reportingů, které se zaměřují na oblasti potřeb dané organizace a slouží jako podklady pro rozhodovací procesy na všech úrovních managementu organizace. Daný reporting zahrnuje širokou oblast od prostých souhrnů (např. výstupní sestavy), až po sofistikované reporty (vytvořené pomocí analytických metod, metodik) s využitím historických souvislostí, které jsou určeny pro strategická rozhodnutí managementu organizace. [9]

1.3 Kvalita dat

Termín kvalita dat jsem v předchozích částech této práce použil několikrát, protože jde o velmi důležitou charakteristiku dat, která má velký vliv na využití dat a informací ve všech procesech organizace. Nekvalitní data mohou vést k mnoha zkreslením v procesech organizace, což může vést k zvýšení nákladů na provoz organizace nebo i v krajním případě k zániku organizace. Oblast kvality dat je nedílnou součástí standardního data managementu každé organizace a je využívaná v celém životním cyklu dat.

Pohled na charakteristiky kvality dat se v čase mění a je závislý na vzniku a vývoji nových IT technologií, proto uvedu několik základních znaků, které se využívají pro charakteristiku kvality dat. Jde o přesnost, platnost, důvěryhodnost, včasnost, jedinečnost, úplnost, konzistenci, použitelnost, flexibilitu, hodnotu (přínos / relevanci). [9]

V současné době datového boomu je nutné používat ucelené datové strategie, abychom v organizaci nezapadli do datových bažin. Pod datovou bažinou si lze představit jakoukoliv datovou platformu nebo množinu dat z datové platformy (datový sklad, datové jezero atd.), která představuje data, jenž nejde dále smysluplně využít nebo jejich výsledná analytika má negativní dopady na celou organizaci. Jak lze poznat, že se z naší datové platformy stává datová bažina? Jde o kombinaci několika příčin. Kdy neznáme způsob vzniku dat a jejich

význam a nemáme reálnou představu, jak chceme uložená data využít. Nebo nevyužíváme metadata (pro popis datových množin a datových transformací, atd.), data nejsou konsolidovaná a kompletní (logicky nebo fyzicky), popřípadě nemají dostatečnou úroveň kvality. Organizace nemá vypracovanou aktuální dokumentaci datové platformy a nedokáže vyspecifikovat, které procesy je možné využít pro tvorbu hodnot z nashromážděných dat, které lze využít pro koncové uživatele, zákazníky atd. Získání dat v požadované struktuře je velmi časově náročné a s daty nelze provádět ad hoc vyhledávání a analýzy. Odstranění datové bažiny je závislé na její šířce a hloubce a je finančně velmi náročné. V krajním případě je vhodnější vytvořit novou datovou platformu tzv. na zelené louce. Proto je výhodnější využívat veškeré nástroje pro správu dat a vyhnou se takové situaci. [9]

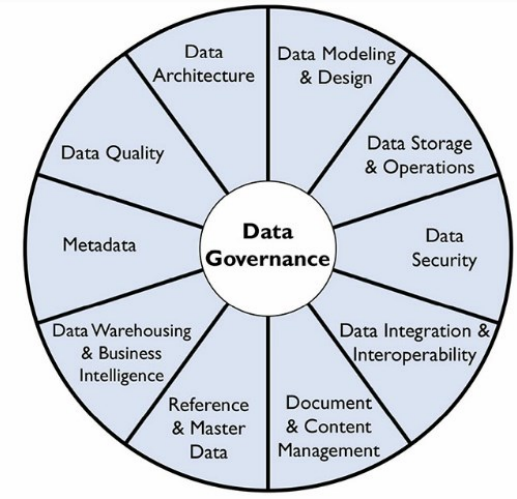
Tabulka 1: Charakteristiky kvality dat

Charakteristika	**Popis charakteristiky**
Přesnost	Udává, do jaké míry data správně popisují daný objekt nebo událost „skutečného světa“.
Platnost	Data jsou platná, pokud odpovídají definici syntaxe (formát, typ, rozsah).
Důvěryhodnost	Data jsou důvěryhodná, pokud byly a jsou při správě dat zavedeny, využívány a dodrženy procesy správy, ochrany a zabezpečení dat
Včasnost	Nám určuje míru, do kdy daná data představují realitu od požadovaného časového bodu.
Jedinečnost	Žádná instance entity (data, informace) nebude zaznamenána více než jednou na základě toho jak toto data (informace) bylo identifikováno (zamezení duplicitám).
Úplnost	Podíl skutečně uložených dat oproti plánovanému potenciálu = ideálně 100 %.
Konzistence	Data jsou konzistentní v případě, že neexistuje rozdíl při porovnání dvou a více reprezentací dat s definicí. Data nejsou konzistentní, pokud se stejná data, která jsou uchovávána na různých místech, neshodují.
Použitelnost	Data jsou použitelná, pokud jsou srozumitelná, jednoduchá, relevantní, přístupná, udržovatelná a jsou na správné úrovni přesnosti.
Flexibilita	Data jsou flexibilní, když jsou srovnatelná a kompatibilní s jinými údaji. Je možné z takových dat vytvářet užitečná seskupování a klasifikování. Mohou být znovu použita a je snadné s nimi manipulovat.
Hodnota (Přínosnost / Relevance)	Zjišťujeme, zda data mají vhodný poměr z pohledu nákladů a přínosů, zda jsou optimálně využita a jestli jejich využívání může mít vliv na bezpečnost a soukromí lidí (GDPR) nebo na právní odpovědnost organizace. Zda data podporují nebo poškozují firemní image atd.

Zdroj : [Vlastní tvorba]

1.4 Data Management

Problematika Data Managementu jak jsem uvedl v úvodu této části je velmi komplexní. Nástroje Data Managementu nám umožňují nejen charakterizovat data v organizaci, ale plně spravovat tuto oblast a vytvářet datové strategie. Datové strategie by měly zahrnovat využití všech dostupných informací k dosažení strategických cílů organizace. Datová strategie musí vycházet z porozumění potřebám dané organizace: jaká data organizace potřebuje, jak data získá, jak je bude spravovat a jak zajistí jejich spolehlivost v průběhu času a jakým způsobem je bude využívat. Pro názornost jsem použil schéma Data Managementu, které je rozčleněno na specializované oblasti této problematiky. Dané oblasti jsou opět velmi komplexní, a proto jsem pro potřeby této práce uvedl jen jejich krátké charakteristiky, viz. Obrázek č.7.



Obrázek 7: Oblasti Data Managementu.

Zdroj : [9]

Tabulka 2: Oblasti Data Managementu

Název oblasti	Popis oblasti Data Managementu
Správa dat (Data governance)	Definuje směr a dohled nad správou dat v organizaci pomocí vytvoření systému rozhodovacích práv nad daty. Účelem správy dat je zajistit jejich řádnou správu podle zásad a osvědčených postupů. Hlavním faktorem je zajistit, aby organizace získala, ze spravovaných dat požadované hodnoty a mohla tak plnit své strategické cíle.
Metadata	Správa metadat zahrnuje aktivity plánování, implementace, kontrolu včetně definic, modelů, datových toků a dalších informací důležitých pro porozumění datům a systémům organizace. Podrobnosti jsou popsány v této práci v části "1.2.1 Typy dat podle datové struktury".

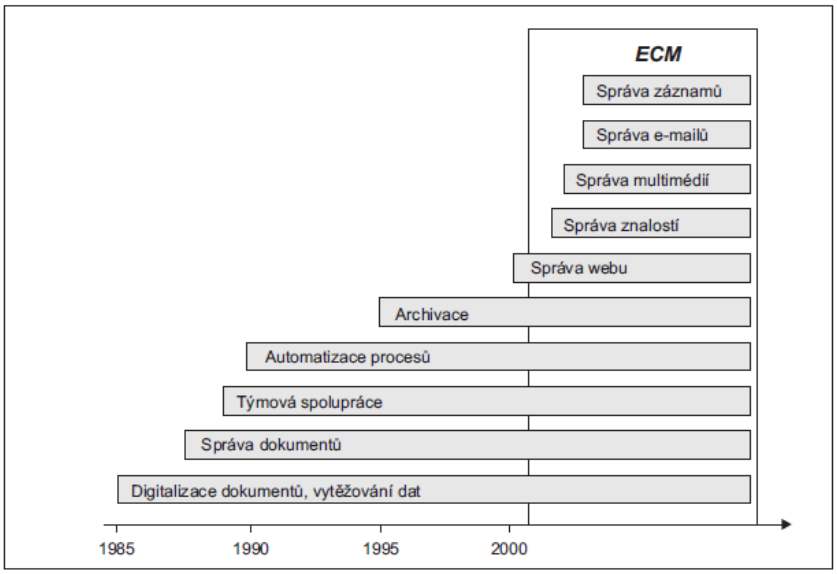
Název oblasti	Popis oblasti Data Managementu
Kvalita dat (Data Quality)	Kvalita dat zahrnuje plánování a implementaci technik řízení kvality pro měření, hodnocení a zlepšování vhodnosti dat pro použití v dané organizaci. Podrobnosti jsou popsány v části "1.2.3 Kvalita dat".
Datová architekture (Data Architecture)	Definuje plán pro správu datových aktiv, který je sladěn se záměry organizace a je vytvořen za účelem stanovení požadavků na specifikaci strategický údajů a návrhů, které vyhovují záměrům a požadavkům pro dosažení strategických cílů organizace. Základem je datový model organizace, který obsahuje názvy dat, komplexní definice dat a metadat, koncepční a logické entity s jejich vztahy a procesní pravidla organizace. Datová architektura organizace je dále popsána integrovanou kolekcí dokumentů tzv. hlavního návrhu na různých úrovních abstrakce, včetně standardů, které určují způsob správy dat v organizaci. Datová architektura definuje datovou strategii organizace.
Datové modely a design (Data Modeling and Design)	V této části Data Managementu jde o procesy zabývající se zjišťováním, analyzováním, reprezentací a komunikací požadavků na data v přesné formě, které vytvářejí datové modely. Datové modelování je důležitou součástí správy dat. Proces modelování vyžaduje, aby organizace objevily a dokumentovaly, jak jejich data do sebe zapadají. Datové modely zobrazují a umožňují organizaci porozumět jejím datovým aktivům.
Úložiště dat a užívání dat (Data Storage and Operatin)	Úložiště a vlastní využívání dat zahrnuje návrhy, implementace a podporu při ukládání dat s cílem maximalizovat jejich hodnotu po celou dobu jejich životního cyklu. Součástí této části Data Managementu mimo tvorby pravidel a metodiky je i navrhování technických a softwarových řešení pro danou problematiku s následnou administrací dat (databáze atd.).
Bezpečnost dat (Data Security)	Zabezpečení dat zahrnuje plánování, vývoj a provádění bezpečnostních zásad a postupů k zajištění řádného ověřování, autorizace, přístupu a auditu dat a informačních aktiv. Specifika zabezpečení dat (které je třeba například chránit) se v různých průmyslových odvětvích a státech mohou lišit. Cíl postupů zabezpečení dat je nicméně stejný: Chránit informační aktiva v souladu s předpisy o ochraně soukromí a důvěrnosti, smluvními dohodami a obchodními požadavky (např. dle GDPR).
Integrace dat a Interoperabilita (Data integration and Interoperbility)	Integrace a interoperabilita dat nám popisuje procesy související s pohybem a konsolidací dat v úložištích dat, aplikacích organizace a mezi nimi. Integrace konsoliduje data do konzistentních forem, ať už fyzických, nebo virtuálních. Datová interoperabilita je schopnost komunikace mezi více systémy. Jde o základní funkce správy dat, na kterých závisí existence většiny organizací.

Název oblasti	Popis oblasti Data Managementu
Správa dokumentů a obsahu (Document and Content Management)	Správa dokumentů a obsahu zahrnuje plánování, implementaci a kontrolní činnosti používané ke správě životního cyklu dat a informací nalezených v řadě nestrukturovaných médií, zejména dokumentů. V mnoha organizacích mají nestrukturovaná data přímý vztah ke strukturovaným datům. Proto pro zajištění bezpečnosti a kvality nestrukturovaných dat je nutno využít efektivní správu dat, spolehlivou datovou architekturu a dobře spravovaná metadata.
Reference a kmenová data (Reference and Master Data)	Reference a kmenová data zahrnují průběžné sladění a údržbu klíčových kritických sdílených dat, aby bylo možné konzistentní používání nejpřesnější, nejaktuálnější a nejrelevantnější verze pravdy o základních procesních entitách napříč všemi systémy v organizaci.
Datové sklady a Business Intelligence (Data Warehousing and Business Intelligence)	Datové sklady and Business Intelligence zahrnuje procesy pro plánování, implementaci a kontrolu správy dat, která podporuje rozhodování a umožnuje znalostním pracovníkům získat požadovanou hodnotu z dat prostřednictvím analýz, reportů a hlášení.

Zdroj : [Vlastní tvorba]

1.5 Tok dokumentů v organizaci (ECM)

Pod Enterprise Content Management (dále jen ECM) jsou zahrnuty veškeré procesy, činnosti, metody a technologie, které organizace využívá pro získávání, správu, ukládání, uchovávání a distribuci (publikování) obsahu. Obsah v tomto kontextu představuje veškeré papírové dokumenty využívané organizací a digitální obsah, který zahrnuje veškerá strukturovaná, nestrukturovaná data a informace, která organizace vytváří a využívá. ECM spravuje v organizaci celý životní cyklus takto specifikovaného obsahu od jeho vzniku přes jeho správu, až po jeho uložení a vyřazení (zánik). ECM má za cíl doručit správný obsah do správných procesů pro dosažení strategických cílů organizace a tak zajistit dostupnost informací, nižší chybovost, vyšší informační bezpečnost s nárůstem rychlosti a kvality zpracovávaných informací. Dalším cílem ECM je digitalizace informací z listinné formy do elektronické formy a jejich následné efektivnější využití v procesech organizace. Vývoj ECM byl závislý na rozvoji informačních technologií, viz. obrázek 16. [42]



Obrázek 8: Historie vzniku ECM.

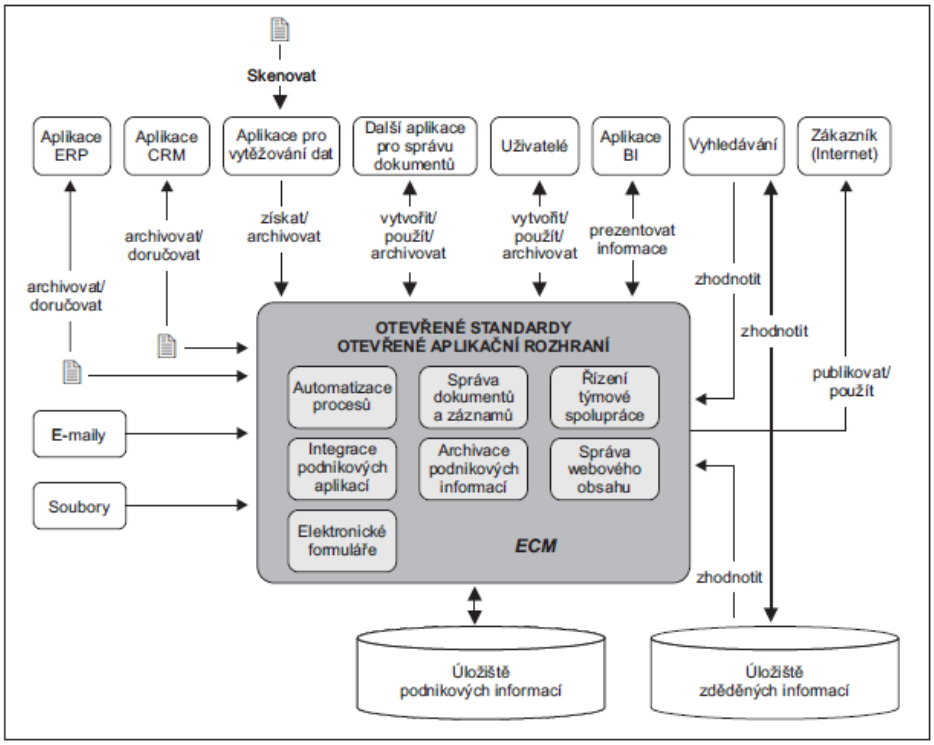
Zdroj : [42]

Běžně se setkáváme s nesprávnou záměnou termínu „obsah“ a „dokument“ při popisu této oblasti využívání dat a informací organizacemi. Vzhledem k tomu, že pod termínem dokument si většina z nás představí jeho listinnou podobu (papírový dokument) nebo text vytvořený v textovém editoru a uložený ve formě souboru je termín „dokument“ velmi omezující (např. pro audio nebo video data se termín „dokument“ běžně nevyužívá).

ECM se zaměřuje na využití nestrukturovaných dat, které jsou stejně jako strukturovaná data nepostradatelná pro činnost a existenci organizace. Strukturovaná data jsou v organizacích standardně zpracovávána informačními systémy (např. ERP, atd.) při každodenních činnostech organizace. Nestrukturovaná a semistrukturovaná data jsou stále častěji zařazována do standardních procesů organizace. Tyto data mohou být využita jak v zdrojovém formátu společně s metadaty nebo mohou být využita, až po procesu vytěžení (pomocí přesně specifikovaných požadavků) ze zdrojového formátu dat do vhodné formy strukturovaných dat. Dále se již nebudu v této oblasti zabývat tzv. strukturovaným obsahem informací z pohledu ERP informačních systému (databází atd.) ale pouze informacemi, které jsou formou dokumentů (strukturované, semistrukturované, nestrukturované) a ostatních nestrukturovaných dat (audio, video a WEB data atd.). Jak jsem již zmínil, problematika ECM by měla být komplexně začleněna do informační struktury organizace – jako např. na obrázku 9. [42]

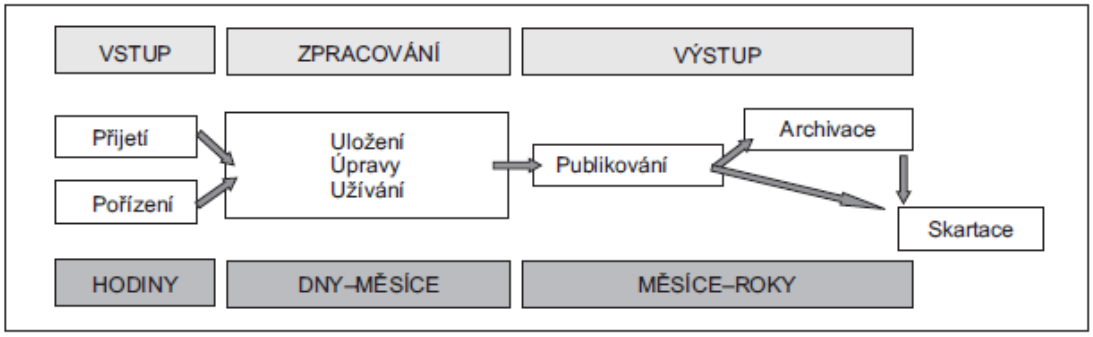
Nejen data a informace, ale i obsah nestrukturovaných dat ECM má svůj životní cyklus, který je zahájen tím, že jsou přijaty (např. uživatel obdrží e-mail, stáhne si dokument z internetu atd.) nebo jsou pořízeny neboli vytvořeny (např. vytvoří v textovém editoru nový dokument, nebo

provede digitalizaci listinného dokumentu na jeho elektronickou formu). Na obrázku 10 je tato problematika velmi přehledně vizualizována i s orientační časovým intervalem, který ohraničuje základní fáze obsahu organizace. Životní cyklus informací obsahu organizace obvykle členíme na fázi vstupu, zpracování a výstupu. [42]



Obrázek 9: ECM v aplikační architektuře informačního systému.

Zdroj : [42]



Obrázek 10: Fáze životního cyklu obsahu organizace.

Zdroj : [42]

V této části se zaměřím na termíny používané v oblasti ECM a to termín dokument a záznam.

- **Dokument:** Je z hlediska českého práva definován zákonem o archivnictví a spisové službě (Zákon 499/2004). Dokument je každá písemná, obrazová, zvuková nebo jiná zaznamenaná informace v analogové (listinné, písemné, atd.) či digitální podobě, která

byla vytvořena svým původcem nebo byla původci doručena. Obecně lze dokumenty v analogové podobě vzájemně spojovat fyzicky, a dokumenty v digitální podobě lze vzájemně spojovat prostřednictvím metadat. Dokumenty lze také konvertovat. Konverze dokumentů je jejich převod z listinné podoby do podoby datové zprávy nebo datového souboru a naopak. Každá takto provedená konverze musí mít přesná pravidla, aby převedený dokument měl stejné právní účinky jako ten původní (např. konverze pro dokumenty právní povahy provádí orgány veřejné moci a advokáti). Digitalizace dokumentů je extrakcí dat z papírových dokumentů (analogových) do digitální podoby, která může být realizována dvěma způsoby. První způsob je vytvoření souboru (např. PDF) a druhý způsob je extrakce obsahu původního dokumentu speciální aplikací s následným uložením digitalizovaných dat přímo do struktury informačního systému organizace (např. převod faktury přímo do systému, kde může být uložena jako soubor i přímo načtena do atributů tabulky databáze informačního systému organizace). Pro digitální dokumenty je vhodné (pro některé druhy dokumentů je to i povinné) zavést pravidla o věrohodnosti a neporušenosti dokumentů např. používání elektronického podpisu, elektronické značky, časového razítka nebo jiný vhodný hardwarový nebo softwarový způsob. [42]

Dokumenty v organizaci vznikají převážně v OIS (Office Information system) organizace, který je zaměřen na automatizaci administrativních činností organizace, jako jsou činnosti uživatele např. využívání elektronické pošty, zpracování textů, manipulaci s dokumenty, digitalizace dokumentů, správa dokumentů atd. Tyto dokumenty provázejí většinu procesů v organizacích. Organizace se z tohoto důvodu musí zabývat problematikou životního cyklu dokumentu. Proto je nutné využívat na správu dokumentů (životní cyklus atd.) vhodný sofistikovaný nástroj nejlépe aplikaci, která by měla vyhovovat požadavkům ECM. [42]

Důvěryhodný dokument: [42]

- Je originálním dokumentem nebo je odvozený z originálu jako stejnopis.
- Je možné jednoznačně určit jeho původ.
- Je možné jednoznačně ověřit, zda někdo dokument nepoškodil (neporušil).
- Je čitelný
- Je možné prokázat existenci dokumentu v časové linii.

- **Záznam:** Vznikne zařazením dokumentu do systému pro správu záznamů, protože již není pod kontrolou autora (Document Management), ale je pod kontrolou organizace (Record Management). Záznam je dokument, který podléhá regulatorním a legislativním předpisům a musí s ním být v organizaci nakládáno jiným způsobem než s dokumentem. Záznam může být libovolného formátu (analogový, digitální). Záznam je složen z obsahu, struktury a kontextu. Záznamem nejsou dokumenty, které jsou modifikovány nebo jsou ve schvalovacích procesech a mohou u nich vznikat nové verze nebo mohou být kdykoliv smazány. [42]

ECM – Legislativa a standardy:

Touto oblastí se budu pro potřeby této práce zabývat pouze obecně, i když je velmi důležitá pro vlastní procesy ECM. Jde převážně o zrovnoprávnění listinných a elektronických dokumentů, kde je toto zrovnoprávnění implementováno zákony a vyhláškami ČR a Evropské unie. V případě využívání ECM v jiných zemích je nutno podrobně nastudovat legislativu v této oblasti, která platí v daném státu. Obecně jde o zákony specifikující svobodný přístup k informacím, ochranu osobních údajů, informační systémy veřejné správy, archivnictví, spisovou službu, autorizovanou konverzi dokumentů, elektronický podpis (kvalifikovaný a další), certifikáty (kvalifikovaný a další), elektronickou značku, časové razítko, dále co je E-podatelna, datová schránka, Terminál Czech POINT atd. [42]

ECM je v současnosti specifikován pomocí standardů ISO, které usnadňují jeho implementaci, protože specifikují jeho implementační standard v organizaci. Tuto oblast zaštiťují např. normy ISO 9000 (System Managementu Quality), ISO 20000 (PDCA(Plan-Do-Check-Act) a standardy pro správu záznamů ISO 15849, ISO 23081. [42]

ECM aplikace: [42]

- Document Management System
- Web Content Management System
- Digital Asset Management System
- E-mail Management System
- Records Management System
- Collaboration Tools
- Workflow

1.5.1 DMS (Document Management System)

Tato oblast se zaměřuje na správu dokumentů v elektronické podobě a na digitalizaci dokumentů v listinné podobě. DMS (Document Management System dále jen DMS) je složen z několika komponent, které může organizace využívat podle svých potřeb. [42]

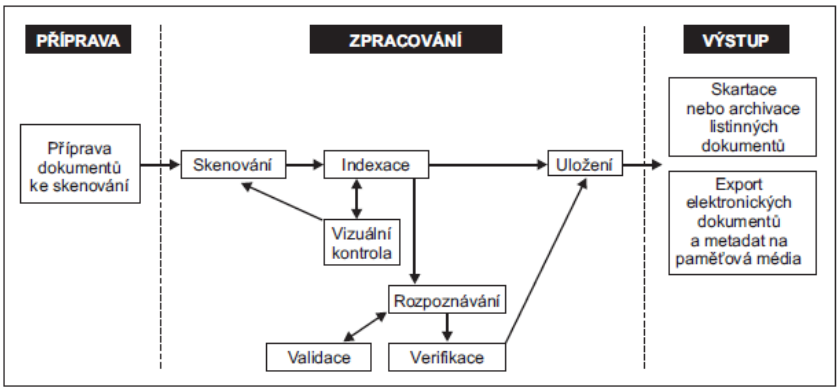
Digitalizace dokumentů (Document Imaging System):

Listinné dokumenty jsou zpracovávány mimo informační systém = jsou bez integrace. Jsou zpracovávány manuálně a po příchodu je jim přiřazen jednoznačný identifikátor v organizaci, který je zaznamenám v informačním systému.

Data z listinných dokumentů mohou být do IS (informační systém – dále jen IS) integrována manuálně nebo pomocí procesu digitalizace. Manuální vkládání je integrací do IS na úrovni dat obsažených v dokumentech oproti digitalizaci ta zajistí plnou integraci dokumentu do IS. Proces digitalizace obecně členíme na dvě varianty. První varianta digitalizuje listinný dokument do souboru, který je přiložen k strukturovaným záznamům v ERP. Druhá varianta je založena na sofistikovanější digitalizaci, která načte data přímo do atributů EPR systému a vytvoří tak standardní součást ERP. Druhá varianta je běžněji používána pro strukturované dokumenty. [42]

Vlastní digitalizace obsahuje tři fáze: [42]

- **1 fáze: Příprava dokumentů** – příprava dokumentů na skenování.
- **2 fáze: Zpracování** – obsahuje skenování (OCR, ICR, OMR BCR), rozpoznání, indexace, verifikace a validace, uložení.
- **3 fáze: Výstup** – Elektronické dokumenty jsou exportovány do úložišť organizace, listinné dokumenty jsou archivovány nebo skartovány.



Obrázek 11: ECM proces digitalizace dokumentů.

Zdroj : [42]

Spisová služba: [42]

Je jedním z procesů, který je realizován v každé organizaci, jde o řízený tok dokumentů v organizaci za přesně stanovených podmínek, který kopíruje procesní potřeby organizace na tuto část ECM.

Tabulka 3: Procesy spisové služby.

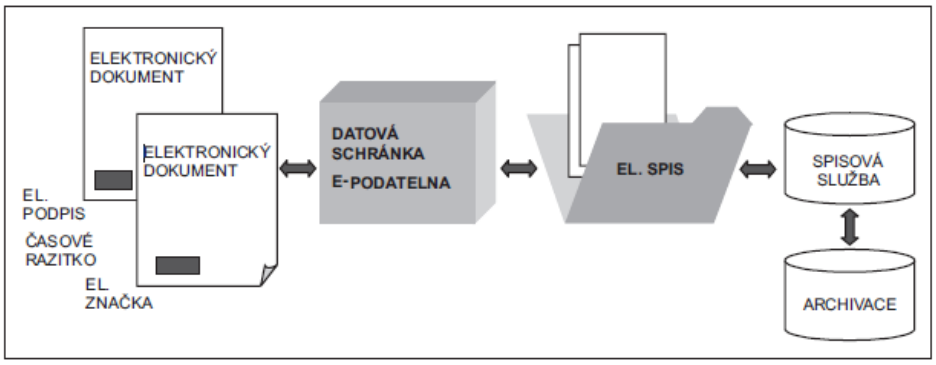
Proces	Popis procesu
Příjem (podatelna, e-podatelna)	Elektronická podatelna (e-podatelna) je zaměřena na příjem datových správ, jejich uložení, evidenci, ověření náležitostí a následnému předání k dalšímu vyřízení. Je řízena přesnými pravidly, viz. vyhláška č. 495/2004 Sb. Podatelna přijímá dokumenty v listinné podobě a provede jejich digitalizaci a ostatní činnosti jako e-podatelna.
Evidence	Všechny přijaté dokumenty jsou zadány do IS s jedinečným identifikátorem, který mají po celou dobu svého životního cyklu. Dokumenty jsou roztříděny a škálovány do skupin podle účelu využití (např. faktury atd.). Dalším způsobem je vytváření tzv. spisu (kde jsou vztaženy dokumenty k jedné věci např. k reklamaci atd.) Vše je řízeno podmínkami spisového řádu původce. Tato činnost může být generována sofistikovaným automatizovaným systémem. Evidence uchovává historii po celou dobu životního cyklu dokumentu.
Rozdělování a oběh dokumentů	Každý dokument má určen svého vlastníka (příjemce) podle předefinovaných účelů použití a každá změna vlastníka je zaznamenána (přidělení, předání, převzetí odmítnutí). Tok dokumentu organizací je na základě předefinovaného účelu použití daného dokumentu. Pokud jsou vyřešeny všechny záležitosti celého spisu, dochází k jeho uzavření a ukončení. Tato část aplikace umožňuje provádět průběžnou kontrolu nad daným spisem.
Odesílání	Odeslání dokumentů je realizováno stejným způsobem jako jejich příjem. Pro každý dokument zařazený do dané skupiny na podatelně existuje pro každý krok tzv. rozdělovník (seznam oprávněných uživatelů) a způsob vypravení (pošta, e-mail, osobně)
Vyřízení, uzavření	Dokument se vyřizuje samostatně nebo jako součást spisu. Kdy vyřízením se rozumí ukončení po splnění všech náležitostí. Pokud je vyřízen i spis je mu přidělena speciální identifikace tu má každý dokument, ze spisu a celý spis je standardně ukončen. K ukončenému dokumentu nebo spisu je přidán příznak „ukončení“ a datum uzavření. Dle pravidel organizace začíná běžet skartační lhůta.
Ukládání, archivace	Pro ukládání listinných dokumentů je zřizována speciální místnost tzv. spisovna, kde jsou dokumenty archivovány podle spisového a skartačního řádu organizace. Pro zajištění co největší automatizace správy toho úložiště (spisovny) je veden systém evidence (archivní kniha) na základě důležitých evidenčních atributů. Pro elektronickou verzi je samozřejmě použit obdobný systém aplikovaný v dané aplikaci. Dále musí být zabezpečena neměnnost obsahu všech dokumentů a spisů se systémem zálohovaní dle Data Managementu organizace. [42]

Proces	Popis procesu
Skartační řízení, vyřazování	Do skartačního řízení jsou přesouvány dokumenty a spisy podle uplynulé skartační lhůty. Následně probíhají standardizované procesy podle skartačního řádu organizace a to jak pro listinné tak pro digitální dokumenty nebo spisy.

Zdroj : [Vlastní tvorba]

Komponeta ECM, která se zabývá vytěžováním dat, je zaměřena na realizaci získání dat z digitalizovaného dokumentu s jejich následným uložením do strukturované datové struktury organizace (např. ERP databáze), a je implementována z pohledu životního cyklu obsahu ECM v části „Pořízení“ (viz. obrázek 10).

Přínosem spisové služby je standardizace, zabezpečení a zefektivnění veškerých jejich procesů s dokumenty tak, aby vyhovovaly podmínkám legislativy. Standardizace zavádí do organizace přehledná a jasná pravidla při toku dokumentů v rámci schvalovacích procesů s možností notifikovat účastníky procesů. Jde o eliminaci chyb při těchto procesech a zajištění standardního auditování této části IS. Další možností je tvorba sofistikovaných dotazů, analýz aj.



Obrázek 12: ECM prvky elektronické komunikace a archivace.

Zdroj : [42]

1.5.2 RMS (Record Management System)

RMS (Record Management System dále je RMS) neboli správa záznamů pokrývá v rámci životního cyklu obsahu organizace fázi publikování a archivace. Základní charakteristika RMS systému je v tom, že soubory do nich vložené jsou již neměnné, slouží jako zdroj informací a jejich zrušení musí odpovídat předpisům a procesům organizace. Zavedením RMS systému organizace máme k dispozici úložiště se standardizovanou, zabezpečenou strukturou, která nám umožňuje optimalizaci nákladů na správu této části dokumentů v organizaci. Obvykle RMS obsahuje záznamy veškerých obchodních aktivit a transakcí, vyjednávání kontraktů, firemní korespondenci, finanční výkazy atd. [42]

1.5.3 E-mail Management System

E-mail je základním pilířem podnikové komunikace dneška. Jeho prostřednictvím jsou předávány důležité dokumenty, je vyřizována obchodní korespondence, jsou distribuovány manažerská rozhodnutí a jsou přes něj sdělovány informace důvěrného charakteru. Cílem komponenty je zamezit ztrátě důležitých informací a využít e-maily jako důležitý informační zdroj organizace. Proto jsou tyto zprávy přesunuty do sdíleného úložiště, kde jsou k dispozici k dalšímu využití. Při přesunu jsou e-maily automaticky zálohovány a do úložiště je uložen e-mail i s přílohami. E-mail je uložen podle klasifikace (účelu využití), jsou vygenerována potřebná metadata a je nastaveno propojení s jinými zprávami dle jejich charakteru a záznamu je přiřazen příslušný skartační plán. Přínosem je omezení ztráty informací, z důležitých činností organizace s vytvořením informační platformy, která je zdrojem pro různé standardní nebo jednorázové analýzy. [42]

1.5.4 CMS (Content Management System)

Tato část je zaměřena na systémy, které pracují převážně s nestrukturovanými daty, která nejsou čistými textovými dokumenty

- **Web Content Management Systém**

Komponenta ECM pro správu WEB obsahu je založena na principu víceuživatelské správy webových stránek. Tato komponenta slouží k publikování informací koncových uživatelů do standardního WEB prostředí organizace. Publikují data, která jsou primárně uložena v různých úložištích a jsou v různých výstupních formátech. [42]

- **Digital Asset Management System**

  Tato komponenta je specializovaná pro práci s multimediálními daty. Tato komponenta umožňuje vkládání, anotaci, kategorizaci, uložení vyhledání a zabezpečený přístup, ke kompletnímu spektru multimediálních dat. Jde o uložení multimediálních dat v úložišti včetně jejich metadat, která slouží pro jejich identifikaci, kdy dále tato komponenta může obsahovat funkcionality pro práci s těmito daty (přehrávání, transformace atd.). [42]

- **Collaboration Tools**

  Je součástí týmové spolupráce integrované v ECM a jde o software, který umožňuje dvěma a více lidem navzájem komunikovat, kooperovat a koordinovat své aktivity. Collaboration tools je založen na třech formách spolupráce a to na komunikaci (psaná,

zvuková výměna informací), kooperaci (víceuživatelská komunikace), koordinaci (vzájemné časové sladění aktivit). [42]

1.5.5 Workflow

Jde o automatizaci celého nebo části procesu organizace, kdy v jeho průběhu jsou přenášeny dokumenty, informace od jednoho účastníka procesu k druhému podle sady procedurálních pravidel. Workflow slouží k monitoringu průběhu jednotlivých procesů v organizaci, protože shodné procesy probíhají jednotným postupem a měly by být realizovány stejným způsobem.

Workflow podle užití členíme na:

- Administrativní workflow (automatizace jednoduchých administrativních činností)
- Produkční workflow (jde o automatizací hlavních podnikových procesů)
- Ad hoc workflow (je jednoduchý automatizační proces vytvářený uživatelem)
- Kolaborativní workflow (procesy zaměřené na posloupnost spolupráce uživatelů)

1.5.6 Dílčí souhrn pro ECM

Tok dokumentů organizací není samovolným procesem, ale v současných organizacích je součástí nejen správy informačního obsahu (ECM), ale především důležitou komponentou celého informačního systému organizace. Dokumenty provází standardizované procesy v celé hierarchii organizace a nejsou jen doplňkovou informací. Dokumenty souží i jako zdrojová data pro různé procesy organizace (např. faktury atd.). Tuto informaci si lze ověřit na Informační pyramidě IS, kdy ECM prolíná dle typu organizace nejen do systémů OIS (Office Information System (dokumenty z kancelářských softwarových balíčků jako např. MS OFFICE), ale z části i do EDI (Electronic Data Interchange: např. e-mail, WEB technologie) a do EIS (Executive Information System), MIS (Management Information System), TPS (Transaction Processing System). Podrobnosti k informační pyramidě viz. část této práce „1.4 Dílčí shrnutí“.

Dokumenty jsou velmi rozmanité, protože mohou obsahovat jak strukturovaná, semistrukturovaná a nestrukturovaná data, proto je důležité vždy provést podrobnou analýzu, která nám upřesní možný způsob využití dat z tohoto uceleného souboru informací.

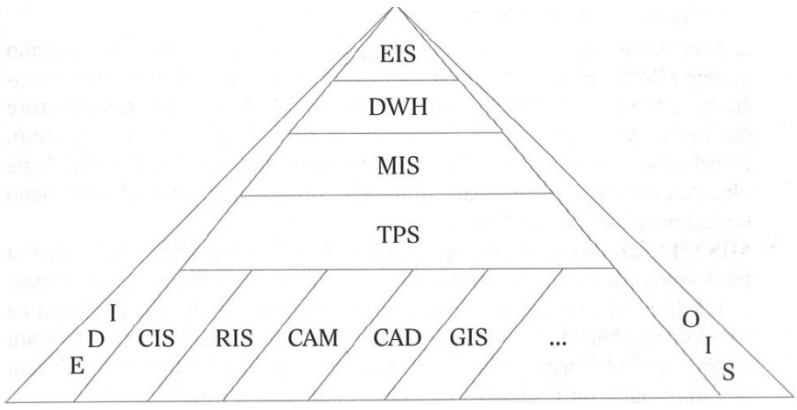
1.6 Dílčí souhrn

Data a informace s rozvojem informačních technologií jsou stále významnější komoditou, která ovlivňuje chod a řízení téměř veškerých organizací. Proto je nutné data sofistikovaně spravovat pomocí standardizovaných aplikací, které využívají metodiky Data Managementu, a využít jejich potenciál pro dosažení strategických cílů organizací nebo konkurenční výhody.

V současnosti je velký rozvoj zpracování a využití semistrukturovaných a nestrukturovaných typů dat. U semistrukturovaných dat jde o globální rozvoj technologií IoT, a to jak v podnikové sféře, tak i v běžných domácnostech. Jde převážně o technologii sběru a správy dat ze "senzorů", což má svá technická specifika podle použitých technologií.

Další skupinou jsou nestrukturovaná data, která dle dnešních odhadů představují cca 80 % všech vytvořených a vygenerovaných dat. Tyto data jsou chápana jako doplnění standardně využívaných strukturovaných dat v současných informačních systémech organizací. Kdy tyto data při správném zpracování a využití mohou vytvořit konkurenční výhodu dané organizace nebo mohou alespoň zefektivnit dosažení strategických cílů organizace. Pro tyto činnosti je možné využít metodiky jako je ITIL (IT Infrastructure Library), COBIT (Control Objectives for Information and Related Technology).

Pro ucelenost pohledu na oblast využití a správy dat v organizaci využijeme pyramidu informačních systémů z hlediska jejich architektury. Kde se zaměřím především na organizaci typu podnik, protože organizace veřejné správy jsou velmi specializované z důvodu účelu, pro jaký byly zřízeny.



Obrázek 13: Informační pyramida.

Zdroj : [42]

Tabulka 4: Systémy Informační pyramidy.

EIS (Executive information system) – tyto systémy podporují vrcholné řízení organizace (např. Strategické řízení, finanční řízení, marketing).
DHW (Data Warehouse) – Datový sklad je vytvořen pro dlouhodobou archivaci vybraných dat na základě sofistikovaných analýz, které byly získány z provozních systémů a slouží i jako zdroj pro IS MIS.
MIS (Management information system) – Tyto systémy podporují převážně taktickou a operativní úroveň řízení v organizacích. Jejich hlavním cílem je podpora rozhodování středního managementu v organizaci. Správnost a rychlost tohoto rozhodování má zásadní vliv na plnění strategických cílů organizace. Při využití DHW je možné získat velmi sofistikované analýzy, které lze využít pro naše rozhodovací procesy.
TPS (Transaction Processing System) – Jsou v něm realizovány každodenní procesy organizace (např. jako podpora výrobních, logistických, účetních procesů atd….)
CIS (Customer Information System) – Zajišťuje bezprostřední styk se zákazníky organizace.
RIS (Reservation Information system) – Správa rezervačních systémů v organizaci (např. Logistika, cestovní kanceláře atd.)
GIS (Geographical Information System) – Správa geografických dat, tvorba map, územních modelů atd.
CAD (Computer Aided Design) – Tyto systémy jsou určeny pro konstrukční a návrhářské procesy, jejímž výsledkem je konstrukční řešení.
CAM (Computer Aided Manufacturing) – Autorizovaná podpora řízení výrobních procesů.
OIS (Office Information System) – Podpora rutinních kancelářských činností (správa dokumentů, elektronická pošta atd.)
EDI (Electronic Data Interchange) – Systémy pro standardizovanou výměnu dat mezi IS systémy. Jsou určeny pro datovou komunikaci mezi obchodními partnery nebo pro komunikaci s bankovními ústavy atd.

Zdroj : [Vlastní tvorba]

Z výše uvedených informačních systémů může být vytvořen celopodnikový informační systém typu ERP (Enterprise Resource Planning), který zajišťuje coloživotní cyklus organizace. ERP systém má obvykle u výrobních organizací specifikován výrobní informační systém, který je obecně identifikován jako MES (Manufacturing Execution Systems) a ten má za úkol integrovat důležité výrobní procesy do celopodnikových procesů.

Charakterizaci dat, která jsou v současnosti prezentována pod pojmem Big Data, se budu věnovat části „2. BIG DATA“.

2. BIG DATA

S rozvojem nových technologií jsou každý den při pracovních a volnočasových aktivitách generovány miliony transakcí, e-mailů, fotografií, multimediálních souborů, dat ze senzorů a zařízení IoT, příspěvků na sociálních sítích atd., které vytvářejí zettabyty dat. Všechna tato data obsahují nepřeberné množství informací. Analýzou takových dat a informací by mohly organizace (veřejná správa, firmy, výzkumné instituce atd.) objevit a získat velké množství zajímavých znalostí, které by mohly využít pro dosažení svých strategických cílů. Vzhledem k velkému objemu dat, který je každodenně generován, je stále obtížnější tato data kontinuálně (ideálně v reálném čase) zachycovat, formovat, ukládat, spravovat, analyzovat, vizualizovat a extrahovat z nich cenné znalosti. Tato oblast je v současnosti označována zavedeným anglickým termínem „Big Data“.

Co vlastně představuje termín Big Data? Velká data je jeden z možných překladů termínu Big Data do českého jazyka, který může zahrnovat velkoobjemová data nebo velké množství malých datových struktur anebo velké množství dat vygenerovaných v krátkém čase, která mohou být různorodá. Ano, tento popis je pravdivý, ale pohled na termín Big Data je nutno chápat komplexněji, protože představuje doménu znalostí, která se zabývá metodikami, technikami, dovednostmi a technologiemi, které jsou určeny k odvození cenných poznatků (znalostí) z obrovského množství dat, které nelze spravovat nebo zpracovávat tradičními softwarovými a hardwarovými prostředky v přijatelné časové míře.

2.1 Historie Big Data

Termín Big Data je intenzivněji využíván od 90. let 20. Století, přesnější termín bohužel není znám. Tento termín zpopularizoval John R. Masheye, který v té době pracoval v společnosti Silicon Graphics. Big Data ve své podstatě není novým termínem, protože v průběhu staletí probíhal při daných technických možnostech vývoj analytických technik s analýzou dat, která sloužila pro podporu rozhodování. V posledních dvou desetiletích došlo k změně objemu a rychlosti generování dat. Celkové množství dat na světě bylo v roce 2013 4,4 zettabytů. Do roku 2020 mělo množství dat prudce vzrůst na 44 zettabytů (44 bilionům gigabajtů). Bohužel i s nejpokročilejšími technologiemi současnosti je nemožné všechna tato data analyzovat. Potřeba zpracovávat tyto stále větší (převážně nestrukturované) soubory dat je způsob, jakým se tradiční analýza dat v posledním desetiletí transformovala na oblast Big Data.[4] Problematiku Big Dat je možné rozdělit na tři hlavní fáze. Každá z fází má své vlastní charakteristiky. Abychom pochopili kontext Big Data dnes, je důležité pochopit, jak jednotlivé fáze přispěly k současnému významu Big Data. [5]

Big Data fáze 1.0 (1990 -1999)

Analýza a analytika Big Dat je založena na znalostech správy databází, kde spoléhá na techniky správy relačních databází (RDBMS). Správa databází a datové sklady jsou považovány za klíčové komponenty Big Data. Pro potřeby Big Data byla vytvořena moderní analýza dat, jaká je známá v dnešní podobě, např. využití známých technik, jako jsou databázové dotazy, online analytické zpracování a standardní nástroje reportingu. [5]

Big Data fáze 2.0 (2000 – 2009)

Na počátku nového milenia vznikají nové online WEB technologie, které vytvářejí nové příležitosti pro analýzu dat (např. Yahoo, Amazon atd.). Tyto technologie umožnily masivní nárůst objemu semistrukturovaných a nestrukturovaných dat. Proto organizace musely hledat nové přístupy a řešení s tvorbou nových úložišť schopných spravovat a efektivně analyzovat tyto datové struktury. Mezi další fenomén tohoto období patří nárůst obliby sociálních sítí a médií, který způsobil nárůst nestrukturovaných dat. Pro tato nestrukturovaná data musely být vyvinuty nové techniky a technologie, které umožnily z nestrukturovaných dat extrahovat smysluplné informace. [5]

Big Data fáze 3.0 (2010 - ….)

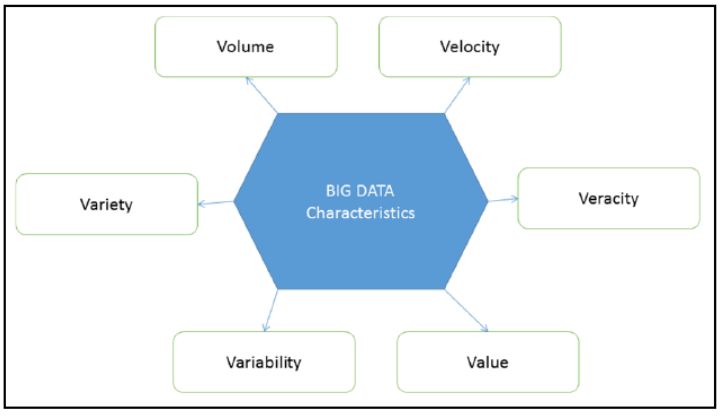
Od roku 2010 do současnosti probíhá tzv. třetí fáze využívání Big Dat z pohledu historického vývoje. WEB nestrukturovaný obsah je stále dominantním zdrojem dat, které mohou organizace využít pro své datové analýzy a další zavedené procesy s Big Data. Mimo WEB technologií nastává velký rozvoj získávání dat a cenných informací z mobilních zařízení a senzorů. Mobilní zařízení také poskytují možnost analyzovat údaje o chování jejich uživatelů (např. využívání mobilních aplikací), ale také umožňují ukládat a analyzovat údaje založené na poloze (data GPS). S pokrokem těchto mobilních zařízení je možné sledovat pohyb, analyzovat fyzické chování a dokonce i údaje týkající se zdraví (Fitness aplikace). Tato data poskytují zcela novou škálu příležitostí. Současný vzestup používání zařízení, založených na senzorech se stálým připojením na internet, zvyšuje generování dat jako nikdy předtím. Miliony různých domácích spotřebičů, termostatů, atd., které jsou známé jako „internet věcí“ (IoT), nyní generují každý den zettabyty dat. A procesy, které mají zajistit získávání smysluplných a cenných informací z těchto nových zdrojů dat jsou na počátku a očekává se jejich velmi dynamický vývoj. [5]

2.2 Charakteristika Big Data

V roce 2001, Doug Laney, který se věnoval problematice data managementu, publikoval první charakteristiku Big Data pomocí charakteristik "V" (zmínil existenci tzv. 3V = Volume, Velocity, Variety) aby je odlišil od ostatních dat. Jde o charakteristiky Big Data, které v anglickém originále začínají na písmeno "V". V následujících letech byly "3V" charakteristiky Big Data na základě rozvoje poznání v této oblasti rozšiřovány o další důležité "V" charakteristiky. [39]

Při studování dostupné literatury jsem dospěl k názoru, že počet využitých "V" charakteristik Big Data je velmi individuální, protože v první řadě je potřeba získat požadovanou kvalitu dat dle pravidel data managementu pro veškeré následné procesy (datové vědy) s Big Data. Kolik a jaké charakteristiky Big Data využijeme pro danou studii (např. 3V, 4V, 5V … 10V …) je důležité pro získání relevantních finálních informací, kdy se konkrétní studie implementace Big Data po ověření může stát součástí informačních systémů organizace. Tato optimalizace "V" charakteristik Big Data může přinést velké úspory nákladů při využití dané studie, ale musí být zachována požadovaná úroveň kvality dat studie tak, aby byly výsledky zpracování a analýz relevantní a srovnatelné v požadovaném rozsahu časové linie, která nám vytváří interval životního cyklu využití dané studie pro potřeby organizace.

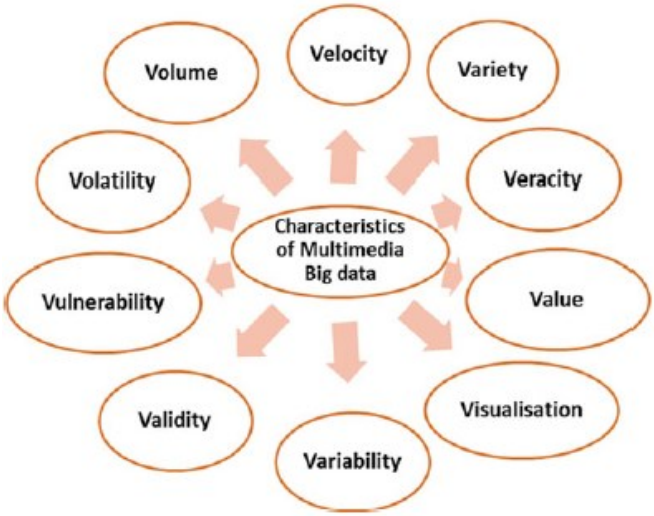
V současnosti je doporučovaný standard pro Big Data 6V (Volume, Velocity, Variety, Veracity, Variability a Value) charakteristik. [41]



Obrázek 14: 6V charakteristiky Big Data.

Zdroj : [41]

Pro multimediální Big Data je doporučeno 10V charakteristik (Volume, Velocity, Variety, Veracity, Value, Visualisation, Variability, Validity, Vulnerability, Volatility). [41, 43]



Obrázek 15: 10V charakteristiky pro multimediální Big Data.

Zdroj : [43]

"V" charakteristiky Big Data:

V této práci budu používat zažité anglické názvosloví, protože české ekvivalenty dle mého názoru potřebují při jejich použití další upřesnění, proto aby nedošlo k nějakému mylnému výkladu.

Volume (Objem)

Volume je nejdůležitější charakteristikou pojmu Big Data, a to nejen proto, že podle velkého objemu dat jsou pojmenována. Jak jsem se již v této práci zmínil, jsou s rozvojem informačních a komunikačních technologií, každý den při pracovních a volnočasových aktivitách generovány velká množství dat, která jsou odhadována v řádů zettabyte (ZB). V budoucnu nás očekávají jistě vyšší množství vygenerovaných dat v řádech yottabyte (YB), brontobyte až geopbyte. Termínem Big Data by měly být označována data v řádech petabyte (PB), ale v prostředí České Republiky je tento termín využívám již pro data v řádech terabyte (TB).

Velocity (Rychlost)

Jde o definování rychlosti nárůstu objemu Big Data s relativní přístupností k datům. Velocity pomáhá organizacím pochopit relativní růst objemu jejich dat a jak rychle jsou tato data dostupná uživatelům, aplikacím a systémům. Velocity zahrnuje faktory, jako je reakce aplikace, čas na provedení transakce, časové nároky na analýzu dat, a automatické a rychlé aktualizace ve všech datových úložištích, kde jsou data uložena. Rychlost přímo souvisí s celou datovou infrastrukturou a architekturou při správě a co nejrychlejším doručování dat příjemcům. Z toho důvodu řešení problematiky Big Data z pohledu charakteristiky Velocity využívá techniky jako

je např. ukládání dat do mezipaměti pro rychlejší přístup k datům důležitých aplikací a služeb, periodickou extrakci a orchestraci dat ve všech úložištích dat s nasazení architektury a infrastruktury s minimální datovou a síťovou prodlevou (latencí).

Způsob zpracování dat je závislý na tom, kdy organizace tato data potřebuje pro svoji činnost. Rozeznáváme tyto základní způsoby zpracování dat: dávkový, near-time, real-time a streaming.

- Dávkové zpracování dat je založeno na zpracovávání ucelených bloků dat v periodách pravidelných (např. denních, týdenních atd.) nebo nepravidelných (např. zpracování proběhne, až po dosažení určitého objemu dat atd.). Data jsou obvykle agregována a ukládána do úložišť, kde slouží k dalším analýzám a reportingu organizace (např. pomocí ETL atd.). [39]

- Near-time zpracování dat je obvykle založeno na událostním zpracování, které může mít např. bezpečností prodlevu trvající kratší dobu (v řádu minut nebo hodin), neboli proces odezvy se zasláním dat je přerušen a až po zpracování jsou data k dispozici. [39]

- Real-time zpracování dat je založeno na událostním zpracování, kdy je zaslaný požadavek zpracován v reálném čase a datová odezva je v řádu sekund. [39]

- Data streaming (datový proud) obecně představuje v problematice Big Data datový proud, který je principiálně založen na zaslání posloupnosti dat v reálném čase (např. ve formě paketů nebo ve formě komunikačního protokolu). Datový proud je vyvolán událostním způsobem, a vlastní zpracování a analýza dat je závislá na využití dat v organizaci. Data jsou obvykle zpracována v real-time a uložena do úložiště, proto aby se uvolnil komunikační kanál pro zaslání nových dat dalšího datového proudu. Následné zpracování dat a jejich analýza je závislá na potřebách organizace (on-line řízení procesů atd.). [39]

Variety (Různorodost)

Tato charakteristika popisuje Big Data z pohledu datových struktur (strukturovaná, semistrukturovaná, nestrukturovaná – podrobnosti viz. „1.2.1 Typy dat podle struktury“), podle místa a zdroje vzniku, na interní a externí (viz. „1.2.2 Typy dat podle místa vzniku).

Veracity (Věrohodnost/Pravdivost)

Veracity se zabývá nejistotou dat, která může být způsobena jejich špatnou interpretací nebo nějakým šumem v datech. Charakteristiky Velocity a Variety jsou závislé na čistých datech před vlastní analýzou, zatím co Variety je odvozena z nejistoty dat a je nutné, určit zda jsou data věrohodné. Bohužel při zpracování např. datových proudů, kdy jsou data generována velmi vysokou rychlostí, nelze zcela eliminovat nejistotu dat tím, že provedeme jejich vyčištění a následně je zpracujeme, protože taková data mohou ztratit vypovídající hodnotu. Proto je nutné stanovit požadovanou mezní úroveň věrohodnosti pro Big Data, aby byla zajištěna jejich správná interpretace a použitelnost. [39]

Variability (Proměnlivost/variabilnost)

Variability je charakteristika, která je založena na změně významů a vývoje významů dat s praktickým využitím např. u analýzy sentimentu. [39]

Value (Hodnota)

Použití daného segmentu Big Data v organizaci by mělo být smysluplné z pohledu nákladů a přínosů, které by měly vytvořit pro organizaci zvýhodnění a tak jí pomoci k dosažení strategických cílů. [39]

Visualization (vizualizace dat)

Zpracovaná a zanalyzovaná Big Data, která nejsou správně vizualizována, nepřináší organizaci žádný benefit. Přímá vizualizace Big Data není možná, a to vzhledem k současným technickým omezením z důvodů nedostačující operační paměti, špatné škálovatelnosti, funkčnosti a doby odezvy zpracování (např. není reálné vytvořit tradiční graf z miliardy údajů …). Proto jsou pro reprezentaci Big Data využívány nové metody, např.: shlukování dat, stromové mapy, metody paralelních souřadnic, kruhové síťové diagramy, grafická prezentace ve formátu růžice slunečních paprsků (sunbursts). Bohužel vzhledem k používanému velkému množství proměnných zkombinovaných s charakteristikami, jako jsou Velocity, Variety a složitých vnitřních vzájemných vztahů v Big Data, je zřejmé, že vývoj jejich smysluplné vizualizace není snadný. [43]

Validity (validita dat)

Tato charakteristika je velmi podobná Veracity, odkazuje na to, jak přesné a správné by měla být Big Data pro jejich předpokládané využití. Odhaduje se, že až 60 % času datového vědce

je věnováno na vyčištění dat, než je bude možné využít pro jakoukoli analýzu. Výhoda z analýzy Big Data je pouze, pokud jsou pro ni využita „kvalitní data“. Proto je nutné využívat osvědčené postupy správy dat, aby byla zajištěna konzistentní kvalita dat (viz. „1.3 Kvalita dat“), společné definice a metadata. [43]

Vulnerability (zranitelnost/náchylnost k chybování)

Charakteristika Vulnerability je v Big Data chápána jako neschopnost (systému, organizace) odolat účinkům nepřátelského prostředí, která je vyvolána softwarovým nebo hardwarovým bezpečnostním problémem. Útočník využívá Vulnerability pro přístup do informačních systémů organizace a následně může organizaci způsobit nenávratné škody na jejich datech. Preventivním opatřením na snížení Vulnerability je vytváření souboru bezpečnostních opatření, které vznikají na základě analýz rizik organizace. Dalším důležitým opatřením jsou pravidelné audity zranitelnosti organizace. [43]

Volatility (těkavost)

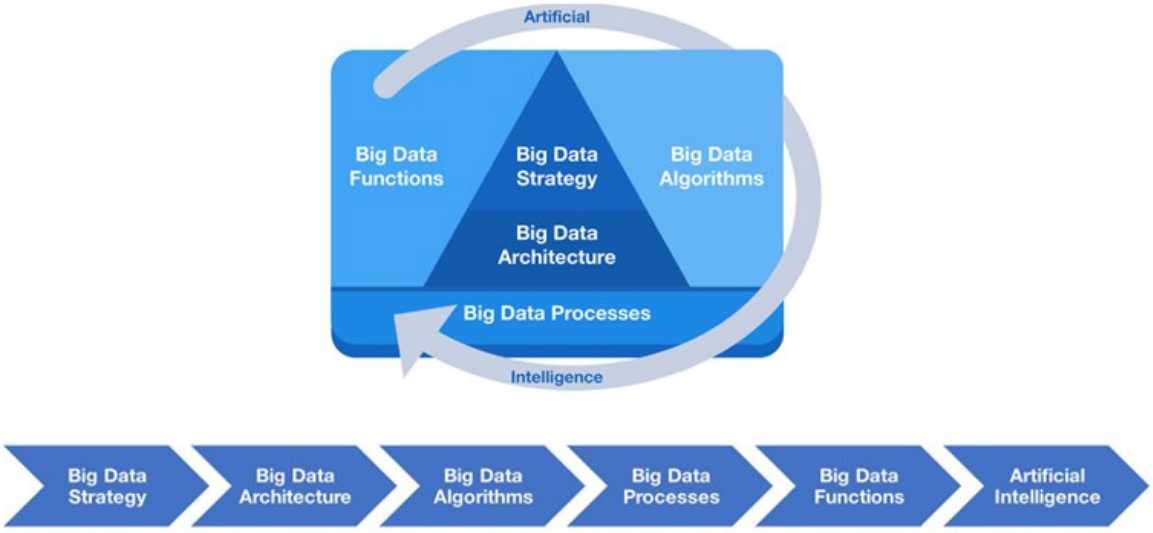
Charakteristika Volatility řeší problematiku, jak dlouho uchovávat Big Data pro potřeby a procesy organizace, než budou považovány za irelevantní. Tato charakteristika Big Data nabývá stále více na důležitosti, protože objem dat neustále stoupá a náklady na čím dál větší datová úložiště rostou. [43]

2.3 Big Data Framework

Pro smysluplné využití Big Data v organizaci je nejprve nutné si ujasnit, co dělá organizaci úspěšnou a jak mohou být technologie a řešení Big Data prospěšné pro zefektivnění procesů organizace a dosažení strategických cílů organizace. Proto je nutné vytvořit v každé organizaci, která má v úmyslu implementovat problematiku Big Data, obecný rámec pro podporu implementace Big Data (Big Data Framework), který specifikuje posloupnost důležitých činností pro implementaci technologií a řešení Big Data do procesů dané organizace tak, aby byly přínosné.

Jak jsem již uvedl, jde o obecný rámec, který popisuje implementaci Big Data (dále budu používat termín Big Data Framework), který je pro každý projekt konkrétní implementace Big Data v organizaci pouze doporučeným rámcem postupů a technik, které by měly zajistit správnou a pro organizaci přínosnou implementaci Big Data.

Obecně je Big Data Framework v literatuře popisován jako otevřený standard pro každou organizaci, který je složen z několika návazných součástí, které bez správného pořadí implementace nemohou správně fungovat. Příklad je Big Data Framework na obrázku č. 11.



Obrázek 16: Obecný rámec implementace Big Data Framework.

Zdroj : [5]

Big Data Framework : (interní členění)[5]

- Big Data Strategy (strategie Big Data)
- Big Data Architecture (architektura Big Data)
- Big Data Algorithms (algoritmy Big Data)
- Big Data Processes (procesy Big Dat)
- Big Data Functions (funkce Big Data)
- Artificial Intelligence (umělá intelligence)

2.3.1 Big Data Strategy (strategie Big Data)

Tato část Big Data Framework má za úkol přesně zmapovat a definovat vlastní potřebu organizace na implementaci Big Data v konkrétní oblasti činnosti organizace. Základem je vytvoření Big Data strategie organizace, která je prosazována od TOP managementu organizace přes všechny její úrovně managementu; jen tento přístup zajistí, že implementace Big Data bude aktivem organizace. Prvním krokem je dokonalé seznámení se se segmentem organizace, kde předpokládáme zavedení Big Data a jeho aktuální analýzu. Pokud výsledky analýzy signalizují přínosnost dané implementace Big Data, je možné přistoupit k definování využitelných oblastí pomocí nástrojů, jako jsou business case a use case s maticí priorit pro všechny možné scénáře

využití Big Data v daném segmentu organizace. Následně vytvoříme Big Data Roadmap (mapu předpokládané implementace komponet Big Data v závislosti na analyzovaném segmentu organizace), která nám nastíní, jaké projekty jsou primární a realizovatelné se zaměřením na data. S popisem jejich rizik vzhledem k zamýšlenému projektu a na použití předpokládané architektury, technologie a nástrojů Big Data. Další oblastí je předpokládaná modifikace všech procesů organizace s proškolením lidských zdrojů, tak aby implementtace Big Data v daném segmentu organizace byla aktivem. Dalšími kroky je stanovení cílů a vytvoření vlastního plánu implementace Big Data a jeho případná realizace. [5]

2.3.2 Big Data Architecture (architektura Big Data)

Pro plné využití potenciálu Big Data je nutné využít správné technologie Big Data s co nejvyšší integrací těchto technologií do IT architektury dané organizace a vytvořit tak společnou IT architekturu celé organizace neboli referenční architekturu.

Referenční architektura je soubor doporučení pro integraci IT produktů a služeb dle požadavků a potřeb dané organizace. Poznatky z oblasti referenční architektury vytvářejí doporučení pro implementaci daných oblastí Big Data na základě zkušeností, svědčených postupů, které jsou zobecněny, zoptimalizovány a slouží jako základní informace pro manažery, vývojáře, IT architekty, tedy pro všechny zainteresované profese, které se podílejí na dané implementaci Big Data. Cílem referenční architektury je vytvoření otevřeného standardu, který každá organizace může využít ve svůj prospěch. Příkladem je referenční architektura NBDRA, která byla vyvinuta v národním institutu pro standardy a technologie NITS, viz. obrázek č.12. [5]

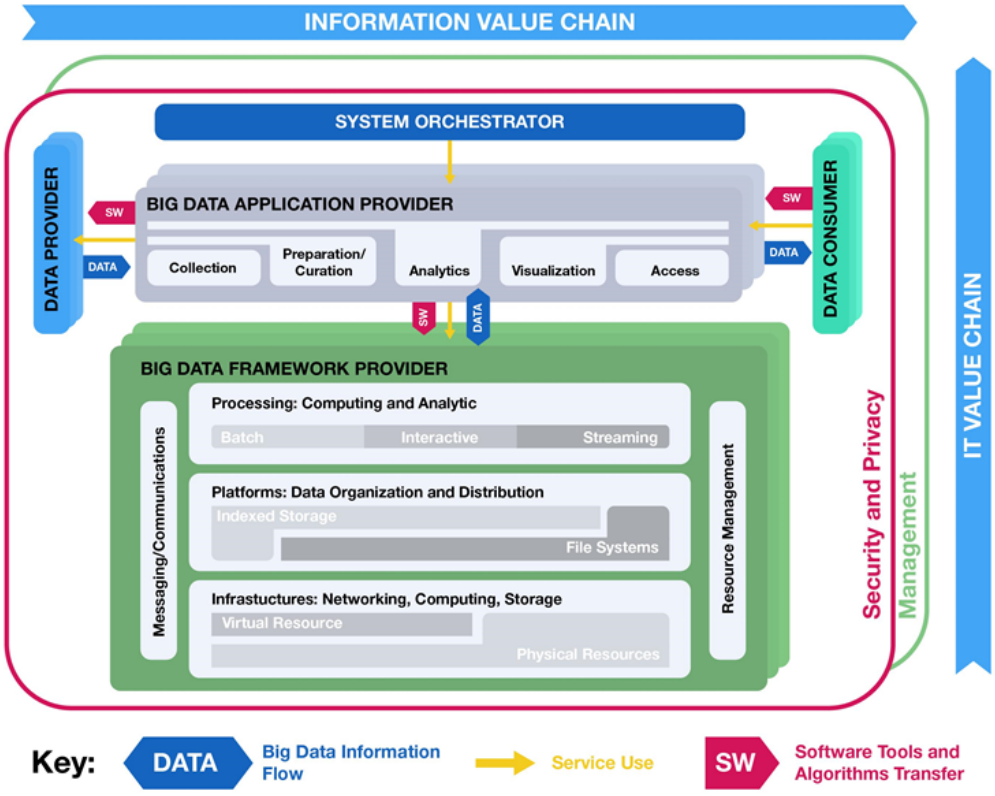
Jaké jsou přínosy použití referenční architektury: [5]

- Společný způsob komunikace (terminologie atd.) pro všechny zúčastněné strany.
- Společné normy, technické specifikace a vzory.
- Konzistentní metody implementace technologií pro řešení obdobných problematik.
- Využití konceptuálního modelu Big Data na základě porozumění různým procesům a systémům Big Data s efektivnějším výběrem finální technologie a dodavatele.
- Analýza standardů vhodných pro interoperabilitu, přenositelnost, opakovatelnost a rozšiřitelnost.

Referenční architektura NBDRA není závislá na prodejci a je využitelná v jakékoliv organizaci, která má za cíl vyvinout architekturu Big Data. NBDRA je složena z pěti logických komponent, které jsou propojeny pomocí rozhraní a služeb. NBDRA ve své struktuře využívá dva

hodnotové řetězce Big Data, a to Informační hodnotu (Information Value Chain) a Informační technologii (IT Value Chain). [5]

Informační hodnota NBDRA je vytvářena pomocí sběru dat, integrace, analýz a aplikací výsledků. Pro rámec IT technologií je hodnota vytvářena pomocí síťové a další infastruktury, aplikačními nástroji, a IT službami pro hostování a provoz Big Data. V průsečíku obou hodnotových řetězců je zřejmé, že jak analýza dat, tak i její implementace (technologie) poskytují hodnotu v hodotových řetězcích. [5]



Obrázek 17: NITS Big Data Reference Architecture.

Zdroj : [5]

Mezi logické komponenty NBDRA v prostředí Big Data, které jsou přítomny v každé organizaci, patří: [5]

- Systémový orchestrátor (Systém orchestrator)
- Poskytovatel dat (Data provider)
- Poskytovatel Big Data aplikací (Big Data application provider)
- Poskytovatel Big Data framevorků (Big Data framework provider)
- Spotřebitel dat (Data consumer)

Systémová orchestrace představuje automatizovanou koordinaci a správu uspořádaných IT systémů a služeb. Zajišťuje synchronní spolupráci všech komponent IT aplikací, dat a infrastruktury v celé organizaci. [5]

Poskytovatel dat umožňuje vstup nových dat, informačních zdrojů pro vyhledávání, přístup a transformace do systému Big Data. Protože jednou z významných charakteristik dat je Variety (různorodost), mohou mít různé datové zdroje rozdílné způsoby zabezpečení a ochrany soukromí. Vlastní přenos dat je složen ze tří fází: zahájení (požadovaná úroveň autentizace), přenos (přenos dat od poskytovatele dat k poskytovateli Big Data aplikací), ukončení (kontrola úspěšnosti přenosu dat). [5]

Poskytovatel Big Data aplikací je nezbytná komponenta architektury Big Data pro transformaci vstupních dat na data požadovaná pomocí extrakce hodnoty z vstupních dat. Obsahuje prvky funkčnosti a logiky používané v dané oblasti organizace (např. obchodní logika). Tato komponenta zahrnuje tyto činnosti: sběr dat, přípravu dat, analytiku dat, vizualizaci dat a přístup k datům. Využití této komponenty je velmi různorodé a je závislé na typu organizace (výrobní organizace ji mohou využít pro optimalizaci dodavatelského řetězce, správu zásob atd.). [5]

Poskytovatel Big Data framevorků, poskytuje zdroje a služby pro základní infrastrukturu architektury Big Data (aplikace, systémy atd.). Prostředí této komponenty je plně optimalizované pro prostředí Big Data. Na tuto komponentu je možné nahlížet z pohled tří vrstev, a to: infrastruktury (počítačové sítě a technika, úložiště), platformy (podle organizace a distribuce dat), zpracování (jak výpočetní tak analytické). [5]

Vrstva infrastruktura pro prostředí Big Data je velmi sofistikovaná, protože musí pracovat s velkým množstvím datových formátů a objemem dat, který není obvykle uložen v jednom úložišti a je nutné s daty pracovat efektivním, bezpečným a škálovatelným způsobem. Infrastruktura se musí stále přizpůsobovat růstu organizace a zachovat požadovaný standard v rychlosti přenosu dat. [5]

Vrstva platforma je kolekce funkcí, které umožní, integraci, správy, použití a výkonné zpracování dat. V oblasti Big Data jde o trend řešení se sjednocením distribuovaného úložiště s distribuovaným zpracováním dat (např. řešení od Hadoop). [5]

Ve vrstvě zpracování probíhá vlastní analýza, která usnadňuje dosažení požadovaných výsledků a hodnoty Big Data. Tato vrstva provádí nejen analýzy, ale také vlastní dotazy na data, které probíhají nad daty za běhu a jsou součástí vlastního zpracování. [5]

Spotřebitel dat (Data consumer)

Pod pojmem spotřebitel dat si lze představit koncového uživatele nebo také systém, kterému jsou data předávána. Pro svoji činnost využívá rozhraní nebo služby nabízené poskytovatelem Big Data aplikací. Tyto prostředky umožňují spotřebiteli získat přístup k požadovaným informacím (získání dat, hlášení o datech, atd.). Mezi činnosti spotřebitele patří: hledání a načítání, stahování, analýza, reporting, vizualizace, práce s daty, která jsou určena pro jeho vlastní procesy. [5]

2.3.3 Big Data Algorithms (algoritmy Big Data)

Tato část se zabývá technikami, které slouží pro odvození poznatků z dat a získání „hodnoty“, která je obsažena v datové sadě. Algoritmy mohou provádět výpočty, zpracování dat a úkoly automatizovaného uvažování. Použitím algoritmů lze získat velké objemy dat a cenné znalosti. (Např. algoritmus najde maximální hodnotu v sadě dat). Algoritmy jsou využity pro manipulaci s daty, pro dotazy nad daty, k vlastní analýze (vizualizace, agregace, využití statistických metodik pro snadnější interpretaci dat) a jejich následnou vizualizaci nebo reporting. Algoritmy jsou důležitou součástí komplexní oblasti Big Data, protože téměř veškeré procesy zpracování dat a jejich následná vizualizace nebo reporting je vytvářena na míru pro dané konkrétní řešení Big Data problematiky. [5]

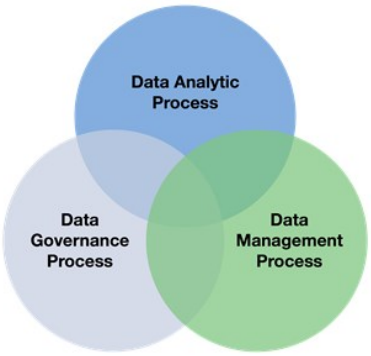
2.3.4 Big Data Processes (procesy Big Data)

Aby se při realizaci projektů zabránilo možným úskalím, které Big Data přináší, mohou být využity procesy, které mohou pomoci organizacím se zaměřením na správný způsob využití Big Data. Procesy v procesně řízené organizaci přinášejí strukturu, měřitelné kroky a lze je efektivně spravovat a řídit každý den. Procesy navíc začleňují odborné znalosti z oblasti Big Data do organizace pomocí postupů, až se z nich stávají standardy organizace. Analýza se stává méně závislou na jednotlivcích, a proto se výrazně zvyšuje šance na dlouhodobé získání hodnoty z Big Data, která vede k dosažení strategických cílů organizace. [5]

Nastavení procesů Big Data v organizaci může být zpočátku časově náročným úkolem, ale rozhodně přináší organizaci výhody v dlouhodobém horizontu. I když jsou procesy problematiky Big Data mezi sebou propojeny a jejich využití je prospěšné v jakékoli organizaci. Je výhodné je rozdělit na dílčí části, do skupin podle jejich zaměření.

Big Data procesy lze rozdělit do tří hlavních dílčích procesů: [5]

- Proces analýzy dat [Data analysiss process] (kontrola)
- Proces správy dat [Data governance process] (dodržování předpisů)
- Proces správy dat [Data management process] (kvalita dat)



Obrázek 18: Data Reference proces.

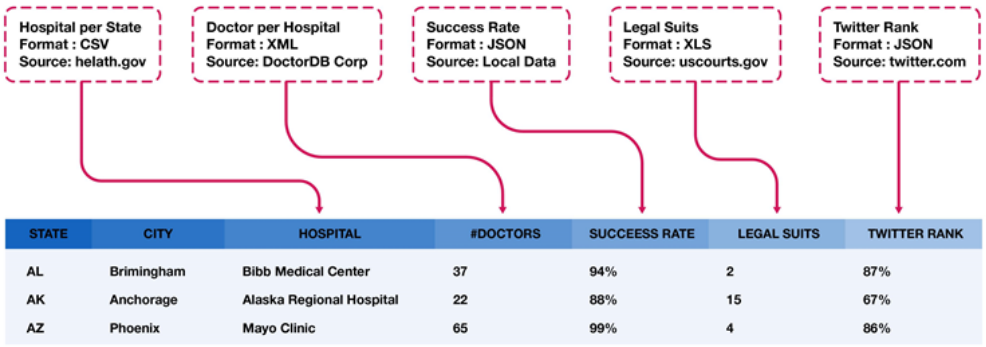
Zdroj : [5]

Proces analýzy dat [Data analysiss process] (kontrola):

Procesní analýzu je vhodné rozložit do několika návazných kroků:

- **Obecný cíl (obchodní cíl):** Tato analýza obvykle proběhne v předstihu, neboť definuje problematiku analýzy (rozsah, obsah, předpokládané náklady atd.), Big Data, která se bude řešit.

- **Identifikace dat:** Jde o specifikaci dat a datových zdrojů pro danou analýzu, zda jsou zpracované (existují v IT systémech organizace) nebo se budou muset zpracovat. Mohou se zakreslit do identifikačního grafu např. obrázek č. 14.



Obrázek 19: Datový identifikační graf.

Zdroj : [5]

- **Sběr a získávání dat:** Již víme, jaká data potřebujeme a kde jsou, jen musíme zajistit jejich sběr pro danou analýzu.

- **Kontrola dat:** Proběhl sběr dat a byly vytvořeny datové struktury pro danou analýzu. Následně určíme proměnné a distribuci dat se zjištěním chybějících hodnot a případné poškození dat, zda existuje vice konfliktních datových sad nebo proměnných s odlišnými údaji v různých systémech a zda jde z takto připravených dat realizovat obecný cíl (první krok).

- **Očištění dat:** Jde o data, která jsou nesprávná, neúplná, nesprávně naformátovaná nebo duplikovaná. Očištění provedeme pomocí interaktivních nástrojů pro čištění dat, pomocí skriptování nebo pomocí dávkového zpracování. Výsledkem je datová struktura, která je konzistentní s podobnými datovými sadami v systému a je připravena na použití.

- **Vytvoření modelu:** Dalším krokem je vytvoření statického modelu, pro který lze použít očištěnou datovou sadu. Pro potřeby modelu mohou být využity matematické vzorce nebo algoritmy a různé statistické metodiky (např. korelace mezi proměnnými). Vztahy mezi proměnnými obecného modelu mohou vytvářet jiné proměnné s chybou, v závislosti na přesnosti daného modelu (např. Data = Model + Chyba).

- **Zpracování dat:** V této části procesu je prováděna skutečná analýza, která obvykle spustí jeden nebo vice (statistických) algoritmů. Tato část procesu může probíhat v iteracích, pokud je analýza průzkumná a není znám vzor (korelace, odchylka), nebo (tato část procesu) může být velmi jednoduchá (jako je dotaz na průměr, medián,…). Vše je závislé na požadavcích dle specifikace konkrétního zadání.

- **Oznámení výsledku:** Proces analýzy Big Data je ukončen oznámením výsledku. Zdá se to jednoduché, ale bohužel není. Protože Big Data a některé algoritmy zpracování jdou někdy obtížněji vysvětlit (z důvodu vnitřní složitosti), proto je dobrá komunikace nezbytná pro úspěch i hodnotu Big Data. Ideální je použít vizualizační techniky (grafy atd.) popřípadě vhodného čísla, které dokáže reprezentovat provedenou analýzu.

Proces správy dat [Data governance process] (dodržování předpisů)

Procesy správy dat zahrnují zajištění kontroly dat během jejich životního cyklu. V tomto pohledu jde také o ochranu osobních dat a důvěrnost dat. Proces správy dat musí nastavit zásady a přiřadit odpovědnosti v rámci celé organizace, je navíc třeba zajistit, aby organizace dodržovaly zákony a předpisy o datech, které jsou platné v místě, kde vykonávají své činnosti. Mezi procesem správy dat a správou dat existuje úzký vztah. Pokud se proces správy dat týká stanovení zásad a odpovědností na strategické úrovni, potom proces řízení provádí a sleduje tyto zásady na provozní úrovni. Mezi těmito dvěma procesy existuje synergie.

Proces správy dat [Data management process] (kvalita dat)

Big Data využívají stejný data management kvality dat a jsou popsány v části "1.3 Kvalita dat" popřípadě v části "1.4 Data Management".

2.3.5 Big Data Functions (funkce Big Data)

Problematika Big Data je velmi komplexní a proto je nutné vytvořit v organizaci tým, který se bude této problematice věnovat, bude vytvářet a spravovat veškerou důležitou dokumentaci. Zajišťovat předávání znalostí ostatním uživatelům, kteří využívají technologie Big Data. Věnovat se rozvoji této problematiky dle požadavků a potřeb organizace se zaměřením na plnění strategických cílů organizace. [5]

2.3.6 Artificial Intelligence (umělá inteligence)

Umělá inteligence je stále častěji využívána pro problematiku Big Data. Umělou inteligenci členíme na: [5]

- Zpracování přirozeného jazyka (Natural language processing)
- Reprezentace znalostí (Knowledge representation)
- Automatické uvažování (Automated reasoning)
- Strojové učení (Machine learning)

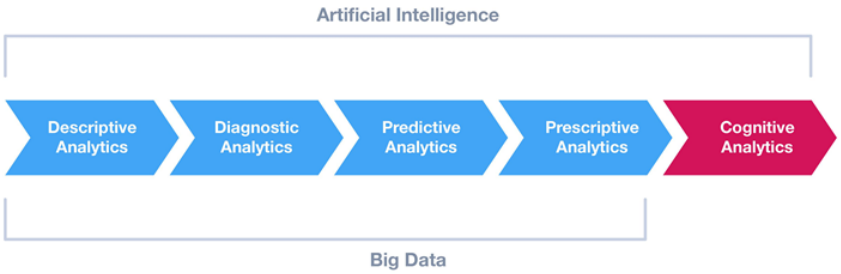
Zpracování přirozeného jazyka a problematiku strojového učení upřesním v části „2.4 Metody zpracování nestrukturovaných dat". Automatizované uvažování je schopnost vytvořit IT systém s takovou úrovní uvažování a rozhodovací logiky, že již nepotřebuje lidskou obsluhu.

Reprezentace znalostí (Knowledge representation)

Jde o část umělé intelligence, která se specializuje na reprezentaci informací, kdy využívá logiku za účelem modelování uvažování s tvorbou logických výroků s využitím ontologie. Tato

technika nám pomáhá např. určit, kde jsou data uložena pro budoucí načtení, jak je načíst a kdy vyžadují zpracování. [5]

Hlavní rozdíl mezi umělou inteligencí a Big Data je v analýze dat, kdy Big Data využívají Deskriptivní (popisnou) analytiku, Diagnostickou analytiku, Prediktivní analytiku, Preskriptivní analytiku, a Umělá intelligence je navíc schopna zahrnou do svých analytických procesů kognitivní vědy. [5]



Obrázek 20: Analýzy Big Data a Umělé inteligence.

Zdroj : [5]

Deskriptivní analytika (Descriptive analytics): se zabývá minulostí, a tedy těžbou historických dat, aby hledala důvody minulého úspěchu nebo neúspěchu. Tuto techniku využívá většina současného reportingu. [5]

Diagnostická analytika (Diagnostic analytics): využívá minulá data a je schopna provádět porovnání a diagnostikovat příčiny odchylek porovnávaných dat.

Prediktivní analytika (Predictive analytics): využívá historická data kombinována s pravidly, algoritmy, aby určila pravděpodobný budoucí výsledek události nebo pravděpodobnost, že nastane nějaká situace. [5]

Preskriptivní analytika (Prescriptive analytics): nejen předvídá, co se stane a kdy se to stane, ale také proč se to stane. Navrhuje možnosti rozhodování o tom, jak využít budoucí příležitosti nebo jak zmírnit budoucí riziko, a ukazuje důsledky každé možnosti rozhodnutí. Protože může neustále přijímat nová data, může u ní dojít k automatickému zlepšení přesnosti predikce. Následně je téměř samozřejmé že, dokáže předepsat lepší možnosti rozhodování. [5]

Kognitivní analytika (Cognitive analytics): aplikuje inteligenci, jako je porozumění nejen slovům v textu, ale také úplnému kontextu toho, co se píše nebo mluví, nebo rozpoznávání objektů v obraze ve velkém množství informací. Pro dosažení tohoto cíle využívá více inteligentních technologií, včetně sémantiky, algoritmů umělé inteligence a učebních technik,

jako je hluboké učení a strojové učení. Použitím těchto technik může být kognitivní aplikace v průběhu času chytřejší a efektivnější v učení než lidé. [5]

2.4 Metody zpracování nestrukturovaných dat

Pro velmi jednoduchou analýzu nestrukturovaných dat ve formě textů lze využít jednodušší metody jako např. Mrak slov, Strom slov atd., které využívají běžné matematické a statistické metodiky. Pro standardní analýzu dle dnešních požadavků je nutné využít sofistikovanější analýzy, založené na principech umělé inteligence, jako je např. pro text analýza sentimentu, nebo analýza multimédií atd.

Umělá inteligence je komplexním vědním oborem, který se zabývá vývojem algoritmů a je součástí nejen software, ale i hardware (dnes je využita nejen v průmyslových zařízeních, ale např. v smartphonech atd.), které vykazující znaky inteligentního chování. Umělá inteligence využívá poznatky z mnoha dalších vědních oborů, a to především z informatiky, matematiky, statistiky, logiky, lingvistiky nebo z neurověd. Běžně se lze setkat také se zkratkovým označením UI nebo AI (Artificial Intelligence). [17]

2.4.1 Umělé neuronové sítě

Představuje jeden z modelů používaných v umělé inteligenci, kdy základem umělé neuronové sítě je jednoduchý procesor neboli percepton (neuron), který představuje matematický model biologického neuronu. Percepton (neuron) má neomezené množství vstupů a jen jeden výstup. Každý percepton (neuron) má svou prahovou hodnotu (potenciál neuronu) a každý vstup do perceptonu (neuronu) má svoji váhu, která je určena synapsí (spojení mezi jednotlivými neurony).

Základní typy neuronových sítí: [17]

- **Vícevrstvé neuronové sítě (MLP, Multi Layer Perception):** Použití: predikce (využívá na vstupu časové řady, vývoje trendu) klasifikace, aproximace.
- **Hopfieldovy sítě:** Použití: asociativní paměť, klasifikátor (rozpoznávání obrazu OCR, robotické vidění, filtrace signálu atd.), řešení optimalizačních problémů (komprimace, problém obchodního cestujícího atd.).
- **Samoorganizující se sítě (SOM, Self Organizing Map):** Použití: shlukování klasifikace.
- **Radiální báze (RBF sítě):** Použití: klasifikační úlohy obdobné jako u MLP, regresní úlohy (odhady na základě historických dat).
- a další.

Základní algoritmy neuronových sítí: [17]

- **učení s učitelem** principiálně jde o srovnání aktuálního výstupu s požadovaným výstupem, kdy se algoritmus snaží danou odchylku minimalizovat. Tuto minimalizaci se snaží zrealizovat pomocí přenastavení váhy a prahu neuronové sítě,
- **učení bez učitele** v tomto algoritmu není znám výstup, protože síť se učí pomocí třídění vstupu tím, že sadu vzorů, které síť obdrží a roztřídí do skupin.

2.4.2 Strojové učení

Umělá inteligence využívá pro zpracovávání nestrukturovaných dat jednu ze svých oblastí, která se označuje pod pojmem strojové učení (včetně umělých neuronových sítí). Strojové učení je jedním z moderních přístupů k technikám umělé inteligence a zabývá se tvorbou tzv. učících se algoritmů a technik, kdy pomocí jejich principů umožní počítačovému systému reagovat na změny. Zjednodušeně, učí zařízení vybavená nějakým softwarem (dnes již nejde pouze o výpočetní techniku) se učit. Tyto techniky jsou využívány v umělých neuronových sítích, kde pomocí strojového učení dokážeme rozpoznávat objekty v digitálních médiích. [29]

Základní algoritmy strojového učení [30]

- **Učení s učitelem (Supervised Learning):** Pro množinu vstupních údajů je definován správný výstup. V případě regresních algoritmů to bývá zpravidla hodnota či interval hodnot, při klasifikačních algoritmech třída.

- **Učení bez učitele (Unsupervised Learning):** Ke vstupním datům není znám výstup, dokonce se předem neví, zda úloha má nějaké řešení, jestli je objekt znám nebo patří do nějaké skupiny předem známých shluků, případně zda je mezi proměnnými charakteristický případ nebo jsou mezi nimi nějaké závislosti. Tato metoda se aplikuje například v různých automatizovaných zařízeních, kde je požadavek rychlé reakce na nepředvídatelnou situaci. Také se často používá i pro screening známých procesů, které neposkytují očekávané výsledky, a proto je třeba najít jiný princip procesu.

- Velmi často se používá kombinace obou metod, takzvaný **semisupervised learning**. Část vstupních dat je k dispozici i se známým výstupem, ale další data takový výstup známý nemají. Kdy analytické algoritmy se „trénují“ na cvičné množině dat, kde jsou známé výstupy.

Metodiky strojového učení: [30]

- Klasifikace
- Regrese (analýzy za účelem předpovědi dalšího chování dat)
- Clusterování (Shlukování)
- Asociace (identifikace pravidel v datech – vztahy mezi množinami dat)

Postup při strojovém učení: [30]

- **Sběr dat:** vytvoří se datový soubor i z několika zdrojů dat s různými formáty.
- **Příprava dat:** je provedena kontrola a očištění dat tak, aby do procesu zpracování byla přijata pouze data s požadovanou úrovní kvality.
- **Trénování modelu:** Důležitá je volba správného algoritmu zpracování a reprezentace dat. Data jsou rozdělena do dvou částí, jedna je trénovací, druhá je testovací.
- **Ohodnocení modelu:** Otestovaný model na trénovací množině ověříme na testovací množině a zjistíme jeho výkon a přesnost.
- **Aplikace / přetrénování modelu:** Model lze aplikovat pouze v případě, že model vyhovuje nárokům na výkon a přesnost. Pokud model požadavkům nevyhovuje, je nutné se vrátit k "Trénování modelu" a využít jiné nebo optimalizované algoritmy a celou činnost zopakovat.

Strojové učení je možné využít pro označování objektů v digitálních obrázcích nebo pro rozeznání optických znaků (OCR) při jejich zpracování. Dalším využitím je textová analytika v digitálních dokumentech jako je analýza sentimentu, extrakce informací, filtrace spamu.

2.4.3 Analýza textu

Analýza textu neboli termín Text mining. Pro zpracování textu existuje značné množství různých metod, přičemž o některých, se pro potřeby této práce, zmíním jen náznakově, protože při zpracování textů je využíváno několik disciplín, které se využívají při zpracování nestrukturovaných dat jako je např. zpracování řeči, tokenizace, normalizace, extrakce vztahů, kategorizace dokumentů, klasifikace a shrnutí textu, stematizace, morfém, lematizace, koreference slov atd. [48]

Použité metodiky pro zpracování textů přirozeného jazyka:

- **Strojový překlad** – překlad z jednoho přirozeného jazyka do druhého.
- **Sumarizace** – vytváření krátkých textových shrnutí za odstavec, článek nebo celý dokument.
- **Rozpoznávání pojmenovaných entit** – identifikace předefinovaných entit v analyzovaném textu s následnou klasifikací do předdefinovaných kategorií.
- **Analýza sentimentu** – jde o metodu založenou na strojovém učení spadající do NLP (Natural Language Processing), kterou je možné aplikovat jak na volný text, tak na snímek (výraz tváře). Jde o velmi populární metodiky, kdy na základě vytvořeného slovníku hledáme, v textu zda obsahuje sentiment (subjektivní text např. slova ze slovníku) a určujeme jeho citové zabarvení (pozitivní, negativní, neutrální).
- **Označování části řeči (POS** – Part of Speech Tagging) – jde o označení slov v textu
- **Vyhledávání** – jde o vyhledávání konkrétních prvků v textu.
- **Generování přirozeného jazyka** – schopnost interpretovat data pomocí přirozeného jazyka.
- **Zodpovídání dotazů –** na základě strojového učení a umělé intelligence jsou vytvořeny systémy, které podle interní logiky a struktury uživatelských dotazů vytváří odpovědi bez zásahu člověka.
- **Jednoduché zpracování textu:** Mrak slov (Word Cloud), Strom slov (Word Tree).

Dalšími metodami zpracování textů je zpracování logových záznamů v reálném čase. Jde o zpracování tzv. datových proudů (stream data) téměř v reálném čase. Způsob zpracování lze samozřejmě upravit podle požadavků organizace na danou oblast, ale je také možno využít existující technologie jako: [34]

- Apache Kafka
- Apache NiFi
- Logstash
- Fluentd
- Apache Flume

2.4.4 Analýza zvuku

Zvuk je mechanické vlnění v látkovém prostředí, které je schopno vyvolat sluchový vjem. Pro naše potřeby využijeme metodiku: **Přepis (Speech To Text – STT)** – tento nástroj je

založen na automatizovaném rozpoznání řeči a následném strojovém převodu řeči na text. Dalším využitím mohou být zařízení, které zajišťují „**Akce zařízení zajišťující ASR, na podnět**„ – jde o ASU (Automatic Speech Understanding), kdy ASR zařízení porozumí např. zvukovému dotazu a vyhledá například odpověď nebo provede jinou činnost. [43]

2.4.5 Analýza digitálního snímku

Principem této analýzy je detekce objektů v daném snímku. Detekované objekty je možno dále snímat, pokud detekujeme osobu, je možné u ní rozpoznat pohlaví nebo ji ztotožnit s údaji v informačním systému organizace.

Analýzy snímků provádíme za účelem:

- **Identifikace objektu na snímku**
- **Zjištění vlastnosti snímku (např. detekce barev na snímku)**
- **Analýza tváře**

Základní kroky analýzy: Segmentace (rozdělení snímku do oblastí), Prahování (zkoumání jasových hodnot snímku), Zpracování binárních obrazů (Matematickou morfologií, Ztenčováním), Metody rozpoznání objektu (Příznaková, Syntaktická), Neuronové sítě (Konvoluční, Kompetitivní, Kohonenovy mapy), Fourierovy transformace průběhu křivosti, Jasové atributy, Transformace, Filtrace, Úpravy barev, Extrakce vlastností, Zhodnocení. [43]

Specializovanou analýzou digitálního snímku je analýza tváře. Základní kroky analýzy: Detekce tváře (Znalostní metody, Invariantní metody, Metody založené na porovnání šablon, Metody založené na strojovém učení), Ohraničení tváře (Umělé neuronové sítě, Vyhledávání SVR (Support Vector Recognition), Metoda ASM (Aktivní tvarový model), AAM (Aktivní vzhledový model), Extrakce rysů (Metoda založená na geometrii tváře), Skupina metod - založená na šablonách - založená na segmentaci barev – založená na vzhledu), Porovnání rysů tváře. [43]

Detekce emocí z výrazu lidské tváře – Základní kroky analýzy: Detekce tváře, 3D modelování tváře s využitím AAM (Active Appereance Model), Klasifikace výrazu tváře. [43]

2.4.6 Analýza videa

Audiovizuální záznam je složen z audio stopy a video stopy. Video stopa je složena ze sekvence jednotlivých snímků a na tyto snímky lze použít analýzu digitálních snímků. Audio stopu je možno zpracovat jako analýzu zvuku. [43]

2.5 Technologie Big Data

V této části aplikace se zaměříme na softwarová řešení vhodná pro problematiku Big Data. Technologie Big Data je možné rozčlenit podle typu (zda generují data) na operativní a analytické.

- **Operativní technologie Big Data** – jde o technologie, které data vytvářejí a to při každodenních pracovních nebo volnočasových činnostech lidí nebo jsou generovány automatizovanými nebo poloautomatizovanými procesy v aplikacích, informačních systémech nebo je generují stroje, zařízení, senzory IoT atd.
- **Analytické technologie Big Data** – jsou určeny ke sběru, zpracování uchování, analýze a vizualizaci pořízených Big Data operativními technologiemi.

Rozčlenění Technologie pro Big Data podle návaznosti na: [3]

- Uskladnění dat (Data Storage)
- Dolování dat (Data Mining)
- Analýza dat (Data Analytics)
- Data Visualisation (Vizualizace dat)

2.5.1 Technologie pro uskladnění Big Data

Pod pojmem technologie pro uskladnění Big Data je skryta komplexní infastruktura datového úložiště, která je schopná spravovat množství Big Data v požadované obslužné rychlosti bez ohledu na fyzické umístění aplikace nebo jejího segmentu. V současnosti má datové úložiště pro Big Data i další funkcionality jako je kontrola a očištění dat před vlastním uložením. Technicky jde převážně o síťová úložiště NAS tedy “servery “, které jsou využity technologiemi k ukládání dat a SAN které slouží jako síťové propojení mezi NAS. Společně mohou tvořit uložiště typu cloudu nebo datového jezera. [3]

Správu dat v celém úložišti řídí systémové soubory HDFS, které v části NameNode mají uložena všechna metadata souborového systému a HDFS je dále ukládá podle datových bloků do DataNode a tam je využívá pro standardizované procesy jako je uložení dat nebo zaslání dat dle požadavku na úložiště. Toto řešení umožní, pokud je klient nastaven na stejném bloku, jako

datový blok může si přečíst daný blok lokálně přes DataNode. Toto řešení velmi zefektivňuje chod datového úložiště. [3]

Hadoop

Hadoop poskytuje komplexní řešení pro Big Data a je v současnosti nejrozšířenějším prostředím pro Big Data. Je navržen pro distribuované data zpracovávané v prostředí s komoditním hardware a jednoduchým programovým modelem. Může ukládat a analyzovat data (velké datové soubory) z různých fyzicky oddělených částí úložiště se zachováním vysoké rychlosti všech činností, a to vše za velmi optimální náklady na provoz. Mezi komponenty Hadoop patří HDFS, MapReduce, YARN. Hadoop je velmi spolehlivé, výkonné a cenově efektivní řešení pro problematiku dnešních Big Data. [3]

Mongo DB

MongoDb je NoSQL dokumentová databáze, která je rychlá, masivně škálovatelná a snadno se používá. MongoDb využívá pro rychlejší a efektivnější správu kolekce, které se odkazují na uložené soubory. Při vyhledávání jsou prohledány kolekce, které mají unifikovanou strukturu a následně jsou provedeny činnosti s dokumenty. Databáze je velmi rychlá, protože vyhledávání v kolekcích, které mají stejnou strukturu, je velmi efektivní. Databáze může archivovat v uložišti různé typy souborů přes svou distribuovanou architekturu a může data uložit i v několika cloudech. [3]

NoSQL

Databáze NoSQL jsou to nerelační nebo distribuované databáze. Patří sem databáze typu klíč-hodnota, dokumentové, sloupcové, grafové. Nemají předdefinované schéma a datovou strukturu, ale mají vyšší nároky na výkon hardware (výkon procesoru) a data management. [3]

Výhody NoSQL:

- Flexibilní horizontální škálovatelnost NoSQL umožňuje přidat do clusterů více serverů a zvýšit tak výpočetní výkon clusterů a celého úložiště
- Flexibilní datový model
- Rychlost přístupu na úložiště (rychlejší čtení a ukládání dat)

Problematiky NoSQL:

- Náročnější administrace
- Prozatím nedostatečná uživatelská podpora (lepší se každým rokem)

- Neexistuje univerzální standardizované prostředí pro přístup k datům (např. každé řešení má svůj vlastní dotazovací jazyk atd.)
- Jde o velmi mladou technologii, která prozatím nemá pověstnou SQL robustnost

2.5.2 Technologie pro Data Minig Big Data

Data Mininig zahrnuje oblast dolování Big Data, která je využívá algoritmy pro vyhledávání, kontrolu a čištění dat s následnou extrakcí a porovnáním dat. Data Minig je založen na sofistikovaných statistických metodách. [3]

Presto

Jde o otevřený zdrojový distribuovaný SQLengine, pro vytváření interaktivních analytických dotazů dat z různých zdrojů. Dokáže pracovat s Gigabyte až Petabyte objemy dat. Přesto umožňuje vytvářet dotazy v Hive, Cassandra, Relation Database a Proprietary Data Stores. [3]

Elasticsearch

Jde o vyhledávací engine založený na knihovně Lucene. Poskytuje plně kompatibilní fultextový HTML vyhledavač a možnost využívat dokumenty JSON bez schématu. [3]

Rapid Miner

Jde o centralizované řešení, které obsahuje velmi robustní grafické rozhraní. Je založen na prediktivní analýze. A umožňuje vytvářet velmi pokročilé skripty v několika programovacích jazycích. [3]

2.5.3 Technologie pro analýzu Big Data

Kafka

Apache Kafka je platformou distribuovaného streamování, která má tři klíčové schopnosti, a to vydavatele, odběratele, spotřebitele, jež jsou obdobou dotazovacím zprávám nebo systému zpráv v organizaci. [3]

KNIME

Umožnuje vytvářet vizuální datové toky pro uživatele s možnou selekcí jednotlivých kroků zpracování a analýzy dat s kontrolou a tvorbou výsledků, modelů a interaktivních pohledů. KNIME je napsán v Javě a je založen na platformě Eclipse s možností přidání dalších pluginů pro rozšíření jeho funkcionalit. [3]

Splunk

Zachycuje, indexuje a provádí korelace v reálném čase nad vyhledávacím úložištěm. Odkud může generovat grafy, reporty, řídící panely a vizualizovat potřebná data. Využívá správu aplikací se zabezpečením a dodržením předpisů. A využívá take obchodní a Web analýzu. [3]

2.5.4 Technologie pro vizualizaci Big Data

Pod pojmem vizualizace si představujeme grafickou interpretaci dat a informací, ve formátu různých grafů a map. Vizualizace může usnadnit porozumění datům. Pro vizualizaci je výhodné využít některý z programovacích nástrojů, který dokáže pracovat s Big Data.

Python

Jde o velmi univerzální programovací jazyk, který umožňuje vytvářet velmi propracovanou vizualizaci. Je standardním programovacím jazykem, který umožňuje tvorbu úplného prostředí případné aplikace, která bude provozována pro potřeby Big Data. Výhodou Python je jeho širší využití, jak na platformě Windows, tak na platformě Linux. Jeho nevýhodou je nižší rychlost jeho aplikací.

Jazyk R

Jazyk R je skriptovacím jazykem, který umožňuje vcelku elegantně vytvářet různé statistické analýzy, a není problém s jeho pomocí vytvořit velmi přehlednou vizualizaci výsledků dané analýzy. R jazyk má velké úložiště balíčků a tak uživatelé mají k dispozici nástroje pro téměř jakoukoliv oblast Big Data. Aplikace vytvořené v jazyku R je možné integrovat do aplikací Apache Handoop, Apache Spark. Problém jazyka R je, že je skriptovacím jazykem a nejde zkompilovat do needitovatelného formátu (např. exe), a proto není vhodný k použití v prostředí organizace jako součást obecné aplikace. Pouze snad jen v případě, že by uživatel měl oprávnění spouštět R skripty např. s náhledem, ale nemohl je editovat.

2.6 Dílčí souhrn

Problematika Big Data je velmi komplexní a nebylo možné obsáhnout celou problematiku. Proto jsem se věnoval jen těm oblastem, které mají souvislost s mou diplomovou prací. Využití Big Data a porovnání rozdílů mezi strukturovanými a nestrukturovanými Big Data popíši v části této práce “3. STRUKTUROVANÁ A NESTRUKTUROVANÁ BIG DATA“.

3. STRUKTUROVANÁ A NESTRUKTUROVANÁ BIG DATA

Charakteristiku strukturovaných, semistrukturovaných a nestrukturovaných Big Data jsem popsal v části této práce „1.2.1 Typy dat podle datové struktury“. Proto se v této části zaměřím z počátku na rozdíly mezi těmito datovými charakteristikami s následným popisem jejich využití v organizaci.

Tabulka 5: Strukturovaná versus nestrukturovaná data.

Vlastnosti	Strukturovaná data	Semistrukturovaná data	Nestrukturovaná data
Technologie	Je založena na relačních databázových tabulkách.	Je založena na XML / RDF (Resource Description Framework).	Je založena na znakových a binárních datech.
Úložiště	Relační databázové systémy (RDBMS). Datový sklad	Souborový systém (perzistentní úložiště) Relační database Nativní XML databáze	Aplikace, NoSQL databáze Datový sklad Data Lake
Získávání dat a manipulace s daty	Je velmi efektivní, protože jde data snadno indexovat. Využití ACID a podpory transakcí.	Je závislé na zdroji a způsobu vzniku dat. Pro efektivní procesy je nutné využit vhodný zpracovatelský modul. Transformace do vhodných struktur.	Je závislé na implementaci vhodného zpracovatelského modulu. Požaduje vyšší výpočetní výkon a výkonnější hardwarové prostředky.
Správa transakcí	Obsahuje vyzrálé transakce a různé techniky souběžnosti.	Transakce je upravená DBMS.	Žádná správa transakcí a žádná souběžnost.
Správa verzí	Přes n-tice, řádky, tabulky.	Je možné vytvářet verze verzí přes n-tice nebo graf.	Verze jako celek.
Flexibilita	Je závislá na předefinovaném schématu a méně flexibilní.	Je flexibilnější než strukturovaná data, ale méně flexibilní než nestrukturovaná data.	Je pružnější a chybí zde předefinované schéma (schéma je dynamické).
Škálovatelnost	Jde velmi obtížné škálovat viz. předdefinované schéma databáze.	Data jsou více škálovatelná než strukturovaná data.	Data jsou více škálovatelná.
Robustnost	Velmi robustní.	Je nižší než u strukturovaných dat – důvodem je rychlý rozvoj technologií a vytváření standardů	Vzhledem k velmi rychlému vývoji v této oblasti je robustnost prozatím nižší.
Výkon dotazu	Strukturovaný dotaz umožňuje složité spojování.	Jsou možné dotazy na anonymní uzly.	Možné jsou pouze textové dotazy
Migrace dat	Snadnější migrace.	Podle způsobu uchovávání dat.	Problematická migrace specializovanými prostředky.

Zdroj : [Vlastní tvorba]

Vztah mezi strukturovanými a nestrukturovanými daty není založen na žádném jejich vzájemném konfliktu. Uživatel aplikace se nezabývá tím, zda jsou data, která využívá, strukturovaná nebo nestrukturovaná, on pouze pracuje s daty a informacemi, které mu tato aplikace poskytne rozhraním aplikace a formou vizualizovaných výstupů (obrazovky, formuláře, grafy, tabulky atd.). Procesy na pozadí aplikace jsou nastaveny tak, aby se co nejlépe přizpůsobily datovým charakteristikám strukturovaných a nestrukturovaných dat, které ovlivňují způsobem jejich zpracování, uchování, analýz a následnou vizualizací. Příkladem je rozdílnost v náročnosti implementace analýzy dat pro strukturovaná data, kdy tato data využívají vyspělé procesy a technologie jako je Business Intelligence, Data Minig atd. Pro analýzu nestrukturovaných dat je nutné investovat do vytvoření datového modelu a vlastní analýzy pro dané řešení např. agregace nestrukturovaných dat do vypovídajících informací (např. obchodních), které následně uložíme do strukturovaných dat nebo je jen jako strukturovaná data dočasně využijeme pro požadované analýzy, reporting a vizualizace dat. Otázkou je vždy konkrétní přínos využití nestrukturovaných dat v dané aplikaci. Neboli přínos by měl převážit nad náklady případného řešení.

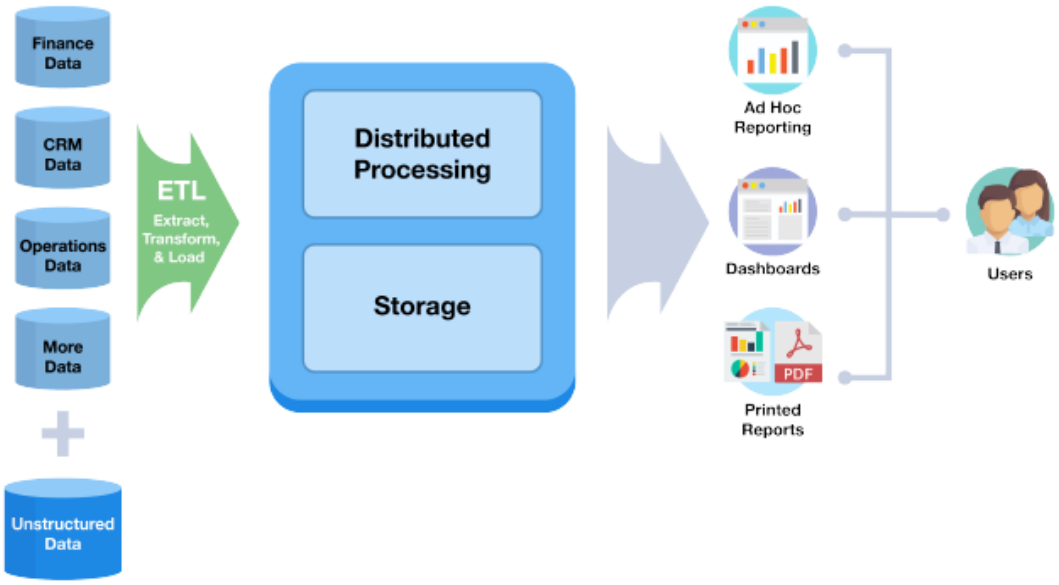
Jako příklad pro zpracování a analýzu jen strukturovaných Big Data s využitím Business Intellingence, která využívá jak analýzu dat, tak analytické techniky, ke konsolidaci a shrnutí informací z různých strukturovaných datových zdrojů standardního IS, je možné využít schéma na obrázku č.21.

Obrázek 21: Standardní zpracování strukturovaných Big Data (Business Intelligence).

Zdroj : [4]

Pro zpracování a analýzu strukturovaných, semistrukturovaných a nestrukturovaných Big Data, je výhodnější využít systémy, které využívají distribuované úložiště, zpracování a analýzu.

Tyto systémy využívají technologii NoSQL databází, distribuovaná souborová řešení např. s HDFS atd. V současnosti je výhodné využít komplexnější distribuované řešení jako je např. Apache Hadoop, které již obsahuje rozhraní podobné SQL pro tvorbu dotazů a analýz. Pro velmi zjednodušený pohled na tuto problematiku je možné využít schéma na obrázku č. 22.



Obrázek 22: Big Data s nestrukturovanými daty vyžadují distribuované úložiště a zpracování.

Zdroj : [4]

Pro smysluplné využití Big Data v organizaci je v prvním kroku nutné, aby si management organizace ujasnit co dělá organizaci úspěšnou a jak mohou být technologie a řešení Big Data prospěšné pro zefektivnění procesů dané organizace. Je vhodné se zaměřit pouze na oblasti činnosti organizace, které jsou přínosné pro plnění jejich cílů s využitím Big Data, která požadovaný přínos dokáží zajistit. Proto je nutná a důležitá analýza, zda zavedení Big Data v daném segmentu činností organizace je opravdu přínosem (benefitem) organizace. Pokud ano, tak je možné zrealizovat implementaci s reálným očekáváním benefitů pro danou organizaci.

3.1 Uplatnění Big Data

Data a Big Data nemají žádný potenciál, pokud s nimi neumíme pracovat, nedokážeme je správně propojit a vytěžit z nich cenné informace, které je možné využít pro dosahování stanovených cílů organizace. Proto je nutné pochopit, že hodnota ukrytá v datech je závislá na jejich efektivním využití, a také to, že nám data poskytují více než jen odpovědi a řešení. Datový způsob myšlení slouží k vytvoření ekonomické nebo společenské hodnoty z analyzovaných dat. Data je nutné chápat jako strategický zdroj.

3.1.1 Přínos uplatnění Big Data

Big Data v současnosti produkuje téměř každá velká i větší organizace (např. firma), která provádí svou činnost např. v oblasti financí, telekomunikací, e-commerce, retailu a nebo ve výrobě.

Přínos uplatnění Big Data pro organizace:

- Podpora rozhodování
- Optimalizace nákladů a procesů organizace
- Analýzy tržních podmínek
- Podpora inovací a vývoje produktů
- Optimalizace akvizicí a retencí klientů/zákazníků
- Podpora marketingu
- Podpora měření veřejné služby

Podpora rozhodování

Zpracování Big Data umožňuje sledovat různé události v okamžitém pohledu, ale i za dlouhou historii. Dále na tyto události je možno vytvářet různé pohledy (např. za nějakou oblast) a získat tak komplexní informace. Data jsou téměř okamžitě analyzována, vizualizována za účelem zprostředkování informací v co nejsrozumitelnější, přehledné a interaktivní formě. Pokud jsou provedeny veškeré procesy analýzy a vizualizace správným způsobem, mohou být tyto informace konkurenční výhodou dané organizace.

Optimalizace nákladů a procesů organizace

Analýzy Big Data, které mohou být využity pro optimalizaci nákladů procesů organizace, jsou prováděny osvědčenými tradičními postupy (matematicko-statistické modely) a také moderními technikami strojového učení (machine learning), hlubokého učení (deep learning) a textové analytiky (NLP).

Analýzy tržních podmínek

Analýzou nákupního chování zákazníků je možné vyspecifikovat na daném trhu produkty, které jdou nejvíce na odbyt např. v daném časovém období a tak např. předpokládanému sezónnímu zvýšenému odbytu přizpůsobit výrobní procesy v dostatečném předstihu a objemu daného produktu. Tyto informace mohou být konkurenční výhodou organizace.

Podpora inovací a vývoje produktů

Využití velkého množství senzorických Big Data z různých analyzátorů a přístrojů, které lze analyzovat. Jde zde o aktuální data nebo historická data, které lze využít při tvorbě souvislostí nebo lze tyto data využít při simulacích. Výsledky analýz jsou využity pro inovační procesy produktů nebo pro vývoj nových produktů.

Optimalizace akvizicí a retencí klientů/zákazníků

Zde jsou využita senzorická Big Data z kamenných prodejen (sledování nákupu produktů, pohybu zákazníka po prodejně atd.), e-commerce v Eshopu (např. čas, rychlost, způsob nakupování atd.), kdy z takto získaných dat lze vytvářet mnoho zajímavých vzorců chování zákazníka. Lze sledovat, zda zákaznický segment, o který organizace usiluje v daném potrfoliu produktů, narůstá a zda dochází k setrvání současných klientů (retenci).

Podpora marketingu

Big Data získaná na základě průzkumu trhu a sledování nákupů a e-commerce nám umožní vytvořit vzorce chování zákazníka a vytvoření predikce z těchto dat pro doporučenou další koupi, která je založena na segmentaci portfolia produktů. Na základě takto vytvořených informací lze zrealizovat marketingové aktivity, jako jsou např. marketingové kampaně cílené na vhodný segment zákazníků (cílení reklamy).

Podpora měření veřejné služby

Zde jsou Big Data získávána od konzumentů veřejné služby jako zpětná vazba. Z takto získaných dat jsou vytvářeny analýzy pro efektivní management veřejné služby (např. pro efektivní veřejné služby v ČT).

3.1.2 Příklady uplatnění Big Data

Pro příklady uplatnění big dat jsem vybral oblasti financí, telekomunikací, maloobchodu (retailu), průmyslu (výrobní data), logistiky a dopravy, energetiky, zemědělství a zdravotnictví.

Finance (Bankovnictví, pojišťovnictví)

Data v této oblasti vznikají v organizacích obvykle v ERP systémech pomocí transakcí, jde o standardní procesy jako je fakturace atd. Specifickou oblastí je bankovnictví, kde transakční data vznikají např. pomocí převodů financí z účtu na účet nebo plateb, popřípadě při výběrech z bankomatů nebo při online platbách v obchodech a na internetu. Analýzou proběhlých transakcí mohou banky segmentovat preference svých klientů. Takto získaná data banka

obvykle využije k zefektivnění svých produktů a služeb tak, aby co nejlépe uspokojovaly požadavky svých klientů.

Pojišťovny se snaží sbírat veškerá dostupná data o chování svých klientů, aby manažeři ve svých pojistných plánech co nejlépe vyspecifikovali případnou rizikovost svých klientů a co nejlépe dokázali detekovat případné pojistné podvody. Dalším využitím Big Data v této oblasti je specifikace nových trendů s vytvořením produktů, které jsou optimalizovány na potřeby a požadavky každého klienta.

Telekomunikace

Big Data v telekomunikacích jsou tvořena převážně ze signálních dat, která vznikají při standardní komunikaci mezi zákaznickým zařízením (např. smartphone atd.) a sítí mobilního operátora. Jde o informace, které běží na pozadí zapnutého zákaznického zařízení, a to i v případě, že neprobíhá žádná aktivní činnost (jako hovor, SMS, využití nějaké mobilní aplikace s požadavkem na datovou výměnu). Tyto informace jsou obvykle využívány pro zefektivnění poskytovaných služeb (např. posílení velmi využívaných segmentů sítě operátora atd.). Dalším využitím je získání informací o pravidelném pohybu zákazníka (identifikace polohy v síti operátora), jeho obchodní návyky, rozsah využívání a preference aplikací, mobilních služeb atd. Pokud operátor tato data dostatečně anonymizuje, případně provede segmentaci, mohou být takto upravená data velmi zajímavou komoditou, kterou může poskytovat třetím stranám.

Maloobchod (Retail)

Sběr Big Data v oblasti maloobchodu je velmi důležitý a probíhá v několika oblastech. Transakční oblast je zaměřena na získávání informací z anonymizovaných nákupů zákazníků, kdy tato data slouží k profilování zákaznického chování s tvorbou vzorců zákaznického chování pro různé marketingové predikce s efektivním cílením reklamy. Současný technický pokrok v oblasti IoT umožňuje využít senzoriku např. pro hlídání zaplněnosti regálů, kontrolu pádu zboží z regálu (jeho případné poškození) nebo smart metering pro nepřetržitý monitoring např. teploty v chladících i mrazících zařízeních dané maloobchodní provozovny. Data o sledování teploty jsou důležitá v tzv. teplotním řetězci, který monitoruje dané zboží od výrobce, až ke konečnému zákazníku, a je možné je využít jako podklad pro reklamační řízení. Další možností IoT je monitoring pohybu zákazníků po maloobchodní provozovně, jde o nejčastější nebo nejdelší zastavení zákazníků v dané oblasti provozovny. Tyto informace slouží opět k vytváření vzorců zákaznického chování. Další oblastí Big Data je sledování

logistických procesů při zásobování maloobchodních prodejen. Tyto informace slouží pro optimalizaci služeb dodavatelských řetězců. Výsledkem je zefektivnění služeb pro koncové zákazníky v maloobchodě.

Průmysl

Využití Big Data v průmyslu je velmi široké a zahrnuje data a informace vznikající v ERP a MES systémech pomocí transakcí, senzorická data ze zařízení monitorující výrobu, firemní logistické procesy atd. Tato data jsou využívána pro optimalizaci a zefektivnění všech důležitých procesů organizace (snižování spotřeby materiálů, energií atd.). Informace ze zařízení jsou využity pro efektivní plánování servisních procesů organizace s minimalizací nečekaných výrobních odstávek a havárií.

Logistika a doprava

Big Data jsou zde využita pro efektivní management veškerých procesů logistického řetězce. Při monitoringu logistického řetězce se zaměřujeme na efektivní přepravu zboží s využitím přepravních možností dané organizace. Jde zde o on-line monitoring (kde se nachází daný přepravní prostředek a zboží) a následné off-line analýzy již proběhlých logistických procesů, se zaměřeném na tvorbu nových efektivnějších predikčních modelů logistického řetězce. Kdy jsou tyto modely následně implementovány do praktického užívání (obvykle jde o přepravní kapacity versus časové požadavky na doručení zboží). Big Data jsou pořizována i senzorikou např. sledování teploty, vlhkosti atd. při celém logistickém procesu. Tato data jsou pak využita jako podklad pro reklamační řízení, která by mohla vzniknout nesprávnou přepravou atd.

Energetika

Oblast energetiky využívá transakční Big Data z obchodní komunikace se svými zákazníky. Z oblasti Smart metering, která se zabývá monitorováním energetických sítí, zde jsou data automatizovaně pořizována z kontrolních čidel a zařízeni. Analýzy z Big Data jsou nejen využívány pro optimalizaci a zefektivnění energetických sítí, ale i pro jejich on-line monitoring, jehož cílem je optimalizovat servisní procesy a minimalizovat havárie energetické sítě. Dalším způsobem využití Big Data je tvorba nových zákaznických produktů, které obsahují lepší zákaznické služby.

Zemědělství

Big Data jsou ve větší míře pořizována pomocí senzoriky a Smart metering z důležitých zemědělských procesů. Samozřejmě, že základem jsou transakční data z IS systémů

využívaných v zemědělství. Jde o využívání dat z GPS naváděcích systémů při procesech obdělávání půdy (např. nastavení hnojení podle polohy, bonity a pěstované komodity atd.) s datovou historií. Nebo senzory umístěné na hospodářských zvířatech (např. v uchu atd.), které monitorují chování zvířat, např. u krav mohou sledovat dojivost jednotlivých kusů v automatizovaných dojících boxech atd. Big Data jsou zde využívána pro optimalizaci zemědělských procesů se záměrem dosažení co nejvyšší efektivity.

Zdravotnictví

V zdravotnictví jsou pořizována Big Data pomocí senzoriky a Smart metering z různých diagnostických přístrojů a zařízení při diagnostikování pacientů s využitím sofistikovaných analýz založených převážně na metodách umělé inteligence. Dalším zdrojem jsou lékařské záznamy pacientů, které obsahují informace o poskytnutých medikamentech, které lze propojit s logistikou medikamentů např. do nemocničních lékáren. Dále tato data slouží jako podklady pro analýzy zdravotních pojišťoven, které je využívají pro zajištění efektivního provozu a na analýzy veřejné služby pro pomocné činnosti v oblasti poskytování zdravotní péče.

3.2 Rizika při využití Big Data

Big Data jsou zdrojem velkého množství užitečných a pro zefektivnění procesů organizací významných informací, ale obsahují velké množství obchodních, osobních aj. údajů, jejichž zneužití může pro organizaci, která se zabývá ať již celým životním cyklem Big Data nebo jen její částí (např. uložištěm, analytikou atd.), být velkým problémem. Tento problém může vyvrcholit až možnými žalobami, pokutami a zánikem organizace, popřípadě trestní odpovědností osob, které mají tyto procesy v gesci.

Příklady základních rizik:

Neorganizovanost nestrukturovaných dat

Pojem neorganizovanost nestrukturovaných Big Data neznamená, že některá data nemají svoji vnitřní logiku nebo strukturu, ale jde zde o jejich charakteristiku popsánu v části této práce “1.2.1 Typy dat podle struktury”. Nestrukturovaná data jsou zpracovávána, uchovávána a analyzována náročnějšími SW a HW prostředky než data strukturovaná. Neorganizovanost dat nám zvyšuje náklady na veškeré procesy, které využijeme při jejich životním cyklu.

Zpracování, ukládání a uchovávání Big Data

Pro zpracování Big Data využíváme velmi sofistikované a efektivní techniky a technologie

podle způsobu jejich vzniku. Odlišné technologie využijeme u Big Dat, která vznikají v datových proudech (např. velký objem rychle vznikajících souborů atd.), a pro data, která vznikají v menších četnostech, která mají velké objemy, jenž způsobují problémy při ukládání na úložišti a je nutno pro jejich uložení využít velmi sofistikované technologie. Současným východiskem jsou cloudová řešení pro problematiku Big Data.

Rizika při analýze Big Data

Aby byla Big Data přínosem pro organizace, je nutné provést smysluplné analýzy, zda budou jejich výstupy přínosem pro plnění strategických (hlavních) cílů organizace (např. optimalizaci procesů, obchodní rozhodnutí, plánování nových strategií atd.).

- **Anonymizace dat** je velmi důležitý proces, který umožňuje organizacím analyzovat osobní informace (např. e-mail a elektronická komunikace, vyhledávače, ukládání dokumentů, multimédií atd.) a obchodní informace důvěrné povahy.
- **Zkreslení výsledků** může nastat u předdefinované automatizované analýzy a následné vizualizace, která odpovídá původnímu testu na datech, ale ty se v průběhu časové osy již změnily. Problémem je období, kdy si tohoto odchýlení nikdo nevšiml a data byla využívaná jako zdroje pro směřování důležitých procesů organizace. Tato problematika se dá částečně eliminovat pomocí umělé inteligence, která je využita při analýze dat.
- **Problematika dehumanizace** může hrát velký vliv u určité skupiny analýz, která jsou citlivé na tzv. osobní pohled experta provádějícího danou analýzu. Proto je nutné věnovat velkou pozornost při tvorbě automatizovaných analýz zaměření na modelové testy, které slouží pro vytvoření technologie pro tuto oblast (použití umělé inteligence).
- **Nesprávná data** mají velký vliv na zkreslení výsledků procesů analýzy a vizualizace. Jde obvykle o zmanipulovaná data vytvořená za účelem ovlivnění zkoumaného cíle, nebo o špatné provázání získaných dat pomocí nesmyslných korelací. Proto je nutné zavést sofistikovaný systém ověřování získaných výsledků automatizovaných analýz, aby byly co nejvíce eliminovány špatné a zkreslené výsledky.

Ochrana osobních údajů

Big Data obsahují velké množství strategických informací, které jsou velmi citlivé k soukromí osob, zákazníkům a procesům obchodní i výrobní povahy (např. ochrana technologií patenty atd.) atd. Zneužití surových dat (raw data) před anonymizací nebo výstupních dat, se špatně použitou anonymizací může vést až k žalobám a soudním sporům. Proto je nutné vyhovět všem zákonným požadavkům v této oblasti (např. GDPR atd.) a zapracovat tyto požadavky

do vnitřních směrnic a pokynů organizace. Následně tyto směrnice dodržovat a kontrolovat jejich dodržování s prováděním interních a externích auditů.

Náklady na problematiku Big Data

V současnosti dochází k velkému rozvoji různých platforem pro zpracování, archivování, analyzování a vizualizaci Big Data, kdy již není problém implementovat vhodnou platformu pro problematiku Big Data i do malých a středních organizací. Důležitý je vlastní přínos implementace komplexního řešení Big Data pro vybraný procesní segment organizace. Proto je nutné provést předimplementační analýzy zamýšleného řešení tak, aby bylo zavedení Big Data benefitem organizace ne její finanční ztrátou.

3.3 Dílčí souhrn

Na základě současných trendů, které směřují k exponencionálnímu zvyšování objemu dat, bude pro jejich zpracování, archivaci atd. využito komplexních cloudových řešení. Pro zpracování, analýzu a vizualizaci Big Data očekáváme rozvoj strojového učení, hlubokých analýz a dalších oblastí umělé intelligence. Bude nutné zabezpečit ochranu osobních údajů, protože se předpokládá vyšší využití dat ze sociálních komunikací (např. e-mail, sociální sítě atd.), s trendem využívat k analýzám dostatečně anonymizovaná data. Správnou úroveň anonymizace dat určí zákony, předpisy normy atd. Také bude nutné zabezpečit ochranu průmyslových dat tak, aby nebyly zneužity. Dojde k nárůstu objemu dat z domácích spotřebičů shromažďovaných pomocí technologií IoT. Tato data musí být využita se správnou úrovní anonymizace tak, aby jejich data a analýzy nebylo možné zneužít.

Pohled na Big Data strukturovaná versus nestrukturovaná není a nebude konfrontační, ale obě datové charakteristiky se budou navzájem doplňovat podle analyzovaných business modelů. Uživatel, který bude užívat výsledky analýz a vizualizací, nebude rozlišovat data podle jejich charakteristiky, ale podle využitelnosti v např. obchodních nebo business modelech s maximálním využitím automatizovaných procesů v celém životním cyklu Big Data.

Vlastní správa procesů v organizaci, které se budou zabývat problematikou Big Data (pořízení, zpracování, uložení, archivace, tvorba metodik analýz a vizualizací), budou v gesci datových odborníků (datových vědců), kteří se budou vzhledem k nárůstu znalostí v oblasti Big Data specializovat na jednotlivé segmenty Big Data. Bohužel takto specializovaných odborníků bude nedostatek vzhledem k velmi dynamickému růstu v oblasti Big Data.

4. MODELOVÁNÍ BIG DATA

Datové modelování je technika, která nám umožní definovat smysluplný pohled na data a následně je kategorizovat se stanovením oficiální definice a deskriptorů, pro následné využívání ve veškerých informačních systémech organizace. Strategické modelování dat nám usnadní vývoj IS v celé organizaci s výběrem vhodných databází. Datové modelování podle strategické osnovy nám navrhne, jaký druh dat bude potřeba pro procesy organizace zaváděné do IS. Jde o pochopení datových toků uvnitř organizace a o analýzu typů dat, které organizace využívá a shromažďuje v svých úložištích. Následně jde o porozumění obchodním procesům a vztahům řešené problematiky v organizaci, kdy tyto znalosti jsou vodítkem při vytváření dat a vztahů v datových modelech. Datové modelování je specifickým druhem dokumentace, která je využívaná v různých podobách, a to jak pro diskuze zainteresovaných stran (investoři, techničtí odborníci atd.), tak i pro vývojáře, kteří jsou zodpovědní za tvorbu IS organizace. Datový model nám v organizaci poskytne společný slovník pro řešenou problematiku, který mohou sdílet různé pracovní role a využít ho pro specifikaci konečné podoby daného řešení např. vývoje nové aplikace, modulu IS nebo celkového IS v organizaci. [26]

Datové modelování v případě Big Data se zaměřujeme na proces hledání podobností mezi daty z různorodých zdrojů, kdy je naším cílem klasifikovat datovou sadu představující Big Data a pomocí modelování formálně prozkoumat podstatu dat. Abychom mohli stanovit vhodný druh úložiště pro daná Big Data a vhodný způsob jejich zpracování. Termín modelování Big Data se častěji objevuje v literatuře od roku 2011. Kde na základě nových přístupů k řešení paradigmat Big Data následně vznikly nové přístupy modelování a správy Big Data v databázích. Mimo již standardně zavedeného relačního modelu databáze nastal velký rozvoj NoSQL databází. [26]

4.1 Základní úrovně datového modelování

I pro modelování Big Data využijeme standardní tři úrovně datového modelování: konceptuální datový model, logický datový model, fyzický datový model, viz. Obrázek č.24.

Konceptuální datový model je prvním krokem v datovém modelování, kdy se snažíme pochopit požadavky organizace na řešenou problematiku. Výsledkem je model, který popisuje realitu s určitou mírou abstrakce (ignorují se technická, implementační specifika) a zahrnuje všechny požadavky k dané problematice na základě zkoumání vztahu mezi entitami dané problematiky. K tvorbě tohoto modelu jsou využívány ER diagramy.

Logický datový model obsahuje podrobnou strukturu popisovaného systému s podrobnými vztahy mezi strukturami. Obsahuje základní entity, atributy, klíčové skupiny a vztahy mezi nimi atd. Logický datový model je definován bez ohledu na typ systému DBMS a je použitelný jak pro MongoDB tak i pro Oracle atd.

Přínosy logického datového modelu:

- Usnadňuje pochopení obchodních dat a jejich požadavků
- Poskytuje základ pro návrh databáze
- Pomáhá předcházet redundanci a nekonzistenci dat u obchodních procesů
- Podporuje opětovné použití dat a sdílení
- Snižuje náklady na vývoj, údržbu a časovou náročnost
- Potvrzuje logický model procesů a pomáhá při analýze dopadů

Logické modelování dat zahrnuje vytvoření konceptuálního datového modelu, normalizaci a abstrakci modelu, určení nejužitečnější formy s přezkoumáním a potvrzením logiky modelu.

Fyzický datový model je optimalizován podle požadavků použité technologie (databáze). Ve fyzickém datovém modelu jsou objekty definovány na úrovni schématu. Kdy schéma je možné definovat jako skupinu souvisejících objektů v databázi. Model lze označit jako fyzický datový model, pokud obsahuje všechny struktury tabulek, včetně názvů sloupců, datové typy sloupců, primární klíče, omezení sloupců a vztahů mezi tabulkami.

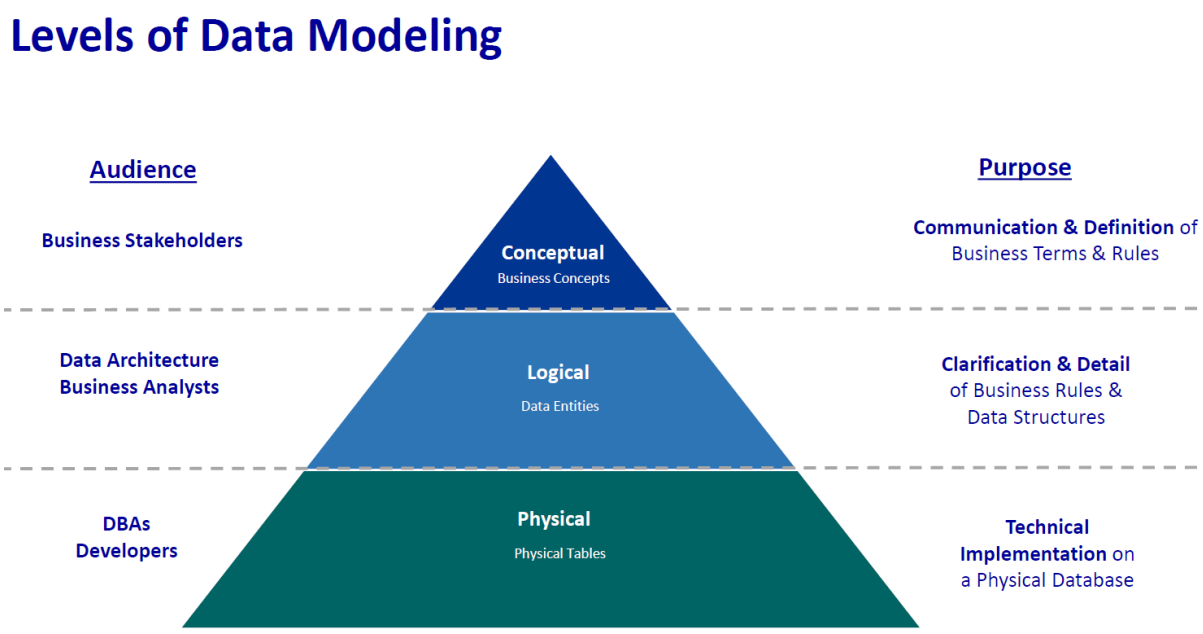
Přínosy fyzického datového modelu:

- Model představuje všechny tabulky a sloupce
- Fyzický model obsahuje všechny cizí klíče, které se používají k rozpoznání vztahů mezi tabulkami
- Fyzický datový model se u různých RDBMS/DBMS liší
- Na základě požadavků uživatele může v tomto modelu dojít k denormalizaci
- Mezi implementační technologie patří relační databáze, schéma XML, MongoDB atd.

Obrázek 23: Porovnání datových modelů.

Dotaz	Konceptuální data model	Logický data model	Fyzický data model
Existují v modelech entity a jejich vztahy?	Ano	Ano	-
Jsou v modelech uvedeny atributy?	Ano	Ano	-
Jsou v modelech zadány primární a cizí klíče?	No	Ano	Ano
Jsou v modelech uvedeny názvy tabulek?	-	-	Ano
Jsou v modelech uvedeny názvy sloupců a datové typy ?	-	-	Ano
Jsou modely zavislé na databázi ?	No	No	Ano
Jsou zadány i jiné databázové objekty jako jsou pohledy (VIEWS)?	No	No	Ano
Na jaké uživatele je model zaměřen? (Obchodní = Business)	Obchod	Obchod	Více na techniky méně na obchod

Zdroj : [Vlastní tvorba podle 26]



Obrázek 24: Pyramida úrovní datového modelování.

Zdroj : [26]

4.2 Datové modelování z pohledu databázového systému

Volbu databázových technologií je nutno vždy přizpůsobit danému řešení. Kdy vybereme vhodnou databázovou technologii podle optimálního modelu využívaných datových struktur v daném segmentu aplikačního řešení (např. pro strukturovaná data relační databázi a pro nestrukturovaná data NoSQL databázi). Výběr databáze je závislý na několika faktorech, jako je např. struktura zpracovávaných dat (např. zda jde o malé datové bloky generované vysokou rychlostí (IoT) nebo jde o zpracování souborů atd.), a také podle systému správy databáze (DBMS).

Pro použití nestrukturovaných dat nebo pro generované datové proudy semistrukturovaných dat (data z čidel (IoT)) není vhodné využívat relační databáze (RDMS), protože pro tuto technologii je problematické zpracovávat příliš velká, rozmanitá a rychle vznikající data. Relační databáze vyžadují před zápisem do databáze schéma, které je příliš rigidní, a vlastnosti ACID jsou pro některé aplikace příliš přísné. Big Data využívají distribuované systémy, a proto byly vytvořen modely CAP a BASE podle konceptu ACID, které umožňují využití vlastností ACID při správě dat a využití transakcí i v distribuovaných systémech.

Mezi základní principy zpracování Big Data patří:

- Škálovatelnost dat
- Konzistence dat
- Distribuce dat (sharding, replikace)

4.2.1 Škálovatelnost dat

Škálovatelnost dat je z pohledu zpracování dat, vlastností systému (sítě, procesu atd.), která umožňuje flexibilní reakce na měnící se požadavky např. na zvyšující se objem dat, zátěž systému, který tyto data zpracovává atd. Kdy v systémech relačních databází (RDBMS) je využíváno vertikální škálování (vertical scaling / scaling up) neboli zvyšování výpočetního výkonu daného serveru pomocí využití robustnějšího hardware. Tato technologie je velmi efektivní, ale bohužel pro zpracovávání Big Data je výkonnost této technologie nedostatečná. Nevýhodou jsou vysoké náklady na hardware serverového řešení.

NonSQL databáze využívají horizontální škálování (scaling out), které umožňuje provést distribuci zpracovávané úlohy v rámci množiny uzlů (clusters), kterou máme momentálně k dispozici s možností rozšířit tuto množinu o další uzly podle našich potřeb. Mohlo by se zdát, že toto řešení je optimální, protože umožňuje zpracování neomezeně velkých dat. Bohužel tato technologie je závislá na mnoha faktorech, jako je sto procentní spolehlivost, bezpečnost, homogenní sítě s nulovými náklady na přenos dat s neomezenou šíří přenosového pásma atd. Bohužel definice tohoto konceptu je v reálných podmínkách nedosažitelná, proto se snažíme tomuto konceptu co nejblíže přiblížit pomocí distribuovaných systémů zpracování dat. [20, 26, 27]

4.2.2 Konzistence dat (ACID, CAP, BASE)

Konzistence dat je důležitá pro efektivní zpracovávání dat. Pro relační databáze podmínka konzistence představuje integritní omezení dat a referenční integritu daného datového modelu

aplikace. Databázový systém následně pracuje na úrovni transakcí (kdy transakce je sekvence logicky souvisejících operací, které převádějí data z jednoho konzistentního stavu do druhého). V průběhu transakce může být referenční integrita dočasně narušena, pokud je to pro danou sekvenci operací potřeba, ale po ukončení transakce musí být veškerá integritní omezení splněna. [20, 26, 27]

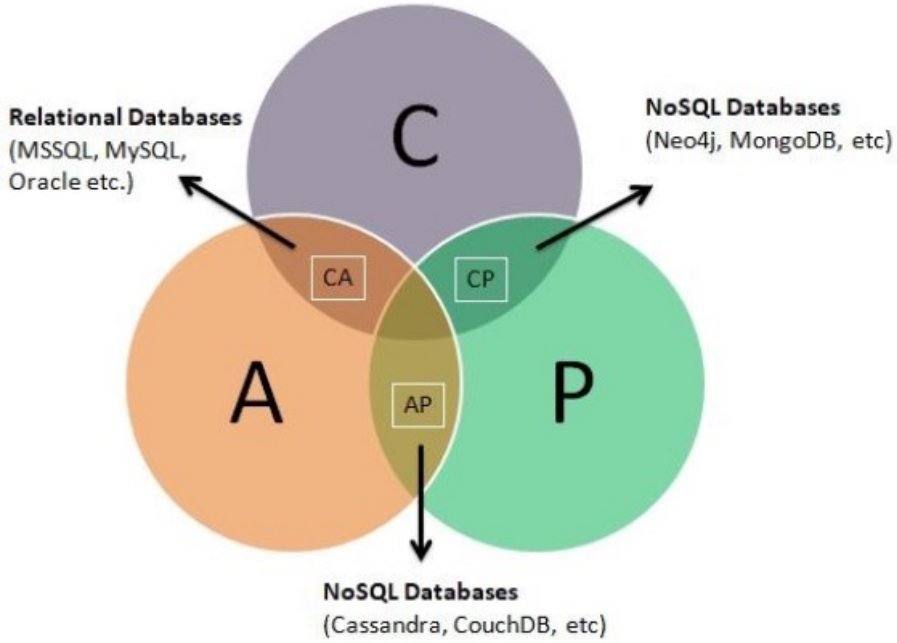
ACID (atomicity, consistency, isolation, durability) je definice souhrnu vlastností, které nám zajistí správné fungování transakcí.

- **Atomicita (atomicity)** je nedělitelnost databázové transakce, má dva stavy proběhne celá nebo neproběhne vůbec.
- **Konzistence (consistency)** zajistí přechod z počátečního konzistentního stavu před začátkem transakce do konečného konzistentního stavu po ukončení transakce.
- **Izolovanost (isolation)** zajistí skrytí veškerých proběhlých operací v průběhu transakce před ostatními probíhajícími transakcemi.
- **Trvalost (duralibity)** nám zajistí uložení výsledků úspěšně proběhlých transakcí do databáze.

Přínosem vlastností ACID je transparentní průběh transakce bez narušení konzistence dat v databázi. Transakční systém je založen na principu souběhu transakcí. Pokud transakce pracují s různými daty nebo je využívají jen pro čtení, je možné tyto transakční operace libovolně prokládat. Problém nastává, pokud různé transakce požadují využít (modifikovat) stejná data v stejném čase, pak dochází ke konfliktu. Obecně jsou popisovány tyto konflikty mezi dvěma transakcemi jako je **„write-read, read-write a write-write“,** kdy je nutné tyto stavy správně ošetřit, tak abychom neztratili konzistenci dat. Správné ošetření je obecně prováděno prostředky s nastavením správné izolace transakcí, a to i pro stejného uživatele, který pracuje s datovými prostředky v několika režimech přihlášení k danému systému. Vlastní pravidla izolace volíme podle úrovně potřeby aplikace, kdy platí pravidlo, čím volnější pravidla stanovíme tím je práce s databází efektivnější. Toto pravidlo je velmi problematické, a proto možné problémy musíme ošetřit jiným způsobem např. na úrovni serveru při ukládání dat atd. Izolace transakcí rozlišujeme v čtyř úrovních: **„read uncommited“** (může nastat čtení nepotvrzených dat, neopakovatelné čtení a výskyt fantomů), následuje **„read commited“** (nemůže nastat čtení nepotvrzených dat, stále může dojít k neopakovatelnému čtení a mohou se vyskytovat fantomy), **„repeatable read“** (nemůže nastat čtení nepotvrzených dat a neopakovatelné čtení, ale mohou se vyskytovat

fantomy), **„serializable“**(nemůže nastat čtení nepotvrzených dat, neopakovatelné čtení a výskyt fantomů), která je nejpřísnější úrovní. Transakce je prostředkem zajištění integrity dat a je ochranou před SW a HW chybou (např. nedostatek místa na disku atd.). [20,26,27]

CAP (consistency, availability, partition tolerance) teorém je popis vlastností transakcí v distribuovaném databázovém systému.



Obrázek 25: CAP teorém.

Zdroj : [41]

- **Konzistence (consistency)** existuje jediná aktuální verze dat.
- **Dostupnost (availability)** všechny požadavky na manipulaci s daty jsou systémem obslouženy.
- **Odolnost (partition tolerance) vůči rozpadu sítě** je u distribuovaného systému zajištěna tím, že funguje, i když se rozpadne do několika separátních částí.

❖ **CA (consistency, availability)** zaručuje konzistenci dat. Tuto činnost zajistí všechny uzly serveru, které jsou k dispozici a mohou spolu komunikovat a budou k dispozici až do úplného splnění požadavku. Po dobu, co tato služba obsluhuje tyto požadavky, není k dispozici pro další požadavky.

❖ **CP (consistency, partition tolerance)** zaručuje konzistentnost dat napříč klastrem (cluster), bez ohledu na geografické umístění fyzických dat.

- **AP (availability, partition tolerance)** nezaručuje, že data jsou vždy konzistentní. Tato služba zajistí, že data budou vždy správně rozděleny a k dispozici.

Ideálním stavem CAP teorému (Brewerův teorém) je dosáhnutí všech vlastností (konzistence, dostupnost, odolnost vůči rozpadu sítě) současně. Bohužel v distribuovaném systému jsme obvykle schopni dosáhnout maximálně dvou uvedených vlastností. V distribuovaném systému je základní vlastností odolnost vůči rozpadu sítě, která musí být vždy splněna, proto budou do určité míry omezeny vlastnosti konzistence nebo dostupnost. K omezení vlastností konzistence a dostupnost dochází pouze v případě, kdy dojde k selhání vlastnosti odolnost vůči rozpadu sítě. V případě, že vše běží OK, není nutno realizovat žádné kompromisy a vše co je nastaveno lze zcela využít. Konzistenci dat tak zajistíme pomocí vhodných modelů transakcí.

Dostupnost systému je obvykle vyjadřovaná pomocí procentuálního podílu času systému s přijatelnou odezvou versus celkový nabízený čas. Hodnotu dostupnosti členíme do několika kategorií, které odpovídají požadavkům na využití daného systému.

Další důležitou vlastností nejen distribuovaných systémů je odolnost vůči chybám, a to nejen způsobených programátorem, ale i vůči chybám, které vznikly selháním modulů systému. Chyby členíme na latentní (prozatím neprojevené) a efektivní (to jsou ty, co se již projevily). Chyby a jejich vliv na chod systému ovlivňují samozřejmě dostupnost systému.

Zpracování transakcí v distribuovaném systému je komplikovanější, protože probíhá na dvou a více různých uzlech clusteru. Hlavním cílem strategií pro řízení a koordinace transakcí v distribuovaném prostředí je zajistit efektivní uspořádanost zpracovávaných dat a zotavitelnost, což je návrat do posledního korektního stavu po proběhlé chybě.

Pro řízení transakcí v distribuovaných systémech je nejčastěji využíván 2PC a 3PC protokol. 2PC protokol pro distribuovanou transakci řídí uzel pojmenovaný „koordinátor“, který má za úkol koordinovat veškeré činnosti transakce v jednotlivých uzlech clusteru. Zjednodušeně, probíhá ve dvou krocích. První krok provede zjištění připravenosti veškerých objektů, s kterými transakce pracuje a zajistí je. Pokud transakce zajistí všechny objekty, je možné pokračovat ve zpracování dat dle procesů transakce s uložením výsledků transakce. Pokud některý z požadovaných objektů zašle informací při kontrole připravenosti, že nebyl připraven, dojde k přerušení transakce. K přerušení dojde také, pokud nebyla doručena odpověď o uložení výsledků transakce, ve všech distribuovaných částech systému.

3PC protokol je rozšířením 2PC protokolu, který je choulostivý na odezvy koordinátora transakce při čekání na pokyn pro ukončení transakce (commit/abort). Proto byl 2PC protokol transakce rozšířen o jeden krok při potvrzování ukládaných výsledků. První krok proběhne jako u 2CP protokolu a nový druhý krok je doplněn o oslovení všech požadovaných objektů na všech uzlech daného řešení. Pokud dostane kladnou odpověď, že jsou připraveny, tak pošle odpověď do třetího kroku, jinak proces transakce přeruší. Na základě výsledků předchozího kroku (nový druhý krok) zašle koordinátor požadavek na uložení výsledků transakce nebo na přerušení transakce. Pokud zde koordinátor selže a ostatní prvky nedostanou informaci o požadavku na uložení výsledků transakce, dojde k uložení výsledků ve všech distribuovaných částech automaticky na základě potvrzení z druhého kroku. V případě, že byl v druhém kroku požadavek na přerušení, dojde k přerušení transakce.

Pro správné využívání transakcí v distribuovaných systémech je důležité správně využívat také optimistické nebo pesimistické off-line zamykání objektů, a časových razítek (daný systém musí být konstruován pro využití časových razítek). Zamykání objektů a systém časových razítek využíváme v dané transakci pro eliminaci problematických situací, které by měly za důsledek přerušení a nedokončení transakce. [20, 26, 27]

BASE (basicalli available, soft state, eventual consistency) jde o využití ACID v distribuovaném systému, kdy při využití nižší míry konzistence získáme vyšší škálovatelnost, která by v ACID nebyla možná.

Charakteristiky BASE:

- **Převážná dostupnost (basicalli available)** znamená, že je systém dostupný s občasnými lokálními výpadky, ale nikdy nedojde k výpadku celého systému.
- **Volný stav (soft state)** představuje dynamický a nedeterministický stav kde neustále dochází ke změnám.
- **Občasná konzistence (eventual consistency)** systém není stále v konzistentním stavu. Konzistence zde není zaručena v průběhu všech procesů jako u ACID. Občasná konzistence je obvykle způsobena ukládáním dat v distribuovaného systému do všech využívaných uzlů s replikacemi dat, a proto může docházet k časovým zpožděním.

Relační databázové systémy používají pesimistické postupy řízení souběžnosti transakcí, které využívají umístění a uvolnění zámků podle dvou fázového protokolu transakcí. Pokud jsou využívány transakce s menšími modifikacemi dotazů, je možné nasadit optimistické metody řízení transakcí. Proto u distribuovaných systémů je možné využívat konzistence dat se

zpožděním, ale pouze podle pravidel CAP. Většina NoSQL systémů využívá optimistického řízení souběžnosti transakcí.

Princip občasné konzistence vychází ze zkušeností při využívání mnoha aplikací, kdy rychlost je považována za mnohem důležitější vlastnost, než je dokonalá konzistence. Protože čím je požadavek na konzistenci v distribuovaných systémech vyšší, tak čas na načítání dat je delší a dochází ke zpomalování dané aplikace. Proto je důležité při tvorbě aplikace, která bude implementovaná jako distribuovaný systém, zvážit jakou úroveň konzistence potřebujeme, zda ACID nebo BASE s využitím CAP. [20, 26, 27]

Tabulka 6: Porovnání ACID a BASE.

ACID	BASE
Konzistence je nejvyšší prioritou (silná konzistence)	Je zajištěna pouze občasná konzistence (slabá konzistence)
Většinou pesimistické souběžné metody s metodami pro uzamykací protokoly	Většinou optimistická souběžnost využívající možnosti různých nastavení.
Dostupnost je zajištěna pro menší a střední objemy dat	Vysoká dostupnost pro distribuovaná datová úložiště
Některé omezení integrity (např. referenční integrity) podle schématu databáze	Některé omezení integrity (např. referenční integrity) podle schématu databáze

Zdroj : [Vlastní tvorba podle,27]

4.2.3 Distribuce dat

Pro optimální zpracování Big Data na distribuovaných systémech je možné využít ortogonální techniky pro zpracování distribuovaných dat. Tyto techniky jsou využívány u většiny NoSQL databází (např. pro grafové databáze využíváme lokální úložiště). Pokud je to samozřejmě technicky možné, tak data nedistribuujeme, protože distribuce dat je náročná jak pro správu datového úložiště a pro zpracování a vizualizaci dat.

Obvykle pro distribuci dat využívám:

- **Rozdělení (sharding)** je proces rozmístění různých částí dat na různé uzly v clusteru pro zvýšení kapacity systému.
- **Replikace** je proces vytvoření přesných kopií dat (neboli replik) na vice uzlů clusteru, pro zvýšení propustnosti daného systému.

Rozdělení (sharding)

Pro horizontální škálování je nutné data rozdělit na vhodné podmnožiny (ne nutně disjunktní), které uložíme na různé uzly v clusteru. Uživatelé podle potřeby přistupují přes aplikaci k datům, která jsou uložena v uzlech daného clusteru. Efektivita systému je spojena s vhodnou strategií uložení dat, proto se zaměřujeme na rovnoměrné rozmístění dat na uzlech a na optimalizací jejich struktury se zaměřením na minimalizaci počtu uzlů, které uživatel použije pro načtení požadovaných dat. Většina NoSQL databází provádí dělení dat automaticky a nabízí uživatelům výběr preferované strategie. [20, 26, 27]

Replikace

Jde o proces vytvoření přesných kopií dat, které jsou dle IT terminologie nazývány replikacemi.

Master-slave replikace je vhodná pro data, která jsou často čtena, ale minimálně modifikována. Když je zvětšen požadavek na čtení daných dat zvýšíme počet replikací dat z primárního (master) uzlu na sekundární (slave) uzly a vytvoříme vetší počet uzlů. V případě že selže primární uzel je tak nahrazen sekundárními uzly, které mají stejnou kopii dat. Nové primární uzly jsou vytvářeny správcem systému nebo automatizovanou předdefinovanou procedurou. Pro tento druh replikace je využita centralizovaná správa primárních uzlů, proto při vyšším počtu požadavků na založení nových primárních uzlů může dojít k ztrátě výkonu. [20,26]

Peer-to-peer replikace řeší problematiku centralizované správy primárních uzlů tak, že všechny uzly jsou si rovny. Kdy při výpadku některého z uzlů neztrácíme možnost čtení a zápisu dat. Tato replikace má vyšší škálovatelnost zápisů, ale bohužel se zvyšuje nekonzistence dat. Konflikty konzistence dat jsou řešeny pomocí koordinace uzlů, kdy tato koordinace zvyšuje nároky na výkon systému. Nebo pomocí replikačního faktoru, který nám definuje minimální počet uzlů, na kterých musí daný process proběhnout. Pokud definujeme např. replikační faktor na hodnotu čtyři. Je následně každá část dat uložena na čtyřech uzlech v clusteru. [20, 26, 27]

Procesy rozdělení a replikace dat probíhají společně. Samozřejmě, že nejprve proběhne rozdělení dat podle strategie, kterou zvolíme a následně provedeme replikace podle zatížení systému. Typ replikací vybereme podle charakteru zpracovávaných dat.

4.3 Datové modelování a Big Data

Pro datové modelování je vhodný již standardně používaný unifikovaný modelovací jazyk UML (dále jen UML). Jazyk UML patří do množiny otevřených průmyslových standardů, které jsou určeny pro visuální modelování dat. UML není metodikou, proto je obvykle při vývoji software doplňován metodikou Unified Process, která je založena na projektovém principu. Kdy každá iterace je samostatným projektem a obsahuje tyto fáze: specifikace požadavků, analýzy, implementace, testování a ostré nasazení aplikace/IS. Tyto iterace probíhají, až do finální podoby vyvíjené aplikace/IS. Podrobnostmi metodiky Unified Process se vzhledem k rozsahu této práce nebudu zabývat. Výhodou UML je, že jde o otevřený standart podporující celý vývojový cyklus aplikace/IS.

Modely jazyku UML jsou zaměřeny na statickou strukturu a dynamické chování, které je popsáno pomocí strukturní abstrakce modelu s relací pro propojení na výstupní diagramy, které vytvářejí podhled na daný model. Základním principem UML modelu je pohled 4+1, který obsahuje logický, procesní, implementační pohled a pohled praktického nasazení, kdy jsou všechny tyto pohledy sjednoceny do pohledu případů užití, který odpovídá požadavkům uživatele na danou aplikaci/IS.[1]

Výhodou jazyka UML je jeho použitelnost jak pro business modelování, tak pro modelování jakékoliv aplikace na jakémkoliv hardware, operačním systému, programovacím jazyku a síti. UML je možné využít jak pro lokální aplikace, tak i pro distribuované aplikace. [1]

Specifikaci softwarových požadavků vyjádříme pomocí metamodelu, který obsahuje model požadavků (requirements model) a model případů užití (use case model). Tento model nám vyspecifikuje funkční požadavky na aplikaci. Dále nám vyspecifikuje případy užití (funkcionality aplikace) s aktéry (role s přímou interakcí s aplikací) a vztahy mezi nimi. Pro získání požadavků na aplikaci je možné využít dotazníkové šetření a požadavky následně pro efektivní využití hierarchicky uspořádáme (vytvoříme taxonomie). [1]

Dalším krokem je analýza, která zahrnuje většinu aktivit, které se týkají tvorby modelů a požadovaného chování aplikace. Analýza obvykle probíhá souběžně se specifikací požadavků. Záměrem analýzy je analytický model, který popisuje, co musí aplikace udělat, ale nezabývá se způsobem provedení. Způsob provedení je řešen v návrhovém modelu. Hranice

mezi analytickým modelem a návrhovým modelem je velmi subjektivní a závisí vždy na konkrétním řešiteli. Pro modelování všech možných procesů v dané aplikaci využíváme diagram aktivit a objektově orientované vývojové diagramy.

V této práci se budu zabývat návrhovými modely, kdy návrhový model vznikne upřesněním a rozpracováním analytického modelu. Model může být pro větší přehlednost rozčleněn do několika podsystémů.

Pro návrhové modely je možné použít tyto UML diagramy: [1]

- Funkční náhled (diagram případů užití (Use Case), business case diagram)
- Logický náhled (diagram tříd, objektový diagram)
- Dynamický náhled (stavový diagram, diagram aktivit, sekvenční diagram, diagram spolupráce)
- Implementační náhled (diagram komponent, diagram rozmístění)

V následující části této práce je mým záměrem použít tyto UML techniky pro obecný popis problematiky pro část zaměřenou na strukturovaná data: [1]

- Business case diagram
- Use Case diagram
- Use Case scénář vybraného případu užití
- Sekvenční diagram pro vybraný případ užití
- Doménový diagram (strukturovaná data)

Pro modelování semistrukturovaných a nestrukturovaných dat je mým záměrem využít: [1, 26]

- data flow diagram,
- vhodné UML diagramy
- CRD (collection relationship diagram)
- myšlenkovou mapu (Mind map)

Myšlenkovou mapu a CRD vytvořím podle doménového diagramu jako alternativu pro NoSQL technologii (např. dokumentovou databázi MongoDB). Kdy struktura myšlenkové mapy je implementována do NoSQL technologií.

Pro modelování Big Data je důležité mít představu, co jsou Big Data a jaké mají vlastnosti. Pokud máme datovou sadu, kterou klasifikujeme jako Big Data, je cílem datového modelování formálně prozkoumat podstatu dat, tak abychom mohli stanovit, jaké úložiště dat potřebujeme

a jaké druhy zpracování nad těmito daty je možné zrealizovat. Big Data kategorizujeme, stanovíme oficiální definice a deskriptory tak, aby data mohla být využita ve všech informačních systémech dané organizace. Z pohledu datové analýzy rozlišujeme datové modelování na modelování podle strategické osnovy a modelování v kontextu analýzy. Strategická osnova nám navrhuje, jaký druh dat bude pro procesy organizace potřeba na rozdíl od modelování v kontextu analýzy, které se zaměřuje na existující data a jejich klasifikaci. V případě Big Data jde o hledání podobnosti dat z různorodých zdrojů. Cílem je generování reprezentace dat, kterou lze replikovat v databázové architektuře dané organizace. Podrobnosti o Big Data jsou v části této práce „2. BIG DATA“.

Pro modelování Big Data využíváme techniky jako je agregace a denormalizace, která je důležitá pro NoSQL distribuované systémy. V této části nebudu rozebírat datové modely pro NoSQL databáze, protože jsou velmi specifické podle použité databázové technologie (např. databáze klíč-hodnota, dokumentové databáze, sloupcové databáze a grafové databáze).

4.4 Datové modelování v relačních databázích

Datové modelování v platformě relačních databází je založeno na principu normalizace. Kdy normalizace je procesem, který návrh database zjednoduší, odstraní redundance a z efektivní využívaní datového modelu (např. rychlejší zápis dat nebo tvorba výstupů z aplikace jako je tvorba reportů a vizualizace analýz).

- **První normální forma 1NF (jedinečnost polí)** musí splňovat podmínky nulté normální formy 0NF (Alespoň jeden atribut obsahuje více než jednu hodnotu) + všechny atributy tabulky musí být atomické, tedy dále nedělitelné.

- **Druhá normální forma 2NF (primární klíč)** musí splňovat podmínky 1NF a každý neklíčový atribut (není součástí žádného primárního klíče) musí být plně závislý na primárním klíči. 2NF klade důraz na odstranění případných duplicit.

- **Třetí normální forma 3NF (funkcionální závislost)** musí splňovat podmínky 2NF a všechny neklíčové atributy musí být vzájemně nezávislé. Jde o dekompozici tabulky s odstraněním tranzitivní závislosti.

- **Čtvrtá normální forma 4NF (nezávislost polí)** musí splňovat podmínky 3NF, atributy primárního klíče musí být vzájemně nezávislé a modifikace libovolného neklíčového atributu (není součástí primárního klíče) nesmí ovlivnit jiné pole tabulky.

- **Pátá normální forma 5NF** musí splňovat podmínky 4NF a pro tabulku platí, že ji není možné již dále bezeztrátově rozdělit, např. využitím dekompozice tabulky pro snížení redundance dat, ale bohužel se ztrátou důležitých relačních vztahů a informací.

Důležitým požadavkem na tvorbu datového modelu je také správné navržení vztahů mezi tabulkami databáze. Protože jen správně navržená databázová struktura je základem pro efektivní a výkonný databázový systém. Dalším krokem po normalizaci tabulek je nastavení vhodných vztahů mezi tabulkami pomocí primárního klíče v primární tabulce a sekundárního klíče v sekundární tabulce. Podmínkou je, že oba klíče musí být stejného datového typu. Vztahy mezi tabulkami nazýváme relace a členíme je na tyto typy 1:1, 1:N, M:N.

Relace jedna k jedné (1:1, one-to-one) vytváří vztah mezi tabulkami, který je založen na předpokladu, že každý záznam z primární tabulky je svázán pouze s jedním záznamem ze sekundární tabulky pomocí jedinečných klíčů v obou tabulkách (primární a sekundární klíče).

Relace jedna ku více (1:N, many-to-one) vytváří vztah mezi tabulkami, který je založen na předpokladu, že každý záznam z primární tabulky je svázán s jedním a více záznamy ze sekundární tabulky pomocí jedinečných klíčů v obou tabulkách (primární a sekundární klíče).

Relace více ku více (M:N, many-to-many) vytváří vztah mezi tabulkami, který je založen na předpokladu, že více záznamů z primární tabulky může být svázáno s více záznamy ze sekundární tabulky pomocí jedinečných klíčů v obou tabulkách (primární a sekundární klíče). V praktických řešeních je tato relace realizovaná pomocí vazební (propojovací tabulky), kde je relace M:N rozložena do dvou vztahů 1:N:M:1.

Dalším druhem relace je **unární relace.** Tato relace vytváří vztahy uvnitř jedné tabulky pro vytvoření hierarchických vztahů mezi záznamy dané tabulky.

Pro modelování dat v relačních databázích využíváme technik UML, které jsem uvedl v části této práce “4.3 Datové modelování”.

4.5 Datové modelování v NoSQL databázích

Datové modelování v NoSQL (Not only SQL) databázích je zcela uzpůsobeno konstrukcí nerelační technologie v daném databázovém prostředku. Existují základní faktory, které odlišují tradiční relační databázi (RDBMS) od NoSQL databází: **rozmanitost, struktura**

(předdefinovaná/ dynamická)**, škálování** (vertikální/horizontální) **a zaměření** (integrita dat/výkon a dostupnost dat). Využití uvedených faktorů je popsáno v předchozích částech této práce např. v "4.2. Datové modelování z pohledu databázového systému". Z tohoto důvodu se této problematice budu věnovat jen krátce.

RDBMS má jen jeden druh struktury (relační), kde jsou jednotlivé záznamy ukládány do řádků tabulky, a každý sloupec tabulky obsahuje konkrétní údaje k danému záznamu. Tabulky je možné navzájem spojovat pomocí vztahů (relací). RDBMS je řešení pro běžný každodenní typ provozního nebo analytického scénáře.

Technologie NoSQL nabízí flexibilní a rozšiřitelný model schématu s dalšími výhodami jako je téměř nekonečná škálovatelnost distribuovaného nastavení a možnost propojení s non-SQL rozhraním. NoSQL databáze lze rozčlenit do čtyř základních kategorií podle využívané technologie:

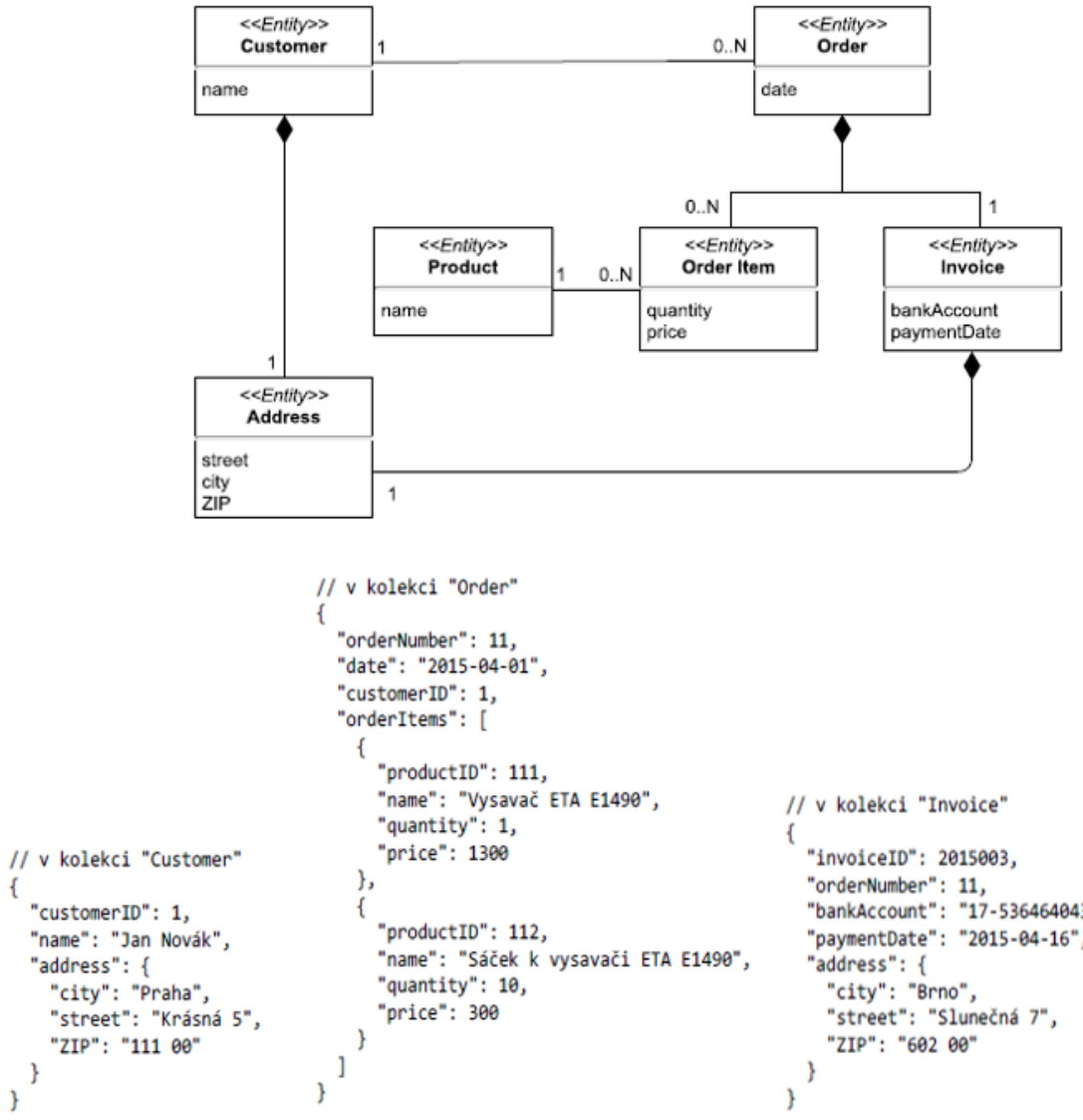
- **databáze klíč-hodnota:** Mezi používané databáze tohoto typu patří například Cassandra, DynamoDB, Azure Table Storage (ATS), Riak, Berkeley DB atd.
- **dokumentové databáze:** Mezi používané databáze tohoto typu patří například MongoDB a CouchDB atd.
- **sloupcové databáze:** Mezi používané databáze tohoto typu patří například HBase a Hypertable atd.
- **grafové databáze:** Mezi používané databáze tohoto typu patří například Neo4j, Polyglot atd.

Kdy každý typ má svůj vlastní způsob správy a ukládáni dat. NoSQL řešení je založeno na efektivním využívání scénářů pro konkrétní typy situací.

Pro datové modelování v NoSQL databázích je možné využít také ER diagramy (entity-relationship) jako v relačních databázích, ale zde je upřednostněn koncept agregací. Tento diagram vychází z konceptu UML ER digramu (doménový diagram) pro strukturovaná data v relační databázi. ER diagram s konceptem agregací využíváme pro pohled na entity, se záměrem vytvořit logické celky entit a ty pak vhodným způsobem zakomponovat do následně vytvářeného databázového schématu. Jde nám o zvýšení výkonnosti a dostupnosti dat, která jsou tak umístěna pohromadě. Jsou sdružena do logických celků podle způsobu užití (dotazy na data atd.) a jsou tak při distribuci na několika uzlech distribuovaného systému snadněji k dispozici procesům každé aplikace, která je používá. Kdy převedení ER diagramu na databázové schéma odpovídá vlastní struktuře NoSQL databáze. Obdobným způsobem je

možné převést standardní ER diagram, doménový diagram, který je konstruován pro relační databáze, do vhodného konceptu pro technologii NoSQL pomocí MindMap (myšlenkové mapy). Přes MindMap a např. pomocí jazyka JSON (JavaScript Object Notation) lze vytvořit databázové schéma s následným vytvořením struktury NoSQL database.

Při použití denormalizace může dojít u schémat databáze k redundanci dat. Tato redundance u Big Data v NoSQL databázích sice přináší vyšší nároky na uložení a správu dat, ale zároveň zajistí rychlejší zpracování a výstup dat. Proto je někdy výhodné implementovat pro některé části databázového schématu přesně specifikované redundance pro zefektivnění vlastní aplikace.



Obrázek 26: Koncept agregace v E-R digramu a schématu v notaci JSON.

Zdroj : [20]

Tabulka 7: Porovnání typů NoSQL databází.

Datový model úložiště	Výkon (Performance)	Škálovatelnost (Scalability)	Flexibilita (Flexibility)	Složitost (Complexity)
Klíč - hodnota	vysoký	vysoká	vysoká	žádná
Sloupcové	vysoký	vysoká	mírná	nízká
Dokumentové	vysoký	variabilní (vysoká)	vysoká	nízká
Graf	variabilní	variabilní	vysoká	vysoká

Zdroj : [Vlastní tvorba podle,41]

V tabulce č.7 jsou porovnány důležité charakteristiky typů NoSQL databází z pohledu jejich výkonnosti, škálovatelnosti, flexibility a složitosti. Flexibilita se zaměřuje na využití strukturovaných a nestrukturovaných dat a případy použití (use case). Kdy část flexibility v této tabulce zahrnuje problematiku technologie Big Data a její analytickou platformu. V části složitost (complexity) je popis, který specifikuje náročnost na složitost vývoje, modelování, provozu a udržování dat atd. v popisovaných typech NoSQL databází. Tyto podklady mohou sloužit jako vstupní informaci při výběru příslušné NoSQL database pro implementaci dané problematiky.

Výběr řešení NoSQL databáze by měl být také spojen s analýzou potřeb na datové úložiště. Kdy pro výběr NoSQL úložiště je možné použít tyto faktory:

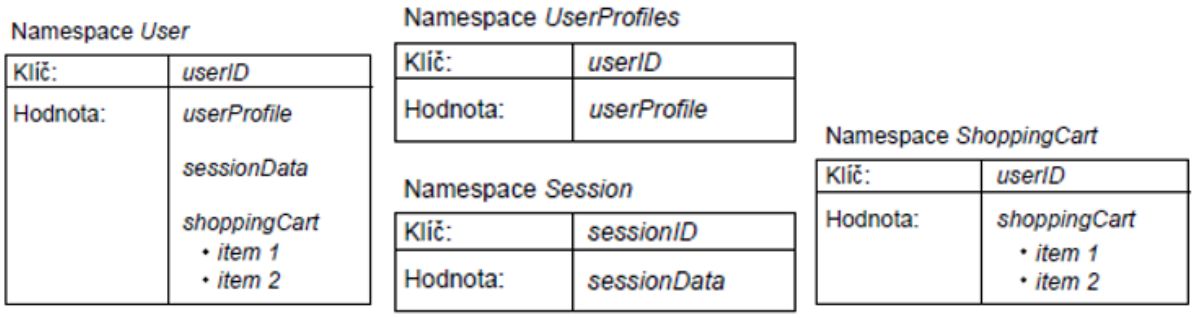
- Rozmanitost vstupních dat
- Škálovatelnost
- Výkon
- Dostupnost
- Náklady
- Stabilita

4.5.1 Databáze typu klíč-hodnota

Tyto databázové systémy přiřazují ukládaným hodnotám unikátní klíče a následně je společně ukládají do jednoduchých tabulek (kdy je zajištěna jedinečná vazba mezi hodnotou a unikátním klíčem). Výhodou tohoto typu databází je vysoká rychlost při přístupu k datům, se snadnou možností distribuování dat. Nevýhodou tohoto typu databází je možnost vyhledávání jen podle

unikátního klíče a ne podle hodnot. Jsou zde povoleny tři základní operace s daty: vložení dat k danému klíči, načtení dat daného klíče a vymazání dat daného klíče. [20, 26]

Datový model je založen na jmenném prostoru klíčů. Konstrukce jmenného prostoru klíčů má velký vliv na vlastní výkonnost dané databáze.



Obrázek 27: Modelová ukázka uložení dat v jednom a ve více jmenných prostorech klíčů.

Zdroj : [20]

4.5.2 Sloupcové databáze

Tyto databáze jsou svojí strukturou z NoSQL nejblíže k standardnímu relačnímu modelu, protože data jsou zde uložena v tabulkách s řádky a sloupci, ale s tím rozdílem, že každý řádek může mít téměř libovolný počet sloupců. Zde každý sloupec má svůj název (column name), hodnotu (column value) a časové razítko, kdy byla daná hodnota uložena do systému (time stamp). Jednotlivé sloupce jsou sdruženy do skupin sloupců (column families). Některé sloupcové databáze využívají další úroveň zanoření zvanou supersloupec. Kdy konstrukci super sloupce odpovídá zanořená rodina sloupců. [20, 26]

Základem datového modelu u tohoto typu databází je řádek, který je identifikován pomocí jedinečného klíče. Struktura by tak mohla odpovídat relační databázi, ale tento koncept může mít pro každý řádek libovolný počet sloupců. Při návrhu schématu mohou být použity pouze sloupce, které existují v rámci dané rodiny sloupců, jenž je definována v schématu k danému řádku. A z takto specifikovaných rodin sloupců lze pro daný řádek vytvářet libovolnou strukturu sloupců (sloupce lze libovolně pojmenovat). Kdy takto definovaný klíč odpovídá primárnímu klíči dokumentu, který je používán např. v MongoDB, a dvojici název-sloupce a hodnota-sloupce, který odpovídá asociativnímu poli {název: hodnota} z JSON objektu. Stejný klíč řádku lze využít v různých rodinách sloupců, kdy řádky se stejným klíčem tvoří jeden logický datový

celek. Každá rodina sloupců obsahuje hodnoty, které spolu logicky souvisí. Bývají obvykle stejného typu s předpokladem, že budou často společně dotazovány. [20, 26]

Tento design je vysoce škálovatelný a nabízí vysoký výkon. Drobnou komplikací je výsledná prezentace a vizualizace dat, která vyžaduje sofistikovanější tvorbu datových sad. Existují dva obecné pohledy na problematiku sloupcových databází. [20, 26]

Jeden z nich spočívá v pohledu na rodinu sloupců jako na relační tabulku. Kdy po vytvoření všech sloupců pro každý řádek nahradíme chybějící hodnoty hodnotou "null". Jeden logický záznam bude tvořen řádky se stejným klíčem pro všechny rodiny sloupců (tabulek). [20, 26]

Druhý pohled je zaměřen na hodnoty v sloupcích a na celou datovou strukturu. Na kterou se dívá jako na multidimenzionální asociativní pole. Kdy mezi dimenze pole patří: klíč řádku, sloupec (popřípadě supersloupec) a časové razítko hodnoty. Takto specifikované pole bývá obvykle seřazeno podle všech dimenzí. [20, 26]

Na obrázku č.28 pro zjednodušení jsou vynechána časová razítka. [20, 26]

```
// rodina sloupců "users"
{
  // řádek s klíčem "honza"
  "honza": {
    "firstName": {
      "1429268224554000": "Jan"
    },
    "lastName": {
      "1429268224554000": "Novák"
    },
    "email": {
      "1429268226436000": "honza@seznam.cz",
      "1429268234554000": "honza@gmail.com"
    }
  },
```

Databáze uživatelů organizovaná jako rodina sloupců

user_id (klíč řádku)	název sloupce	název sloupce	...
	hodnota sloupce	hodnota sloupce	...
1	login	first_name	...
	honza	Jan	...
4	login	age	
	david	35	...
5	first_name	last_name	...
	Jana	Novotná	...
...			...

Obrázek 28: Ukázka skupiny sloupců (column families) users.

Zdroj : [20]

4.5.3 Dokumentové databáze

Dokumentové databáze jsou vytvořeny pro správu a ukládání různých druhů strukturovaných dokumentů, které zde představují datovou strukturu. U této datové struktury předpokládáme samopopisný charakter, protože obsahuje kromě dat i metadata, která popisují význam jednotlivých částí této datové struktury. Typickým příkladem takové datové struktury je formát JSON nebo XML. Tyto formáty obsahují stromové datové struktury a asociativní pole (název, hodnota), seznamy a základní datové typy. V této části se budu věnovat souborové databázi MongoDB, která je vhodná pro nasazení v IS organizace.

Souborová databáze je založena na volném datovém modelu, proto je možné v dokumentové databázi libovolně ukládat různorodé dokumenty. Je doporučené ukládat dokumenty stejného druhu společně, protože tento způsob ukládání dat zjednodušuje další manipulaci a zpracování dat. V dokumentové databázi lze uložená data v ideálním případě použít pro přímou komunikaci s ostatními komponentami IS, protože dokumentové formáty přirozeněji odpovídají struktuře tříd objektového programování a to vzájemně zjednodušuje jejich datovou konverzi (neboli objektově dokumentové mapování ODM).[11, 18, 20, 26]

MongoDB nemá omezení datového schématu pro dokumenty v kolekci, kdy využívá jedinečného ID, který slouží jako primární klíč v kolekci. Toto ID (primární klíč) je obvykle vygenerováno při ukládání dokumentu do databáze MongoDB. Zobecníme-li pohled na primární klíč, jde o typ databáze klíč-hodnota, kdy v hodnotě jsou ukládány JSON dokumenty, ale MongoDB tyto klíče využívá sofistikovanějším způsobem.).[11, 18, 20, 26]

Budu se zabývat konceptem datového modelování v MongoDB, který je založen na kolekci, dokumentech, polích, datových typech, klíčích atd. z pohledu fyzického modelu. Pro pojmenovávání kolekcí, dokumentů, polí atd. v MongoDB se nesmí používat speciální znaky jako je tečka, čárka, znak dolaru, null atd. MongoDB rozlišuje v názvech kolekcí, dokumentů, polí atd., malá a velká písmena, proto je nutné zavést standardizaci pojmenovávání těchto a dalších datových struktur v organizaci. Dále nejsou povoleny duplicitní názvy ve stejné úrovní hloubky datové struktury, ale je povoleno použít stejný název v jiných úrovních datové struktury databáze MongoDB. [11, 18, 20, 26]

Dokument MongoDB je srovnatelný v prostředí RDBMS (relační databáze) se záznamem (jde o instanci entity na fyzické úrovni). Představuje obecný dokument (fakturu, dodací list, nabídku, stvrzenku, zprávu atd.). Dokumenty jsou složeny z polí (field), kdy dokument začíná a končí složenými závorkami {} a hranaté závorky nám definují pole []. Dokumenty se ukládají podle souvisejících dat, které často zobrazujeme společně do kolekcí.

Kolekce MongoDB je srovnatelná v prostředí RDBMS s tabulkou (entita na fyzické úrovni). Kolekce je sada jednoho nebo vice dokumentů. V MongoDB je možné definovat strukturu i data současně. Jde o dynamické schéma, které nám umožní díky své flexibilnosti tzv. přírůstkové změny pro snadné přidání dat, které jsme opomněli zahrnout do daného schématu v předchozí implementaci, nebo potřeba na jejich přidání vznikla později. Toto schéma je možné také využít pro testování různých struktur a na základě těchto testů vybrat ty nejlepší.

Atribut je základní informace důležitá pro obchodní (business) pohled, který identifikuje, popisuje nebo měření instance entity. Např. autor knihy je atributem, protože má obchodní význam a to ve všech způsobech záznamu, ať je to v papírové podobě, relační databázi např. Oracle nebo v MongoDB. Atribut fyzického datového modelu představuje v RDBMS sloupec databáze nebo v MongoDB pole.

Pole MongoDB představuje v RDBMS sloupec databáze. MongoDB pole je složeno ze dvou částí, a to z názvu pole a hodnoty pole. Kdy pole v MongoDB může obsahovat jednoduchou hodnotu nebo mnoho dalších polí nebo dokumentů a tak umožnuje více úrovňové ukládání dat. MongoDB má robustní možnosti dotazování uvnitř polí.

Doména v MongoDB, je úplná sada všech hodnot, které obsahuje atribut. Neboli atribut nemůže obsahovat hodnoty mimo přiřazenou doménu. Doména je tedy definovaná zadáním skutečného seznamu hodnot nebo souborů pravidel (např. doména pohlaví může nabývat hodnot žena,muž).

V relačních databázích využíváme tři typy domén – formát (omezuje délku a typ datového prvku např. character(15)), seznam (definuje seznam povolených hodnot) a rozsah (omezuje hodnoty datových prvků na interval od-do (např. datum zahájení – datum ukončení)).

MongoDB datový typ představuje v RDBMS datový typ.

MongoDB Parent Reference představuje v RDBMS Self Referencing neboli rekurzi na fyzické úrovni pomocí rodiče.

Primární klíč v MongoDB je obdobou primárního klíče v RDBMS a může být složen z jednoho nebo více atributů, které jsou jedinečné pro instanci entity. Primární klíč by měl obsahovat pouze atributy, které jsou potřeba pro specifikaci jednoznačné instance entity. Obecný příklad pro primární klíč “db.collection.ensureIndex({Attrib1:1},{“unique”:true})“, kde se využije vzestupné třídění atributu “Attrib1”, kdy “1” představuje vzestupné třízení a “-1” sestupné třízení. A následně klíč v nastavení unique = true musí mít záznamy v atributu “Attrib1” jedinečné.

MongoDB, dále využívá náhradní klíče (surrogate key), které nejsou spojeny s entitami, ale jsou jedinečnými identifikátory v řádku v tabulce (počítadlo). Tyto klíče jsou důležité pro vytvoření jediné konzistentní verze dat. Aplikace jako datové sklady obsahují data z více aplikací/systémů a náhradní klíče nám umožňují propojit tyto informace i když jsou v každém

zdroji identifikovány odlišně. Náhradní klíče mohou být globálně jedinečnými identifikátory GUID. Kdy GUID identifikátor je dlouhý a často náhodný.

Mongo ObjectID představuje v RDBMS GUID (náhradní klíč na fyzické úrovni) snižuje možnost duplicit. Každý dokument MongoDB je definován jedinečným polem "_id". Kolekce nemůže obsahovat dva dokumenty se stejným klíčem, ale v kolekci, která je např. označená A a B, mohou být dokumenty se stejným klíčem.

MongoDB reference představuje v RDBMS cizí klíč. Kdy je na jedné straně "mateřská entita" a na druhé straně je ve vztahu podřízená entita. V relačních databázích nám cizí klíč umožňuje přecházet z jedné tabulky do druhé. Cizí klíč a MongoDB reference poskytují způsob navigace z jedné struktury do druhé. Kdy cizí klíč v relační databázi zjišťuje podmínku, zda každá hodnota existuje také v primárním klíči = kontrola referenční integrity. Mongo reference je z tohoto pohledu pouze jednoduchým způsobem odkazu pro přechod do jiné struktury.

MongoDB nejedinečný index (Non-unique Index) představuje v RDBMS sekundární klíč. Tyto klíče se přidávají z důvodu zefektivnění načítání dat.

Kolekce skenování v MongoDB představuje dotaz v RDBMS. Pokud kolekce skenování nepoužívá žádný klíč tak server musí procházet všechna data a požadavek na data (dotaz) má velmi vysoké nároky na čas zpracování a výkon serveru.

Několik příkladů příkazů v MongoDB databázi:

Obecně lze popsat strukturu příkazů v MongoDB takto:

Ukazatel na aktuální databázi.kolekce.příkaz(…) …. db.collection.command(…)

- *db.collection.insert({document})* : insert = vložení nového dokumentu, {document} = identifikace vkládaného dokumentu.
- *db.collectio.update({criteria},{changes},{upset,multi}) :* update = změna dokumentů, {criteria}specifikace/výběr dokumentů, které se budou měnit, {changes}specifikace změn (co se bude měnit), {upset,multi} pokud podmínce změny bude vyhovovat vice jak jeden dokument tak tento příkaz provede aktualizaci všech dokumentů.
- *db.collection.remove({criteria}, justOne) :* {criteria} specifikace/výběr dokumentů, které budou vymazány,justOne odstraní document, který vyhovuje podmínkám kritérií.
- *db.collection.find({criteria}, {projection}) :* {criteria}specifikace/výběr dokumentů**,** které budou vyhledány, {projection} nám specifikuje, která pole by se měla ve výběru dokumentů zobrazovat**.**

Tabulka 8: Datový model v RDBMS vs. MongoDB.

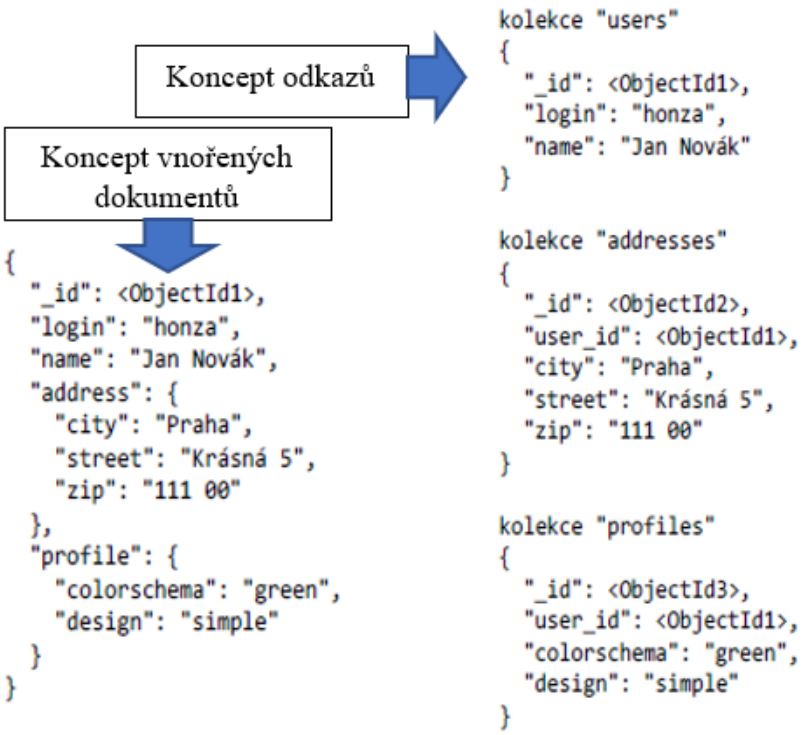
Datový model	RDBMS	MongoDB
Instance entity	Záznam/řádek	Dokument
Entity	Tabulky	Kolekce
Atributy / datové prvky	Datový prvek, pole, sloupec	Pole
Hodnota atributu /datového prvku	Pole	Hodnota pole
Doménový formát	Datový typ	Datový typ
Relační vztah	Omezení / nastavení vztahů mezi tabulkami	Zachyceno, ale nevynuceno ani prostřednictvím odkazu (obdoba cizího klíče) nebo prostřednictvím vložených dokumentů.
Primární klíč	Jedinečný klíč	Jedinečný klíč
Cizí klíč	Cizí klíč	Odkaz /Reference
Sekundární klíč	Sekundární klíč	Nejedinečný klíč (Non Unique index)

Zdroj : [Vlastní tvorba podle,11, 16, 18, 41]

V dokumentových databázích využíváme dva základní typy přístupů, které nám umožní rozdělit data do kolekcí dle metodologie NoSQL datových modelů. Jde o využití vnořených dokumentů (vnořených objektů) nebo odkazů.

Datový model s využitím vnořených dokumentů nám zajistí manipulaci se všemi daty kolekce v rámci jedné operace (zápis, aktualizace, čtení atd.). Ale tento model má omezení ve vztahu mezi nadřízeným a vnořeným dokumentem, kdy je podporován vztah mezi nadřízeným a vnořeným dokumentem 1:1 nebo 1:N. Takto definovaný model může mít v budoucnu problémy s nároky na výkon systému při manipulaci s daty. Neboli když vzroste počet vnořených dokumentů nad přijatelnou míru tak dochází k zpomalení výkonu systému.[11, 18, 20, 26]

Datový model s vnořenými objekty nám prováže dokumenty v dokumentové databázi pomocí odkazů s jejich generovanými jedinečnými "_id". Tento způsob odpovídá využití cizích klíčů v relačních databázích. Struktura dat v takto koncipovaném datovém modelu vytváří flexibilnější schéma, které odpovídá normalizovanému schématu dat v relačních databázích a je vhodná pro modelování vztahů M:N. Tento model má zamezit duplicitním uložením vnořených dokumentů (dat). Jistým omezením takto koncipovaného modelu je problematičtější vytváření modelu vlastní provázanosti, založené na závislosti dokumentů. [11, 18, 20, 26]



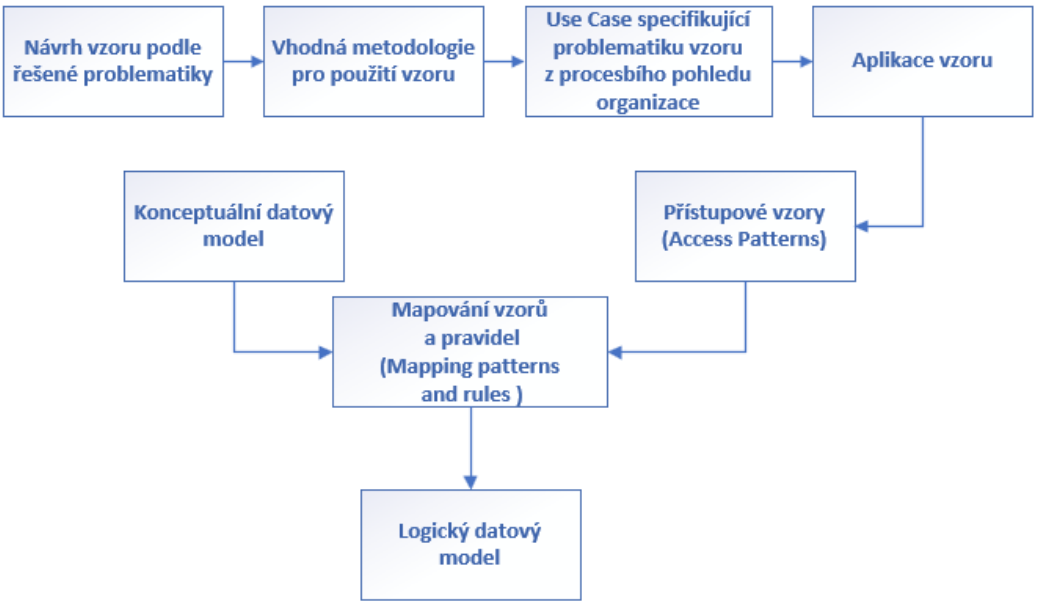
Obrázek 29: Příklad pro porovnání konceptu vnořených dokumentů a odkazů.

Zdroj : [Vlastní tvorba podle 20]

Mnoho pravidel v datovém modelu je možné zachytit pomocí vztahů, kdy vztah je v modelu čára spojující dvě entity, která zachycuje pravidlo nebo jen cestu mezi nimi. Kdy využití vnořených dokumentů s jejich možnou duplicitou je někdy výhodnější pro následné operace s daty. Proto se snažíme pro tuto problematiku využít principů denormalizace s nedodržením třetí normální formy a vytvořením žádoucích duplikací dokumentů, které nám následně velmi zefektivní vyhodnocovací a vizualizační procesy. Pro dotazy nad dokumentovými databázemi je velmi výhodné mít data společně uspořádaná podle logické potřeby (např. v kolekci) a velmi efektivně je využívat. Vyšší nároky na zpracování a uložení duplicitních dat jsou převáženy vyšší efektivitou ve vyhodnocovacích procesech.

Dokumentové databáze jsou vhodné pro procesy v organizaci, které jsou přímo navázány na reálné dokumenty využívané při činnostech organizace (např. faktury, články atd.). V těchto případech je možné využít výhod takto koncipovaného datového schématu, které jsou založeny na flexibilnosti, dynamičnosti a reálnosti (přímý vztah mezi modelem a realitou) datového modelu. Proto je tvorba datového modelu silně ovlivněna pravidly a zákonitostmi reálných procesů organizace než teoretickými pravidly pro správný návrh relační databáze.

Pro modelování v prostředí MongoDB jsou často využívány přístupové vzory, kdy je zobecněné vzorové řešení dané problematiky, naimplementováno s parametrickými modifikacemi na konkrétní řešený případ. Kdy datový model reprezentuje využití dat a vzor řešení pomocí šablon (template). Vzory mohou být typu vzorové řešení standardních opakujících se problematik při zpracovávání dat, dizajnové nebo pro modelování dat např. denormalizace dat, agregace dat, sloučení vnořených dokumentů atd.[11, 18, 20, 26]



Obrázek 30: Tvorba logického datového modelu s pomocí vzorů a pravidel.

Zdroj : [Vlastní tvorba podle 11, 16, 18, 20, 41]

Odbjednávky (kolekce)

{ID_Objednávky: '1', Číslo objednávky : '202100001', Odběratel:{ID_Odběratele:'1', Název odběratele:'xxx'}, Produkty :{ID_Produktu: '1', Název produktu:'ZZZ', Cena:'100.99'}}
{ID_Objednávky: '2', Číslo objednávky : '202100002', Odběratel:{ID_Odběratele:'2', Název odběratele:'xxx1'}, Produkty :{ID_Produktu: '1', Název produktu:'ZZZ', Cena:'100.99'}}
{ID_Objednávky: '3', Číslo objednávky : '202100003', Odběratel:{ID_Odběratele:'1', Název odběratele:'xxx'}, Produkty :{ID_Produktu: '2', Název produktu:'ZZY', Cena:'1000.99'}}
{...}

Odběratel (kolekce)

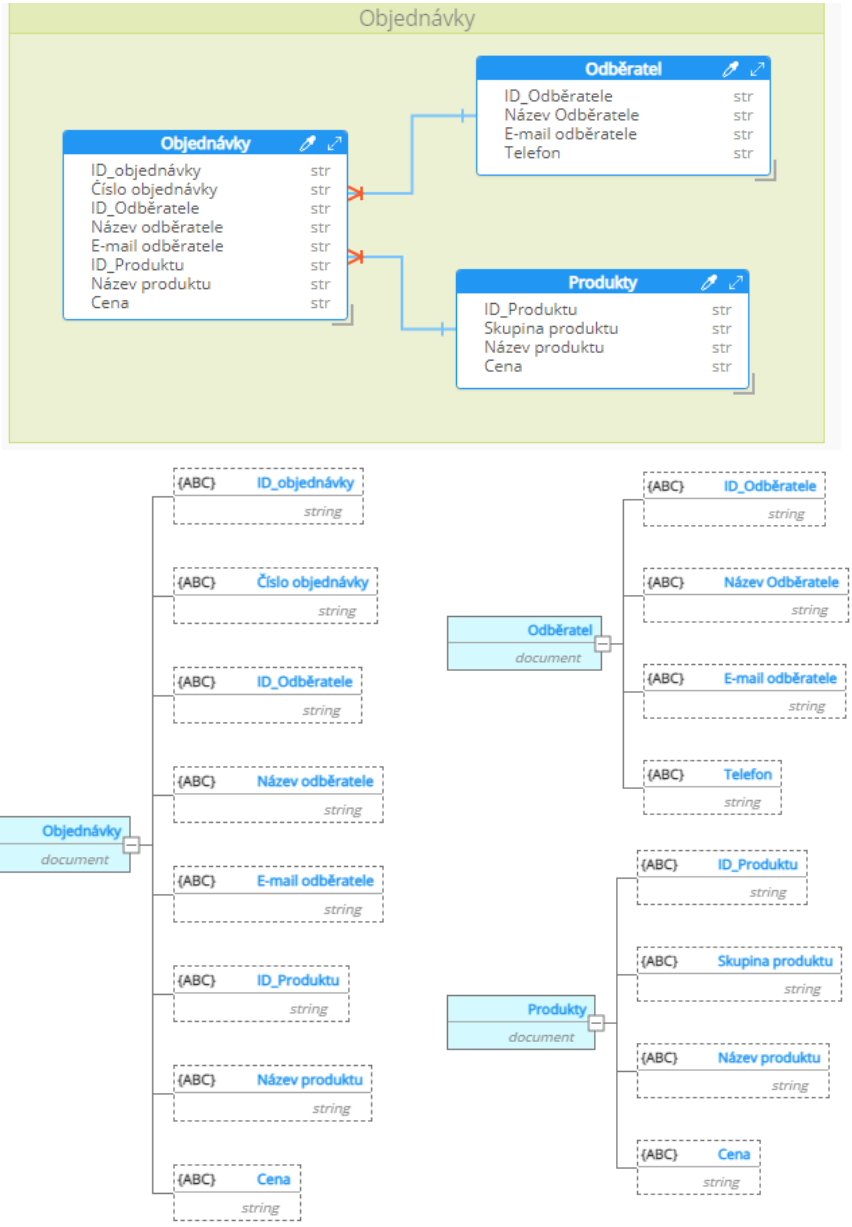
{ID_Odběratele:'1', Název odběratele:'xxx', E-mail odběratele:'xxx@xxx.com',Telefon:'123'}
{ID_Odběratele:'2', Název odběratele:'xxx1', E-mail odběratele:'xxx1@xxx.com',Telefon:'456'}
{...}

Prokukty (kolekce)

{ID_Produktu: '1', Název produktu:'ZZZ', Cena:'100.99'}
{ID_Produktu: '2', Název produktu:'ZZY', Cena:'1000.99'}
{...}

Obrázek 31: Ukázka duplicity v kolekcích u příkladu Objednávky.

Zdroj : [Vlastní tvorba podle 11, 16, 18, 20, 41]



Obrázek 32: Ukázka tvorby modelu u příkladu Objednávky.

*Zdroj : [Vlastní tvorba v **Hackolade Studio 4.3.13 trial version** podle 11, 16, 18, 20, 41]*

4.5.4 Grafové databáze

Konstrukce grafové databáze je založena na množině uzlů, které jsou vzájemně propojeny hranami. Obecně jde o uložiště typu databáze, do kterého máme záměr uložit informace o reálném procesu organizace, jako např. do relační nebo jiné databáze. Je pravdou, že i do relační databáze jsme schopni persistentně uložit informace, které jsou reprezentované pomocí grafu. Bohužel takto uložené informace jsou zpracovávány s vysokými nároky na technické řešení (např. z pohledu výkonu a rychlosti odezvy při dotazech na data).

Z pohledu datového modelování proces tvorby grafů obsahuje množinu objektů a množinu vztahů mezi nimi. Kdy uzel v grafové databázi odpovídá jednomu objektu, který může obsahovat množinu atributů (vlastností). Hrany představují vztahy mezi objekty a každý objekt může mít několik hran (vztahů) s dalšími objekty. Hrany mohou také obsahovat atributy s podmínkami platností definovaných vztahů. Vlastní obecná reprezentace grafů v grafových databázích je založena na definici dvojic G = (V,E), kde "V" je množina uzlů (vrcholů) a "E" je množina hran, která odpovídá podmínkám "E = Vx*V", kde je hrana vymezena dvojicí uzlů "(v1,v2) ∈ E ⋀ v1,v2 ∈ V". Počet uzlů obvykle označujeme jako "n" a počet hran jako "m".

Pro uložení grafu je možné využít několik datových struktur : [20, 35, 37]

- **Matice souslednosti:** Jde o dvourozměrné pole "A", které obsahuje " n" x "m" Booleovských hodnot (0,1), kdy hodnota "1" na pozici A_{ij} nám identifikuje stav, kdy mezi uzly "i" a "j" existuje hrana. Pokud "A" nabývá hodnoty "0" tak hrana neexistuje. Konstrukce takového grafu může mít neorientované hrany nebo hrany mohou mít určité váhy, pak jde o vážený graf, kdy místo hodnot <0,1> jsou u hran uváděny hodnoty jednotlivých vah.

- **Seznam sousedů:** Jde o datovou strukturu, která odpovídá vektoru "n" ukazatelů na seznam sousedních uzlů s kterými má výchozí uzel nějaký vztah (hranu grafu). Tato reprezentace grafů je velmi často využívána pro analýzy pravidelnosti grafu, opakovatelnosti částí jeho struktur nebo pro porovnání rozdílů mezi přesně definovanými strukturami grafové databáze.

- **Matice incidence:** Jde zde o dvourozměrné pole Booleovských hodnot ve formátu "n" x "m" (řádek, sloupec). Sloupce této matice reprezentují hrany a hodnota "1" označuje uzly, které danou hranu tvoří. Kdy řádek této matice reprezentuje uzel a hodnota "1" je přiřazována každé hraně, která náleží k danému uzlu.

- **Laplaceova matice:** Jde zde o dvourozměrné pole hodnot ve formátu ”n” x “n” (čtvercová matice). Kdy na diagonále této matice je uveden stupeň příslušného uzlu (neboli počet hran daného uzlu). Pokud do dalších pozic matice pokud mezi uzly vede hrana, zapisujeme hodnotu “-1”, a pokud mezi uzly hrana nevede, tak zapisujeme hodnotu “0”. Výhodou tohoto řešení je možnost provádění složitějších matematických operací, pomocí kterých je možné analyzovat strukturu daného grafu.

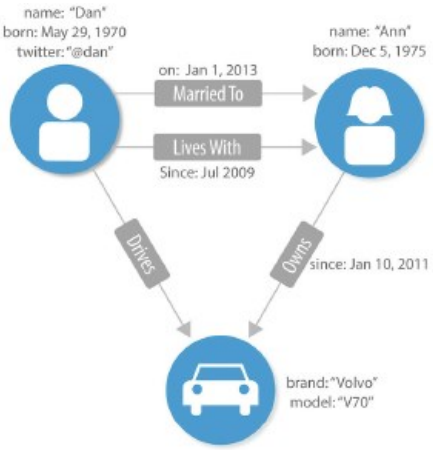
Důležitým klasifikačním kritériem grafové databáze je, jaké typy grafů je schopna využívat. Zda grafy orientované, neorientované nebo oba typy. Kdy při průchodu grafem je možné stanovit délku na základě počtu hran, kterými jsme prošli od počátečního do koncového uzlu cesty. V orientovaném grafu je možné rozlišovat cesty např. ve směru orientace hran. Grafy se člení na jednovztahové a vícevztahové (ohodnocené - labeled) grafy. U jednovztahových grafů jsou hrany stejného typu (homogenní) nebo mohou být nespecifikované. Vícevztahové grafy, které mají různé typy hran, nám vytvářejí multigrafy. Graf je multigrafem, když mezi dvěma uzly vede více hran různých typů. Pokud máme případ, že hrana spojuje dva uzly, které mohou být totožné, jde o smyčku. Pokud hrana vytváří vztah mezi více jak dvěma uzly, tak dochází k vytváření hypergrafů. Dalším typem grafů mohou být atributové grafy, které mají uvedeny atributy u uzlů a hran, jak jsem již výše uvedl. [20, 35, 37]

Databáze grafů jsou obecně konstruovány pro podporu transakčních systémů OLTP, ale bez problému dokáží podporovat i OLAP transakční systémy. Výkon grafové databáze na rozdíl od relační databáze je v závislosti na růstu datové sady relativně konstantní. Důvodem je, že dotazy jsou lokalizované obvykle jen na část grafu a časová náročnost na jejich provedení odpovídá části grafu, kterou jsme pro jejich provedení prošly. Grafy jsou přirozeně aditivní, proto je možné přidávat nové uzly, vztahy (druhy vztahů), nové atributy vztahů/uzlů, nové podgrafy k již vydefinované struktuře bez rušení existujících dotazů a funkcionalit aplikace. Flexibilní model grafové databáze nám umožní, že nemusíme podrobně definovat domény datového modelu a společně s aditivní povahou této technologie je možné snížit režii a rizika údržby tohoto úložiště. Vytvořený datový model grafové databáze je vhodný pro dnešní rychle se měnící obchodní prostředí, kdy daný model je možné pružně modifikovat a vytvářet tak různé simulační testy pro nasazení optimálního řešení do produktivního IS sytému organizace.

Výrazným představitelem grafových databází je Neo4j, která je implementována na technologii JAVA, a proto je přenositelná mezi systémovými platformami. Kdy z pohledu technologie Big Data má omezenou škálovatelnost, protože nedokáže vytvořený graf distribuovat a využívá

transakční vlastnosti ACID. Datový model Neo4j je založen na standardním využitím uzlů a hran. Kdy typ hrany je určen svým názvem a spojuje pouze dva uzly. Proto Neo4j může vytvářet grafy a multigrafy. Hrany jsou vždy orientované, a proto musíme rozlišovat, zda se jedná o počáteční nebo koncový uzel hrany. V případě, že pro naše řešení není nutné použít orientovaný graf, je možné orientaci hran ignorovat, protože algoritmy tohoto systému dokáží procházet graf po směru i proti směru orientace jeho hran. Uzly i hrany mohou obsahovat atributy. Atribut je vytvářen pomocí klíče a hodnoty. Klíč je řetězec a hodnota může být jednoduchý datový typ nebo i pole hodnot dle standardních datových typů jazyka JAVA. V Neo4j na rozdíl od relačních databází není nadefinována speciální prázdná hodnota „null“. Pro vyjádření, že daný atribut nemá hodnotu (klíč, hodnota) jej prostě neuvedeme. Neo4j v dotazech může vracet informaci o neexistenci nějakého atributu tedy, že jeho hodnota je „null“. [20, 35, 37]

Ukázkový jednoduchý příklad konstrukce uzlů a vztahu mezi nimi z pohledu grafové databáze s atributy uzlů i vazeb v provedení multigraf. Neboli Dan narozen 29.05.1970 má účet na Twitter @dan, žije od 07.2009 s Ann, a je ženatý od 01.01.2013 s Ann narozenou 5.12.1975. Dan řídí uvedené auto Volvo V70. Ann vlastní uvedené auto Volvo V70 od 10.01.2011. Atributy u každého uzlu a vztahu nám upřesňuje zobrazený datový model.[35]



Obrázek 33: Modelový příklad grafové databáze.

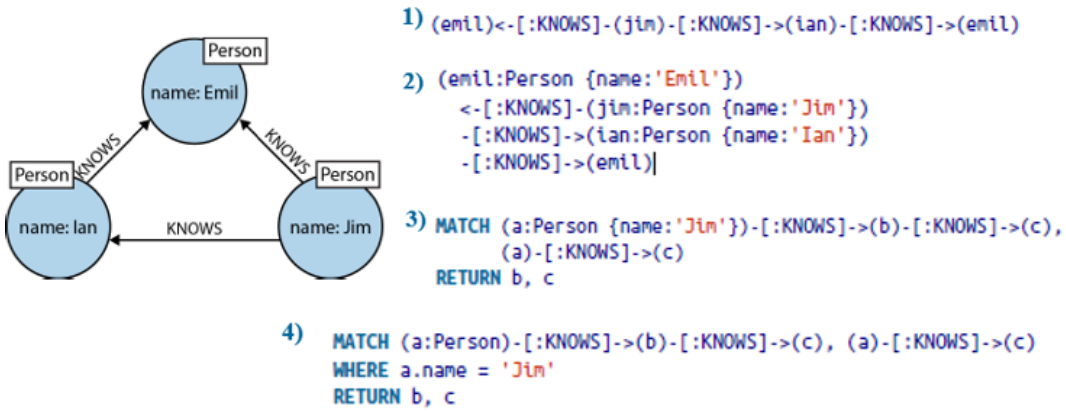
Zdroj : [35]

Cypher je dotazovacím jazykem v Neo4j, který umožní uživatelům ukládat a načítat (manipulovat s daty) data z databáze grafů. Cypher je založen na konceptu snadné použitelnosti a srozumitelnosti pro každého uživatele. Jde o deklarativní jazyk inspirovaný SQL, který je určen pro popis vizuálních vzorů v grafech pomocí syntaxe ASCI-Art. Pomocí kterého mohou uživatelé vytvářet expresivní a efektivní dotazy pro vytváření, čtení, aktualizaci a mazání dat.

Cypher není pouze velmi dobrým způsoben interakce s daty a Neo4j, ale je také open source technologií, která je k dispozici široké veřejnosti. [35, 37]

Cypher klauzule: [35, 37]

- **MATCH:** Tato klauzule specifikuje směr postupu přes množinu uzlů a vztahů systémem od – do, kdy klíč-hodnota je vlastností uzlu a vztahu a jsou definované uvnitř složené závorky (obdobně jako objekt v Java-scriptu).
- **RETURN:** Určuje, které uzly a vlastnosti v datech by měly být navráceny uživateli.
- **WHERE:** Poskytuje kritéria pro filtrování výsledků, porovnáním vzorů.
- **CREATE a CREATE UNIQUE :** Slouží pro vytvoření nových uzlů a vztahů.
- **MERGE:** Zjistí, zda v grafu existuje zadaný vzor (obsahující dané uzly a vztahy), a pokud odpovídá zadaným predikátům, umožní jeho použití nebo vytvoří nový.
- **DELETE :** Odebere uzly vztahy a vlastnosti.
- **SET:** Nastaví hodnoty vlastností.
- **FOREACH :** Provede aktualizaci každého prvku v seznamu.
- **UNION:** Sloučí výsledky ze dvou nebo více dotazů.
- **WITH:** Předá výsledky z části dotazu do další části – zřetězení dotazů.
- **START:** Určuje jeden nebo více počátečních uzlů nebo vztahů.



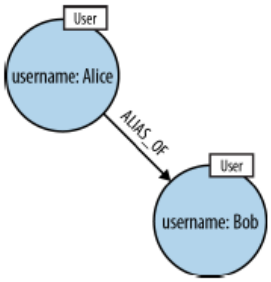
Obrázek 34: Modelový příklad proNeo4j.

Zdroj : [37]

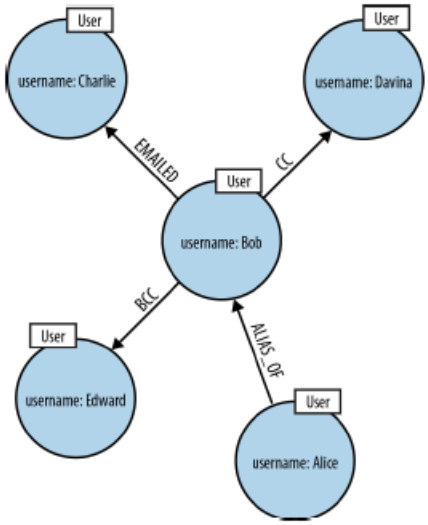
Na obrázku č.34 je popsán modelový příklad vyjádření vztahu mezi uzly Jim, Emil a Ian čtyřmi různými způsoby syntaxe, kdy způsob 3) a 4) je možné považovat za vzor, který po zobecnění může být opakovaně použit pro dotazy v strukturách grafové databáze.

Obrázek č.34 popisuje osobu Jim, která zná Emila, zná také Iana a Ian zná Emila. [37]

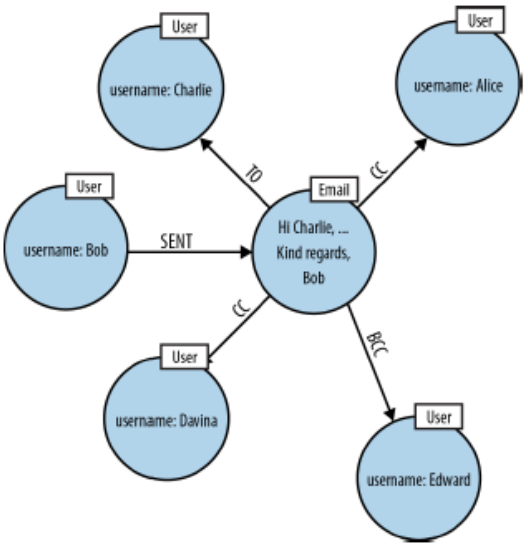
```
CREATE (alice:User {username:'Alice'}),
       (bob:User {username:'Bob'}),
       (charlie:User {username:'Charlie'}),
       (davina:User {username:'Davina'}),
       (edward:User {username:'Edward'}),
       (alice)-[:ALIAS_OF]->(bob)
```




```
MATCH  (bob:User {username:'Bob'}),
       (charlie:User {username:'Charlie'}),
       (davina:User {username:'Davina'}),
       (edward:User {username:'Edward'})
CREATE (bob)-[:EMAILED]->(charlie),
       (bob)-[:CC]->(davina),
       (bob)-[:BCC]->(edward)
```




```
CREATE (email_1:Email {id:'1', content:'Hi Charlie, ... Kind regards, Bob'}),
       (bob)-[:SENT]->(email_1),
       (email_1)-[:TO]->(charlie),
       (email_1)-[:CC]->(davina),
       (email_1)-[:CC]->(alice),
       (email_1)-[:BCC]->(edward)
```



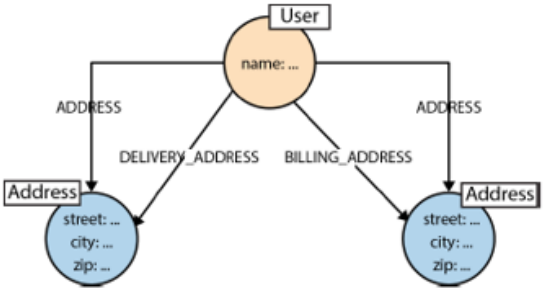
Obrázek 35: Modelový příklad aditivnosti v Neo4j složený ze tří kroků.

Zdroj : [37]

Na obrázku č.35 je popsán příklad aditivnosti v Neo4j, kdy vytvoříme v prvním kroku čtyři uzly a vztah ALIAS_OF z uzlu Alice do uzlu Bob. Ostatní uzly Charlie, Davina, Edvard nejsou zobrazeny, protože jsou bez vztahu.

V druhém kroku je graf z prvního kroku rozšířen o uzly Charlie, Davina, Edvard a o vazby mezi Bob-EMAILED-Charlie, Bob-CC-Davina a Bob-BCC-Edward. Uzel Alice s vazbou ALIAS_OF z uzlu Alice do uzlu Bob zůstává z prvního kroku.

V třetím kroku dojde k modifikaci předchozího grafu podle nových požadavků, Bob posílá email s uvedeným obsahem vazbou SENT a následně tento email má vztah TO z Charlie a vztah CC s Davina a vztah CC s Alice a vztah BCC s Edward.[37]




```
MATCH (user:User {id:{userId}})
MATCH (user)-[:DELIVERY_ADDRESS]->(address:Address)
RETURN address

MATCH (user:User {id:{userId}})
MATCH (user)-[:ADDRESS]->(address:Address)
RETURN address
```

Obrázek 36: Modelový příklad různých typů vztahů mezi uzly vNeo4j.

Zdroj : [37]

Na obrázku č.36 zjistíme nejdříve pro daného userId (uživatele) z uzlu User doručovací adresu z uzlu Address a následně v druhé sekvenci příkazů zjistíme pro daného userId (uživatele) adresy v uzlu Address. V schématu grafu je naznačena další možnost, která není popsána v kódu, jde o zjištění fakturační adresy BILLING_ADDRESS v uzlu Address. [37]

4.6 Dílčí souhrn

Datové modelování je process, který nám umožní definici datových požadavků a funkcionalit, které by mělo obsahovat dané řešení problematiky. Datové modelování je založeno na sběru požadavků od zadavatele (který by měl být dokonale obeznámen se všemi aspekty řešené problematiky) a následné kooperaci mezi zadavatelem a vývojářem při přípravě, vývoji a implementaci dané problematiky. Datové modelování pro strukturovaná data má v současnosti velmi efektivní nástroje jako je UML atd. založené na modelu relačních databází, který je ve své obecné koncepci využitelný téměř pro veškeré relační technologie. Pro NoSQL tato univerzálnost neplatí a je nutno vždy postupovat v souladu s technickými prostředky použitého NoSQL řešení, které po analýze vstupních informací dané problematiky využijeme.

5. PŘÍKLAD SBĚRU DAT A NÁVAZNÉHO MODELU

Návazný model je zaměřen na popis procesů plánovaného nákupu materiálu dle ISO 9000 pro výrobní i nevýrobní (např. organizaci z oblasti státní správy) organizaci. Kdy metodika nákupu materiálu dle ISO 9000 je založena na principu co nejefektivnějšího nákupu. Efektivitu nákupu zajistíme pomocí srovnání nabídek oslovených dodavatelů. Metodika nákupu materiálu dle ISO 9000 je založena na následných procesech poptávky, nabídky, vyhodnocení nabídky, objednávky a dodávky požadovaného materiálu v daném množství, dohodnuté cenové hladině s požadovanou úrovní jakosti a v požadovaném termínu dodání.

Model popisuje plánované nákupní procesy v organizaci, kdy vznikne požadavek (např. ve výrobě) na materiál, který je nutný pro zajištění chodu organizace nebo pro její výrobní procesy. Organizace má implementovaný systém ISO 9000 (a popřípadě další ISO atd.) a proto jsou její veškeré procesy řízeny a monitorovány dle vypracovaných metodik, postupů a interních směrnic atd. V organizaci jsou zavedeny materiálové specifikace, které specifikují pomocí parametrů (např. jakosti) požadované vlastnosti každého používaného materiálu. Každá vlastnost (parametr) má definovány akceptovatelné intervaly hodnot. Pro materiály od různých výrobců, které je možno zaměňovat, jsou vytvořeny substituční skupiny. Vlastní tvorba substitučních skupin je v daném systému povinná. Každý materiál musí být přiřazen do substituční skupiny (substituční skupina může obsahovat 1 až n materiálů), proto je možné provádět plánovaný nákup na substituční skupinu materiálů, což je výhodné, jak popíši v další části.

Pro účely plánovaného nákupu jsou udržovány číselníky se substitučními skupinami materiálů, s materiály, se specifikacemi materiálů a s dodavateli. Organizace má v svém managementu jakosti vypracována pravidla pro kvalifikaci materiálů od různých dodavatelů, které splňují nejen jakostní, účelové, technologické a cenové požadavky. Kvalifikace materiálů v organizaci je proces, který na základě kontrol a testování (např. malosériové výrobní testy s využitím nového materiálu, který prochází procesem kvalifikace) stanoví, zda je materiál pro procesy organizace vhodný či nikoliv. Kvalifikované materiály jsou založeny v číselníku materiálů s přiřazením specifikace materiálu, informacemi o dodavateli (adresa, atd.) a následně jsou přiřazeny do příslušné substituční skupiny.

Vzniklý požadavek materiálové potřeby (např. nová výroba dle výrobního plánu atd.) je řešen v prvním kroku dotazem na skladové zásoby materiálu v organizaci. Pokud je materiál dostupný, může být pro danou materiálovou potřebu zarezervovaný, pokud ne, je nutné jej

objednat tak, aby byl dodán v požadovaném termínu, aby mohl být předán pro uspokojení potřeb daných oddělení v organizaci. Dle plánů předpokládané spotřeby materiálu jsou vytvářeny skladové zásoby (např. dle výrobního programu atd.)

Plánované objednání požadovaného materiálu např. pro doplnění skladových zásob je realizováno pomocí vytvoření plánu nákupu materiálů na substituční skupinu daného materiálu. Kdy plán nákupu obsahuje požadované množství materiálu a datum požadovaného dodání materiálu. Využitím substituční skupiny je zajištěna efektivita nákupu, protože jsou následně vygenerovány položky daného plánu nákupu materiálu, které odpovídají záznamům přiřazených materiálů pro dané substituční skupiny (např. substituční skupina materiálů obsahuje tři materiály, na základě této konfigurace jsou založeny tři položky daného plánu nákupu materiálu.). Vygenerovaný počet položek plánu nákupu materiálu může pracovník nákupu zredukovat dle aktuálních potřeb organizace. Pro každou položku plánu nákupu materiálů je následně vygenerovaná poptávka na daný materiál. Tento způsob nám zajistí, že je možné oslovit všechny dodavatele, kteří mají daný materiál kvalifikován pro použití v procesech naší organizace a využít tak nejnižší cenové nabídky pro uspokojení našich potřeb. Samozřejmě, se splněním dodání daného materiálu do požadovaného datumu dodání, který je velmi důležitý pro zajištění procesů naší organizace (např. splnění smluvních závazků pro dodávky výrobků naší organizace atd.).

Každý dodavatel obdrží na předefinovanou adresu (e-mail) poptávku po materiálu, která specifikuje požadavek na materiál (materiálová specifikace), jeho množství a požadovaný termín dodání. Dodavatel vypracuje nabídku, která obsahuje podmínky z poptávky. Doplní ji o cenu a zašle ji na adresu nákupního oddělení (e-mail). Po shromáždění všech nabídek, které mohou odpovídat počtu zaslaných poptávek (závisí na dodavateli, zda má zájem nebo možnost dodat zboží v daném termínu atd.), provede pracovník nákupu organizace vyhodnocení nabídek pro daný plán nákupu, kdy do vyhodnocení jsou zařazeny pouze položky plánu nákupu materiálu, ke kterým byly zaslány nabídky tak, aby byly splněny podmínky standardu ISO 9000.

Je vyhodnocena optimální nabídka, která splňuje požadované množství, cenový rozsah, jakostní parametry a datum dodání. Ta je následně přiřazena k danému plánu nákupu materiálu. Pokud by ani jedna nabídka nesplňovala požadavky poptávky, je nutné přikročit k specifickému postupu, kterým se vzhledem k zaměření této práce již nebudu zabývat, protože jde zde jen o modelový příklad.

Následně je vytvořena objednávka materiálu (která obsahuje veškeré náležitosti dle obchodních zvyklostí v ČR) pro dodavatele, který byl vybrán ze zaslaných nabídek. Pokud objednávku dodavatel potvrdí, potom vyčkáme na její realizaci. Realizace je vlastní dodávka materiálů v požadovaném množství, úrovni jakosti, ceně a v požadovaném termínu. Dodávka materiálu je upřesněna dle obchodních zvyklostí v dodacím listu, certifikátu jakosti a faktuře (podmínky platby).

Po dodání materiálu provedeme porovnání mezi požadovaným množstvím a skutečně dodaným. Pokud je požadavek splněn, je možné daný plán nákupu uzavřít. V případě, že nejsou požadavky na množství materiálu zcela uspokojeny, je nutno postupovat dle metodik v dané organizaci (touto problematikou se dále v této práci nebudu zabývat z důvodu využití této problematiky jen pro jednodušší modelový příklad).

Daná aplikace bude provádět i sběr dokumentů jako je poptávka, nabídka, objednávka, dodávka (dodací list, certifikát jakosti, faktura atd.).

Nashromážděná data z plánů nákupu materiálu budou ukládána do archívu a následně budou využity pro analýzy a reporting v organizaci. Tato data budou využita pro zvýšení efektivity nákupu materiálu (např. plánování materiálových potřeb v časové linii, se zaměřením na nákupní procesy při, kterých je možno daný materiál získat za nižší cenu atd.).

5.1 Příklad sběru dat u návazného modelu

Popisovaná problematika je založena na sběru dat ze zrealizovaných plánů nákupu materiálů, které obsahují strukturovaná, semistrukturovaná a nestrukturovaná data. Proto je výhodné využít pro danou problematiku metodiky pro Big Data (datová různorodost a větší objem dat).

Strukturovaná data jsou uložena v tabulkách popisovaného řešení a obsahují veškeré důležité informace o zrealizovaných plánech nákupu materiálů, které jsou generovány danou aplikací.

Poptávka a nabídka je vázána na položku plánu nákupu materiálů a obvykle tyto dokumenty mají semistrukturovanou část (např. množství atd.), ale i nestrukturovanou část, která specifikuje komunikaci s dodavatelem a může být velmi důležitá pro vytváření budoucích obchodních vztahů.

Objednávka je vázaná na plán nákupu materiálu, obsahuje semistrukturovanou část (např. množství atd.) a nestrukturovanou část, která obsahuje komunikaci s dodavatelem a je zaměřena na realizaci daného obchodního případu.

Dodávku obvykle provázejí semistrukturované dokumenty, jako je dodací list, atest jakosti, faktura atd. Ale může obsahovat i např. průvodní dopis, ve kterém mohou být důležité obchodní informace.

Pro sběr dat v souborovém formátu, která jsou generována v průběhu realizace plánu nákupu materiálů, je možné použít relační technologie (databáze). Tyto relační technologie budou využívat sofistikovaného programového rozhraní, které bude extrahovat data z poptávek, nabídek, objednávek, dodávek (dodací list, certifikát jakosti a faktura) a následně je bude ukládat do databázové struktury k ostatním datům dané aplikace. Tento způsob lze využít při nižších objemech dat a následného nevyužití potenciálu textových nestrukturovaných dat v souborové části aplikace (data již byla extrahována v předdefinované struktuře, kterou lze obtížně měnit).

Další možností je využití relačních technologií pro práci s finálními daty a pro sběr, zatímco pro správu a uložení dat v souborovém formátu je efektivnější využít NoSQL technologii jako je souborová databáze (např. MongoDB). Kdy tato databáze může být konstruována jako distribuované úložiště, které využívá technologie Big Data, pro ukládání a správu veškerých souborů vznikajících v procesu plánování nákupu materiálů pro danou organizaci. Do relační databáze se budou ukládat pouze vybrané extrahované údaje ze souborů jako nabídka, dodací list, certifikát jakosti, faktura atd. (např. množství, cena atd.). Rozdělení systému na relační a No-SQL část způsobí vyšší nároky na dotazovací procesy, které tak budou při extrakci dat z No-SQL a relačního prostředí komplikovanější.

V případě, že požadujeme sofistikovanější využívání textových částí uložených souborů k plánům nákupu materiálů nebo při větším objemu dat, je efektivnější pro danou problematiku sběru dat využít NoSQL technologii s naimplementovanými vzory pro zpracovávání Big Data. Pro tyto účely je možné využít NoSQL databázového systému s distribuovaným úložištěm jako je např. souborová databáze MongoDB atd. Data jsou zde ukládána v kolekcích podle předpokládaného způsobu použití v dotazech a porušují tak pravidla 3 normální formy, protože vznikají duplicity, které však velmi zrychlí dotazování a analýzu takto uložených dat.

5.2 Příklad návazného modelu

Pro popis návazného modelu plánovaného objednání požadovaného materiálu jsem použil standardní metodiky jazyka UML, které nám umožní nejdříve obecný pohled na danou problematiku od základních procesů, přes případy užití, koncepční model, logický model,

fyzický model, až k specifikaci ER diagramu s konceptem agregací, která vytváří logické datové celky mezi entitami. Takto specifikované logické datové celky jsou základem pro datové struktury využívané v NoSQL technologiích.

- Business case diagram:. ……podrobnosti příloha A
- Use Case diagram:…….podrobnosti příloha B
- Use Case scénář vybraného případu užití (Dodavatel):……podrobnosti příloha C
- Sekvenční diagram pro vybraný případ užití (Dodavatel):.. podrobnosti příloha D

Pro strukturovaná data – relační model:

- Koncepční model:……podrobnosti příloha E
- Logický model:……podrobnosti příloha F
- Fyzický model:……podrobnosti příloha G
- ER diagram s konceptem agregací…….podrobnosti příloha H

Pro Semistrukturovaná a nestrukturovaná data - NonSQL model :

- Data flow diagram:……podrobnosti příloha CH
- UML Objektový diagram…….……..…podrobnosti příloha I
- Myšlenkovou mapu (Mind map):……..podrobnosti příloha J
- CRD (collection relationship diagram) - MongoDB:.....podrobnosti příloha K
- MongoDB kód pro CRD z přílohy K ……podrobnosti příloha L
- Schéma modelu grafové databáze …….podrobnosti příloha M

Prvním krokem pro NoSQL datové modelování je identifikace entit a atributů. Následně je nutné stanovit postup, které entity jak seskupit, kdy je žádoucí aby toto seskupení vycházelo z modelu, jak se daná data využívají. Protože správné seskupení je důležité pro datové procesy v distribuovaném úložišti. Proto je důležité při modelování Big Data porozumět zpracovávaným datům a používané databázové struktuře s využitím agregace a denormalizace dat. Důležité je také správné seskupení datových entit s využitím datových vzorů, jako je např. agregace, denormalizace, připojení aplikace, stromová agregace, vnoření dokumentu atd.

ZÁVĚR

Práce byla zaměřena na charakterizování typů dat v organizaci, toku dokumentů v organizaci se zaměřením na uplatnění nestrukturovaných dat a na porovnání využití strukturovaných a nestrukturovaných dat v organizaci. Pro praktičtější část této práce byl vytvořen návrh postupu sběru dat s návazným modelem pro následné využití Big data v organizaci.

Nejdříve jsem se zaměřil na charakteristiku dat v organizaci dle typů (strukturovaná, semistrukturovaná, nestrukturovaná), podle místa vzniku a na popis charakteristiky „kvalita dat“. Dále jsem se zaměřil na data management a tok dokumentů v organizaci, kde jsem popsal datovou strukturu dokumentu. A jak je tato struktura využívaná v IT prostředcích organizace.

V oblasti Big Data jsem se věnoval charakteristice základních vlastností při pohledu na Big Data 6V a 10V. V další části jsem popsal otevřený standard pro každou organizaci Big Data Framework, který je souhrnem informací, jak implementovat Big Data do procesů organizace. Následně jsem uvedl příklady metod zpracování nestrukturovaných dat, jako jsou umělé neuronové sítě, strojové učení, analýzy textu, analýzy zvuku, analýzy digitálního snímku a analýzy videa. Popsal jsem některé technologie pro uskladnění Big Data, DataMinig pro Big Data, technologie pro analýzu a vizualizaci Big Data.

V další části této práce jsem popsal rozdíly mezi strukturovanými, semistrukturovanými a nestrukturovanými daty. Popsal jsem přínos uplatnění Big Data a rizika při využití Big Data.

Při popisu modelování Big Data jsem se věnoval základním úrovním datového modelování, a to z pohledu databázového systému jeho škálovatelnosti, distribuce, rozdělení, replikace a konzistence dat (ACID, CAP, BASE). V oblasti datového modelování jsem uvedl základní typy NoSQL databází (klíč-hodnota, sloupcové, dokumentové, graf). Podrobněji jsem se věnoval základním vlastnostem dokumentové databáze MongoDB s porovnáním k relační databázi a grafové databázi Neo4j.

Pro praktické znázornění modelování dat jsem vytvořil modelový příklad, se sběrem dat s návazným modelem, který jsem popsal na modelovém příkladu. Tento příklad je zaměřen na procesy plánovaného nákup materiálu v organizaci dle ISO 9000 s generováním poptávky, příjmem nabídky, vyhodnocením nabídek s vytvořením objednávky a dodávkou materiálu.

Pro popis výše uvedené části procesů organizace jsem využil standardně běžně používaných metodik modelování pro relační databáze. Kdy jsem popis modelu směřoval na obecný popis procesu s využitím Business case diagram, případy užití (Use Case diagram), Use Case scénář,

a to pouze pro vybraný případ užití (Dodavatel) a sekvenční diagram pro vybraný případ užití (Dodavatel). Následně jsem začal vytvářet relační model pomocí koncepčního, logického a fyzického modelu dané problematiky. Tuto část jsem zakončil ER diagramem s konceptem agregací, který slouží k určení logických datových struktur a je využíván jako vstupní popis datového modelování pro NoSQL databáze s využitím pro semistrukturovaná a nestrukturovaná data. Určení logických datových struktur je důležité pro porozumění datům. Pro správné porozumění datům jsem vytvořil Data flow diagram modelového příkladu, který je zaměřen na plánovaný nákup materiálu v organizaci dle ISO 9000. Doplnil jsem jej UML Objektovým diagramem a myšlenkovou mapou (Mind Map). Tyto prostředky jsem použil, protože porozumění datům a datové struktuře je důležité pro vybrání správného NoSQL softwarového řešení dané problematiky. Proto jsem vybral pro řešeného modelového příkladu NoSQL databázi MongoDB zaměřenou na uchovávání souborových systémů. Pro modelování v databázi MongoDB jsem použil software Moon Modeler. Kde jsem využil CRD (collection relationship diagram) pro MongoDB, který je založen na modelování s objekty jako je databáze, kolekce a soubor. S možnou tvorbou relací mezi kolekcemi. Zde je využita část MongoDB, kde jsou uchovávána metadata pro které lze využít techniky používané v ER modelech. Pro porovnání jsem vytvořil obecný model pro grafovou databázi Neo4j v software Hackolade (Omezená verze pro studenty), která využívá vlastností ACID.

Pokud známe strukturu dat a máme znalosti o databázové struktuře, na kterou je záměr aplikovat dané řešení, lze použít vhodné škálování dat a následně využít vhodnou agregaci s dekompozici datové struktury. Kdy použití dekompozice zvýší objem dat (jejich duplikacemi) a nároky na jejich správu, ale také velmi zefektivní zpracování dotazů a vizualizaci Big Data. Pro zpracovávání Big Data je dále důležité využití distribuce dat podle použitého No-SQL řešení, kdy jsou data škálována do množin atributů podle jejich využití. Vlastní použití rozdělení dat (sharding) s replikací dat je vždy specifikováno použitým softwarovým prostředkem (např. využití HDFS architektury v Apache Hadoop), kdy některé prostředky mají tyto procesy přímo integrované a u jiných je nutno tuto část aplikace naprogramovat.

V této diplomové práci jsem se zabýval otázkou, jak řešit obecný problém modelování dat pro semistrukturovaná a nestrukturovaná Big Data. Obvykle je tento problém řešen vhodnou modifikací datového digramu typu ER pro tu část dat, kterou lze takto využít (metadata) s návazností na ostatní data řešené problematiky.

Závěrem lze konstatovat, že cíl práce byl naplněn.

POUŽITÁ LITERATURA

[1] ARLOW, Jim a NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací*. Dotisk 1. vyd. Brno: Computer Press, a.s., 2011. 567s. ISBN 978-80-251-1503-9.

[2] ASHISH, Kumar, AVINASH, Paul. *Mastering Text Mining with R.* Birmingham, UK: Packt Publishing, 2016. 259 s. [cit. 2020-10-10]. ISBN 978-1-78355-181-1. Dostupné z: https://www.packtpub.com/catalogsearch/result?q=Mastering+Text+Mining+with+R

[3] BAGHA, Arshdeep, MADISETTI, Vijay. *Big Data Analytics: A Hands-On Approach.* USA: Arshdeep Bahga & Vijay Madisetti. 2019. 542 s. [cit. 2020-10-10]. ISBN: 978-1-949978-00-1. Dostupné z: https://book4you.org/book/3669607/48b4d6

[4] BAWDEN, David, ROBINSON, Lyn. *Úvod do informační vědy.* Vyd. 2. Český Těšín: Těšínská tiskárna a.s. 2017. 452s. ISBN: 978-80-88123-10-1.

[5] BIG DATA FRAMEWORK. *Enterprise Big Data Professional.* Bonn, Germany: Big Data Framework, 2018. 121 s. ISBN 978-90-828958-0-3.

[6] BIG DATA FRAMEWORK. *Enterprise Big Data Analyst.* Ver. 1.0. Bonn, Germany: Big Data Framework, 2019. 121 s. ISBN 978-90-828958-10.

[7] BUREŠ, Vladimír. *Znalostní management a proces jeho zavádění. Průvodce pro praxi.* Vyd. 1. Praha: Grada Publishing, 2007. 216 s. ISBN 978-80-247-6717.

[8] CAIBURRO, Giuseppe, JOSHI, Prateek. *Python Machine Learning Cookbook.* Second Edition. Birmingham, UK: Packt Publishing, 2019. 604 s. [cit. 2020-10-10]. ISBN 978-1-78980-845-2. Dostupné z: https://www.packtpub.com/product/python-machine-learning-cookbook-second-edition/9781789808452

[9] DAMA International. *DAMA-DMBOK, Data Management body of knowledge.* Vyd. 2. Basking Ridge, USA: DAMA International. 2017. 778 s. [cit. 2020-10-10]. ISBN: 978-1634622349. Dostupné z: https://book4you.org/book/5694489/aa584a

[10] DESHPANDE, Anand, KUMAR, Manish. *Artificial Intelligence for Big Data.* Birmingham, UK: Packt Publishing, 2018. 372 s. [cit. 2020-10-10]. ISBN 978-1-78847-217-3. Dostupné z: https://www.packtpub.com/free-ebook/artificial-intelligence-for-big-data/9781788472173

[11] EDWARD, Shakuntala Gupta, SABHARWAL, Navin. *Practical MongoDB.* California USA: Apress Media, 2015. 263 s. [cit. 2020-10-10]. ISBN 978-1-4842-0647-8. Dostupné z: https://book4you.org/book/2657293/ef9cd8

[12] FLECKENSTEIN, Mike, FELLOWS, Lorraine. *Modern Data Strategy.* Cham, Switzerland: Springer International Publishing AG, 2018. 269 s. [cit. 2020-10-10]. ISBN 978-3-319-68993-7. Dostupné z: https://book4you.org/book/3494505/53d43c

[13] FRYMAN, Lowell, LAMPSHIRE, Gregory, MEERS, Dan. *The Data and Analytics Playbook.* Cambridge, USA: Elsevier Inc. 2017. 275 s. [cit. 2020-10-10]. ISBN: 978-0-12-802307-5. Dostupné z: https://book4you.org/book/2769505/7cc187

[14] GÁLA, Libor, POUR, Jan a Zuzana ŠEDIVÁ. *Podniková informatika. Počítačové aplikace v podnikové a mezipodnikové praxi.* Praha: Grada Publishing, 2015. 240 s. ISBN 978-80-247-5457-4.

[15] GORELIK, Alex. *The Enterprise Big Data Lake.* Sebastopol, RU: O'Reilly Media, Inc. 2019. 218 s. [cit. 2020-10-10]. ISBN 978-1-491-93155-4. Dostupné z: https://book4you.org/book/3714632/11db78

[16] GUPTA, Sumit, SAXENA, Shilpi. *Real-Time Big Data Analytics.* Birmingham, UK: Packt Publishing, 2016. 299 s. [cit. 2020-10-10]. ISBN 978-1-78439-140-9. Dostupné z: https://book4you.org/book/2971340/9b6810

[17] HEATON, Jeff. *Deep Learning and Neural Networks.* Chesterfield, USA: Heaton Resaarch, Inc. 2015. 268 s. ISBN 978-1505714340 Dostupné z: https://book4you.org/book/2656808/e5ac18

[18] HOBERMAN, Steve. *Data modeling for MongoDB. Building Well-Designed and Supportable MongoDB Databases.* Basking Ridge, USA: Technics Publications, LLC, 2014. 252 s. [cit. 2020-10-10]. ISBN 978-1-935504-71-9. Dostupné z: https://book4you.org/book/2524049/c47c1d

[19] HOFMAN, Markus, CHISHOLM, Andrew. *Text Mining and Visualization.* Boca Raton, USA: CRC Press Taylor & Francis Group. 2016. 337 s. [cit. 2020-10-10]. ISBN 978-1-4822-3758-0. Dostupné z: https://book4you.org/book/2654319/b07be7

[20] HOLUBOVÁ, Irena, KOSEK, Jiří, MINAŘÍK, Karel a David NOVÁK. *Big Data a NoSQL databáze.* Praha: Grada Publishing, 2015. 288 s. ISBN 978-80-247-5466-6.

[21] ISSON, Jean Paul. *Unstructured Data Analytics.* Hoboken. New Jersey, USA: John Wiley & Sons, Inc, 2018. 432 s. [cit. 2020-10-10]. ISBN 978-1-119-32549-9. Dostupné z: https://book4you.org/book/3517554/df7504

[22] KHAN, Muhammad Usman Shahid, KHAN, Samee U. and Albert Y. ZOMAVA. *Big Data-Enabled Internet of Things*. London, UK: The Institution of Engineering and Technology. 2019. 492 s. [cit. 2020-10-10]. ISBN 978-1-78561-637-2. Dostupné z: https://book4you.org/book/5422945/dae45b

[23] KUNSTOVÁ, Renata. Efektivní správa dokumentů. Praha: Grada Publishing, a.s. 2011. 204 s. ISBN 978-80-247-6651-5.

[24] KWARTLER, Ted. *Text Mining in Practice with R.* Hoboken, USA: John Wiley & Sons Ltd, 2017. 310 s. [cit. 2020-10-10]. ISBN 978-1-119-28201-3. Dostupné z: https://book4you.org/book/2939431/370f70

[25] LEA, Perry. *Internet of Things for Architects.* Birmingham, UK: Packt Publishing, 2018. 515 s. [cit. 2020-10-10]. ISBN 978-1-78847-059-9. Dostupné z: https://book4you.org/book/3698164/005e4d

[26] LEE, James, WEI, Tao and Suresh Kumar MUKHIYA. *Hands-on Big Data Modeling.* Birmingham, UK: Packt Publishing, 2018. 306 s. [cit. 2020-10-10]. ISBN 978-1-78862-090-1. Dostupné z: https://www.packtpub.com/product/hands-on-big-data-modeling/9781788620901

[27] MEIER, Andreas, KAUFMANN, Michael. *SQL & NoSQL Databases.* Wiesbaden, Germany: Springer, 2019. 238 s. [cit. 2020-10-10]. ISBN 978-3-658-24549-8. Dostupné z: https://book4you.org/book/5214039/107cf0

[28] MAKHABEL, Bater, MISHRA, Pradeepta, DANNENMAN, Nathan, HEIMANN Richard. *R: Mining Spatial, Text, Web, and Social Media Data.* Birmingham, UK: Packt Publishing, 2016. 651 s. [cit. 2020-10-10]. ISBN 978-1-78829-374-7. Dostupné z: https://book4you.org/book/3558587/29cca9

[29] MARIN, Ivan, SHUKLA, Ankit and Sarang VK. *Big Data Analysis with Python.* Birmingham, UK: Packt Publishing, 2019. 256 s. [cit. 2020-10-10]. ISBN 978-1-78995-528-6. Dostupné z: https://www.packtpub.com/product/big-data-analysis-with-python/9781789955286

[30] MARSLAND, Stephen *Machine Learning : An Algorithmic Perspective.* Second Edition. Boca Raton, USA : CRC Press, 2015. 452s. ISBN 978-1-4665-8333-7 Dostupné z: https://book4you.org/book/2543746/ef80cb

[31] MAYER-SCHÖNBERGER, Viktor a Cukier KENNETH. *BIG DATA: Revoluce, která změní způsob, jak žijeme, pracujeme a myslíme.* Brno: Computer Press, 2014. 256 s. [cit. 2020-10-10]. ISBN 978-80-251-4119-9.

[32] MILLER, James D. *Big Data Visualization.* Birmingham, UK: Packt Publishing, 2017. 285 s. [cit. 2020-10-10]. ISBN 978-1-78528-194-5. Dostupné z: https://www.packtpub.com/product/big-data-visualization/9781785281945

[33] MINTEER, Andrew. *Analytics for the Internet of Things (IoT).* Birmingham, UK: Packt Publishing, 2017. 363 s. [cit. 2020-10-10]. ISBN 978-1-78712-073-0. Dostupné z: https://book4you.org/book/4990340/d43150

[34] NATARAJ, Dasgupta. *Practical Big Data Analytics.* Birmingham, UK: Packt Publishing, 2018. 389 s. [cit. 2020-10-10]. ISBN 978-1-78355-439-3. Dostupné z: https://www.packtpub.com/product/practical-big-data-analytics/9781783554393

[35] NEEDHAM, Mark, HODLER, Amy E. *Graph Algorithms*, Sebastopol, USA: Published by O'Reilly Media, Inc.,2019. 266 s. ISBN 978-1-492-04768-1

[36] RIORDAN, Rebecca M. Vytváříme relační databázové aplikace. 1.vyd. Brno: Computer Press, a.s., 2000. 280 s. ISBN 80-7226-360-9.

[37] ROBINSON, Ian, WEBBER, Jim, EIFREM, Emil, *Graph Databases*, . Second Edition, Sebastopol, USA: Published by O'Reilly Media, Inc.,2015. 238s. ISBN 978-1-491-93200-1

[38] RYŽKO, Dominik. *Modern Big Data Architectures*. New Jersey, USA: John Wiley & Sons, Inc, 2020. 199 s. [cit. 2020-10-10]. ISBN 978-1-119-59794-0. Dostupné z: https://book4you.org/book/5558752/1dd69e

[39] SAXENA, Shilpi, GUPTA, Saurabh. *Practical Real-Time Data Processing and Analytics.* Birmingham, UK: Packt Publishing, 2017. 422 s. [cit. 2020-10-10]. ISBN 978-1-78728-120-2. Dostupné z: https://book4you.org/book/3430479/e2c490

[40] SRIDHAR, Alla. *Big Data Analytics with Hadoop 3.* Birmingham, UK: Packt Publishing, 2018. 456 s. [cit. 2020-10-10]. ISBN 978-1-78862-884-6. Dostupné z: https://www.packtpub.com/free-ebook/big-data-analytics-with-hadoop-3/9781788628846

[41] SYED, Muhammad, FAHAD, Akhtar. *Big Data Architect's Handbook.* Birmingham, UK: Packt Publishing, 2018. 460 s. [cit. 2020-10-10]. ISBN 978-1-78883-582-4. Dostupné z: https://www.packtpub.com/product/big-data-architect-s-handbook/9781788835824

[42] ŠVARCOVÁ, Ivana, RAIN, Tomáš. *Informační management.* Praha: Alfa Naklada-telství, s.r.o., 2011. 183 s. ISBN 978-80-87197-40-0.

[43] TANWAR, Sudeep, TYAGI, Sudhanshu, KUMAR, Neeraj. *Multimedia Big Data Computing for IoT Applications.* Singapore, Singapore: Springer Nature Singapore Pte Ltd, 2020. 477 s. [cit. 2020-10-10]. ISBN 978-981-13-8759-3. Dostupné z: https://book4you.org/book/5247228/e11620

[44] TOMCY, John, PANKAJ, Mistra. *Data Lake for Enterprises.* Birmingham, UK: Packt Publishing, 2017. 585 s. [cit. 2020-10-10]. ISBN 978-1-78728-134-9. Dostupné z: https://book4you.org/book/3413406/e1c5b1

[45] TRIPATHY, B.K., ANURADHA, J. *Internet of Things (IoT).* Boca Raton, USA: CRC Press Taylor & Francis Group. 2018. 359 s. [cit. 2020-10-10]. ISBN 978-1-138-03500-3. Dostupné z: https://book4you.org/book/3419078/472f62

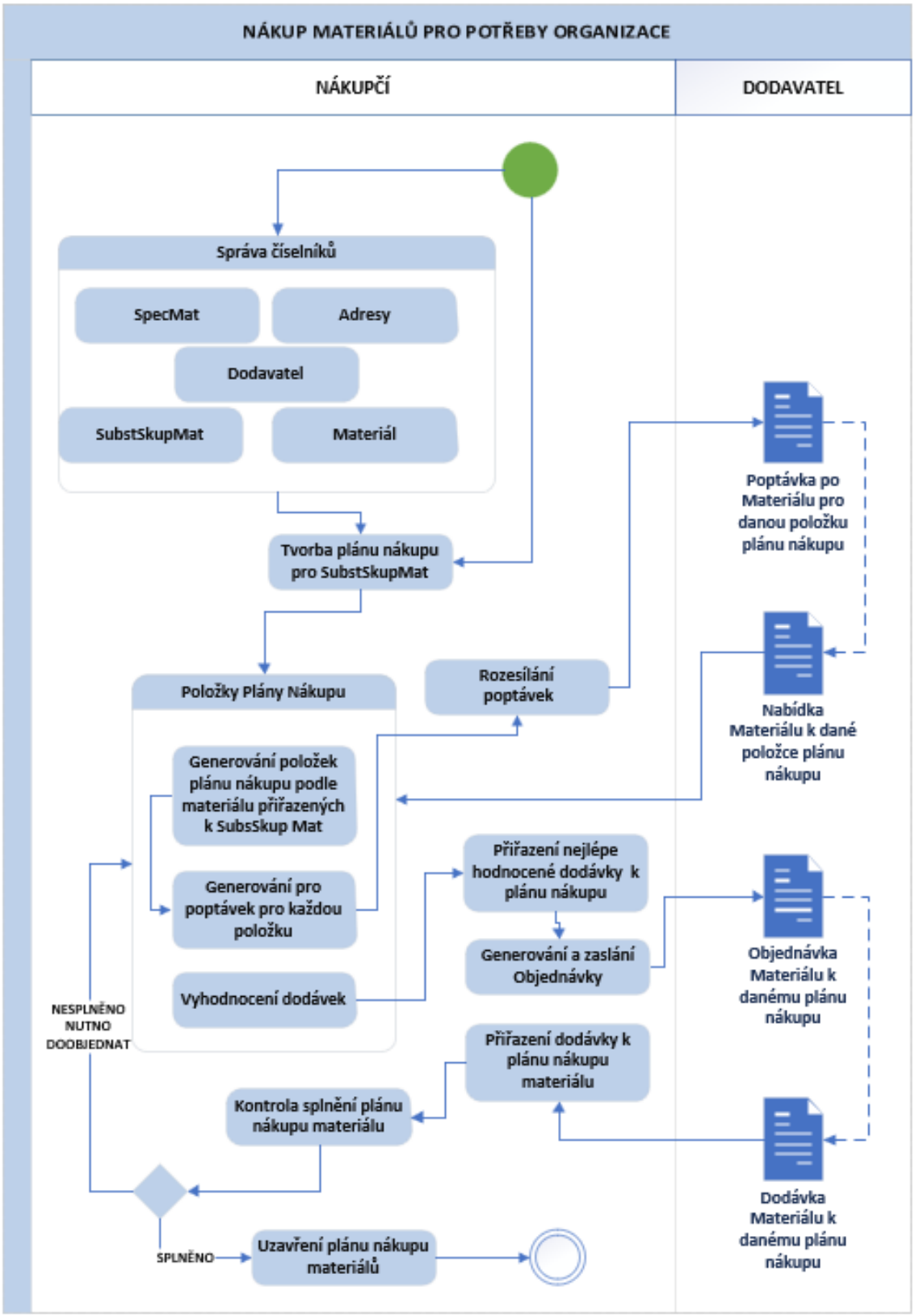
[46] TVRDÍKOVÁ, Milena. *Aplikace moderních informačních technologií v řízení firmy. Nástroje ke zvyšování kvality informačních systémů.* Vyd. 1. Praha: Grada Publishing, 2008. 176 s. ISBN 978-80-247-6298-2.

[47] WALKOWIAK, Simon. *Big Data Analytics with R.* Birmingham, UK: Packt Publishing, 2016. 485 s. [cit. 2020-10-10]. ISBN 978-1-78646-645-7. Dostupné z: https://www.packtpub.com/product/big-data-analytics-with-r/9781786466457

[48] ZHAI, ChengXiang, MASSUNG, Sean. *Text Data Management and Analysis.* San Rafael USA: Computing Machinery and Morgan & Claypool Publishers, 2016. 531 s. [cit. 2020-10-10]. ISBN: 978-1-97000-117-4. Dostupné z: https://book4you.org/book/2956688/aa4ae8

PŘÍLOHY

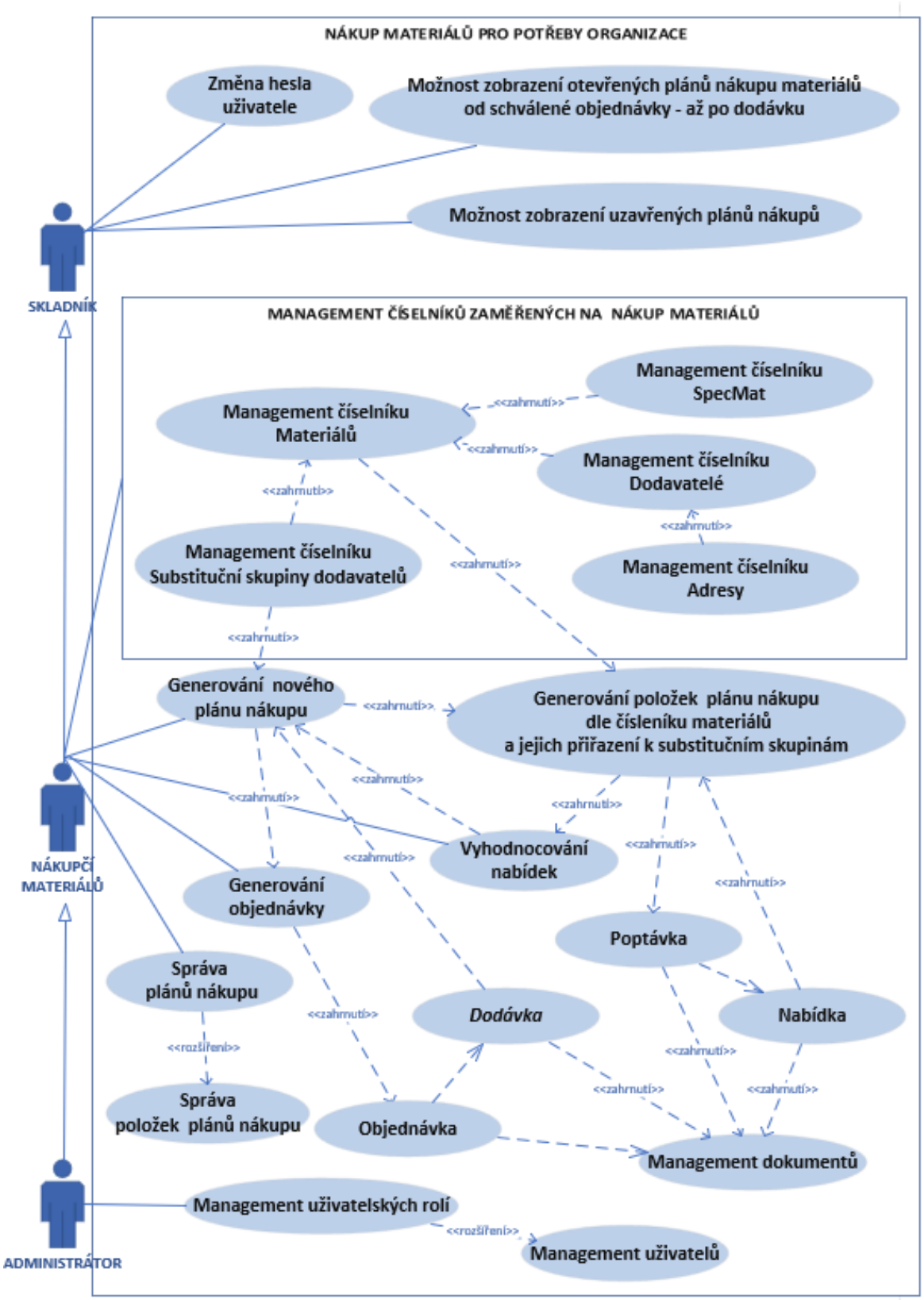
PŘÍLOHA A – BUSINESS CASE



Business case modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

PŘÍLOHA B – USE CASE DIAGRAM



Use case modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

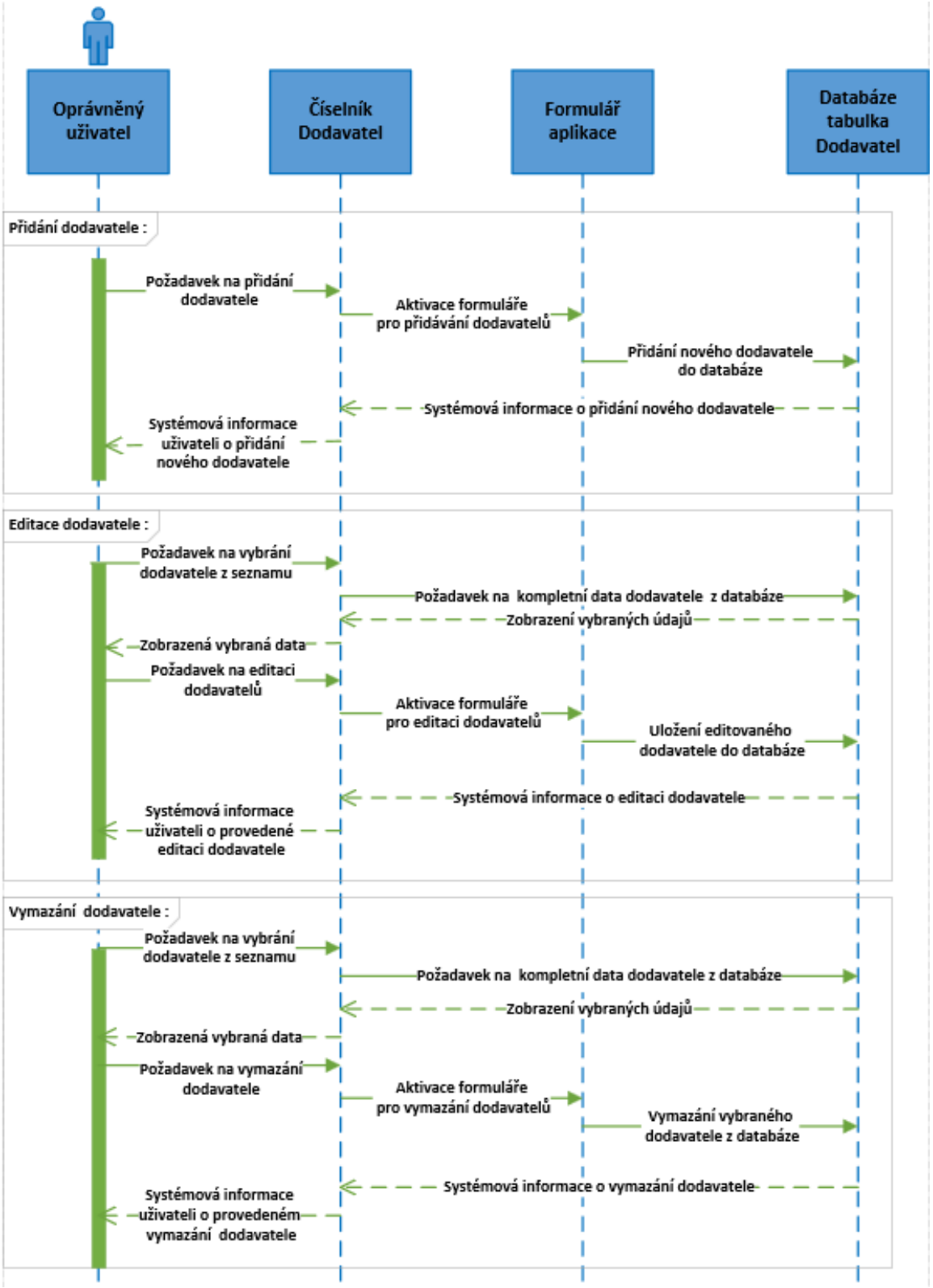
PŘÍLOHA C – USE CASE SCÉNÁŘ

Případ užití: Spravovat číselník dodavatelů:
Účastníci: Editor (nákupčí materiálu), Administrátor
Vstupní podmínky: Editor (nákupčí materiálu) je přihlášen v aplikaci a je v části číselníky.
Hlavní scénář: 1. Oprávněný uživatel se nachází v části aplikace, číselníky a otevírá číselník dodavatelů. 1. Systém zobrazí seznam uložených dodavatelů. 2**. KDYŽ** oprávněný uživatel zvolí možnost přidání dodavatele: 2.1. Systém aktivuje zobrazený formulář, pro přidání nového dodavatele. 2.2. Oprávněný uživatel vyplní veškeré údaje. 2.3. Zadané informace potvrdí ovládacím prvkem pro uložení do tabulky databáze. 2.4. Následně je do tabulky **„Dodavatel“** přidán nový záznam dodavatele. 3. **KDYŽ** oprávněný uživatel zvolí možnost editace dodavatele: 3.1. Oprávněný uživatel vybere ze seznamu dodavatelů záznam, který bude editovat. 3.2. Provede požadovanou editaci dat vybraného dodavatele. 3.3. Provedené změny potvrdí ovládacím prvkem pro uložení dat do databáze-tabulky **„Dodavatel“.** 4. **KDYŽ** oprávněný uživatel zvolí možnost vymazat dodavatele: 4.1. Oprávněný uživatel vybere ze seznamu dodavatelů záznam, který bude odstraněn. 4.2. Provede ovládacím prvkem jeho odstranění. 4.3. Oprávněný uživatel je vyzván systémovým oknem aplikace k potvrzení požadavku na odstranění záznamu vybraného dodavatele. Po potvrzení je dodavatel odstraněn z databáze-tabulky **„Dodavatel“.** Rozšiřující bod: Správa importu dat.
Výstupní podmínky: Žádné
Alternativní scénáře: Importování dat do podnikového systému přes *CSV, XML, JSON.

Use case – scénář pro správu číselníku dodavatelů modelového příkladu.

Zdroj: Vlastní tvorba.

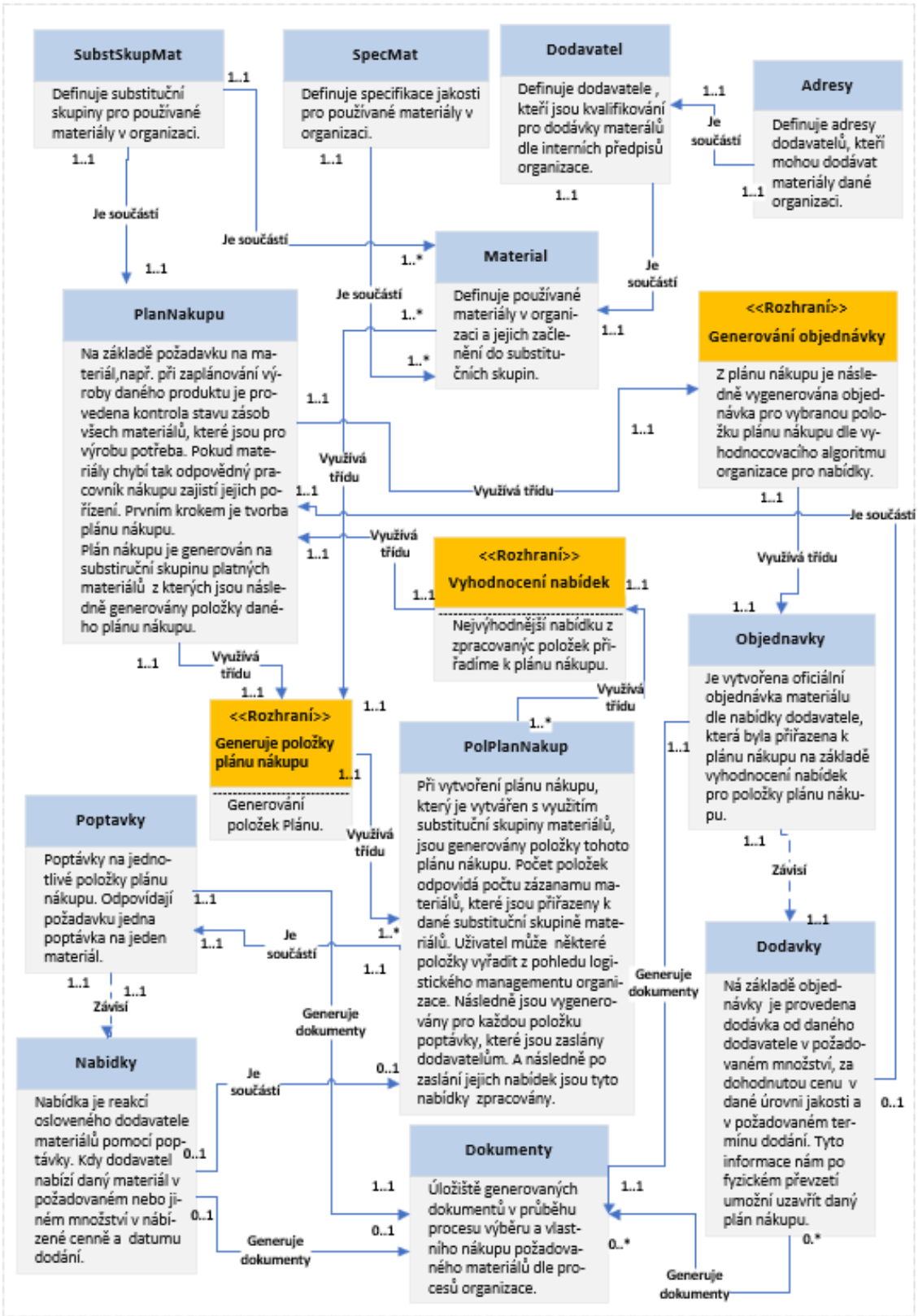
PŘÍLOHA D – SEKVENČNÍ DIAGRAM



Sekvenční diagram pro správu číselníku dodavatelů modelového příkladu.

Zdroj: Vlastní tvorba v MS-Visio 2016.

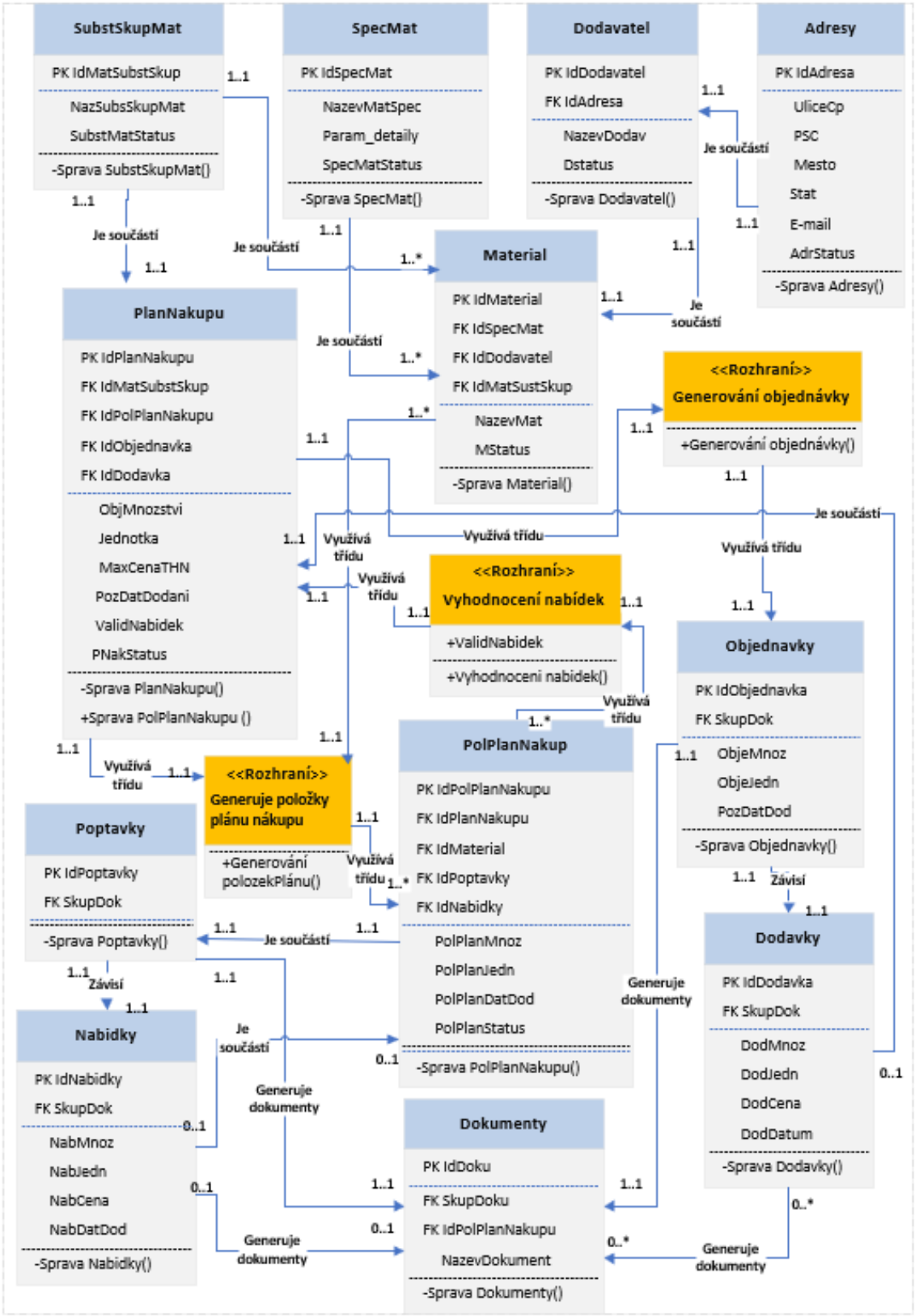
PŘÍLOHA E – KONCEPTUÁLNÍ MODEL



Konceptuální model modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

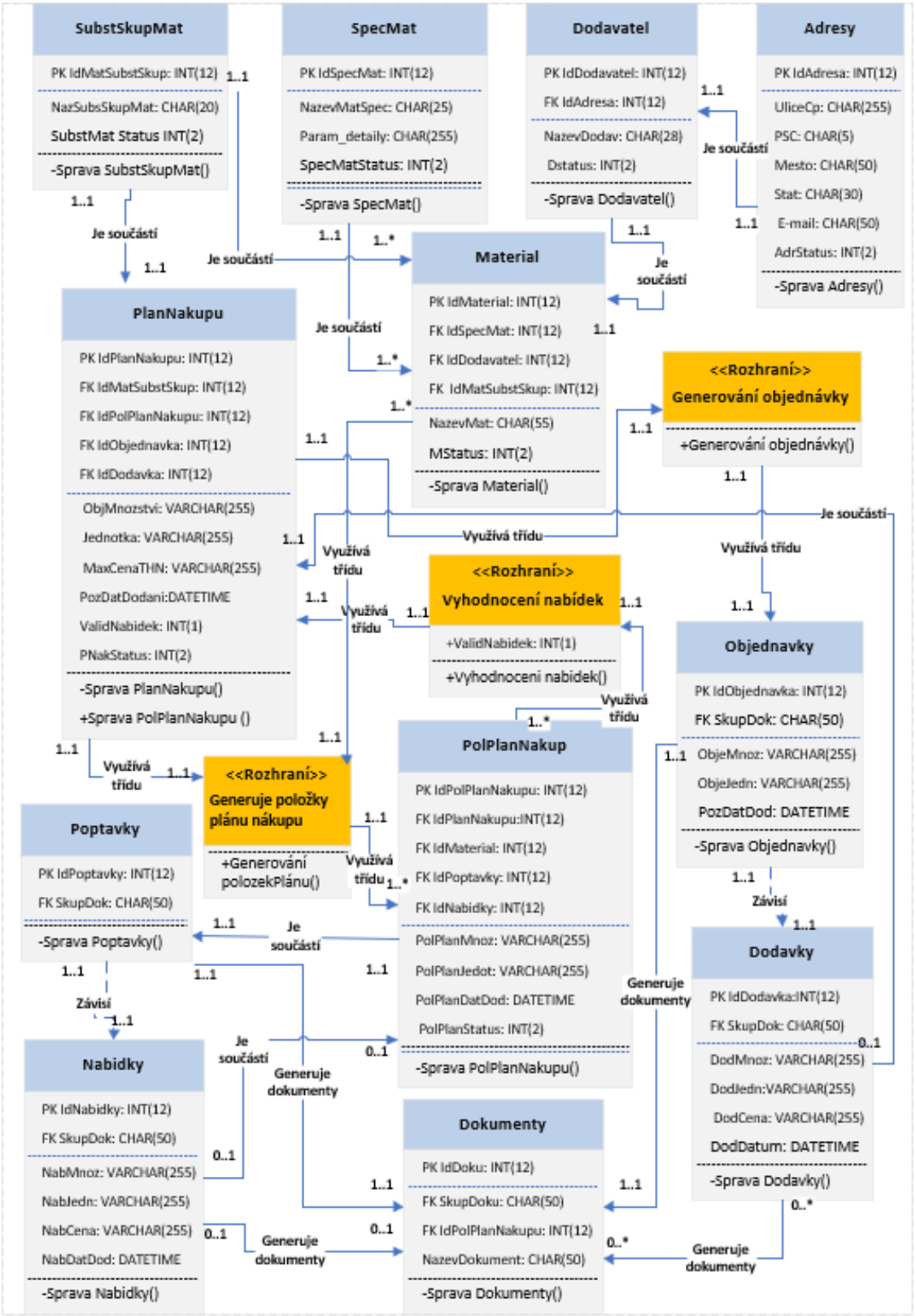
PŘÍLOHA F – LOGICKÝ MODEL



Logický model modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

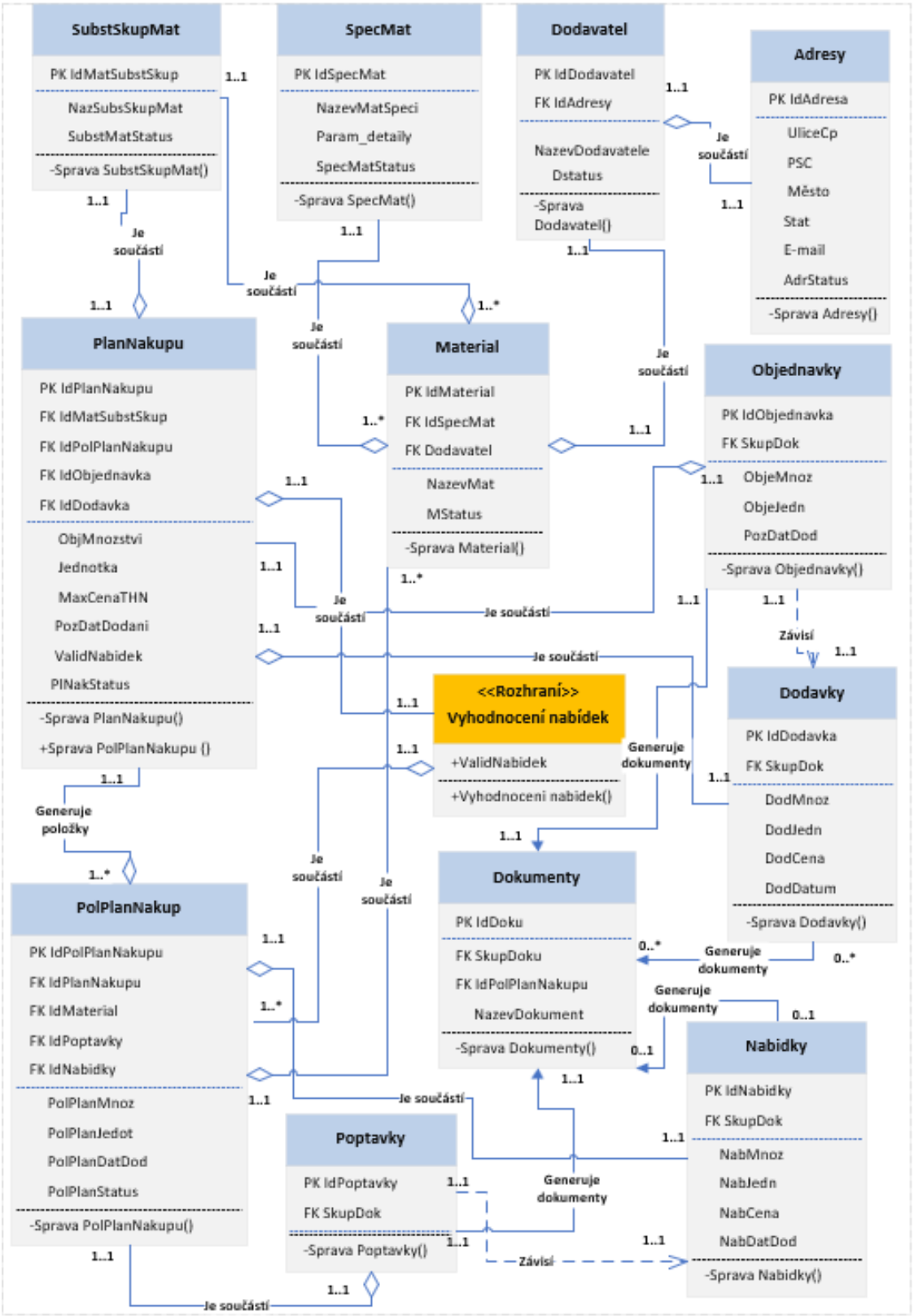
PŘÍLOHA G – FYZICKÝ MODEL



Fyzický model modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

PŘÍLOHA H – E-R DIAGRAM S KONCEPTEM AGREGACÍ



E-R diagram s konceptem agregací modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

PŘÍLOHA CH – DATA FLOW DIAGRAM (DFD)

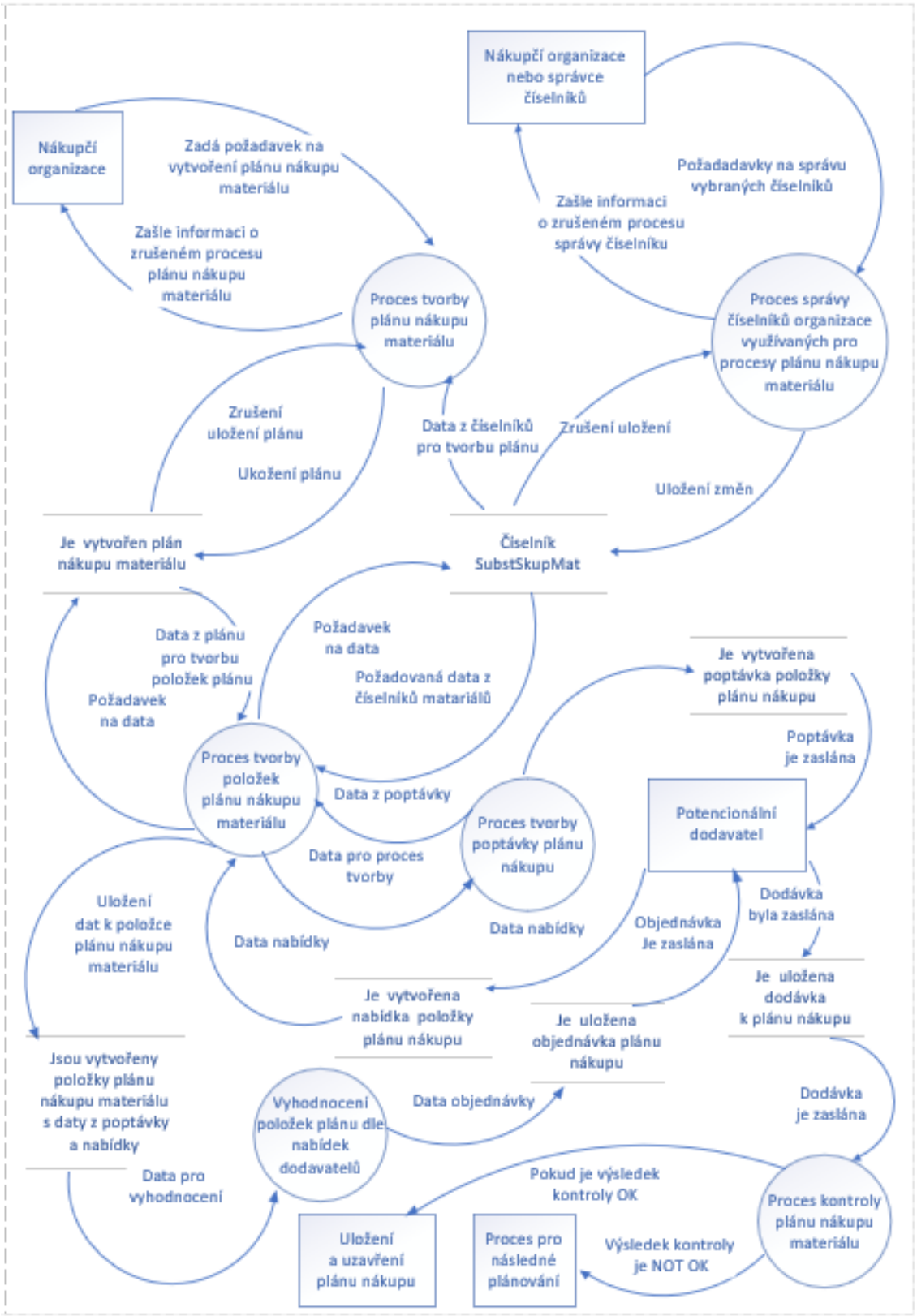
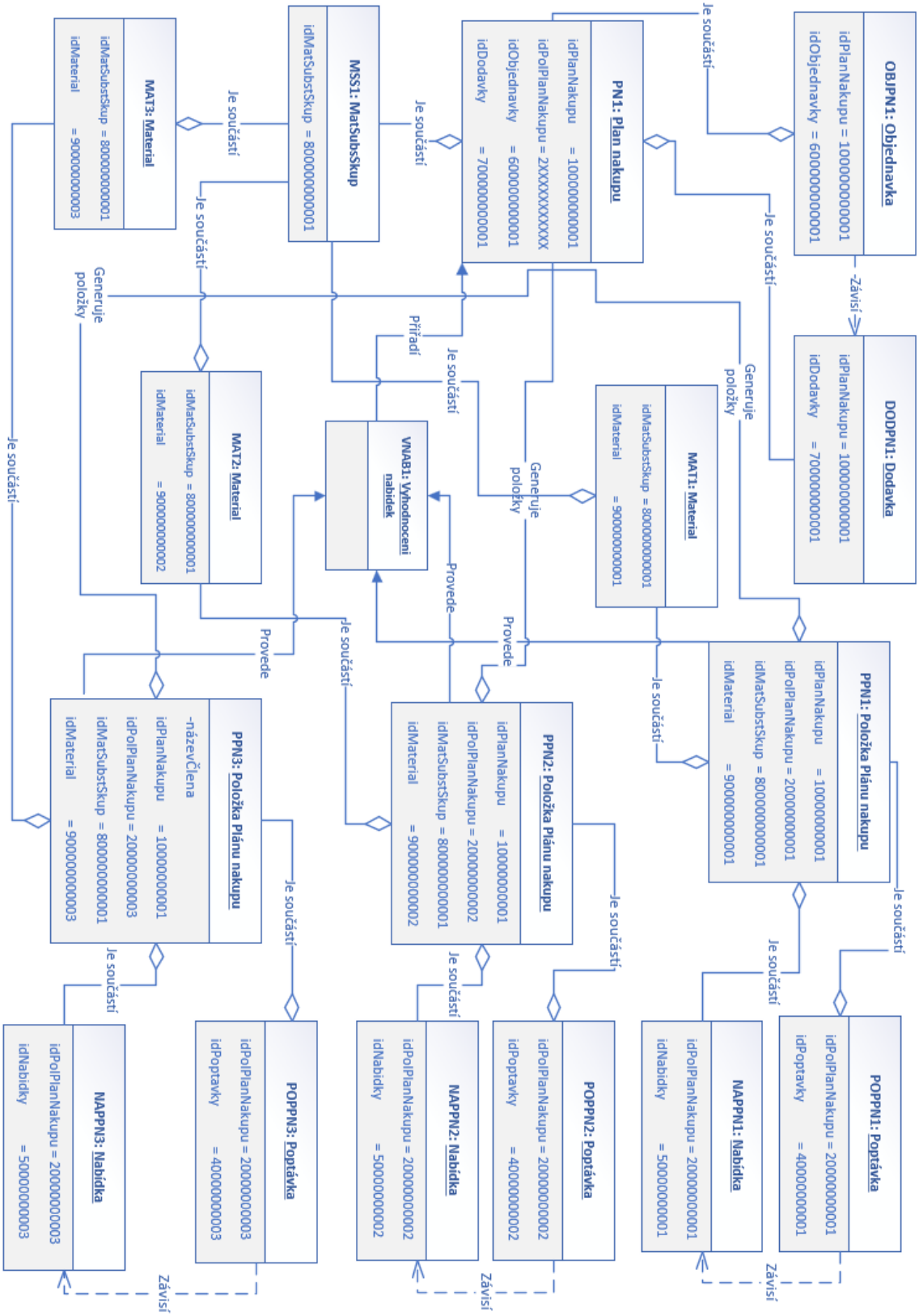


Diagram toku dat (DFD) modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

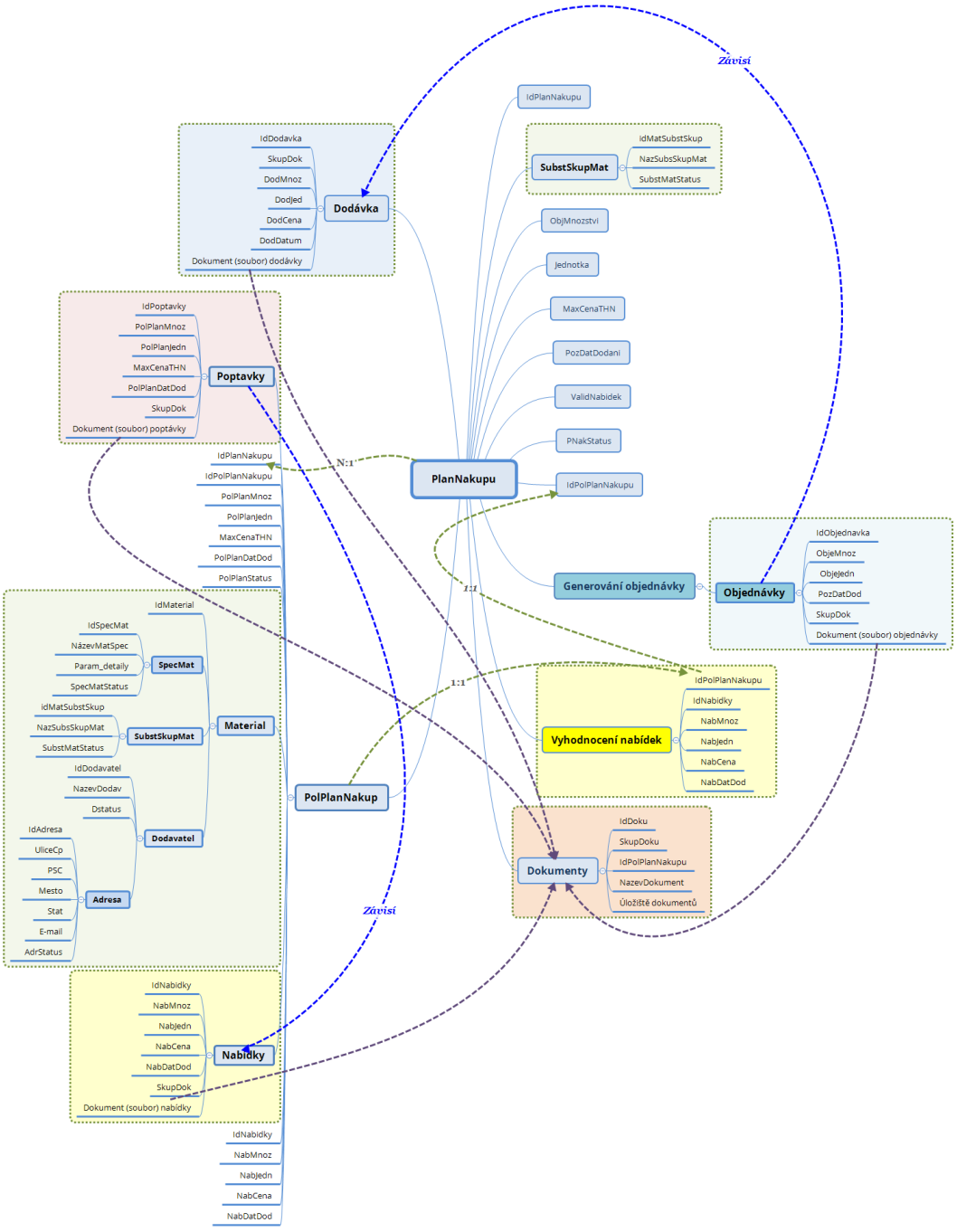
PŘÍLOHA I – UML OBJEKTOVÝ DIAGRAM



UML Objektový diagram modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v MS-Visio 2016.

PŘÍLOHA J – MYŠLENKOVÁ MAPA (MIND MAP)

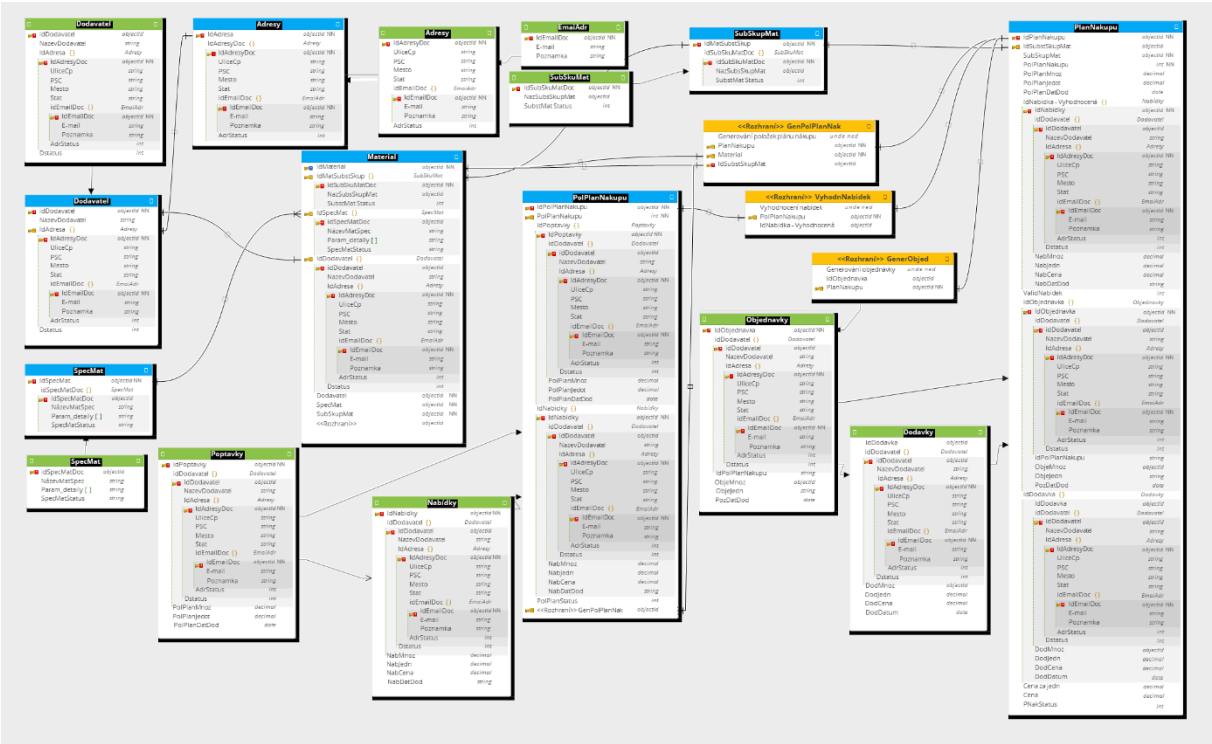


Myšlenková mapa (Mind Map) modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Pro přehlednou práci s Mind Map nastavte prosím ZOOM na 200 %.

Zdroj : Vlastní tvorba v XMind 8 Update 9 (R3.7.9.201912052356) .

PŘÍLOHA K – CRD (collection relationship diagram) -MongoDB



CRD (collection relationship diagram) - MongoDB modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Pro přehlednou práci s CRD (collection relationship diagram) - MongoDB nastavte prosím ZOOM na 300 %.

Zdroj: Vlastní tvorba v Moon Modeler 2.9.6 .

PŘÍLOHA L – MongoDB KÓD PRO CRD Z PŘÍLOHY K

```
db.createCollection('Material', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'Material',
      required: ['idMaterial', 'Dodavatel', 'SpecMat', 'SubSkupMat'],
      properties: {
        idMaterial: {
          bsonType: 'objectId'
        },
        IdMatSubstSkup: {
          bsonType: 'object',
          title: 'SubSkuMat',
          required: ['idSubSkuMatDoc'],
          properties: {
            idSubSkuMatDoc: {
              bsonType: 'objectId'
            },
            NazSubsSkupMat: {
              bsonType: 'objectId'
            },
            SubstMat Status: {
              bsonType: 'int'
            }
          }
        },
        IdSpecMat: {
          bsonType: 'object',
          title: 'SpecMat',
          properties: {
            idSpecMatDoc: {
              bsonType: 'objectId'
            },
            NázevMatSpec: {
              bsonType: 'string'
            },
            Param_detaily: {
              bsonType: 'array',
              items: {
                bsonType: 'string'
              }
            },
            SpecMatStatus: {
              bsonType: 'string'
            }
          }
        },
        IdDodavatel: {
          bsonType: 'object',
          title: 'Dodavatel',
          properties: {
            idDodavatel: {
              bsonType: 'objectId'
            },
            NazevDodavatel: {
              bsonType: 'string'
            },
            IdAdresa: {
```

```
              bsonType: 'object',
              title: 'Adresy',
              required: ['IdAdresyDoc'],
              properties: {
                IdAdresyDoc: {
                  bsonType: 'objectId'
                },
                UliceCp: {
                  bsonType: 'string'
                },
                PSC: {
                  bsonType: 'string'
                },
                Mesto: {
                  bsonType: 'string'
                },
                Stat: {
                  bsonType: 'string'
                },
                idEmailDoc: {
                  bsonType: 'object',
                  title: 'EmaiAdr',
                  required: ['IdEmailDoc'],
                  properties: {
                    IdEmailDoc: {
                      bsonType: 'objectId'
                    },
                    E - mail: {
                      bsonType: 'string'
                    },
                    Poznamka: {
                      bsonType: 'string'
                    }
                  }
                },
                AdrStatus: {
                  bsonType: 'int'
                }
              }
            },
            Dstatus: {
              bsonType: 'int'
            }
          }
        },
        Dodavatel: {
          bsonType: 'objectId'
        },
        SpecMat: {
          bsonType: 'objectId'
        },
        SubSkupMat: {
          bsonType: 'objectId'
        },
        << Rozhraní >>: {
          bsonType: 'objectId'
        }
      }
    }
  }
});
```

```
db.createCollection('Adresy', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'Adresy',
      required: ['IdAdresa'],
      properties: {
        IdAdresa: {
          bsonType: 'objectId'
        },
        IdAdresyDoc: {
          bsonType: 'object',
          title: 'Adresy',
          required: ['IdAdresyDoc'],
          properties: {
            IdAdresyDoc: {
              bsonType: 'objectId'
            },
            UliceCp: {
              bsonType: 'string'
            },
            PSC: {
              bsonType: 'string'
            },
            Mesto: {
              bsonType: 'string'
            },
            Stat: {
              bsonType: 'string'
            },
            idEmailDoc: {
              bsonType: 'object',
              title: 'EmaiAdr',
              required: ['IdEmailDoc'],
              properties: {
                IdEmailDoc: {
                  bsonType: 'objectId'
                },
                E - mail: {
                  bsonType: 'string'
                },
                Poznamka: {
                  bsonType: 'string'
                }
              }
            },
            AdrStatus: {
              bsonType: 'int'
            }
          }
        }
      }
    }
  }
});
```

```
db.createCollection('Dodavatel', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'Dodavatel',
      required: ['idDodavatel'],
      properties: {
        idDodavatel: {
          bsonType: 'objectId'
        },
        NazevDodavatel: {
          bsonType: 'string'
        },
        IdAdresa: {
          bsonType: 'object',
          title: 'Adresy',
          required: ['IdAdresyDoc'],
          properties: {
            IdAdresyDoc: {
              bsonType: 'objectId'
            },
            UliceCp: {
              bsonType: 'string'
            },
            PSC: {
              bsonType: 'string'
            },
            Mesto: {
              bsonType: 'string'
            },
            Stat: {
              bsonType: 'string'
            },
            idEmailDoc: {
              bsonType: 'object',
              title: 'EmaiAdr',
              required: ['IdEmailDoc'],
              properties: {
                IdEmailDoc: {
                  bsonType: 'objectId'
                },
                E - mail: {
                  bsonType: 'string'
                },
                Poznamka: {
                  bsonType: 'string'
                }
              }
            },
            AdrStatus: {
              bsonType: 'int'
            }
          }
        },
        Dstatus: {
          bsonType: 'int'
        }
      }
    }
  }
});
```

```
db.createCollection('SubSkupMat', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'SubSkupMat',
      required: ['IdMatSubstSkup'],
      properties: {
        IdMatSubstSkup: {
          bsonType: 'objectId'
        },
        idSubSkuMatDoc: {
          bsonType: 'object',
          title: 'SubSkuMat',
          required: ['idSubSkuMatDoc'],
          properties: {
            idSubSkuMatDoc: {
              bsonType: 'objectId'
            },
            NazSubsSkupMat: {
              bsonType: 'objectId'
            },
            SubstMat Status: {
              bsonType: 'int'
            }
          }
        }
      }
    }
  }
});

db.createCollection('SpecMat', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'SpecMat',
      required: ['IdSpecMat'],
      properties: {
        IdSpecMat: {
          bsonType: 'objectId'
        },
        idSpecMatDoc: {
          bsonType: 'object',
          title: 'SpecMat',
          properties: {
            idSpecMatDoc: {
              bsonType: 'objectId'
            },
            NázevMatSpec: {
              bsonType: 'string'
            },
            Param_detaily: {
              bsonType: 'array',
              items: {
                bsonType: 'string'
              }
            },
            SpecMatStatus: {
              bsonType: 'string'
            }
          }
```

```
        }
      }
    }
  }
});

db.createCollection('PlanNakupu', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'PlanNakupu',
      required: ['IdPlanNakupu', 'SubSkupMat', 'PolPlanNakupu'],
      properties: {
        IdPlanNakupu: {
          bsonType: 'objectId'
        },
        IdSubstSkupMat: {
          bsonType: 'objectId'
        },
        SubSkupMat: {
          bsonType: 'objectId'
        },
        PolPlanNakupu: {
          bsonType: 'int'
        },
        PolPlanMnoz: {
          bsonType: 'decimal'
        },
        PolPlanJedot: {
          bsonType: 'decimal'
        },
        PolPlanDatDod: {
          bsonType: 'date'
        },
        IdNabidka - Vyhodnocená: {
          bsonType: 'object',
          title: 'Nabídky',
          required: ['IdNabidky'],
          properties: {
            IdNabidky: {
              bsonType: 'objectId'
            },
            idDodavatel: {
              bsonType: 'object',
              title: 'Dodavatel',
              properties: {
                idDodavatel: {
                  bsonType: 'objectId'
                },
                NazevDodavatel: {
                  bsonType: 'string'
                },
                IdAdresa: {
                  bsonType: 'object',
                  title: 'Adresy',
                  required: ['IdAdresyDoc'],
                  properties: {
                    IdAdresyDoc: {
                      bsonType: 'objectId'
                    },
                    UliceCp: {
```

```
              bsonType: 'string'
            },
            PSC: {
              bsonType: 'string'
            },
            Mesto: {
              bsonType: 'string'
            },
            Stat: {
              bsonType: 'string'
            },
            idEmailDoc: {
              bsonType: 'object',
              title: 'EmaiAdr',
              required: ['IdEmailDoc'],
              properties: {
                IdEmailDoc: {
                  bsonType: 'objectId'
                },
                E - mail: {
                  bsonType: 'string'
                },
                Poznamka: {
                  bsonType: 'string'
                }
              }
            },
            AdrStatus: {
              bsonType: 'int'
            }
          }
        },
        Dstatus: {
          bsonType: 'int'
        }
      }
    },
    NabMnoz: {
      bsonType: 'decimal'
    },
    NabJedn: {
      bsonType: 'decimal'
    },
    NabCena: {
      bsonType: 'decimal'
    },
    NabDatDod: {
      bsonType: 'string'
    }
  }
},
ValidNabidek: {
  bsonType: 'int'
},
IdObjednavka: {
  bsonType: 'object',
  title: 'Objednavky',
  required: [' IdObjednavka'],
  properties: {
    IdObjednavka: {
      bsonType: 'objectId'
```

```
},
idDodavatel: {
  bsonType: 'object',
  title: 'Dodavatel',
  properties: {
    idDodavatel: {
      bsonType: 'objectId'
    },
    NazevDodavatel: {
      bsonType: 'string'
    },
    IdAdresa: {
      bsonType: 'object',
      title: 'Adresy',
      required: ['IdAdresyDoc'],
      properties: {
        IdAdresyDoc: {
          bsonType: 'objectId'
        },
        UliceCp: {
          bsonType: 'string'
        },
        PSC: {
          bsonType: 'string'
        },
        Mesto: {
          bsonType: 'string'
        },
        Stat: {
          bsonType: 'string'
        },
        idEmailDoc: {
          bsonType: 'object',
          title: 'EmaiAdr',
          required: ['IdEmailDoc'],
          properties: {
            IdEmailDoc: {
              bsonType: 'objectId'
            },
            E - mail: {
              bsonType: 'string'
            },
            Poznamka: {
              bsonType: 'string'
            }
          }
        },
        AdrStatus: {
          bsonType: 'int'
        }
      }
    },
    Dstatus: {
      bsonType: 'int'
    }
  }
},
IdPolPlanNakupu: {
  bsonType: 'string'
},
ObjeMnoz: {
```

```
      bsonType: 'objectId'
    },
    ObjeJedn: {
      bsonType: 'string'
    },
    PozDatDod: {
      bsonType: 'date'
    }
  }
},
IdDodavka: {
  bsonType: 'object',
  title: 'Dodavky',
  properties: {
    IdDodavka: {
      bsonType: 'objectId'
    },
    idDodavatel: {
      bsonType: 'object',
      title: 'Dodavatel',
      properties: {
        idDodavatel: {
          bsonType: 'objectId'
        },
        NazevDodavatel: {
          bsonType: 'string'
        },
        IdAdresa: {
          bsonType: 'object',
          title: 'Adresy',
          required: ['IdAdresyDoc'],
          properties: {
            IdAdresyDoc: {
              bsonType: 'objectId'
            },
            UliceCp: {
              bsonType: 'string'
            },
            PSC: {
              bsonType: 'string'
            },
            Mesto: {
              bsonType: 'string'
            },
            Stat: {
              bsonType: 'string'
            },
            idEmailDoc: {
              bsonType: 'object',
              title: 'EmaiAdr',
              required: ['IdEmailDoc'],
              properties: {
                IdEmailDoc: {
                  bsonType: 'objectId'
                },
                E - mail: {
                  bsonType: 'string'
                },
                Poznamka: {
                  bsonType: 'string'
                }
```

```
                    }
                  },
                  AdrStatus: {
                    bsonType: 'int'
                  }
                }
              },
              Dstatus: {
                bsonType: 'int'
              }
            }
          },
          DodMnoz: {
            bsonType: 'objectId'
          },
          DodJedn: {
            bsonType: 'decimal'
          },
          DodCena: {
            bsonType: 'decimal'
          },
          DodDatum: {
            bsonType: 'date'
          }
        }
      },
      Cena za jedn: {
        bsonType: 'decimal'
      },
      Cena: {
        bsonType: 'decimal'
      },
      PNakStatus: {
        bsonType: 'int'
      }
    }
  }
});


db.createCollection('PolPlanNakupu', {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'PolPlanNakupu',
      required: ['IdPolPlanNakupu', 'PolPlanNakupu'],
      properties: {
        IdPolPlanNakupu: {
          bsonType: 'objectId'
        },
        PolPlanNakupu: {
          bsonType: 'int'
        },
        IdPoptavky: {
          bsonType: 'object',
          title: 'Poptavky',
          required: ['IdPoptavky'],
          properties: {
            IdPoptavky: {
              bsonType: 'objectId'
```

```
},
idDodavatel: {
  bsonType: 'object',
  title: 'Dodavatel',
  properties: {
    idDodavatel: {
      bsonType: 'objectId'
    },
    NazevDodavatel: {
      bsonType: 'string'
    },
    IdAdresa: {
      bsonType: 'object',
      title: 'Adresy',
      required: ['IdAdresyDoc'],
      properties: {
        IdAdresyDoc: {
          bsonType: 'objectId'
        },
        UliceCp: {
          bsonType: 'string'
        },
        PSC: {
          bsonType: 'string'
        },
        Mesto: {
          bsonType: 'string'
        },
        Stat: {
          bsonType: 'string'
        },
        idEmailDoc: {
          bsonType: 'object',
          title: 'EmaiAdr',
          required: ['IdEmailDoc'],
          properties: {
            IdEmailDoc: {
              bsonType: 'objectId'
            },
            E - mail: {
              bsonType: 'string'
            },
            Poznamka: {
              bsonType: 'string'
            }
          }
        },
        AdrStatus: {
          bsonType: 'int'
        }
      }
    },
    Dstatus: {
      bsonType: 'int'
    }
  }
},
PolPlanMnoz: {
  bsonType: 'decimal'
},
PolPlanJedot: {
```

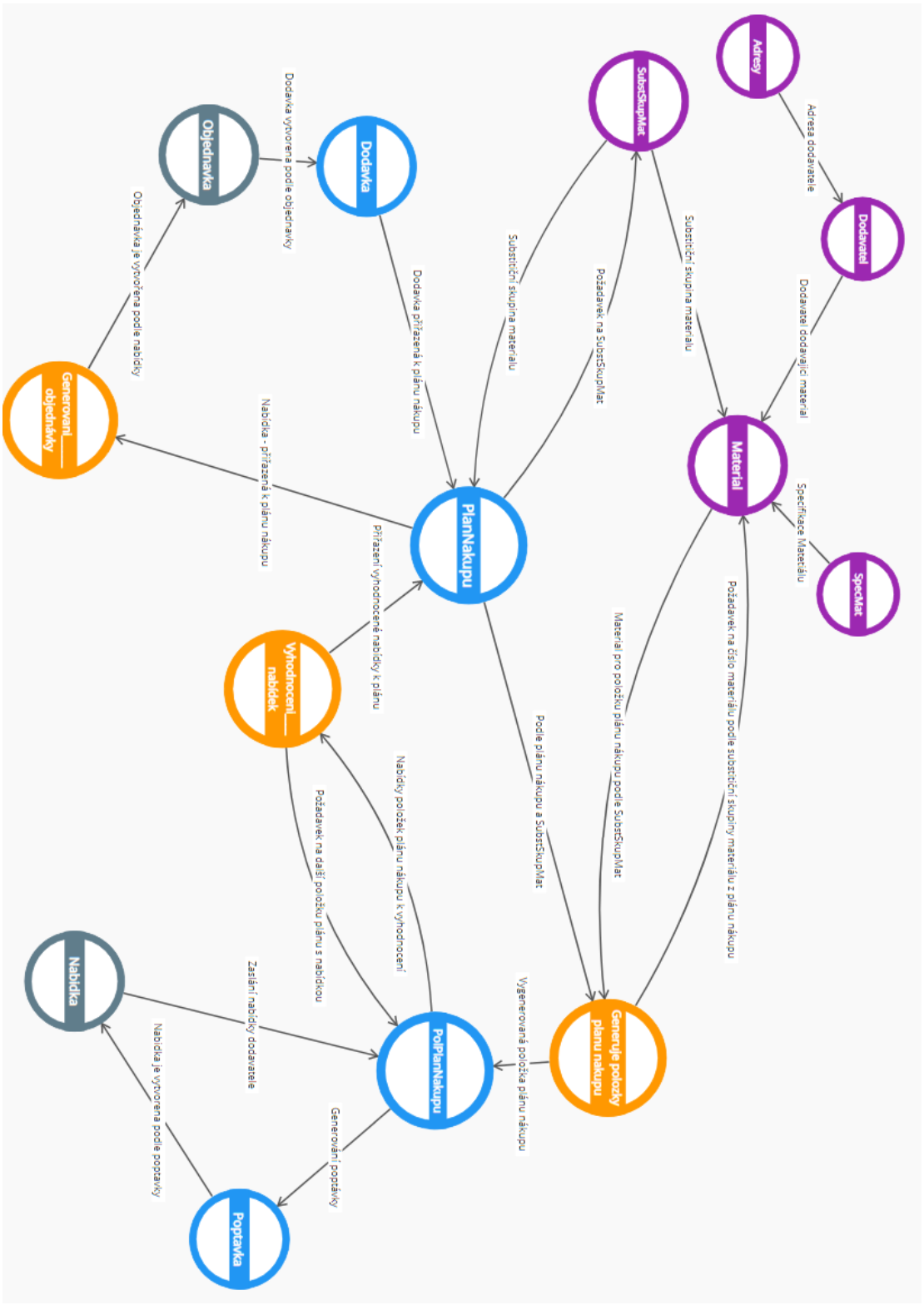
```
      bsonType: 'decimal'
    },
    PolPlanDatDod: {
      bsonType: 'date'
    }
  }
},
IdNabidky: {
  bsonType: 'object',
  title: 'Nabídky',
  required: ['IdNabidky'],
  properties: {
    IdNabidky: {
      bsonType: 'objectId'
    },
    idDodavatel: {
      bsonType: 'object',
      title: 'Dodavatel',
      properties: {
        idDodavatel: {
          bsonType: 'objectId'
        },
        NazevDodavatel: {
          bsonType: 'string'
        },
        IdAdresa: {
          bsonType: 'object',
          title: 'Adresy',
          required: ['IdAdresyDoc'],
          properties: {
            IdAdresyDoc: {
              bsonType: 'objectId'
            },
            UliceCp: {
              bsonType: 'string'
            },
            PSC: {
              bsonType: 'string'
            },
            Mesto: {
              bsonType: 'string'
            },
            Stat: {
              bsonType: 'string'
            },
            idEmailDoc: {
              bsonType: 'object',
              title: 'EmaiAdr',
              required: ['IdEmailDoc'],
              properties: {
                IdEmailDoc: {
                  bsonType: 'objectId'
                },
                E - mail: {
                  bsonType: 'string'
                },
                Poznamka: {
                  bsonType: 'string'
                }
              }
            },
```

```
                    AdrStatus: {
                      bsonType: 'int'
                    }
                  }
                },
                Dstatus: {
                  bsonType: 'int'
                }
              }
            },
            NabMnoz: {
              bsonType: 'decimal'
            },
            NabJedn: {
              bsonType: 'decimal'
            },
            NabCena: {
              bsonType: 'decimal'
            },
            NabDatDod: {
              bsonType: 'string'
            }
          }
        },
        PolPlanStatus: {
          bsonType: 'int'
        },
        << Rozhraní >> GenPolPlanNak: {
          bsonType: 'objectId'
        }
      }
    }
  }
});
```

Zdrojový kód vybraných kolekcí z CRD (collection relationship diagram) - MongoDB modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Zdroj: Vlastní tvorba v Moon Modeler 2.9.6.

PŘÍLOHA M – SCHÉMA MODELU GRAFOVÉ DATABÁZE



Model grafové databáze pro Noe4j modelového příkladu pro plánovaný nákup materiálů v organizaci s využitím ISO 9000 postupů (poptávka, nabídka, objednávka, dodávka).

Pro přehlednou práci s modelem grafové databáze nastavte prosím ZOOM na 200 %.

Zdroj : Vlastní tvorba v Hackolade Studio 4.3.17 (Licence – Student).