

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

DIPLOMOVÁ PRÁCE

Automatizace řízení výroby v kontextu Industry 4.0



2022

Vedoucí práce:
RNDr. Arnošt Večerka

Bc. Martin Stojmenov

Studijní obor: Aplikovaná informatika,
prezenční forma

Bibliografické údaje

Autor: Bc. Martin Stojmenov
Název práce: Automatizace řízení výroby v kontextu Industry 4.0
Typ práce: diplomová práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2022
Studijní obor: Aplikovaná informatika, prezenční forma
Vedoucí práce: RNDr. Arnošt Večerka
Počet stran: 159
Přílohy: 1 CD/DVD
Jazyk práce: český

Bibliographic info

Author: Bc. Martin Stojmenov
Title: Automation of manufacture control in the context of Industry 4.0
Thesis type: master thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2022
Study field: Applied Computer Science, full-time form
Supervisor: RNDr. Arnošt Večerka
Page count: 159
Supplements: 1 CD/DVD
Thesis language: Czech

Anotace

Diplomová práce se zabývá teoretickým rozбором vybraných standardů pro automatizaci a řízení procesů výroby. Rozbor je realizován v kontextu Industry 4.0 a v souladu s normami ANSI/ISA-88 a ANSI/ISA-95, které jsou v práci rovněž představeny. Součástí práce je i krátké uvedení podnikových informačních systémů a zejména technologií, které realizují samotné Industry 4.0.

Synopsis

This master's thesis deals with theoretical analysis of certain automation and manufacturing control standards. The analysis is realized in the context of Industry 4.0, while abiding by ANSI/ISA-88 and ANSI/ISA-95 requirements, which are both elaborated on in this thesis as well. The thesis also includes a brief introduction to enterprise information systems as well as to technologies, which enable Industry 4.0.

Klíčová slova: Enterprise resource planning, Internet of Things, průmyslová revoluce, Industry 4.0, RAMI 4.0, ANSI/ISA-88, ANSI/ISA-95, řízení výroby, automatizace výroby, workflow, business proces, Workflow Reference model, Petriho sítě, Workflow sítě, Business Process Model and Notation

Keywords: Enterprise resource planning, Internet of Things, industrial revolution, Industry 4.0, RAMI 4.0, ANSI/ISA-88, ANSI/ISA-95, manufacturing control, manufacturing automation, workflow, business process, Workflow Reference model, Petri nets, Workflow nets, Business process Model and Notation

Rád bych poděkoval RNDr. Arnoštu Večerkovi za cenné rady a připomínky, a rovněž za shovívavý přístup při časovém rozvržení tvorby této diplomové práce. Rovněž bych chtěl poděkovat Mgr. Martinu Kotyzovi za věcné a přínosné konzultace, které pomáhaly korigovat směr a vznik této diplomové práce.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval/a samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

I	Úvod	11
1	Úvod a motivace	11
2	Enterprise Resource Planning	12
2.1	Informační systém	12
2.1.1	Podnikový informační systém	12
2.2	Material Requirements planning	13
2.3	Manufacturing resource planning	14
2.4	Enterprise Resource Planning	14
2.4.1	Architektura ERP systémů	17
II	Technologie a koncepty	19
3	Internet of Things	19
3.1	Cloud	22
3.2	Industrial Internet of Things	24
3.3	Big data	24
4	Průmyslové revoluce	25
4.1	Historie průmyslových revolucí	25
4.2	Cyber-physical system	27
4.3	Digital Twin	29
4.4	Iniciativa Industry 4.0	29
4.4.1	Základní stavební kameny	30
4.4.2	Charakteristiky	34
III	Normy a standardy	37
5	ANSI/ISA-88	37
5.1	Model dávkového procesu	39
5.2	Fyzický model	39
5.3	Model procedurálního řízení	41
5.4	Receptury	43
5.5	Vztahy mezi modely	45
5.6	Stavy a módy	46
5.7	Aktivity a funkce řízení dávkové výroby	47
5.8	Modely objektů	48
5.9	Implementace normy	49

6	ANSI/ISA-95	49
6.1	Hierarchie řízení	50
6.1.1	Úroveň 3	51
6.2	Hierarchie zařízení	53
6.3	Model funkcí	54
6.4	Manufacturing Operations Management	56
6.5	Výměny informací a modely objektů	58
6.5.1	Modely objektů informací v úrovni 3	61
6.6	Komunikace (mezi a napříč úrovněmi)	63
6.7	Korelace norem ANSI/ISA-88 a ANSI/ISA-95	65
7	RAMI 4.0	66
7.1	Pravá horizontální osa	68
7.2	Levá horizontální osa	69
7.3	Vertikální osa	69
7.4	Model komponenty Industry 4.0	70
IV	Workflow systémy	71
8	Obecně	71
8.1	Základní terminologie	71
8.2	Modelování	72
8.3	Jazyky workflow systémů	73
8.4	Typy workflow systémů	73
9	Workflow Reference Model	74
9.1	Obecná architektura workflow systémů	74
9.2	Generický model workflow systémů	75
9.3	Workflow Reference Model	78
9.3.1	Systém řízení workflow	78
9.3.2	WAPI a výměna informací	79
9.3.3	1. rozhraní: import/export definice procesů	79
9.3.4	2. rozhraní: klientské workflow aplikace	79
9.3.5	3. rozhraní: vyvolané aplikace	80
9.3.6	4. rozhraní: interoperabilita workflow systémů	80
9.3.7	5. rozhraní: monitorování a administrace	81
9.4	Notace	82
9.5	Definiční jazyky	82
9.6	Shrnutí	83
10	Petriho síť	84
10.1	Rozšíření Petriho sítě	88
10.1.1	Barevné Petriho sítě	88
10.1.2	Petriho síť rozšířená o čas	89

10.1.3 Hierarchické Petriho sítě	90
10.2 Workflow sítě	90
10.2.1 Konstrukty řízení toku ve Workflow sítích	91
10.2.2 Provádění aktivit	93
10.3 Shrnutí a závěr	94
11 Business Process Model and Notation	95
11.1 Modely	96
11.2 Prvky a jejich notace	97
11.2.1 Tokové objekty	98
11.2.2 Spojovací objekty	104
11.2.3 Plavecké dráhy	105
11.2.4 Datové objekty	105
11.2.5 Artefakty	106
11.3 Decision Model and Notation	107
11.4 XPDL a BPMN	107
11.5 Exekutivní BPMN	108
11.6 Shrnutí	109
12 Doporučení standardů	110
12.1 Standard pro obecnou implementaci	110
12.2 Standard pro definici a modelování	110
13 Usazení do kontextu Industry 4.0 a norem ANSI/ISA-88 a ANSI/ISA-95	111
13.1 Propojení Industry 4.0 a norem S88 a S95	112
13.2 BPMS v Industry 4.0 a v souladu s normou S95	113
13.3 BPMN v Industry 4.0 a v souladu s normou S95	114
13.3.1 Diagramy	115
13.3.2 Tokové objekty	115
13.3.3 Spojovací objekty	116
13.3.4 Plavecké dráhy	116
13.3.5 Datové objekty	116
13.3.6 Artefakty	117
13.3.7 Napojení na S95	117
13.4 WRM	117
Závěr	119
Conclusions	120
A Analýza systému v UML	121
B Příklad zdrojového kódu XPDL	129

C	Formátování a slovníky přeložený pojmů	132
C.1	Formátování pojmů	132
C.2	Tabulky s pojmy	132
D	Vysvětlení ERD diagramů	141
E	Vysvětlení diagramů UML	142
E.1	Diagramy tříd	142
E.1.1	Třídy	142
E.1.1.1	Výčtový typ (enumeration)	143
E.1.2	Vztahy mezi třídami	143
E.2	Doménové diagramy	143
F	Použité nástroje	145
G	Obsah přiloženého CD/DVD	146
	Seznam zkratk	147
	Literatura	150
	Rejstřík	157

Seznam obrázků

1	Vajíčko MRP, MRPII a ERP	15
2	Přehled ERP systémů	16
3	Třívrstvá architektura ERP	18
4	Cluster tree topologie	21
5	Miniaturní RFID tag od výrobce Hitachi	21
6	IaaS , PaaS a SaaS	23
7	Průmyslové revoluce	27
8	Digital Twin	29
9	Dávkový proces – ERD	40
10	Fyzický model 2010 – ERD	40
11	Složky fyzického modelu – ERD	41
12	Model procedurálního řízení – ERD	42
13	Druhy receptur 2010 – ERD	44
14	Vztahy mezi modely – ERD	45
15	Procedurální stavy 2010 – ERD	46
16	Model řídicích aktivit – ERD	47
17	Hierarchie řízení	50
18	Hierarchie zařízení v kontextu rolí	53
19	Model funkcí	54
20	Univerzální model aktivit	57
21	Informace předávané mezi úrovněmi 3 a 4	59
22	Modely <i>vybavení</i> , <i>fyzických aktiv</i> a vztah mezi nimi	60
23	Model definice práce	62
24	Workflow model	62
25	Příklad transakce	64
26	Korelace fyzického modelu (S88) a modelu hierarchie zařízení (S95)	65
27	Korelace modelu řídicích aktivit (S88) a modelu aktivit řízení výrobních operací (S95)	66
28	Vztahy mezi modely objektů S88 a S95	67
29	Reference Architecture Model for Industrie 4.0	67
30	RAMI 4.0 hierarchie (pravá horizontální osa)	68
31	Industry 4.0 komponenta	70
32	Základní pojmy a jejich vztahy	73
33	Charakteristiky workflow systémů	75
34	Implementační model	76
35	Workflow Reference Model	78
36	Příklad Petriho sítě	86
37	Příklad Petriho sítě – faktura	89
38	Petriho sítě – hierarchické rozšíření	90
39	Workflow sítě – AND a notace	92
40	Workflow sítě – OR v klasické notaci	92
41	Workflow sítě – OR v notaci	93

42	Workflow síť – AND/OR	93
43	Workflow síť – Iterace	93
44	Workflow síť – Aktivity	94
45	BPMN – události	100
46	BPMN – základní aktivity	101
47	BPMN – rozšířené a označené aktivity	102
48	BPMN – aktivity choreografie	103
49	BPMN – brány	104
50	BPMN – toky	105
51	BPMN – plavecké dráhy	105
52	BPMN – datové objekty	106
53	BPMN – artefakty	106
54	BPMN – příklad s DMN	107
55	Doménový model – propojení Industry 4.0 komponenty, RAMI 4.0 a S95	121
56	Vztahy mezi jednotlivými balíčky	122
57	Doménový model – BPMN Diagram	122
58	Doménový model – Tokové objekty	122
59	Doménový model – Události	123
60	Doménový model – Aktivity	124
61	Doménový model – Brány	125
62	Doménový model – Propojení	126
63	Doménový model – Plavecké dráhy	126
64	Doménový model – Data	127
65	Doménový model – Artefakty	127
66	Doménový model – napojení BPMN na S95	128
67	Doménový model – WRM	128
68	Notace ERD diagramů	141
69	Notace UML diagramů – třídy	142
70	Notace UML diagramů – vztahy	144

Seznam tabulek

1	On-premise nasazení	17
2	Cloud nasazení	18
3	Dvacet profesí s největším indexem ohrožení digitalizací	30
4	Slovesa	63
5	PUSH zprávy	64
6	Elementy XPDL 1.0	84
7	Elementy XPDL 2.2	108
8	Slovník přeložený pojmů S88	132
9	Slovník přeložený pojmů S95	134
10	Slovník přeložený pojmů RAMI 4.0	137
11	Slovník přeložený pojmů WRM	137

12	Slovník přeložený pojmy BPMN	138
----	--	-----

Seznam vět

1	Definice (Informační systém)	12
2	Definice (Business proces)	13
3	Definice (Cyber-physical system)	27
4	Definice (Hodnotový řetězec)	35
5	Definice (Výrobní proces)	38
6	Definice (Dávka (batch))	38
7	Definice (Dávková výroba, dávkový proces (batch process))	38
8	Definice (Dávkové řízení (batch control))	42
9	Definice (Workflow)	71
10	Definice (Definice procesu)	71
11	Definice (Aktivita)	72
12	Definice (Pracovní položka)	72
13	Definice (Seznam prací)	72
14	Definice (Petriho síť)	85
15	Definice (Stav Petriho sítě)	86
16	Definice (Proveditelnost přechodu)	86
17	Definice (Provedení přechodu)	86
18	Definice (Dosažitelnost stavu)	87
19	Definice (Živá Petriho síť)	87
20	Definice (Omezenost Petriho sítě)	87
21	Definice (Bezpečnost Petriho sítě)	88
22	Příklad (Příklad Petriho sítě – faktura)	88
23	Definice (Čas proveditelnosti přechodu)	89
24	Definice (Cesta v Petriho síti)	90
25	Definice (Workflow sítě)	91
26	Definice (Business Process Management)	95

Seznam zdrojových kódů

1	Příklad procesu v jazyce WPDL	83
2	Příklad v jazyce XPDL	129
3	Příklad aktivit v jazyce XPDL	130
4	Příklad přechodů v jazyce XPDL	131

Část I

Úvod

1 Úvod a motivace

Podniky jsou od pradávna nuceny inovovat a patřičně zefektivňovat výrobu. Koneckonců napříč historií bylo identifikováno několik průmyslových revolucí, které zásadně proměnily způsoby a postupy výroby. V dnešní době jsou neodmyslitelnou součástí podniků informační systémy, díky kterým lze podnik efektivně řídit, a to i jeho „výrobní“ část. Již v průběhu minulého století se začal objevovat [SW](#), jež měl za úkol zefektivnit samotnou výrobu. Postupem let jsme se dostali až k dnešním systémům, nazývaným [ERP](#). Může se jednat o velké [SW](#) balíky pro větší podniky, či naopak balíky menší pro podniky malé.

Dnes se nacházíme na sklonku 4. průmyslové revoluce, která má a zajisté bude mít dopad na řízení chodu podniku, a tudíž i jeho způsobu výroby. V duchu této revoluce se dostávají do popředí i nové technologie a koncepty, které se snaží její myšlenku uchopit a využít. Jedná se zejména o [IoT](#) a iniciativu Industry 4.0.

Pro uvedení do kontextu si v této diplomové práci rozebereme podnikové informační systémy a jejich historický vývoj, avšak samotným cílem práce bude představit a rozebrat standardy pro řízení a automatizaci procesu výroby, a to [Workflow Reference Model](#), [Coloured Petri Nets](#) a [Business Process Model and Notation](#). Tento rozbor bude proveden v duchu zmíněné 4. průmyslové revoluce. Z toho důvodu bude v práci přestavena myšlenka a historie průmyslových revolucí. Zejména bude kladen důraz na iniciativu Industry 4.0, jež konceptualizuje onu 4. průmyslovou revoluci. Rovněž bude kladen důraz na další koncepty a technologie, na kterých Industry 4.0 stojí, jako jsou [IoT](#) či [CPS](#). Rozbor by měl rovněž být v souladu s normami [ANSI/ISA-88](#) a [ANSI/ISA-95](#), které standardizují přístup k tvorbě systémů výroby, tudíž na ně bude brán patřičný zřetel, skrze představení jejich nejdůležitějších myšlenek a částí.

Výsledkem práce bude doporučení, které z představených standardů je vhodné použít pro implementaci automatizovaného systému řízení výroby. Rovněž bude součástí analytický rozbor systému s těmito standardy, a to právě v kontextu Industry 4.0 a norem [S88](#) a [S95](#).

Tato diplomová práce vznikala za konzultací s Mgr. Martinem Kotyzou, jehož věcné připomínky pomáhaly korigovat směr a celé téma práce.

2 Enterprise Resource Planning

K tomu, abychom vůbec mohli definovat, co je to vlastně **Enterprise Resource Planning (ERP)**, je potřeba představit několik dalších pojmů. Jednak jsou potřeba pro zasazení do kontextu a zároveň je záhodno ukázat, jak se historie k ERP vlastně dopracovala.

2.1 Informační systém

Dnes se jedná o velmi často skloňovaný pojem. Ačkoliv většina z nás dokáže informační systém správně identifikovat, my si zde uvedeme pevnou definici.

Definice 1 (Informační systém)

Informační systém^{1} je kombinací lidí a IT¹, jež vytváří, shromažďuje, zpracovává, ukládá a distribuuje užitečná data a informace. [1] [2]

Na základě této definice je dobré rozlišit pojmy *data* a *informace*. Data chápeme jako shluk symbolů bez hlubšího významu a formátu. Když data zpracujeme (zformátujeme a přiřadíme k nim nějaký smysl), pak můžeme mluvit o informaci. Například podíváme-li se na následující data:

- 13051997,
- f000054,

tak nevíme, co znamenají ani jaký mají účel. Ovšem když je zformátujeme a přiřadíme k nim smysl, tak například můžeme dostat:

- 13. 5. 1997 – datum,
- F-000054 – číslo faktury,

čímž nám z dat vzniknou informace.

2.1.1 Podnikový informační systém

Podnikový informační systém (PIS)^{2}, anglicky Enterprise information system, je informační systém, který je nasazen v podniku za účelem zefektivnění a podpory jeho business procesů. Pro zjednodušení, bude-li se nadále hovořit o informačním systému, pak budeme mít namysli podnikový informační systém. Takový systém slouží podnikům v mnoha ohledech, avšak pro jeho efektivnost a správnou funkčnost jsou klíčovým prvkem lidé. Bez zaměstnanců, kteří by systému rozuměli, plnili jej daty, efektivně ho využívali a samozřejmě i vývojářů a administrátorů, kteří by jej dále vyvíjeli, udržovali a spravovali, by se takový systém mohl stát spíše pomyslnou koulí u nohy. Při práci s takovým systémem jsou vykonávány určité kroky pro dosažení kýženého cíle, jakým může být například

¹Informační Technologie – zahrnují software, hardware a telekomunikační sítě

vystavení faktury. Tyto kroky se dají označit termínem business proces (neboli podnikový proces)².

Definice 2 (Business proces)

Business proces^{3} (česky *podnikový proces*^{4}) je sada vzájemně navazujících kroků, jejichž cílem je dosažení určitého výsledku [3] [4] (např. služba či produkt pro zákazníka).

Každý business proces má tedy své kroky, kterými mohou být aktivity vykonávané zaměstnanci, externími systémy či právě informačním systémem (business procesy budou důležité zejména v části IV o systémech workflow).

Z dosavadního popisu lze nabýt dojmu, že informační systém je pouze nějaký větší, komplexní SW. Konec konců, očekává se od něj, že procesy budou sjednoceny, budou na sebe navazovat a tomu bude odpovídat i UI³. Nicméně faktem je, že se vůbec nemusí jednat pouze o jeden ucelený SW, nýbrž o vícero programů, které jsou patřičně zintegrovány. Například se může jednat o vzájemně integrované CRM⁴, SW pro řízení výroby a skladových zásob, SW pro řízení financí aj., přičemž každý z jmenovaných může běžet nezávisle na ostatních, ba dokonce na odděleném HW. Avšak uživateli se vše jeví, jakoby pracoval pouze s jedním systémem. Účelem zavádění takových systémů je zejména zefektivnění chodu onoho podniku.

2.2 Material Requirements planning

Pomyslná etapa vývoje ERP začala v 60. letech 20. století, kdy vznikl SW zvaný **Material Requirements planning (MRP)**^{5}[5], jelikož neexistoval efektivní způsob, jak uhlídat potřebný materiál pro výrobu. Na svou dobu se jednalo o velmi pokročilý SW, který zefektivnil samotnou produkci. Úkolem MRP je sledovat skladové zásoby, a proto si pokládá základní otázky pro výrobu [6]:

- Co a kdy budeme vyrábět? (Jaký produkt)
- Jaký materiál je potřeba k výrobě onoho produktu? (BOM⁵)
- Kolik takového materiálu je potřeba? (A máme už nějaký skladem?)

Účelem kladení těchto otázek je zefektivnění výroby. MRP na základě obsáhlých vstupních dat vydává doporučení, jaké produkty a v jakém objemu je potřeba nakoupit, či vyrobit, pro splnění výroby. Bere samozřejmě v potaz také výrobky, které již může podnik mít na svém skladě. Pro efektivitu v podniku je potřeba

²V této práci se budeme držet termínu business proces, jelikož jeho český ekvivalent nutně nevystihuje jeho pravý význam. Rovněž lze říct, že termín business proces je dnes již pojmem standardně používaným.

³User Interface – uživatelské rozhraní; skrze UI lidé pracují se softwarem

⁴Customer relationship management – slouží pro řízení vztahu mezi podnikem a zákazníky

⁵Bill Of Material – jedná se o seznam materiálu potřebného pro výrobu produktu

udržovat zásoby dostatečné, avšak nikoliv přespříliš velké. Materiál je potřeba nakupovat včas, avšak nikoliv příliš brzy, a to samé platí i o samotné výrobě produktu. Na vše se totiž váží finance. Uskladnění zabírá prostor a čas pracovníků a nakoupený materiál bez přidané hodnoty představuje také nemalý náklad. Pokud k tomu všemu budou produkty ležet ladem, než se prodají zákazníkovi, pak bude cashflow podniku poněkud nevyrovnaný, jelikož jeho finanční prostředky budou ležet v materiálu a produktech na skladech.

2.3 Manufacturing resource planning

Dalším stupněm ve vývoji byl tzv. **Manufacturing resource planning (MRP II)**^[6], a to v 80. letech 20. století [5]. Jelikož se jednalo o nástupce a rozšíření původního **MRP**, označuje se jako **MRP II**. Jak lze vyčíst již z názvu, tak se již nejedná o pouhopouhé plánování materiálu, avšak o plánování všech prostředků, které mohou zasahovat do výrobního procesu, kam patří například i samotný nákup či prodej. Zároveň se pokouší předpovídat a tím odpovídat na otázky „Co když?“ Důležité je vzájemné propojení celé funkcionality, jež **MRP II** nabízí.

Zahrnuje následující funkcionalitu [4]:

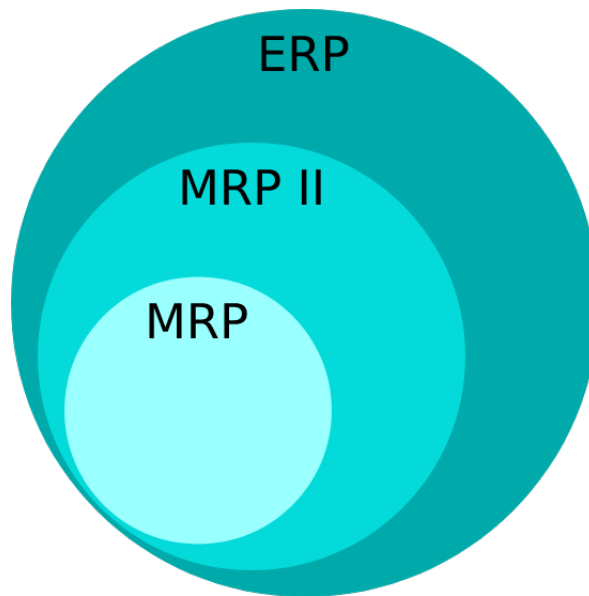
- obchodně-provozní plánování,
- hlavní plánování výroby,
- **Material Requirements planning (MRP)**,
- plánování kapacity,
- různé podpůrné systémy pro realizaci výše zmíněného.

2.4 Enterprise Resource Planning

Pojem **Enterprise Resource Planning**^[7] se poprvé objevil na začátku 90. let 20. století. S vývojem trhu, potřeb zákazníků, náročností řízení podniku a vzrůstající byrokracií přestaly systémy **MRP II** dostačovat. Vznikala potřeba pro podporu, integraci a automatizaci procesů napříč celým podnikem od financí, přes HR až po prodej a účetnictví. Samozřejmostí byla potřeba zachovat podporu výroby známou ze systémů **MRP II**, potažmo **MRP**. Hierarchii těchto systémů (resp. generací) zachycuje obrázek 1.

Již existovaly různé samostatně stojící moduly pro zmíněné části řízení v podniku, avšak jejich vzájemná integrace by podnik přišla na nemalé prostředky (jak finanční, tak personální a časové). Velcí dodavatelé **SW**, kteří již takové moduly podnikům dodávali, proto začali vzájemně integrovat své moduly a začali je nabízet pod jednou značkou. Mezi typické moduly, jež **ERP** zahrnuje, patří [7] [8]:

- prodej a marketing,
- finance,



Obrázek 1: Vajíčko MRP, MRPII a ERP

- lidské zdroje (HR),
- CRM,
- výroba,
- zásobování,
- řízení chodu podniku.

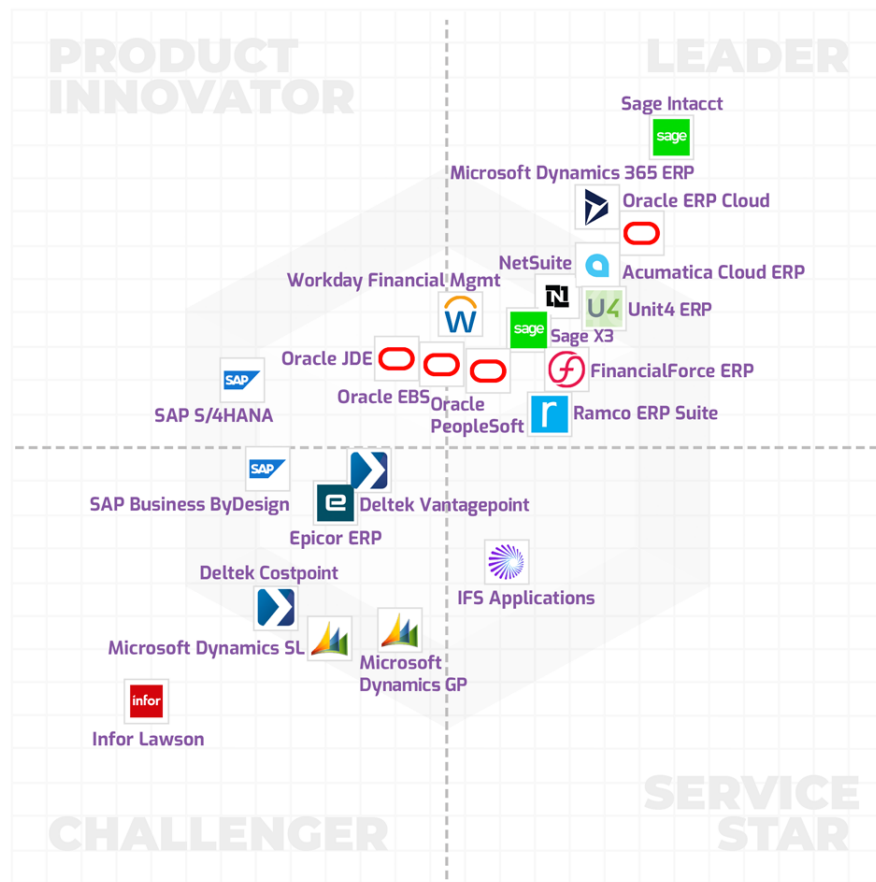
Můžeme tedy říct, že se jedná o mnoho zaintegrovaných modulů, které využívají jednotná data a tváří se jako jeden SW. Nicméně ERP systémy jsou nesmírně objemné. Vyvinout takový systém trvá poměrně dlouho a je na to potřeba mnoho prostředků, a to jak finančních, tak i personálních. Pro podnik je proto v podstatě nemožné, nechat si vyrobit svůj vlastní ERP na míru. Z toho důvodu dodavatelé poskytují již připravené systémy, za které podniky zpravidla platí předplatné. Výhodou je, že si podnik může takový ERP dále „nakonfigurovat“ podle potřeby, navíc se jedná o praxí otestovaný SW. Tudíž nepotřebuje-li podnik nějaký modul, nemusí za něj nutně platit. Dodavatelů ERP je mnoho, avšak za zmínku určitě stojí největší hráči na trhu, a to Microsoft, SAP a Oracle, jak lze vidět na obrázku 2. Avšak existují desítky dalších, přičemž v ČR se jedná například o Asseco Solutions (a jejich ERP Helios) [9] či ABRA Software (a jejich ERP Abra) [10].

Ačkoliv business procesy^[8] mohou být svým způsobem standardizovány (např. vystavení faktury), každý podnik může mít tyto procesy nějak specifické. Může se jednat o interní politiku, ale i o specifika dané zákony země, kde se podnik nachází. Proto je potřeba ERP patřičně upravit, aby vyhovovalo business procesům toho či onoho podniku. Na tyto úpravy si zpravidla podnik najímá externí



8.6

PRODUCT FEATURES AND SATISFACTION



6.5

Y
X

5.2

VENDOR EXPERIENCE AND CAPABILITIES

9.4

Obrázek 2: Přehled ERP systémů

Zdroj: <https://www.softwarereviews.com/awards/data-quadrant-awards-2021-enterprise-resource-planning>

Tabulka 1: On-premise nasazení

Výhody	Nevýhody
Kontrola nad nasazením	Velké počáteční náklady
Kontrola nad přístupem	Rozšíření (výkon, úložiště) \implies dodatečné náklady a starosti
Nižší/„žádné“ provozní náklady	Časově náročné nasazení a správa
Kontrola nad aktualizacemi	Personálně náročné
	Nutná údržba
	Offline přístup
	Riziko ztráty dat

dodavatele, kteří se specializují právě na úpravu **ERP** podle potřeb. Takoví dodavatelé většinou navazují spolupráce přímo s dodavatelem **ERP**. Přesto vše je pořízení **ERP** nákladné a je potřeba zvolit ten správný systém v závislosti na potřebách a velikosti podniku.

2.4.1 Architektura **ERP** systémů

Architektura **ERP**^{9} systémů se liší v na závislosti dodavateli a způsobu nasazení, přičemž existují dva základní [11]:

1. On-premise
2. Cloud

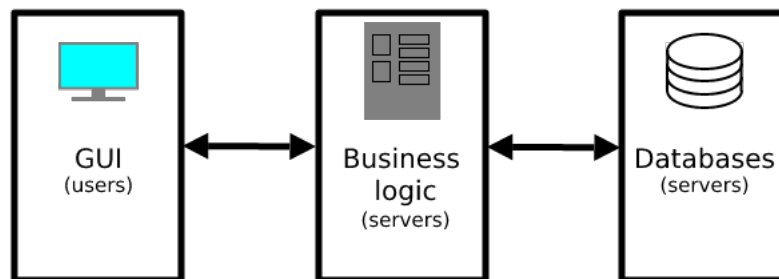
On-premise způsob nasazení byl hojně využíván dříve. Avšak s dnešními možnostmi **HW** a s dostupností a rychlostí internetu, se čím dál častěji využívá Cloudového nasazení.

On-premise

On-premise nasazení^{10} **ERP** se dá přeložit jako „na místě.“ Jak z názvu vyplývá, veškerý **HW** i **SW** běží na infrastruktuře podniku. Podnik tudíž musí zajistit dostatečný výpočetní výkon, dostatek úložného prostoru, ale i pravidelné zálohování a záložní servery v případě výpadku. Architektura tohoto způsobu nasazení je znázorněna na obrázku 3, přičemž jednotlivé vrstvy jsou znázorněny obdélníky a datové toky šipkami. Je zřejmé, že má určitou podobnost s velmi známou architekturou **MVC**⁶. Uživatel pracuje s **ERP** skrze **Graphical User Interface (GUI)**⁷. Na základě akcí uživatele se vykonává logika, o jejíž vykonání se může starat vícero serverů. Samotná logika dále může přistupovat k datům, které jsou uloženy na databázových serverech. Výhody a nevýhody tohoto způsobu nasazení jsou shrnuty v tabulce 1.

⁶**Model-View-Controller** – **SW** architektura, jejíž hlavní podstatou je oddělení logiky aplikace od prezentace obsahu

⁷**GUI** – Grafické uživatelské prostředí



Obrázek 3: Třívrstvá architektura ERP

Tabulka 2: Cloud nasazení

Výhody	Nevýhody
<ul style="list-style-type: none"> Nízké počáteční náklady Levnější a jednodušší rozšíření (výkon, úložiště) Časově nenáročné nasazení a správa Personálně nenáročné Vždy aktuální (aktualizované) Automatické zálohování Ochrana proti nenadálým událostem Dodatečný přístup k dalším Cloud rozšířením 	<ul style="list-style-type: none"> Náklady za hostování Ztráta kontroly nad přístupem Nutnost připojení k internetu Ztráta kontroly nad aktualizacemi

Cloud

Jak lze již z názvu odvodit, tak tzv. cloudové nasazení^[11] se odehrává mimo podnik, tudíž není potřeba zajišťovat vnitřní infrastrukturu, jako tomu bylo u on-premise nasazení. Systém je umístěn ve velkých datových centrech, přičemž přístup je zajištěn skrze webové rozhraní. Vyznačuje se velmi nízkými vstupními náklady (oproti on-premise odpadá nutnost zajištění vlastní infrastruktury). Funguje na principu pay-as-you-use⁸. Výhody a nevýhody cloudového nasazení lze porovnat v tabulce 2. Samotný pojem Cloud je více rozveden v kapitole 3.1.

⁸Pay-as-you-use – cena pronájmu je úměrná výši užití, tedy čím více člověk službu využívá, tím více platí (a obráceně)

Část II

Technologie a koncepty

V této části budou představeny technologie a koncepty, jež hrají a budou hrát zásadní roli v podniku (ať už ve výrobě, či v jiných jeho částech). Zprvu bude uveden pojem [Internet of Things](#) a stěžejní technologie, o které se opírá. Následuje letmý průřez průmyslovými revolucemi, který vede k nynější, poslední, 4. průmyslové revoluci. V duchu této průmyslové revoluce budou představeny stěžejní technologie pro koncept Industry 4.0, jež bude následně rozveden.

3 Internet of Things

^{12} Jedná se o vzájemně propojená zařízení či objekty (označované jako things, česky **věci**), které jsou vybaveny senzory^{13} a mají schopnost síťové konektivity [12]. Díky velikosti dnešních senzorů, můžeme být *věcí* např. senzor samotný, stroj, žárovka v domě, auto, avšak i člověk (např. s nějakým transplantátem), přičemž každá taková *věc* má svůj unikátní identifikátor ([GUID](#)⁹). *Věci* mezi sebou mohou komunikovat bez jakéhokoliv lidského zásahu, a to právě skrze síť. Síťové technologie se různí, přičemž mezi nejnámější patří [13]:

- Wi-Fi Backscatter,
- Bluetooth Low Energy,
- ZigBee ([IP](#)),
- [RFID](#),
- [NFC](#).

Na jednotlivé jmenované se podíváme blíže.

Wi-Fi Backscatter

Wi-Fi^{14} je velmi dobře známá skupina standardů pro bezdrátový přenos dat mezi zařízeními, které jsou založeny na standardu IEEE 802.11. Typicky je Wi-Fi signál generován routerem, který slouží jako spoj mezi lokální sítí a internetem. Připojená zařízení poté mohou data jak odesílat, tak i přijímat, přičemž pro komunikaci dvou (a více) zařízení v síti funguje router jako prostředník při přenosu dat. **Wi-Fi Backscatter**^{15} zachovává samotnou funkcionalitu Wi-Fi pro přenos dat a připojení, avšak zabývá se i myšlenkou přenosu energie pomocí rádiových vln. Router tedy při odesílání dat dokáže zároveň odeslat i energii potřebnou pro chod zařízení, pro které jsou data určena. Tudíž odpadá nutnost napájení mnoha zařízení, a to ať už z baterie, či ze sítě.

⁹[Global Unique Identifier](#) - jedná se o unikátní číslo pro jednoznačnou identifikaci v systému

Bluetooth Low Energy

Bluetooth^{16} je také velmi dobře známou technologií pro bezdrátový přenos dat. Avšak oproti Wi-Fi funguje na principu P2P, kdy dvě zařízení na přímo komunikují mezi sebou, zpravidla na kratší vzdálenosti. Zároveň je určen pro mnohem menší objemy dat a je více energeticky efektivní. **Bluetooth Low Energy**^{17} je odnož pro **IoT**, jejíž hlavním cílem je maximální energetická efektivita, jelikož **IoT** zařízení mohou být malá a může jich být mnoho, tudíž jejich napájení může být značnou překážkou. Bluetooth Low Energy je určený pro zařízení, která nemusí být neustále dostupná, mají nízký odběr energie a zpravidla jsou napájena mincovou baterií. Komunikace zařízení probíhá na základě Master-Slave schématu¹⁰.

ZigBee & ZigBee IP

^{18} Jedná se o volně dostupnou technologii pro bezdrátový přenos dat, jež je energeticky efektivní a má malou propustnost dat, podobně jako Bluetooth. Staví na standardu IEEE 802.15.4, který definuje fyzické a MAC vrstvy při přenosu dat. ZigBee navíc přidává vrstvy další, a to síťovou a aplikační. Jednotlivé uzly v síti mají 64bitové IEEE adresy (unikátní identifikace) a také 16bitové krátké adresy. **ZigBee IP**^{19} přidává podporu pro **Internet Protocol**¹¹, tudíž velmi zjednoduší propojení s vnějším světem. Samotné nasazení ZigBee technologie se odehrává ve čtyřech možných topologiích, jež první tři jsou velmi známé [14]:

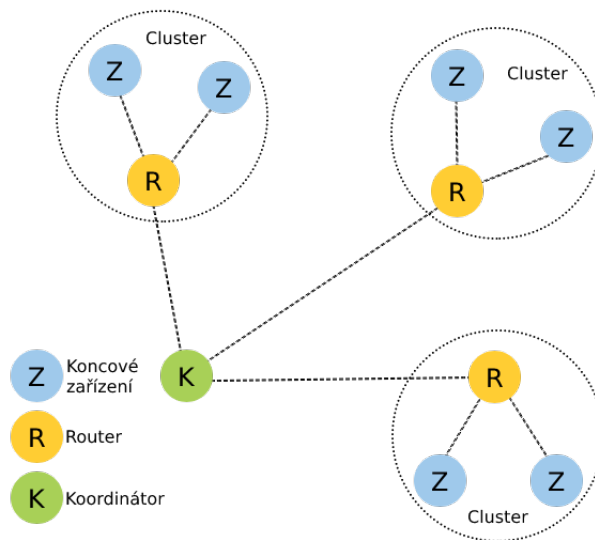
- star (hvězda),
- mesh,
- tree,
- cluster tree – viz obrázek 4.

Radio Frequency Identification (RFID)

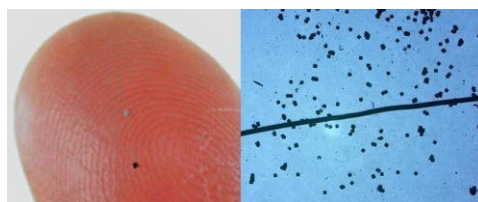
RFID^{20} je bezdrátová technologie, která využívá tzv. **RFID** tagy^{21}, ve kterých jsou uložena data. Tato data mohou být vyměněna s **RFID** čtečkami bez jakéhokoliv fyzického kontaktu, a to za pomoci elektromagnetických polí. Velkou výhodou je absence nutnosti přímé viditelnosti **RFID** tagu, tudíž čtení dat je možné i skrze obaly a zároveň stačí pouhý okamžik pro spolehlivé přečtení. Najednou lze načíst stovky tagů, které mohou být aktivní (mají nějaké napájení, jako např. baterie), či pasivní, které při kontaktu využívají energii emitovanou čtečkou (vyžadována mnohem kratší vzdálenost mezi tagem a čtečkou kvůli přenosu energie). Díky dnešním technologiím mohou být **RFID** tagy opravdu velmi malé viz obrázek 5.

¹⁰Master-Slave schéma – komunikace mezi zařízeními, při které jedno zařízení (Master) má plnou kontrolu nad zařízením druhým (Slave)

¹¹**Internet Protocol** – základní síťový protokol pracující na 3. (síťové) vrstvě ISO/OSI modelu



Obrázek 4: Cluster tree topologie
 Zdroj: [14]



Obrázek 5: Miniaturní RFID tag od výrobce Hitachi
 Zdroj: <https://thefutureofthings.com/3221-hitachi-develops-worlds-smallest-rfid-chip/>

Near Field Communication (NFC)

NFC^{22} je dnes hojně rozšířenou bezdrátovou technologií, známou zejména na poli bezkontaktního placení (např. pomocí chytrých zařízení). Technologie se vyvinula z RFID^{23}, přičemž operační vzdálenost je zde mnohem menší. Pro přenos dat jsou zapotřebí dvě různá zařízení, která když se dostanou vzájemně do napětí, tak mohou přenést krátkou informaci.

Lze si povšimnout, že všechny zmíněné technologie fungují na principu bezdrátového přenosu dat. Tento fakt je poněkud žádoucí, jelikož v opačném případě by nutnost zavést drátové připojení k jednotlivým IoT zařízením, značně zkomplikovala nasazení, údržbu, ale i obsluhu. Na téma využití IoT architektury vzniklo několik studií. Pěkným příkladem může být studie na téma chytré kanceláře, viz [15].

3.1 Cloud

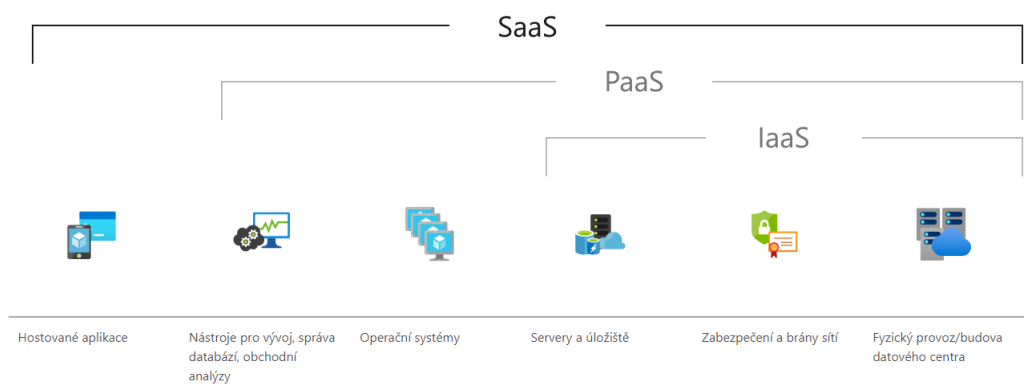
Pojem Cloud^{24} byl lehce uveden v kapitole 2.4.1. Jak již bylo zmíněno, data jsou z pohledu uživatele mimo jeho infrastrukturu, a to tzv. „v oblacích“. Prakticky jsou data uložena ve velkých datových centrech, přičemž přístup k nim je zajištěn skrze webové technologie (protokoly a služby). Největšími hráči na trhu jsou obecně známí giganti, jako:

- Microsoft – a jeho platforma Microsoft Azure [16],
- Amazon – AWS [17],
- Google – Google Cloud [18].

Cloud je dnes velmi rozšířený a využívá jej velké množství uživatelů s přístupem k internetu, i když si to nemusí nutně uvědomovat. Tradiční cloudové služby, které jsou mezi spotřebiteli hojně využívány, jsou různé zálohovací datové služby, které rovněž poskytují možnost sdílet a synchronizovat soubory (mezi zařízeními), a to např. OneDrive [19], DropBox [20] či iCloud. Avšak může se jednat i o emailové služby, jako například Gmail, ba dokonce o synchronizaci prohlížečů napříč zařízeními (Google Chrome). Faktem je, že odpadá nutnost kdejaké lokální synchronizace zařízení. Naopak mít jeden a týž soubor na mobilní zařízení, počítači a zároveň někde zálohovaný, je otázkou využití Cloudu (např. mít soubor v OneDrive složce, přičemž mít OneDrive nainstalovaný na všech takových zařízeních). Změny, které na souboru uživatel provede, jsou propány mezi všechna zařízení, jakmile dojde k jejich synchronizaci, pro kterou je samozřejmě zapotřebí (kvalitní) internetové připojení.

Avšak Cloud nezjednodušuje život pouze jednotlivcům, ba dokonce z něj mohou těžit a těží i podniky všech velikostí, státní instituce apod. V tomto ohledu s sebou Cloud přináší další tři důležité pojmy, které si více rozvedeme [13]:

- Infrastructure as a Service (IaaS) [21]



Obrázek 6: IaaS, PaaS a SaaS

Zdroj: <https://azure.microsoft.com/cs-cz/overview/what-is-saas/>

- Platform as a Service (PaaS) [22]
- Software as a Service (SaaS) [23]

Infrastructure as a Service

^{25} Díky vymoženostem dnešní doby není nutné, aby podnik měl svou vlastní infrastrukturu (servery, výpočetní výkon, úložný prostor, sítě, ...), jelikož díky „infrastruktuře jako službě“ je možné si vše potřebné pronajmout od poskytovatele. Jedná se tudíž o pronájem HW (tím pádem i jeho výpočetního výkonu), se kterým je možné svévolně (v rámci určitých podmínek sjednaných při pronájmu) nakládat. Velkou výhodou jsou velmi nízké vstupní náklady, nízké režijní náklady, podpora od poskytovatele (včetně snížení rizik v případě chyb, požárů apod.) a náklady rovnající se pouze míře využití a vytížení. Limitujícím faktorem je nutnost stabilního internetového připojení (pojem latence) a již zmíněné náklady za pronájem.

Platform as a Service

^{26} „Platforma jako služba“ staví na již představené IaaS^{27}, kde si podnik pronajímá nejen samotnou infrastrukturu, nýbrž i dodatečný SW, což může zahrnovat operační systém, databáze, různé nástroje či programy. Jedním z příkladů použití může být využít takové prostředí pro vývoj SW. Nesmírnou výhodou je absence nutnosti nákupu potřebné infrastruktury, vývojových nástrojů a licencí. Je možné si tudíž přímo pronajmout prostřední pro platformu, pro kterou chceme vyvíjet (Android¹², Microsoft Windows¹³, apod.).

Výhody a nevýhody tohoto modelu plynou z IaaS (jelikož PaaS zahrnuje i IaaS). Navíc dodavatel tohoto cloudového řešení zodpovídá nejen za infrastrukturu, nýbrž i za samotný SW, jež je součástí pronájmu, přičemž se jedná zároveň

¹²Android – operační systém pro chytrá zařízení (tablety, telefony, aj.)

¹³Microsoft Windows – operační systém pro počítače

o výhodu i nevýhodu (odběratel má o starost méně, avšak je dodavatelem odstíněn od infrastruktury).

Software as a Service

^{28} Volně přeloženo jako „software jako služba“ je nasazení **SW**, při kterém není **SW** nainstalován na zařízení klienta, nýbrž na infrastruktuře poskytovatele, přičemž přístup je zajištěn skrze webové služby. Klient nemusí ani vlastnit licenci na **SW**, jelikož tu si pronajímá spolu se **SaaS**. Tomu odpovídá i cena, jelikož klient platí pronájem v takové výši, v jaké **SW** skutečně používá. Nejčastější měrnou jednotkou bývá čas. Výhody a nevýhody plynou ze zahrnutí **IaaS**^{29} a **PaaS**^{30}, přičemž mezi další výhody patří zejména odpadnutí nutnosti nákupu licence samotného **SW**, která by mohla představovat velmi velkou investici. Shrnutí **IaaS**, **PaaS** a **SaaS** lze pěkně vidět na obrázku 6.

3.2 Industrial Internet of Things

IoT samo o sobě proniklo i do průmyslové sféry, kde je neustálá potřeba efektivních způsobů, jak optimalizovat samotný chod podniku. Pro podnik, ať už výrobní či jiný, zněla myšlenka **IoT** lákavě, jelikož zpravidla provozuje nespočet (výrobních) strojů, počítačů a jiných zařízení. Dokonce v **IoT** hojně skloňované senzory^{31} jsou v podnicích přítomny již dekády, ba dokonce i samotné stroje mezi sebou musely komunikovat dávno předtím, a to skrze **Machine-to-Machine (M2M)**^{32}. **M2M** je typ komunikace mezi stroji, a to bez napojení na okolní svět. **IIoT**^{33} ovšem nejvíce těží právě z napojení na onen vnější svět. Skrze myšlenky z **IoT** lze v **IIoT** shromažďovat velká kvanta dat (**big data**) z nejrůznějších senzorů či systémů, a ta poté pomocí cloudových technologií (= vnější svět) ukládat, potažmo **analyzovat** pro zefektivnění chodu podniku. Tyto analýzy vedou k optimalizaci nejrůznějších procesů, avšak dokážou i „předpovídat“ např. životnost strojů, jejich údržbu, a tím zvýšit produktivitu. Ve výsledku **IIoT** slouží i pro lepší pochopení byznysu.

3.3 Big data

^{34} Jedná se o strukturovaná a nestruturovaná data v takových objemech, že jejich zpracování není možné konvenčními databázemi a procesními nástroji, kterými disponuje infrastruktura běžného podniku. **Strukturovaná data** jsou zpravidla nějakým způsobem organizovaná (např. relační databáze) – data lépe čitelná a použitelná pro stroj. Pod **nestruturovanými daty** si lze představit nejrůznější formáty textu (word, pdf, příspěvek na sociální sítích), multimedia (zvuk – mp3, obrázek – png, video – avi) aj. – data lépe čitelná pro lidi.

V **IIoT**^{35} se zpravidla jedná o data ze senzorů^{36} a dalších systémů v podniku (např. i včetně dat z webových stránek). Díky velikosti nejnovějších senzorů a jednoduchého (bezdrátového) propojení, je tento pojem stále aktuálnější. Pakliže se nachází na jednom místě velké množství nestruturovaných dat, pak

se bavíme o **data lake**. Big data lze charakterizovat do několika tzv. „V“ [13] (vycházejí z pojmů v angličtině):

1. Volume – množství; čím větší objem, tím větší vypovídající hodnota
2. Velocity – rychlost příjmu a zpracování dat (na finanční data stačí batch job¹⁴, avšak data z motoru stroje je potřeba zpracovávat ihned)
3. Variety – data přichází z různých zdrojů (systémová, webová, data ze senzorů)
4. Veracity – pravdivost (nechybovost) dat; špatná data \implies špatný výsledek po zpracování
5. Value – užitečnost dat; ne všechna data jsou užitečná \implies nutnost rozpoznat data hodnotná pro podnik
6. Visibility – podnik nechce data sdílet s neautorizovanými osobami (např. konkurencí)

Big data mohou být pro podnik velmi cenná. Avšak bez správného přístupu mohou být jen nevyužitým potenciálem. Podnik musí vědět, k čemu jaká data sbírá a analyzuje. Krom již zmíněného zefektivnění procesů či údržby strojů, lze taková data použít i k trénování **Artificial intelligence (AI)**¹⁵. Díky nutnosti vytěžit z big dat tu pravou hodnotu, vzniká v podniku nové pracovní místo, tzv. data scientist, který má za úkol data analyzovat, hledat v nich vzorce, korelace a ptát se na ty správné otázky.

4 Průmyslové revoluce

Průmyslové revoluce^{37} jsou období v čase, ve kterém se měnily charakteristiky způsobu výroby. Od 18. století do nynějška byly identifikovány právě 4 průmyslové revoluce, přičemž každá se z nich má jiné charakteristiky. Společným jmenovatelem je fakt, že každá z nich zásadně mění způsob výroby. Pro pochopení kontextu si uvedeme stručnou historii a vývoj. Následně se zaměříme na nejdůležitější koncepty, na kterých staví Iniciativa Industry 4.0, která bude následně také představena.

4.1 Historie průmyslových revolucí

První průmyslová revoluce

První průmyslová revoluce^{38} se datuje do 18. století (1784), v tehdejší Anglii [24]. Revoluce se nesla v duchu **industrializace**, tedy přechodu z ruční výroby na výrobu za pomoci nejrůznějších strojů. Objevení a využití **parní energie**

¹⁴Batch job – dávkové zpracování; úloha, která je naplánovaná a zpracována v určitém čase

¹⁵**Artificial intelligence** – umělá inteligence

sloužilo jako hlavní zdroj energie jak pro stroje, tak budoucí dopravní prostředky. Nejvíce zřetelné byly změny zejména v textilním průmyslu, díky vynálezu mechanického tkalcovského stavu, který byl následně poháněn právě parními motory [25]. Suroviny pro výrobu byly získávány a dováženy zejména z kolonií, kam se následně revoluce rozšířila. Dalším velkým milníkem byl vynález parní lokomotivy a parní lodi [26]. Díky efektivnější dopravě byl jednodušší import surovin a export zboží, stejně tomu bylo s přepravou lidí. Revoluce měla i sociální dopady, jelikož díky vzniku manufaktur se zvyšovala koncentrace obyvatel ve velkých městech, spolu s obecně rostoucí populací a dožití se vyššího věku [27].

Druhá průmyslová revoluce

Druhá průmyslová revoluce^{39} byla zahájena objevem **elektrické energie**, a to v 19. století [24]. Její další charakteristikou je tzv. masová výroba \implies **montážní linky**. Jako první ji instalovala společnost Cincinnati, avšak nejznámější pohyblivou montážní linkou je ta z automobilového průmyslu, jmenovitě z automobilky Ford (Henry Ford). V České republice měl podobný úspěch obuvnický podnik Baťa [28]. Tato revoluce se vyznačovala mnoha vynálezy, jako např. žárovka (Thomas Edison) [29] či transformátor (Nikola Tesla) [30].

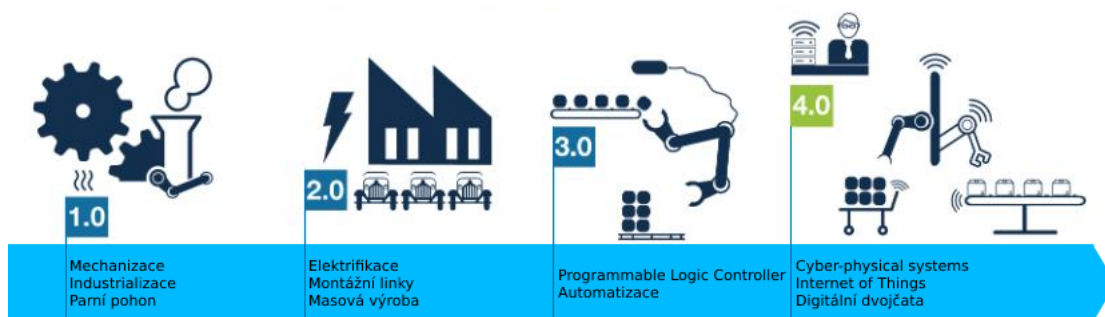
Třetí průmyslová revoluce

Ve 20. století nastala třetí průmyslová revoluce^{40}, jež se vyznačovala zejména **automatizací** [24]. Konec konců, stroje byly v továrnách již zavedeny, avšak k vykonávání svých funkcí potřebovaly lidskou pracovní sílu. Poprvé se zde začalo hovořit o pojmu počítač. Hlavní zvrát přišel s představením prvního **Programmable Logic Controller (PLC)**^{16{41}{42}}. Obecně se v této etapě velmi hojně adaptují počítače v nejrůznějších oblastech výroby a dalších podpůrných procesech.

Čtvrtá průmyslová revoluce

Aktuálně probíhající průmyslová revoluce^{43}, někdy zaměňována s pojmem Industry 4.0^{44}. Pro ni nejdůležitější technologií je **internet** \implies propojení nejrůznějších věcí/strojů (nejen uvnitř podniku). Staví na již představeném **IoT**^{45}, ale i na dosud nediskutovaných **CPS**^{46}, **DT**^{47} či **smart produktech**^{48}. Před představením samotné iniciativy Industry 4.0 si tyto pojmy probereme. Lze si všimnout, že s každou průmyslovou revolucí se měnily potřeby na zaměstnance. Čím dál tím méně jsou od nich potřeba fyzické úkony, naopak začíná převažovat potřeba intelektuálních. Naprostou většinou manuálních prací zvládnou stroje (dokonce i lépe a rychleji), avšak je potřeba lidí, které tyto stroje programují, sbírají data aj. Všimněme si, že stroj může klidně postavit další stroj. Přibývá tedy potřeby zaměstnávat kvalifikované zaměstnance, a naopak upadá potřeba nekvalifikovaných. Letmý průřez revolucemi lze vidět na obrázku 7.

¹⁶Programmable Logic Controller – Programovatelný logický automat je malý programovatelný počítač, jež se používá právě pro automatizaci strojů. [31]



Obrázek 7: Průmyslové revoluce

Zdroj: <https://www.oecd.org/innovation/the-next-production-revolution-9789264271036-en.htm>

4.2 Cyber-physical system

Jak již název může napovídat (česky Kyberfyzikální systém), jedná se o spojení dvou zdánlivě odlišných světů, a to kybernetického (popř. digitálního) a fyzického. Prakticky to znamená, že bavíme-li se o **CPS**^{49}, pak takový systém musí „pracovat“ v obou zmíněných světech \implies musí mít jak stránku výpočetní, tak i fyzickou. Tyto dvě stránky jsou úzce spjaté ve smyslu vzájemného ovlivňování (na základě rozhodnutí výpočetní stránky může být ovlivněn fyzický proces a naopak) [32]. Výpočetní stránka je poměrně intuitivní, jedná se totiž o jakýsi mozek (analogie s člověkem), kde probíhá zpracovávání, rozhodování, výpočty aj. Naopak fyzickou stránku zastávají senzory a aktuátory. Tyto dva členy si lehce definujeme, jelikož ať už v **IoT**, kybernetickém prostoru obecně, či v samotné výrobě (můžeme rozumět přímo výrobní linku v podniku), hrají stále významnější roli.

Senzor^{50} je součástka, která dokáže sledovat a měřit rozdíly určitých sledovaných vlastností okolí či předmětu (např. vlhkost vzduchu, tlak, opotřebení, světlost aj.) [4]. Tyto informace poté dokáže v digitální formě předat dál pro zpracování [33].

Aktuátor^{51} je díl, typicky zakomponovaný v mechanických zařízeních, který slouží k vykonání nějaké pohybové činnosti [34]. např. senzor aktivně měří světlost v pokoji a předává tuto informaci dál do systému. Pakliže světlost přesáhne určitou úroveň, systém vydá pokyn aktuátoru, který zatáhne žaluzie.

Měli-li bychom popsat **CPS** formálněji:

Definice 3 (Cyber-physical system)

Cyber-physical system je systém, který má následující tři schopnosti/vlastnosti [13] [35]:

1. Fyzikální – řízení, sledování (aktuátory^{52}, senzory^{53})
2. Síťovou – schopnost komunikace s okolím (další **CPS**, jiné stroje, systémy, ...)

3. Výpočetní – rozhodování, výpočty, pokyny aj.

Klíčová je integrace již zmíněné výpočetní a fyzické komponenty (resp. jejich světů). Všimněme si, že 2. bod nám umožňuje začlenění CPS do IoT \implies komunikace s ostatními zařízeními přes cloud [36]. Není totiž náhodou, že tyto dva pojmy spolu pěkně korelují (a společně tvoří jedny z pilířů Industry 4.0). Pro upřesnění, CPS není podmnožinou IoT ani na něm není nijak nutně závislý. Co se týče interakce či zásahů člověka, tak není definováno žádné omezení, tudíž CPS může fungovat jak s nimi, tak i bez.

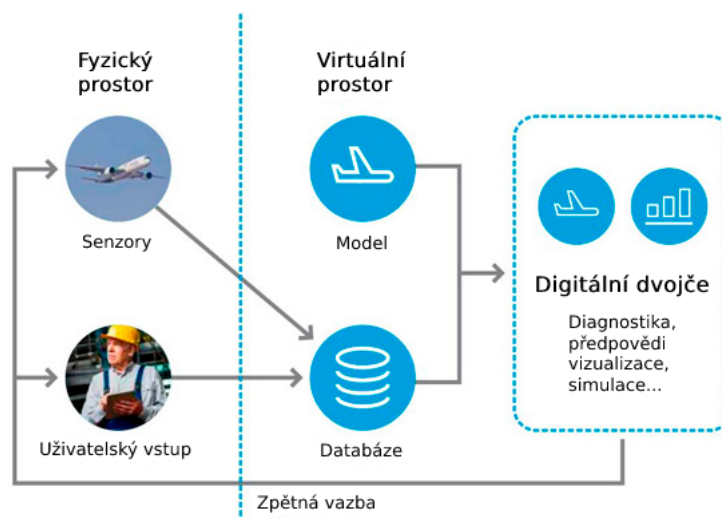
Můžeme si uvést pár příkladů CPS, přičemž pro identifikaci jednotlivých schopností/vlastností zachováme číslování z formální definice výše viz definice 3:

- **Systém hlídání jízdních pruhů ve vozidle**, přičemž jeho součásti jsou následující:
 1. Sensory pro hlídání čar na silnici
 2. Napojení FBT¹⁷, u vozidel konkrétně na CAN [13]
 3. Mikroprocesor, jednotka, ...
- **Robot na výrobní lince**:
 1. Sensory a aktuátory
 - senzor pro hlídání opotřebení robota,
 - senzor pro kontrolu dílů, součástek, výrobku,
 - aktuátory pro vykonávání pohybu.
 2. např. M2M¹⁸, nebo i Wi-Fi aj.
 3. Řídící jednotka robota
- **Člověk?**:
 1. Ruce
 2. Komunikace s ostatními lidmi (řeč, gesta) a stroji (HMI¹⁹)
 3. Mozek

¹⁷Field Bus Technology – jednotné označení pro technologie, které pracují na principu komunikace přes sběrnici (každé zařízení je na ni napojeno a může komunikovat s ostatními)

¹⁸Machine-to-Machine – komunikace pouze mezi jednotlivými stroji, žádné povědomí o okolním světě

¹⁹Human Machine Interface – Rozhraní sloužící pro interakci člověka se strojem, systémem, počítačem.



Obrázek 8: Digital Twin

Zdroj:

<https://www.anylogic.com/upload/medialibrary/1b6/1b6b62f3bfae9c36aa7ae00f324f6f4e.jpg>

4.3 Digital Twin

Digital Twin (DT)^{{54}{55}} je jeden ze základních pilířů Industry 4.0^{56}. Jako první tento koncept představila a následně využila NASA, pro své kosmické lodě [37]. Myšlenka spočívá v tom, že kosmická loď měla své tzv. digitální dvojče, které mělo za úkol simulovat chování svého fyzického dvojčete. Simulace chování byly prováděny na základě jak historických dat, tak i real-time dat, tedy dat ze senzorů. Na digitálním dvojčeti mohla být kromě simulací prováděna i diagnostika či předpovědi nutnosti údržby aj.

Zatím nejmenované průmyslové koncepty počítají s aplikací konceptu DT právě v podniku. Díky myšlenkám IoT, tedy zejména hojnému zapojení senzorů^{57} a sběru velkého množství dat (big data^{58}), může mít každý stroj, v tomto ohledu lze již říci že i každý CPS^{59}, své vlastní digitální dvojče, na kterém může podnik simulovat nejrůznější situace. Nicméně nejedná se jen o jednosměrnou cestu dat od fyzického stroje ke svému digitálnímu dvojčeti. Na základě real-time hodnot ze senzorů, může digitální dvojče vyhodnocovat aktuální stav a předávat cenné informace svému fyzickému protějšku. Zároveň do tohoto konceptu může vstupovat i samotný uživatel. Tato myšlenka je znázorněna na obrázku 8.

4.4 Iniciativa Industry 4.0

Původním názvem Industrie 4.0^{60}, nebo také česky Průmysl 4.0, je iniciativa, která původně pochází z Německa. Na základě zadání tehdejší německé vlády vznikl dokument popisující Industrie 4.0 (původní název), který adresoval 4. průmyslovou revoluci. Následně se ke 4. průmyslové revoluci postavily i další státy

Tabulka 3: Dvacet profesí s největším indexem ohrožení digitalizací

Název profese	Index ohrožení digitalizací
Úředníci pro zpracování číselných údajů	0,98
Všeobecní administrativní pracovníci	0,98
Řidiči motocyklů a automobilů (kromě nákladních)	0,98
Pokladníci a prodavači vstupenek a jízdenek	0,97
Kvalifikovaní pracovníci v lesnictví a příbuzných oblastech	0,97
Kováři, nástrojaři a příbuzní pracovníci	0,97
Ostatní úředníci	0,96
Sekretáři (všeobecní)	0,96
Obsluha pojiždných zařízení	0,96
Chovatelé zvířat pro trh	0,95
Pomocní pracovníci v zemědělství, lesnictví a rybářství	0,95
Obsluha zařízení na těžbu a zpracování nerostných surovin	0,94
Obsluha strojů na výrobu a zpracování výrobků z pryže, plastu a papíru	0,94
Úředníci v logistice	0,94
Montážní dělníci výrobků a zařízení	0,93
Obsluha strojů na výrobu potravin a příbuzných výrobků	0,93
Pracovníci s odpady	0,93
Pokladníci ve finančních institucích, bookmakeři, půjčovatelé peněz, inkasisté pohledávek a pracovníci v příbuzných oborech	0,93
Strojvedoucí a pracovníci zabezpečující sestavování a jízdu vlaků	0,92
Ostatní obsluha stacionárních strojů a zařízení	0,92

Zdroj: [39]

Evropy, Čína, Indie a další, kteří tuto myšlenku dále rozvíjeli. V České republice byl v roce 2017 představen dokument vypracován Ministerstvem průmyslu a obchodu, nazvaný *Iniciativa průmysl 4.0*^[61] [38]. Počítá se zejména s digitalizací a autonomizací, což bude mít i vliv na samotný trh práce. Tabulka 3 ukazuje 20 profesí, které jsou digitalizací nejvíce ohroženy.

Nejdříve se podíváme na jednotlivé stavební kameny, které jsou a budou v Industry 4.0 hojně využívány. S pochopením těchto pojmů si následně představíme charakteristiky Industry 4.0 a tím i jeho cíle.

4.4.1 Základní stavební kameny

Staví se na několika pojmech, které již byly představeny. Mezi hlavní pilíře patří zejména CPS^[62], IoT^[63] a DT^[64]. Na základě těchto tří konceptů se nám otevírají široké možnosti propojení jak ve fyzickém, tak i digitálním světě. Podívejme se na další stavební kameny Industry 4.0:

- Smart products,
- Autonomní roboti/stroje,
- Simulace,
- [Augmented Reality](#),

- Big Data a analytika \implies Cloud,
- Kybernetická bezpečnost,
- Additive manufacturing (aditivní výroba),
- Smart Factory (chytrá továrna).

Lze vidět, že se počítá s masivní digitalizací. Data se stávají nepomíjitelnou součástí strategie podniku. Díky efektivnímu sběru a analýze dat ze všech koutů továrny, lze zefektivnit proces výroby napříč celým hodnotovým řetězcem (pojem je definován v definici 4). V následujících odstavcích se více podíváme na ony stavební kameny a usadíme je do celého konceptu.

Smart product

Začíná se již u samotného produktu. Doteď byl výrobek „hloupý“, jelikož v sobě nenesl žádnou informaci při svém výrobním procesu. Idejí je, aby každý vyráběný produkt dokázal identifikovat, čím vlastně je, v jakém stavu výrobního procesu se nachází, jaký je jeho stav, co potřebuje apod. Díky dnešním technologiím lze tohoto dosáhnout poměrně snadno, a to za pomoci již zmíněných miniaturních RFID tagů^{65} diskutovaných v kapitole 3. Díky nim bude moci produkt aktivně vstupovat do výrobního procesu, který nemusí mít jasně danou „cestu“ (představme si pás na výrobní lince), nýbrž může sám komunikovat se stroji, které vstoupí do onoho procesu výroby. Tím dostaneme něco, co Industry 4.0 nazývá tzv. **Smart product**^{66} (česky přeloženo jako chytrý produkt).

Zde není omezení pouze na RFID tagy. Lze využít i jinou, mnohem starší a známější technologii, jakou jsou čárové kódy. Nicméně technologie pokročily, a jak již bylo diskutováno, dnešní RFID tagy mohou být miniaturní a přesto pojmu větší objem informací.

Existují specifické případy, kdy není možné využít žádnou fyzickou technologii, která by na výrobku byla přítomna. Například při výrobě výkovek (fyzické opracování, teploty aj.). V takovém případě lze využít DT^{67} produktu, které by daná data uchovávalo. Jakmile nebude výrobek podléhat dalšímu fyzickému opracování, je možné ho RFID tagem opatřit a přenést do něj informace z jeho DT. Samotný výrobek totiž ještě musí být distribuován k zákazníkům.

Autonomizace (výrobního procesu)

^{68} Dalším faktorem při výrobě jsou samotné stroje, které jsou přímo i nepřímo zodpovědné za výrobu daného produktu. Aby *smart product* dostal svého potenciálu, je potřeba využít jeho možnosti do něho zapisovat a z něho číst informace. Samozřejmě by něco takového mohl dělat člověk za pomoci RFID čtečky, avšak účelem není přidat zaměstnancům další manuální a repetitivní práci. Mnohem lepším způsobem je, když s produktem přímo komunikuje výrobní linka. Ba dokonce, *smart product* si může říct, co přesně potřebuje ke své vlastní výrobě a dle

toho je nastaven výrobní proces, to vše autonomně. Produkt by jednak určoval, co potřebuje ke své výrobě a zároveň by uchovával celý svůj výrobní proces, včetně případných chyb, které by jej degradovaly (kontrola kvalita napříč výrobním procesem). Tím pádem by dokázal informovat i o své chybovosti, aby se tím pádem nedostal do rukou zákazníků. Všechny tyto informace by mohly být (klidně i real-time) analyzovány díky sensorům a uloženým datům (včetně procesu výroby). Díky **autonomii strojů**, potažmo celého **výrobního procesu**, by pracovníci nemuseli vykonávat nudné, repetitivní, manuální práce. Z „pouhých strojů“, které mezi sebou již nějakým způsobem komunikovaly (M2M), se nám stanou komplexní **CPS**^{69}, s vlastními **DT**^{70}, které nejenže komunikují mezi sebou, ale pracují nezávisle na člověku, také se učí, ale i dokážou předvídat na základě dat. Dokonce tato data umí aktivně komunikovat a např. oznámit potřebu nadcházející údržby či možnou závadu. V reakci na to nemusí být zastavena celá výrobní linka, nýbrž bude tok výroby pouze přeměrován a úkol vyřazeného stroje zastane stroj jiný.

Simulace

V ruku v ruce s **DT**^{71} přichází i rozšířená možnost **simulací**^{72}. Díky přítomnosti a využití **DT**^{73} lze mít kopii továrny v digitálním světě. Na této virtualizované továrně je posléze možné provádět nejrůznější komplexnější simulace (třeba procesu výroby), čímž odpadá nutnost tzv. „pokus-omyl“ přístupu. Rovněž lze simulovat i různé situace, jako je výpadek stroje, defektní produkt, či dokonce přiřazení nového stroje do výroby. Tímto způsobem lze zlevnit a zjednodušit rozšiřování dané továrny.

Augmented Reality

Je zřejmé, že digitalizací a automatizací vzniká tlak na zaměstnávání více kvalifikovaných zaměstnanců, jejichž trénink a školení mohou být pro podnik velmi nákladné, a hlavně časově náročné. Nicméně i tento problém Industry 4.0 částečně adresuje, a to díky již známé technologii **Augmented Reality (AR)**^{74}. Tato technologie spočívá v přidání digitálního objektu do fyzického prostředí, a to například za pomoci kamer a obrazovek, či například brýlí. Na základě skenu fyzického prostředí ve spolupráci s nejrůznějšími senzory lze potom dosazovat objekty, ale třeba i instrukce. Například je možné naskenovat stůl s několika součástkami a následně v **AR** přehrát simulaci skládání součástí do finální podoby pro vznik nějaké produktu. Spolu s tím lze promítat různé doplňující informace a instrukce. Tudiž je možné, aby se zaměstnanci učili tzv. v reálném čase. Faktem nicméně zůstává, že bude kladen větší důraz na intelektuální práci. Nicméně se předpokládá, že díky Industry 4.0 by měly vzniknout lepší pracovní podmínky a měl by s sebou přinést i lepší work-life balance, což by se následně mělo promítnout i do životní úrovně.

Využití lze samozřejmě nalézt i při samotné výrobě. **AR** se dá využít i ke zkoumání výrobků a snížení chybovosti při výrobě. Takové informace lze poté

v reálném čase předávat přímo zaměstnancům, například přímo do zmíněných brýlí.

Cloud, IoT

Samozřejmě tohle vše by nebylo možné bez již diskutovaného **IoT**^{75} a hlavně možností **Cloudu**^{76}, který byly rozebrány v kapitole 3, potažmo 3.1. Díky jeho výpočetním a hlavně datovým možnostem, podniku neutěchou drahocenná data (**big data**^{77}) z procesu výroby, která nejenže poslouží samotným strojům apod., avšak lze je i zpětně analyzovat a porovnávat za účelem zvýšení efektivity či zkoumání různých možností rozšířený továrny.

Kybernetická bezpečnost

Spolu s napojením na vnější svět ovšem přichází řada rizik. **Kybernetická bezpečnost**^{78} je velkou otázkou posledních několika desítek let. Je velmi důležité, aby přenos dat byl bezpečný a spolehlivý. Kdekdo by se mohl domnívat, že se bavíme pouze o strojích, kabelech, internetu atd., avšak zabezpečení musí začínat již u samotných zaměstnanců. Velké část útoků na podnik dnes probíhá skrze tzv. sociální inženýrství. V dnešní době dokážou být phishingové²⁰ emaily natolik přesvědčivé, že právě jeden ze zaměstnanců může být tzv. vstupním bodem pro útočníka, skrze který se mnohem jednodušeji dostane k datům podniku. K čemu nám je šifrovaná komunikace, pokud se útočník dostane k samotnému zdroji, a tudíž ji ani nepotřebuje odchyťovat. Pakliže Industry 4.0 počítá s přesunem zaměstnanců od manuální k intelektuální práci, pak je zřejmé, že přibude zaměstnanců s přístupem do informační sítě podniku (zaměstnanec u pásu dříve nutně nepotřeboval místní firemní účet pro přístup do sítě, avšak přesune-li se na pozici, kdy bude pouze dohlížet na stroje, které přebraly jeho manuální práci, pak bude zajisté potřebovat místní účet). Proto musí být kladen velký důraz na kybernetické zabezpečení firmy počínaje zaměstnanci, přes vlastní firemní infrastrukturu, avšak i na přenos dat do Cloudu. Poslední část by měl obstarat právě poskytovatel onoho Cloudového řešení, avšak tuto problematiku nelze brát na lehkou váhu, jelikož datová centra mohou být v úplně jiných geografických oblastech s různými legislativami ohledně zacházení s daty [38].

Additive manufacturing (aditivní výroba)

S nástupem Industry 4.0 se počítá i s větším využitím tzv. **Additive manufacturing (aditivní výroba) (AM)**^{79}, zpopularizované pod názvem **3D tisk**^{80}. Tato technika výroby spočívá v nanášení jednotlivých vrstev materiálu na sebe, což vede ke vzniku 3D objektu. Před samotnou výrobou je ovšem

²⁰Phishing - jedná o jeden z kybernetických útoků, při kterém útočník na základě sociálního inženýrství zjistí patřičné detaily o oběti, které následně formou elektronické komunikace odešle natolik přesvědčivou a autenticky se tvářící zprávu, že oběť nebude mít nutný důvod domnívat se, že se může jednat o podvod

potřeba daný objekt vymodelovat. Jednou z možností je použít 3D skener pro naskenování již existujícího objektu, avšak takový postup je možný pouze v případě, že takový objekt již máme. Pakliže chceme **AM** využít k prototypování, pak je třeba objekt vymodelovat. Tento proces zpravidla probíhá v nejrůznějších 3D modelovacích programech či CAD programech, přičemž existují různé způsoby modelování (např. množinové operace nad objekty, modelování na základě deformací objektu, ...). Velkým přínosem pro podnik může být již zmíněné prototypování. Využitím **AM** lze takto ušetřit čas, a hlavně snížit náklady. Další výhodou může být možnost vyrobit specifické produkty v malém množství, avšak za přijatelnou cenu.

Smart Factory (chytrá továrna)

Všechny zmíněné stavební kameny dohromady utvoří něco, co Industry 4.0 nazývá **Smart Factory (chytrá továrna) (SF)**^{81}. Cílem chytré továrny je navýšení efektivity a snížení množství odpadu (špatné výrobky, odpadní materiál z výroby, spotřeba energií, ...). Použitím již zmíněných konceptů a technologií toho lze dosáhnout. Nicméně **SF** jde ještě dál. Očekává se, že taková továrna bude natolik flexibilní a inteligentní, že bude dosahovat za zdi továrny, bude vstupovat do produktu a celého hodnotového řetězce. Bude umět reagovat na vnější podněty, jako jsou výše objednávek, schopnosti dodavatelů, ceny energií atd. Avšak je potřeba si uvědomit, že **SF** již existují a jsou mezi námi, i když ne nutně se všemi popsány vlastnostmi. Jako v každém procesu a adaptaci, se musí jednat o postupný vývoj.

4.4.2 Charakteristiky

Industry 4.0^{82} má 3 základní charakteristiky:

- Vertikální integrace (chytrých) produkčních systémů
- Horizontální integrace prostřednictvím globálních sítí hodnotového řetězce
- Průběžné inženýrství v celém hodnotovém řetězci

přičemž samotným cílem je zejména zrychlení výroby a její zefektivnění bez toho, aniž by byla kompromitovaná kvalita produktů. Tyto charakteristiky a cíle patřičně zohledňuje i model **RAMI 4.0**, který je rozveden v kapitole 7. Níže si jednotlivé charakteristiky více rozvedeme.

Vertikální integrace

^{83} Dnešní (chytré) produkční systémy mají standardizovanou podobu, která vychází z normy **S95**, která je podrobněji rozvedena v kapitole 6. Jedná se o hierarchii subsystémů (v produkčním systému) a jejich vzájemnou komunikaci. Industry 4.0 má potenciál a tendenci tuto hierarchii pozměnit. S příchodem **CPS**, které by měly být autonomní, může být výrobní proces mnohem flexibilnější,

jelikož jednotlivé CPS budou moci komunikovat mezi sebou, přímo reagovat na přání zákazníků, změnu skladových zásob, či na různé poruchy, ale například budou i moci komunikovat přímo s cloudovými službami.

Horizontální integrace

Abychom mohli rozebrat horizontální integraci^{84}, je potřeba se nejdříve podívat, co znamená pojem hodnotový řetězec.

Definice 4 (Hodnotový řetězec)

Hodnotový řetězec^{85}, definován Michaelem Porterem, je soubor aktivit, které podnik vykonává za účelem dodání svého produktu zákazníkům [40]. Má dva základní typy činností, horizontální (primární) a vertikální (podpůrné).

Hodnotový řetězec slouží podniku k identifikování a rozebrání jednotlivých aktivit, které vstupují do produktu. Je třeba je patřičně zanalyzovat a docílit co nejvyšší efektivity v každé z nich včetně vzájemné provázanosti a propojení mezi nimi, čímž podnik dosáhne konkurenční výhody. Horizontální činnosti jsou následující:

- Příchozí logistika
- Výroba a provoz
- Odchozí logistika
- Prodej a marketing
- Servis

Naopak do vertikálních, tedy podpůrných činností, patří:

- Infrastruktura podniku
- Personalistika
- Technologický rozvoj
- Materiálové hospodářství

Veškeré činnosti podniku by měly zapadat do jedné z uvedených činností. Právě provedením analýzy hodnotového řetězce může podnik dospět k závěru, že pro něj není výhodné vyrábět určitou část pro svůj produkt a je výhodnější takovou část nechat vyrobit u jiného subjektu \implies navázání partnerství. A to je také podstatnou horizontální integrace. Jedná o **propojení hodnotových řetězců napříč podniky** (dodavatelé, výrobci, distributoři), čímž budou moci všichni jednotně reagovat na změny v požadavcích zákazníků, změny skladových zásob, či poruchy aj. Tím se může podnik soustředit na užší výrobní proces, zjednodušit jej a snížit své náklady.

Průběžné inženýrství v celém hodnotovém řetězci

Spolu s dvěma předchozími body jde v ruku v ruce tzv. průběžné inženýrství^{86}. Je potřeba, aby byl produkt sledován od návrhu, prototypů, materiálu potřebného k výrobě, přes výrobu samotnou a distribuci, až k samotnému zákazníkovi, pro zajištění patřičné kvality a poprodejního servisu. Tudíž je důležité mít informace o **celém životním cyklu produktu** \implies zpětná vazba, analýza dat.

Část III

Normy a standardy

V této části budou zprvu uvedeny dvě normy, jež měly zásadní vliv na proces výroby (resp. standardizovaly pohled na přístup a systémy, jež výrobu řídí). Tyto normy si představíme a uvedeme jejich fundamentální principy, jejich návaznost a myšlenky. V závěru bude uveden standard, jež formalizuje myšlenky představeného konceptu Industry 4.0.

Norma je sada vysvětlení, rad, doporučení aj., ohledně určité problematiky. Zpravidla jsou poměrně detailní a definují mnohé pojmy vázající se k dané problematice. Cílem norem není říct, jak se co **musí** dělat, nýbrž co bychom **měli** udělat, abychom dosáhli kýženeho cíle [41]. Na vývoji norem se zpravidla podílí široké spektrum odborníků z mnoha různých podniků. Normy jsou důležité pro, pokud možno, udržení jednotného přístupu k dané problematice. Zároveň usnadňují práci a jsou již ověřenými nástroji, jelikož do jejich vývoje vstupuje mnoho odborníků a jsou patřičně promyšleny.

5 ANSI/ISA-88

Norma [ANSI/ISA-88 \(S88\)](#)^{87} definuje způsoby návržení a vytvoření systému pro řízení a implementaci **dávkové výroby**. Původně vznikla v Americe, nicméně je aplikována a využívána i v Evropě, a to pod označením IEC 61512 [42]^{88}. Aktuálně se skládá z následujících 4 částí:

1. Část 1. – Modely a terminologie
2. Část 2. – Datové struktury a metodické pokyny pro jazyky
3. Část 3. – Modely a reprezentace obecných a místních receptur
4. Část 4. – Záznamy dávkové výroby

Norma je spíše obecným, dostatečně detailním, popisem problematiky implementace dávkové výroby. Je aplikovatelná na širokou škálu různých implementací. Normu vydal a udržuje výbor SP88^{89} (patřící přímo pod [ISA](#)), který se skládá z mnoha odborníků z nejrůznějších podniků. Je určena pro lidi zapojené do návrhu, vývoje či přímo provozování dávkových výroby; zodpovědné za specifikaci řízení a souvisejících aplikačních programů pro dávkové výroby; či pro lidi zapojené do návrhu a marketingu produktů v oblasti dávkové výroby [43].

[S88](#) cílí zejména na dávkový proces výroby, avšak aplikace normy je možná i na zbylé dva druhy výrobního procesu, které jsou představeny níže.

Před samotným pohledem na normu [S88](#), je potřeba vysvětlit a definovat několik nezbytných pojmů, jež se v normě používají. Prvním pojmem je **výrobní proces**^{90}.

Definice 5 (Výrobní proces)

Proces, při kterém je na vstupní materiál aplikována řada kroků, jež vyústí ve výstup v podobě polotovarů či produktů.

Z hlediska normy lze výrobní proces rozdělit na tři různé:

1. Kontinuální výroba
2. Diskrétní výroba
3. Dávková výroba

Kontinuální výroba ^{91} je takový výrobní proces, při kterém materiál nepřetržitě prochází přes jednotlivá výrobní/zpracovatelská zařízení, např. výroba energií, paliv, ... [43]

Diskrétní výroba ^{92} je výrobní proces, při kterém součástka (či více součástek) prochází přes vícero různých výrobních/zpracovatelských zařízení, přičemž každé z nich mu přidá určitou „hodnotu“. Každý takový výrobek či přidaná součástka se vyznačuje svou vlastní unikátní identitou, kterou si zachovává po celou dobu výrobního procesu. Typicky výrobní linka, tedy např. výroba automobilů [43].

Pro definici samotné **dávkové výroby**, je potřeba nejdříve definovat samotný pojem dávky:

Definice 6 (Dávka (batch))

Dávka^{93} (batch)^{94} je výstupní produkt, či polotovar, dávkového procesu (výroby) (batch processingu). Exaktní definice jsou [43]:

1. Vyráběný nebo již vyrobený materiál v rámci jednoho provedení dávkové výroby.
2. Subjekt představující výrobu materiálu v jakémkoliv bodě výrobního procesu

Samotný dávkový proces je definován následovně:

Definice 7 (Dávková výroba^{95}, dávkový proces^{96} (batch process^{97}))

Proces, který vede k výrobě konečného množství materiálu, při podrobení určitého množství vstupního materiálu konečné uspořádané množině zpracovatelských aktivit během konečného časového úseku, za použití jednoho či více zařízení [43].

Při dávkové výrobě je tedy vstupní materiál (typicky se jedná o materiál, který lze měřit v litrech, kilech apod.), podroben určitým krokům, čímž nám

vznikne dávka. Taková dávka ovšem nemusí být konečným produktem, nýbrž polotovarem a může být použita jako vstup pro další dávku. Na jedné výrobní lince může být vyráběno více podobných produktů, pouze za změny receptury (jiný vstupní materiál, poměry, jiné kroky, ...). Dávkové procesy jsou typicky vykonávány v cyklech. Konečná dávka (tudíž dávka, jež je již produktem), se nazývá **šarže** (anglicky lot).

5.1 Model dávkového procesu

{98} Norma dále rozšiřuje vymezení pojmu dávkové výroby, přičemž identifikuje jednotlivé podčásti, ze kterých se tento proces skládá. Jedná se o hierarchické uspořádání s určitými pravidly představenými níže.

Fáze procesu {99} Dávkový proces, jako takový, se skládá z neprázdné uspořádané množiny **fází daného** procesu. Tyto fáze mohou být jak paralelní, tak sériové, přičemž v nich dochází k chemickým či fyzickým změnám vyráběného produktu (např. *namíchání určité substance – na základě různých chemických reakcí, vytvarování produktu – různým ohýbáním*).

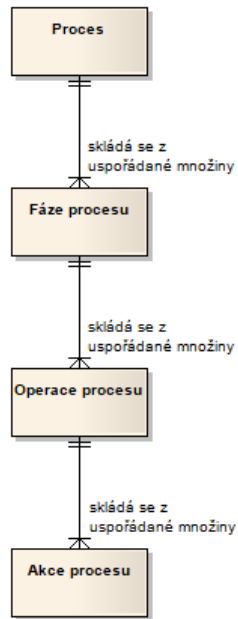
Operace procesu {100} Jednotlivé fáze procesu se nadále skládají z **operací** v dané fázi, které vyústí v určitou chemickou či fyzickou změnu vyráběného produktu (např. *chemická reakce na produktu – přidáním ingrediencí, ohnutí produktu – zahřátím a ohnutím*). Opět se jedná o neprázdnou uspořádanou množinu.

Akce procesu {101} Nejmenší jednotka, na kterou lze dávkových proces rozložit, je **akce** (např. *přidat ingredienci, zahřát povrch, ohnout výrobek*). Pro množinu akcí platí totožná podmínka, jako u operací (a fází). Jednotlivé části dávkového procesu lze vidět na obrázku 9.

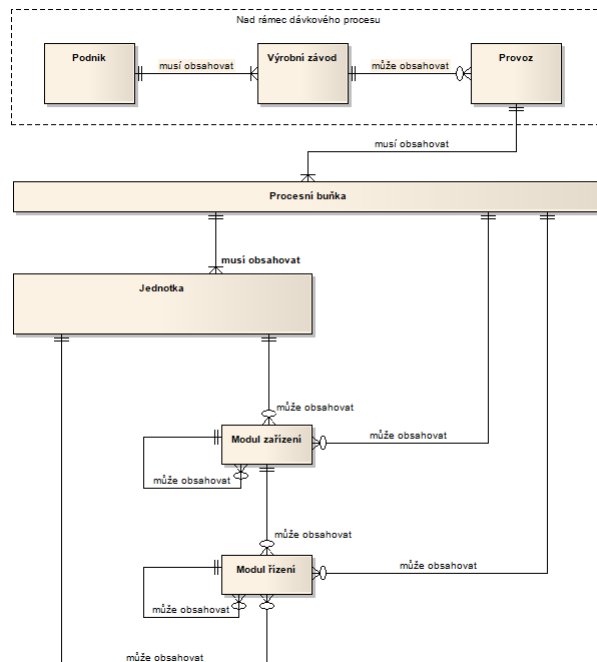
5.2 Fyzický model

{102} Norma dále uvádí, jak zařadit a zorganizovat fyzická aktiva v podniku. Vychází z nejobecnější jednotky (*podnik*) po nejspecifičtější. Nicméně jelikož první tři jsou chápány nad rámec dávkové výroby, a tedy i celé normy, tak jsou uvedeny jen pro zasazení do kontextu celého podnikové výrobního procesu. Hierarchie je analogická modelu dávkového procesu a lze ji vidět na obrázku 10. Jednotlivé složky modelu a jejich popis s příkladem následuje:

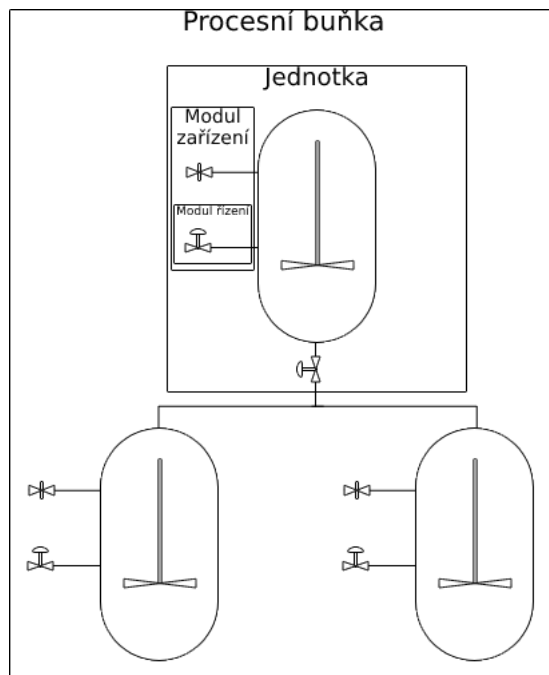
- Podnik^{103} – soubor jednoho či více závodů, ve kterých koordinuje výrobu, např. Siemens
- Výrobní závod^{104} – fyzické, geografické či logické seskupení určené podnikem, např. továrna v Mohelnici; továrna v Brně



Obrázek 9: Dávkový proces – ERD
Zdroj: [43]



Obrázek 10: Fyzický model 2010 – ERD
Zdroj: [44]



Obrázek 11: Složky fyzického modelu

- Provoz^{105} – fyzické, geografické či logické seskupení určené závodem, např. provoz výroby elektromotorů
- Procesní buňka^{106} – logické seskupení různých zařízení potřebných k výrobě jedné či více dávek, např. výrobní linka
- Jednotka^{107} – uskupení modulů zařízení a řízení vykonávajících určitou procesní aktivitu na dávce, přičemž vždy pracuje jen na jedné dávce; soustředí se na určitou část výrobního procesu
- Modul zařízení^{108} – vykonává konečné množství malých výrobních aktivit v dané jednotce, např. vážení, dávkování, přidávání; může se skládat i z dalších modulů zařízení
- Modul řízení^{109} – seskupení senzorů^{110}, aktuátorů^{111} a dalších modulů řízení, např. zařízení kontrolující průtok

Procesní buňky, jednotky, moduly zařízení a moduly řízení, spolu se svými „řídícími systémy“, jsou označovány jako **subjekty vybavení**^{112}. Pro lepší pochopení si lze jednotlivé složky prohlédnout na obrázku 11.

5.3 Model procedurálního řízení

^{113} Jedná se v podst. o návod, jak zorganizovat jednotlivé procesy výroby, včetně receptů, zařazení do fyzického modelu (fyzická aktiva podniku) atd. Tuto organizaci definuje jako dávkové řízení:



Obrázek 12: Model procedurálního řízení – ERD
Zdroj: [43]

Definice 8 (Dávkové řízení (batch control))

^{114} ^{115} Kontrolní aktivity a funkce, které poskytují prostředky k zpracování konečného množství vstupního materiálu, při podrobení konečné uspořádané množině zpracovatelských aktivit během konečného časového úseku, za použití jednoho či více zařízení [43].

Jsou identifikovány tři typy dávkového řízení, přičemž se nejedná o pouhé rozdělení, nýbrž jeden typ může mít vliv na druhý:

1. Základní řízení
2. Procedurální řízení
3. Koordinační řízení

Základní řízení ^{116} je typ řízení, při kterém se dosáhne a udržuje specifický stav zařízení či procesu.

Procedurální řízení ^{117} ve své podstatě definuje co a jak má jednotlivé zařízení (chápejme stroj apod.) udělat k dosažení dané úlohy výrobního procesu, tedy k výrobě nějakého produktu. Tento typ řízení je charakteristický pro dávkové procesy, přičemž se skládá ze 4 složek (**procedurálních prvků** ^{118}),

příčemž jejich hierarchické uspořádání je analogické tomu u dávkového procesu, jak lze vidět na obrázku 12. Jednotlivé složky jsou definovány následovně:

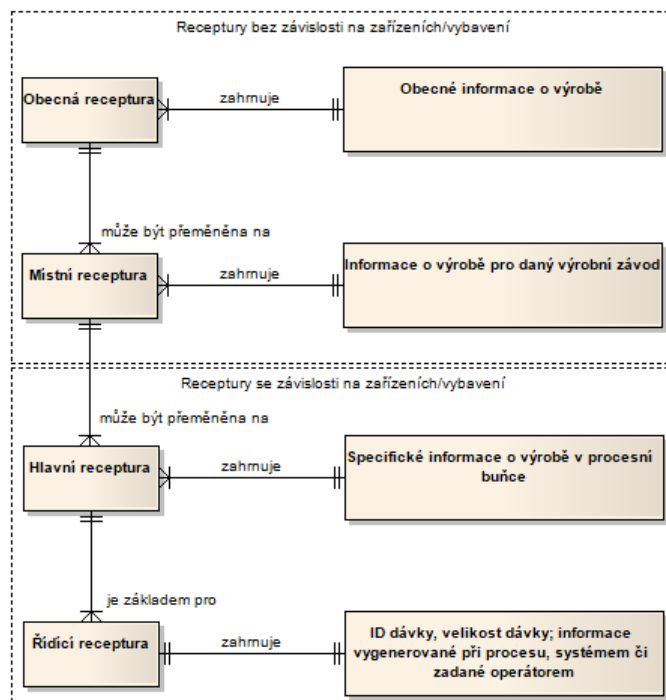
- Procedura^{119} – jedná se o hlavní jednotku v hierarchii, jedná se o plán (složený z jednotlivých jednotkových procedur) pro výrobu dané dávky, např. *vyrobiť jednorázové rukavice*
- Jednotková procedura^{120} – složena z jednotlivých operací, které právě v jedné jednotce (jednotka viz obrázek 10) uskutečňují dávkový výrobní proces \implies více jednotkových procedur může být zároveň vykonáváno ve více jednotkách, např. *chemickou reakcí vytvořit materiál pro rukavice, nasekat materiál na potřebnou velikost, ...*
- Operace^{121} – uspořádaná množina fází, mění stav materiálu (chemická či fyzická změna), např. *chemickou reakcí vytvořit materiál pro rukavice, začít tento proces, (chemická) reakce, ...*
- Fáze^{122} – jeden či více „příkazů“, vyvolává nějakou akci (změny algoritmů, čtení hodnot, měnění hodnot, ...), přímo využívá dostupné zařízení/vybavení (stroje), např. *přidat, zahrát*

Koordinační řízení ^{123} řídí, iniciuje či modifikuje vykonávání procedurálního řízení a využívání subjektů vybavení (zjednodušeně můžeme chápat jako jednotlivé stroje, i když je to nepřesné – viz definice *subjektů vybavení*^{124} výše). Dá se tedy chápat, jako dodatečné řízení k řízení procedurálnímu, které nadále usměrňuje. např. alokování vybavení pro výrobu dávek, koordinace vstupního materiálu.

5.4 Receptury

^{125} Za recepturu je považován soubor minimálních informací potřebných k přesnému stanovení výrobního procesu daného produktu, jež je tudíž možné identifikovat a odlišit od jiných. Říká systému dávkového řízení, jak přesně daný produkt vyrobit \implies návod. Skládá se z 5 částí:

- Hlavička^{126} – obsahuje identifikaci procedury, verzi procedury, datum, avšak např. i informace z jaké receptury pochází
- Formule^{127} – obsahuje **vstupy** (množství a typ materiálu pro výrobu), **parametry** (pro zpracování vstupního materiálu) a **výstupy** daného procesu (definuje množství a jaký materiál/energie má vzejít z dané receptury)
- Požadavky na vybavení^{128} – jaké vybavení (stroje) jsou nutná pro aplikování dané receptury (výrobu)
- Procedura receptury^{129} – jak je daná receptura vykonávána – postup; obecné a místní receptury jsou popsány z pohledu hierarchie dávkového



Obrázek 13: Druhy receptur 2010 – ERD

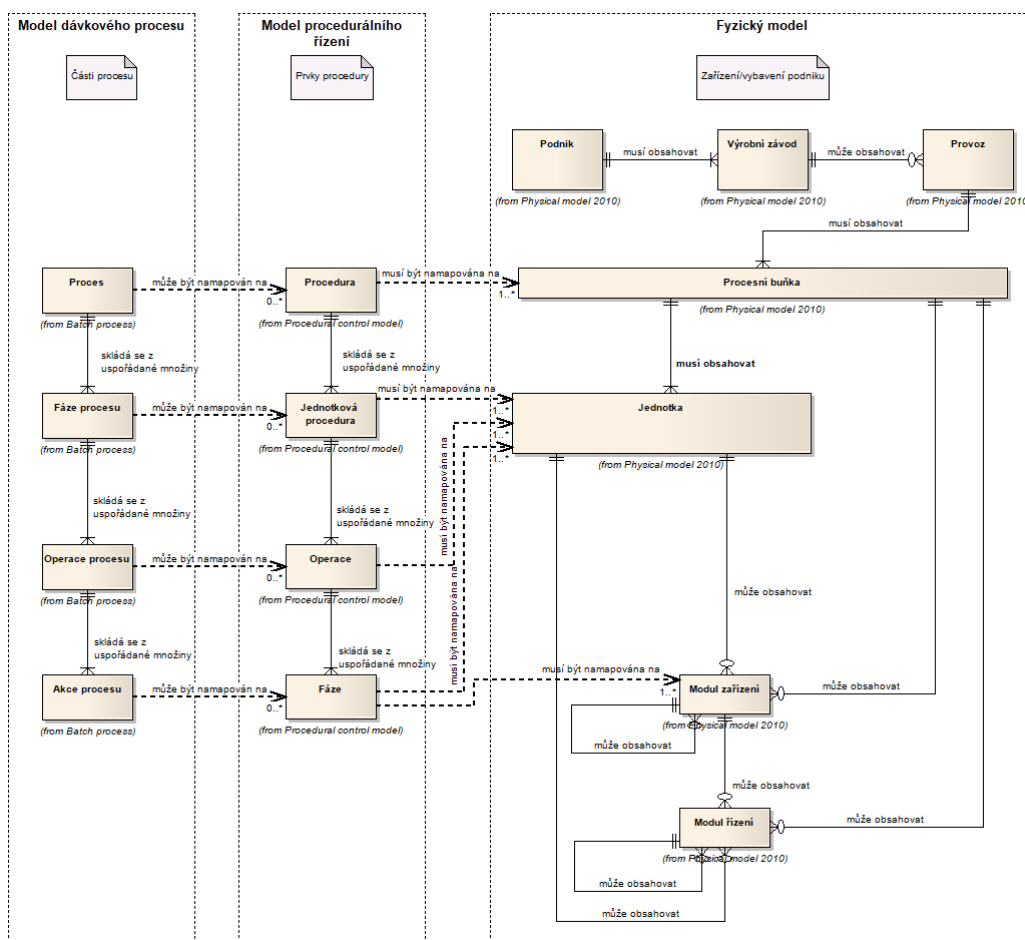
Zdroj: [44]

procesu, viz obrázek 9 (kvůli obecnosti); naopak hlavní a řídicí receptury z pohledu hierarchie procedurálního modelu řízení, viz obrázek 12 (kvůli vztahu k vybavení [stroje])

- Dodatečné informace^{130} – další informace neobsažené v předchozích částech, např. bezpečnostní informace, způsob balení a štítkování atd.

V jednotlivých částech byly zmíněny různé druhy receptur. Norma diskutuje právě 4. druhy, které lze přehledně vidět na obrázku 13. Jak je již zvykem, hierarchie postupuje od nejobecnější po nejspecifičtější druh receptury, přičemž se liší následovně:

1. Obecná receptura^{131} – nejobecnější typ receptury na podnikové úrovni, např. receptura na jednorázové rukavice
2. Místní receptura^{132} (v kontextu výrobního závodu) – receptura určena pro různé výrobní závody (každý může mít dostupné mírně rozdílné vstupní materiály, či vyrábět lehce odlišný druh produktu), např. jednorázové rukavice modré barvy vs zelené barvy
3. Hlavní receptura^{133} (v kontextu procesní buňky) – receptura pro jednotlivé procesní buňky (můžeme si představit výrobní linku), každá buňka (linka) může být něčím specifická, tudíž se receptura může mírně lišit (bere v potaz dostupná zařízení)



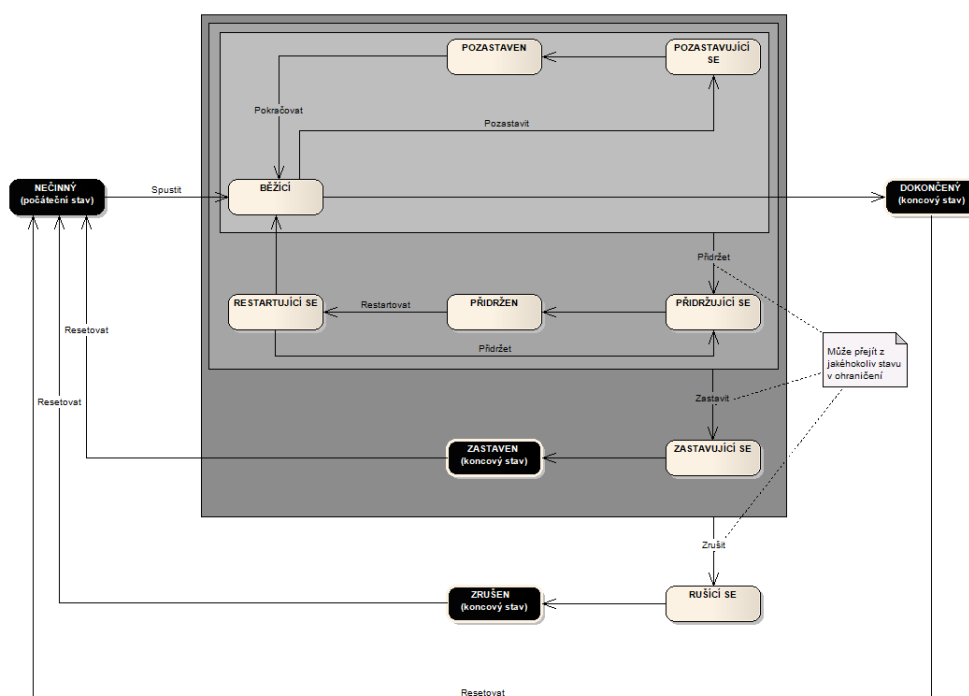
Obrázek 14: Vztahy mezi modely – ERD

- Řídicí receptura^{134} – receptura se specifickými informacemi pro jednotlivé dávky, které budou vyráběny (v počátku kopie hlavní receptury)

5.5 Vztahy mezi modely

Vztahy mezi jednotlivými modely definovanými v předchozích podkapitolách lze vidět na obrázku 14. Z tohoto diagramu lze vidět, že jednotlivé části dávkového procesu jsou spjaty s procedurálními prvky, kde jsou definovány jednotlivé kroky, které poté vykonávají *subjekty vybavení*^{135}.

Modely procedurálního řízení a fyzický model spolu mohou „splývat“, kdy jednotlivé procedurální prvky mohou být zakomponované přímo do zařízení (na určité úrovni). Nicméně na čím nižší úrovni zařízení tato návaznost je, tím je receptura flexibilnější.



Obrázek 15: Procedurální stavy 2010 – ERD

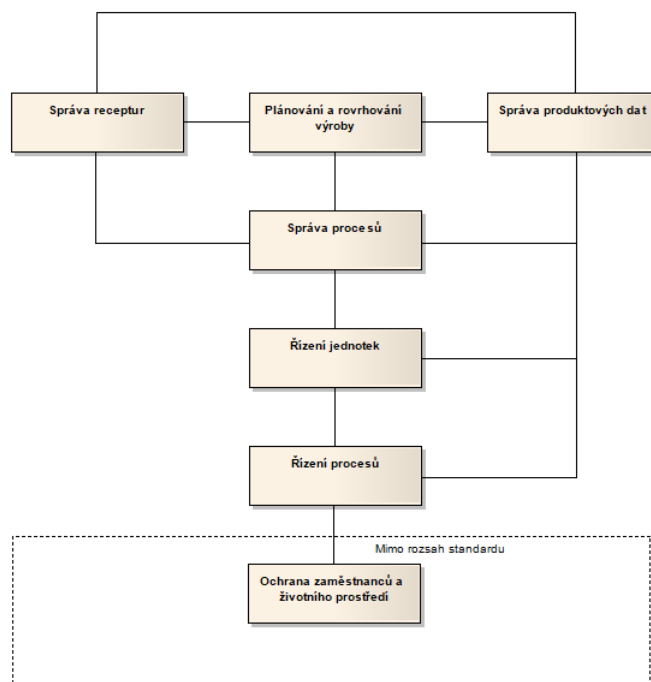
Zdroj: [44]

5.6 Stavy a módy

Jak procedurální prvky^[136], tak subjekty vybavení^[137], mohou nabývat určitých stavů. Tyto stavy nejsou nutně pevně definovány. Naopak jsou závislé na požadavcích daného výrobního procesu. Přejechy mezi jednotlivými stavy jsou prováděny na základě příkazu nebo svévolně při vykonávání daného výrobního procesu (např. po dokončení přejde ze stavu „běžící“ do stavu „dokončený“). Koncept si lze ukázat na příkladu na obrázku 15. Jednotlivé stavy digramu korespondují se stavy procedurálních prvků. Přejechy mezi stavy, které mají přiřazen název, korespondují s příkazy, přičemž přechody bez názvu jsou „volné“ přechody na základě vykonávání daného výrobního procesu. Nicméně norma navíc diskutuje způsoby, jakými procedurální prvky a subjekty vybavení reagují na jednotlivé příkazy, a to skrze módy:

1. Automatický
2. Polo-automatický
3. Manuální

Přičemž subjekty vybavení mohou být pouze v módech 1 a 3. Ve své podstatě módy určují, jak ovlivnitelné prvky jsou z pohledu operátora.



Obrázek 16: Model řídicích aktivit – ERD

Zdroj: [43]

5.7 Aktivity a funkce řízení dávkové výroby

S88 diskutuje také tzv. řídicí aktivity, které vstupují do dávkové výroby, či ji ovlivňují. V tomto tématu nebudeme zabíhat do podrobností. Představíme samotný model a krátce uvedeme každou aktivitu v něm uvedenou. Model^{138} samotný se dle normy skládá z právě 7 částí, které lze vidět na obrázku 16. Diagram ukazuje pouze hlavní vztahy mezi jednotlivými aktivitami, nikoliv všechny. Mezi jednotlivými aktivitami kolují informace (ve formě dat), přičemž jejich struktura by měla být čitelná pro všechny aktivity, a to kvůli přístupu k relevantním datům pro danou aktivitu. Co se týče obsahu, pak to záleží na potřebách daného procesu. Důležitým faktorem je zabezpečení těchto dat, a to jak ze stránky přístupu (pouze oprávněné osoby, proces aj. by měly mít přístup), tak i ze stránky samotného přenosu (včetně ztráty dat). Mezi daty by měly být vedeny záznamy o případných změnách a odkazech na jiná data. V neposlední řadě je důležitá také archivace, přičemž taková data by opět měla být v čitelné a použitelné podobě. Taková data mohou sloužit například pro extrakci informací z minulosti (o receptuře, chybách atd.).

Správa receptur^{139} slouží pro vytváření, ukládání a spravování *obecných*^{140}, *místních*^{141} a *hlavních receptur*^{142}. Výstup aktivity je *hlavní receptura*, která je předána aktivitě *Správa procesů*, jež z ní vytváří *řídicí receptury*^{143}.

Plánování a rozvrhování výroby ^{144} je aktivita, jejíž hlavním úkolem je plánování výroby jednotlivých dávek, jež předává aktivitě *Správa procesů*.

Správa produktových dat ^{145} shromažďuje, ukládá, zpracovává a reportuje produktová data, a to pouze taková, jež se přímo vztahují k dávkové výrobě. Zejména pracuje s daty, jež se týkají historie jednotlivých dávek.

Správa procesů ^{146} je aktivita, jež zahrnuje nejrůznější funkce pro správu jednotlivých dávek v *procesní buňce*^{147}. Využívá data z prvních dvou nadřazených aktivit pro vytvoření *místních receptur*, přičemž posílá jednotlivá data o dávkách aktivitě *Správa produktových dat*. Identifikuje tři hlavní funkce:

- správa dávek,
- správa zdrojů procesní buňky,
- shromažďování dat z dávek a procesních buněk.

Řízení jednotek ^{148} je aktivita, jejímž hlavním úkolem je propojení řídicí receptury spolu s vybavením. Také identifikuje právě tři funkce:

- Příjem a provádění *procedurálních prvků*^{149} – provádění jednotlivých kroků *receptury* v dané *jednotce* (data přichází ze *Správy procesů*), a to *jednotkových procedur, operací, i fází*,
- správa zdrojů *jednotky*^{150},
- shromažďování dat z dávek a *jednotek*.

Řízení procesů ^{151} zahrnuje *procedurální*^{152} a *základní řízení*^{153}. Bývá rozprostřena mezi několik různých subjektů vybavení (*jednotek*^{154}, modulů *zařízení*^{155} i *řízení*^{156}). Je charakterizována třemi funkcemi:

- provádění *fází*^{157} v zařízení
- provádění *základního řízení*^{158},
- sběr dat.

5.8 Modely objektů

^{159} Norma S88 pro představené koncepty definuje i jejich modely objektů, a to v jazyce UML. Tyto modely objektů přesně definují strukturu jednotlivých konceptů a slouží jako přímý návod pro implementaci. Jsou následující:

- Obecný model pro přehled
- Modely receptur

- Modely vybavení
- Modely plánování a rozvrhování
- Modely pro popis dat z výroby

Pro potřeby této diplomové práce není nutné jednotlivé modely více rozvádět či předvádět.

5.9 Implementace normy

Za nejznámější implementaci normy S88 je považována úplná implementace [Batch Markup Language \(BatchML\)](#)^{160}, která je ve formátu [XML](#)²¹ v [XSD](#) schématu²². Přesněji implementuje jednotlivé modely objektů (viz kapitola 5.8). Jedná se o volně dostupnou implementaci, kterou je možné využívat či použít jako referenci při tvorbě vlastní implementace.

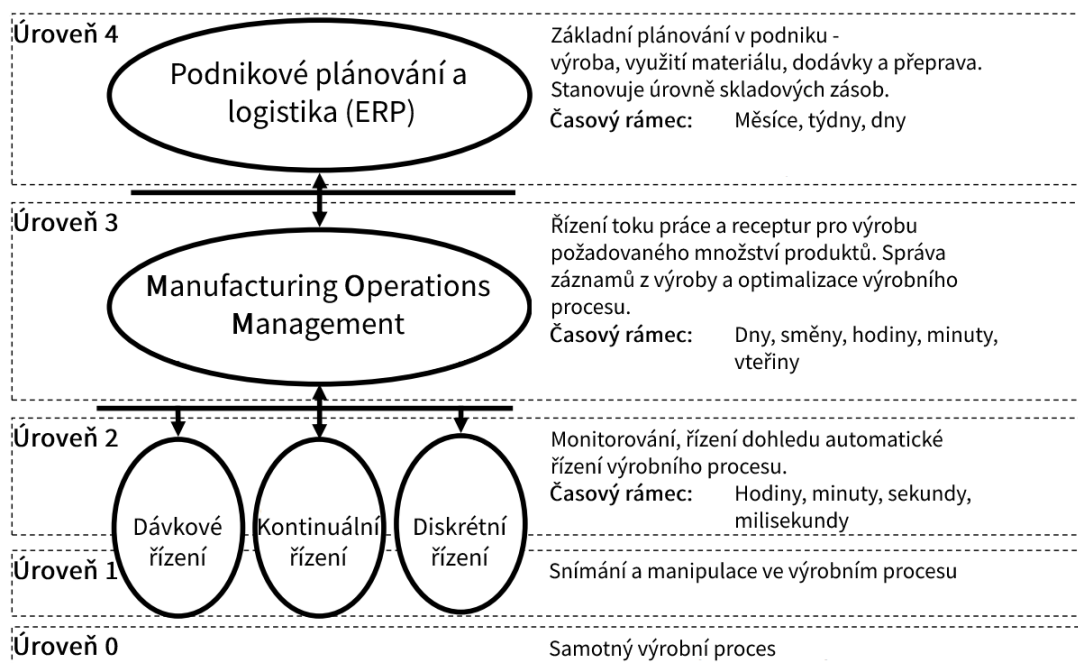
6 ANSI/ISA-95

Norma [ANSI/ISA-95 \(S95\)](#)^{161} adresuje rozdělení jednotlivých funkcí v podniku a komunikaci mezi nimi. Původně vznikla v Americe, nicméně je aplikována a využívána i v Evropě, a to pod označením IEC 62264 [45]^{162}. Aktuálně se skládá z následujících 8 částí, přičemž mnohé části již v průběhu let prošly aktualizacemi:

1. Část 1. – Modely a terminologie
2. Část 2. – Objekty a atributy pro integrované systémy podnikového řízení
3. Část 3. – Modely činnosti managementu výrobních provozů
4. Část 4. – Objekty a atributy pro integraci [Manufacturing Operations Management](#)
5. Část 5. – Transakce mezi obchodem a výrobou
6. Část 6. – Model služby zasílání zpráv
7. Část 7. – Model služby alias
8. Část 8. – Profily výměny informací

²¹[Extensible Markup Language](#) – textový formát dat ve strukturované podobě

²²[XML Schema Definition](#) – definice schématu (pravidla pro tvorbu [XML](#) dokumentu) vydané [World Wide Web Consortium \(W3C\)](#). Důležité pro validitu dokumentu



Obrázek 17: Hierarchie řízení

Zdroj: [46]

Stejně jako u normy S88^{163}, tak i zde se jedná spíše o obecné, dostatečně detailní, popis problematiky integrace podnikových řídicích systémů. Je aplikovatelná na širokou škálu různých implementací. Normu vydal a udržuje výbor SP95^{164} (patřící přímo pod ISA), který se skládá z mnoha odborníků z nejrozličnějších podniků. Je určena pro lidi zapojené do návrhu, vývoje či přímo provozování výrobních systémů; zodpovědné za návrh rozhraní komunikace mezi jednotlivými (řídicími) systémy v podniku; zapojené do návrhu, vývoje, marketingu a integrace automatizace výroby či pro lidi zapojené do návrhu či řízení výroby, skladování a přesunů uvnitř podniku[46].

6.1 Hierarchie řízení

^{165} Nejjednodušší model této normy jasně vymezuje, jak v podniku rozdělit jednotlivé prvky vstupující a ovlivňující výrobní proces. Je zřejmé, že do něj vstupují fyzická zařízení (výrobní stroje), řídicí systémy a tomu podobné, avšak s výrobou jsou spjaty i zásoby materiálu, jejich plánování (nákupy) a potřebné receptury. Nicméně nesmíme opomenout ani zaměstnance a jim přidružený management (mzdový, směnný, ...). Zmíněné rozdělení lze vidět na obrázku 17, jež jej prezentuje jako hierarchii. Půjdeme-li od nejnižší vrstvy, tak **0. úroveň** nám značí samotný fyzický výrobní proces. Nad ní, **1. úroveň**, se zabývá aktivitami v samotném procesu, které jsou prováděny skrze již definované aktuátory^{166}, spolu se senzory^{167}, které slouží pro sledování a měření. Do **úrovně 2** spadají aktivity monitorování (dohled) a řízení výrobního procesu. Rozsáhlejší pokrytí

má **úroveň 3**, které zajišťuje samotnou koordinaci (v) procesu, kam patří i správa receptur a uchovávání záznamů o výrobě. Nejširší je **úroveň 4**, která zahrnuje všechny obchodní aktivity potřebné k zajištění samotného výrobního procesu. Zde se můžeme odkázat na již představený pojem **ERP** v kapitole 2.4, který tuto oblast pokrývá a koresponduje s ní.

Pro samotné rozlišení, které aktivity spadají do jaké úrovně, představuje norma následující poučku. Pakliže aktivita přímo zasahuje do výrobního procesu a obsahuje informace o personálu, zařízeních/vybavení či materiálu a zároveň je nezbytná alespoň pro jednu z následujících činností:

- bezpečnost provozu,
- spolehlivost provozu,
- efektivitu provozu,
- kvalitu vyráběných produktů,
- plnění předpisů,

pak spadá do úrovně 3, 2, nebo 1 [46].

Je důležité si uvědomit, že nejde pouze o samotné rozdělení do jednotlivých oddělených úrovní, nýbrž je velmi důležitá i **komunikace a předávání informací** vně úrovní, avšak i mezi **úrovněmi**, což norma patřičně adresuje.

6.1.1 Úroveň 3

Hlavní podstata normy **S95** leží právě v úrovni 3 a v komunikačním rozhraní mezi 4. a 3. úrovní (potažmo 2. a 3. úrovní). Jedním z hlavních přínosů a myšlenek této normy bylo jasné vymezení pojmů a zařazení funkcionality. Již před samotným zveřejněním byl znám a definován pojem **MES**, a to asociací **MESA. Manufacturing Execution System (MES)**^{168} je systém, který propojuje **PIS** (jakým jsou právě zmíněná **ERP**^{169}) spolu se systémy pro automatizaci výroby. **MESA** pro **MES** identifikovala právě 11 funkcí (budou rozvedeny níže), které měl každý **MES** splňovat.

Kdežto v této normě je jako mezičlánek mezi **ERP** a systémy pro automatizaci výroby uveden pojem **Manufacturing Operations Management (MOM)**^{170} (detailněji rozebraný v kapitole 6.4), zavedený **ISA**. Není tomu náhodou, jelikož **ISA** vzala již existující definici **MES**, usadila ji a zahrнула do svého pojmu **MOM**, resp. do 3. úrovně, která sama o sobě pokrývá ještě širší oblast. Nicméně je přirozené, že všechny funkce **MES** jsou zde zahrnuty také. Úroveň 3. adresuje následujících 12 hlavních funkcí, přičemž prvních 11 koresponduje s definicí funkcí **MES** od **MESA** [47]:

Řízení a přidělování zdrojů se zabývá správou zdrojů (materiál, zařízení, pracovní síly, dokumenty, ...) potřebných k početí a dokončení výroby.

Dispečerské řízení výroby má na starosti přerozdělování aktivit jednotlivým zařízením a pracovníkům.

Shromažďování a získávání dat získává data z výroby.

Řízení kvality se zabývá hlídáním kvality při výrobě na základě dat (včetně měření) z výroby (může i identifikovat problém a doporučit nápravu).

Řízení procesů má na starost monitorování výrobního procesu, automaticky opravuje či pomáhá v rozhodování operátorům.

Sledování výroby poskytuje aktuální status výrobního procesu (kdo je přiřazen k výrobě, aktuální podmínky, výstrahy, . . .), zaznamenává proces výroby pro zpětnou analýzu.

Analýza výkonnosti podává aktuální hlášení o výkonnosti výrobního procesu (může k tomu používat data z různých jiných funkcí), porovnání s minulými výsledky a s předpokládanou výkonností.

Provoz a podrobné plánování vytváří finální plány/rozvrhy pro výrobu, bere v potaz různé parametry výrobků a materiálu a plány/rozvrhy vytváří na základě dostupnosti zdrojů, spolu s překrývajícími se/paralelními procesy.

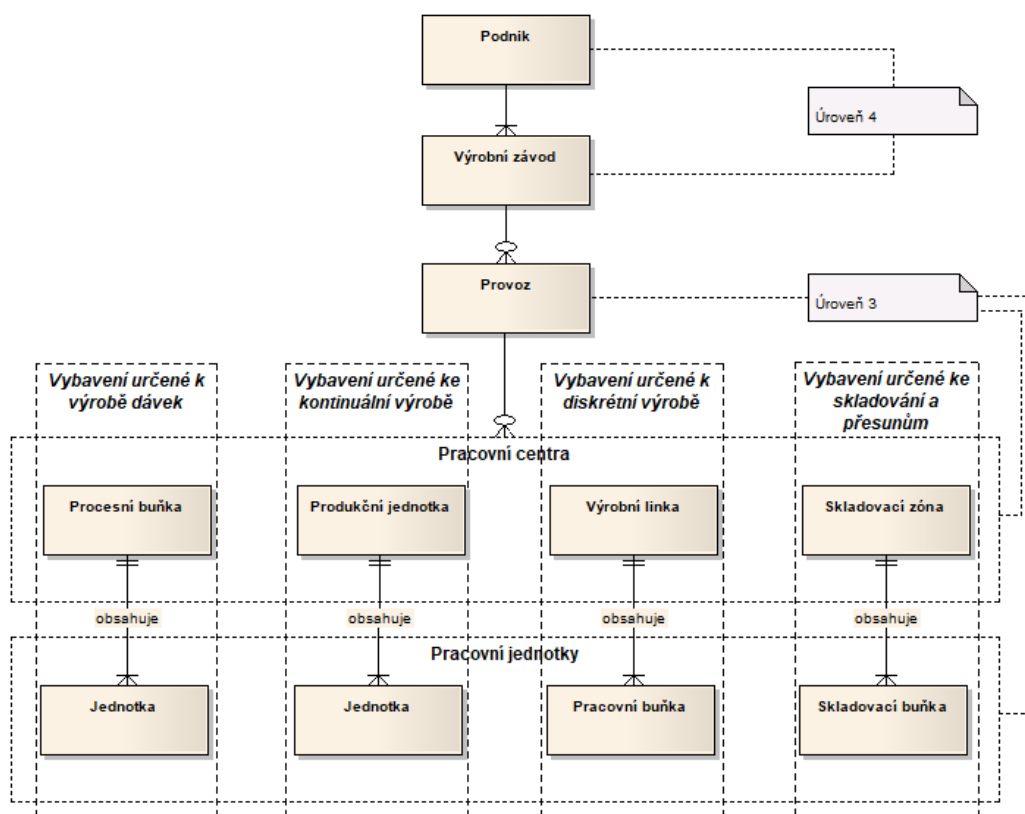
Správa dokumentace spravuje dokumenty vázající se k výrobě, jako jsou např. receptury, [SOP²³{171}](#), záznamy z dávek, ale třeba i BOZP pokyny aj.

Řízení pracovních sil sleduje zaměstnance, jejich příchody a odchody, certifikace, či druh práce, který vykonávají. Zároveň může doporučovat určité pracovníky pro určitý druh úkonu za účelem optimalizace.

Řízení údržby se zabývá údržbou zařízení/vybavení a nástrojů pro zajištění dostupnosti k výrobě. Archivuje poruchy, může pomáhat s diagnostikou a může predikovat nutnost údržby.

Přesun, uskladnění a sledování materiálu se zabývá skladováním a sledováním materiálu (přesuny do výroby, mezi sklady, zařízeními aj.), ale i polotovarů a výrobků.

²³[Standard Operating Procedure](#) – přesné instrukce pro provedení daného úkonu krok za krokem

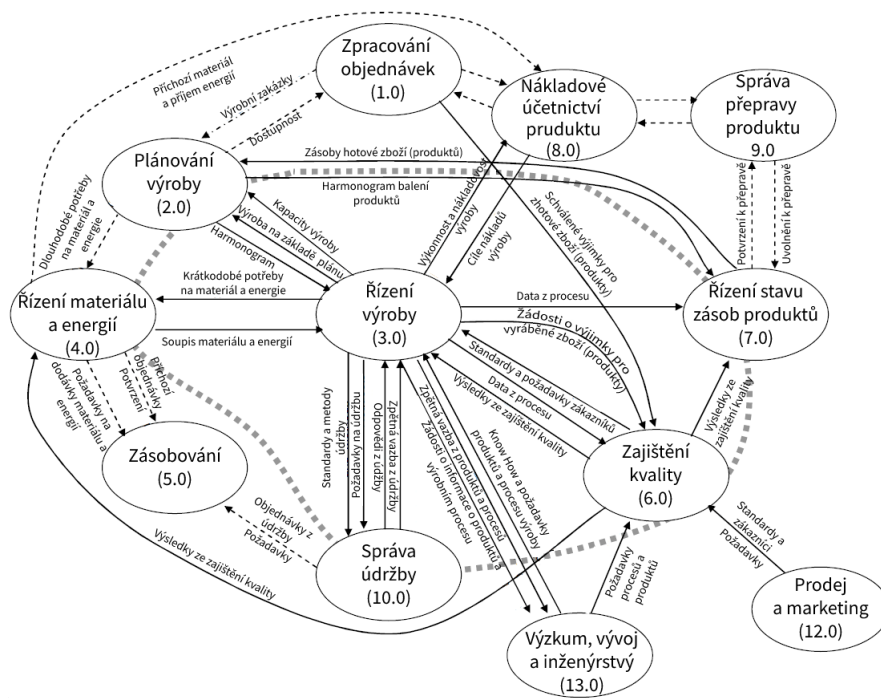


Obrázek 18: Hierarchie zařízení v kontextu rolí
Zdroj: [46]

6.2 Hierarchie zařízení

Prvky vstupující do výroby v jednotlivých úrovních prezentovaných na obrázku 17 jsou rozděleny do dalšího hierarchického modelu^[172]. Tento model koreluje, ba dokonce rozšiřuje, model představený normou S88, přesněji fyzický model^[173] z kapitoly 5.2, vizualizovaný na obrázku 10. Toto rozšíření můžeme vidět na obrázku 18. Jak lze vidět, **podnik**^[174] a **výrobní závod**^[175] spadají do úrovně 4 v hierarchii řízení a jejich definice, spolu s **provozem**^[176] přímo koresponduje s definicemi ve fyzickém modelu normy S88 z kapitoly 5.2. Nicméně zde představená hierarchie se na model dívá s větším odstupem a definuje dva zcela nové pojmy, a to **pracovní centra** a **pracovní jednotky** (v diagramu označeny tečkovaným ohraničením). Samotné **pracovní centrum**^[177] je uskupení zařízení a jejich plánování spadá do úrovně 4 nebo 3. **Pracovní jednotky**^[178] jsou jednotlivá zařízení, která spadají pod pracovní centra. Vykonnávají funkce definované pro úroveň 3, v níž je prováděno i jejich plánování. V tomto ohledu se jedná spíše o abstrakci, která je využita, pakliže není nutné hovořit o specifickém druhu *pracovního centra*, respektive *pracovní jednotky*.

Diagram dále vyobrazuje příklady jednotlivých *pracovních center* a *jednotek*, přičemž je zde zastoupení pro všechny tři typy výrobních procesů (definovaných



Obrázek 19: Model funkcí
Zdroj: [46]

v kapitole 5), a to pro *dávkový*^{179}, *kontinuální*^{180} a *diskrétní výrobní proces*^{181}, plus vybavení určené ke skladování a k přesunům. Na diagramu jsou jednotlivé typy označeny čárkovaným ohraničením. Mohou existovat i další (v našem diagramu nevyobrazená) pracovní centra, jako např. laboratoř, spolu s vlastními pracovními jednotkami. Avšak je důležité zachovat zmíněnou hierarchii (tj. provoz → pracovní centrum → pracovní jednotka), přičemž každá jednotka smí patřit právě pod jedno pracovní centrum. Zmíněný *provoz*, *pracovní centra* a *jednotky*, jsou v hierarchii řízení zakotveny v úrovni 3, tudíž činnosti, které (alespoň částečně) korespondují s činnostmi představenými v předchozí podkapitole o hierarchii řízení ve 3. úrovni (viz 6.1.1).

6.3 Model funkcí

^{182} Představuje 12 základních funkcí v podniku, přičemž opět bere v potaz rozdělení na jednotlivé úrovně, představené v hierarchii řízení (viz kapitola 6.1). Zároveň definuje datové toky mezi jednotlivými funkcemi. Tento model je znázorněn na obrázku 19, přičemž tučná tečkovaná čára odděluje funkce úrovně 4 a 3, přičemž zároveň představuje komunikační rozhraní mezi těmito úrovněmi. Jednotlivé funkce jsou vyobrazeny v elipsách, mezi kterými probíhají datové toky znázorněny čarami (čárkované čáry identifikují datové toky, které nejsou významné pro tento model), jež nesou pojmenování datového toku. Jednotlivé funkce a jejich popis následují.

Zpracování objednávek (1) se zabývá obecně správou (přijímání, zpracovávání, potvrzování, . . .) objednávek od zákazníků, generuje výkazy hrubé marže, prognóz prodeje aj.

Plánování výroby (2) plánuje harmonogramy výroby, vyhodnocuje potřebu surovin pro výrobu (dlouhodobě), množství dostupného produktů k prodeji.

Řízení výroby (3) řídí výrobu samotných produktů (ze surovin do hotového produktu), přičemž bere v potaz harmonogram, dostupné zdroje, překážky, vyžádání surovin. Samotnou výrobu monitoruje, optimalizuje a vydává výkazy. Monitoruje zařízení, přičemž vydává žádosti o údržbu, kterou následně připravuje a koordinuje.

Řízení materiálu a energií (4) obsluhuje sklad, převody (včetně příjmu) a kvalitu materiálu, spolu s příjmem a kvalitou energií. Zároveň vykazuje skladové zásoby a ztráty při výrobě. Generuje žádosti o nákup materiálu či energií.

Zásobování (5) na základě přijatých požadavků podává a monitoruje objednávky na nákup surovin, součástek, nástrojů a přístrojů dodavatelům, včetně fakturace.

Zajištění kvality (6) nastavování standardů kvality, včetně jejich udržování a aplikování \implies testování jak surovin pro výrobu, tak vyrobených produktů (produkty porovnává s požadavky zákazníka). Zároveň tato data sbírá a případně poskytuje.

Řízení stavu zásob produktů (7) správa a evidence vyrobených produktů, jejich zarezervování (pro prodej), provádí balení a zařizuje přepravu na základě požadavků.

Nákladové účetnictví produktu (8) provádí kalkulace a poskytuje výkazy o celkové nákladovosti produktu, stanovuje nákladové cíle pro dodávku materiálu, energií a distribuci.

Správa přepravy produktu (9) organizuje přepravu produktu na základě požadavků v objednávce, přičemž vyhotovuje i potřebné dokumenty, vyjednávání a objednávání přepravy u přepravních společností, kterou následně potvrzuje a předává ji pro fakturaci. Vytváří výkazy o nákladovosti přepravy *Nákladovému účetnictví produktu*.

Správa údržby (10) poskytuje údržbu jak na vyžádání, tak i preventivní, monitoruje zařízení, podává objednávky náhradních součástí (pro zařízení), koordinuje externí dodavatele (z oblasti údržby), vykazuje náklady na údržbu, efektivitu a spolehlivost zařízení.

Prodej a marketing (12) vytváří prodejní a marketingové plány, nastavuje ceny, stanovuje požadavky na produkty (jak na základě požadavků zákazníků, tak i standardů pro produkty), komunikuje se zákazníky.

Výzkum, vývoj a inženýrství (13) vývoj nových produktů, definuje požadavky procesu výroby, včetně požadavků na materiál a výrobní zařízení.

6.4 Manufacturing Operations Management

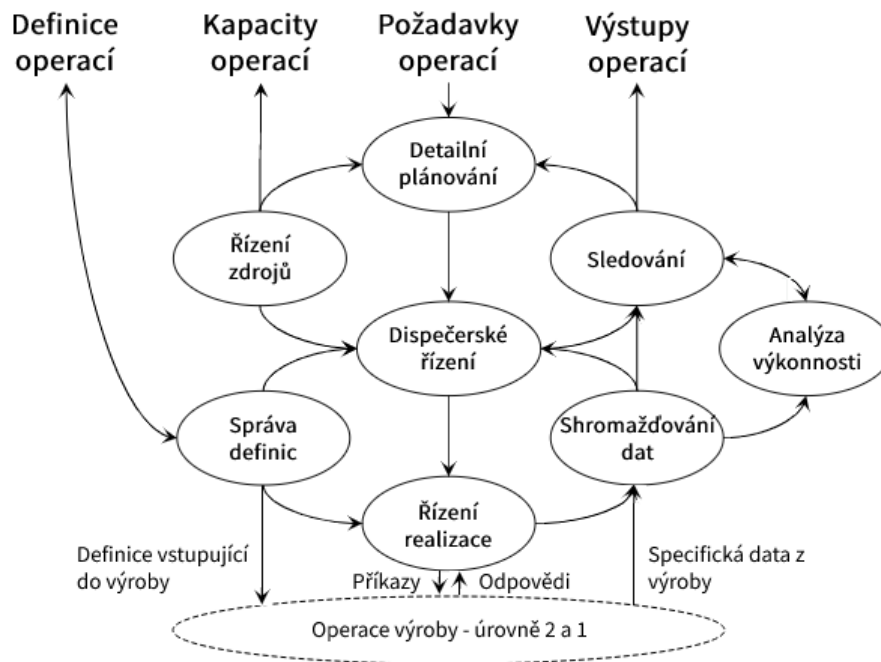
Jak již bylo lehce nastíněno, **Manufacturing Operations Management (MOM)**^{183} je soubor činností a aktivit výrobního podniku, které jsou vykonávány stroji, lidmi či informačními systémy. Tyto činnosti jsou ty, jež koordinují zaměstnance, zařízení, materiál a energii při přeměně surovin na polotovary či produkty, a zároveň zahrnuje správu informací o harmonogramech, využití, schopnostech, definicích, historii a stavu všech **zdrojů podniku** (zaměstnanci, zařízení, suroviny a materiál) [48]. Norma rozděluje činnosti **MOM** do právě 4 kategorií:

1. Řízení výrobních operací
2. Řízení operací údržby
3. Řízení operací kvality
4. Řízení skladových operací

Je možné si všimnout návaznosti a korelace s modelem funkcí z předchozí kapitoly, který je vyobrazen na obrázku 19. Tlustě přerušovaná čára nám totiž vyznačuje funkce, které spadají do **MOM**, třebaže částečně. Jednotlivé kategorie a jejich případná korelace jsou popsány níže.

Řízení výrobních operací jsou aktivity, které koordinují, usměrňují, řídí a sledují funkce využívající suroviny, energie, zařízení, personál a informace nutných k výrobě produktu [48]. V modelu funkcí se jedná o *řízení výroby*.

Řízení operací údržby jsou aktivity, které koordinují, usměrňují, řídí a sledují funkce k údržbě zařízení, nástrojů a přidružených aktiv tak, aby byly připraveny k výrobě [48]. V modelu funkcí se jedná o *správu údržby*, avšak v **MOM** je zahrnuta pouze určitá část.



Obrázek 20: Univerzální model aktivit
Zdroj: podle [48]

Řízení operací kvality jsou aktivity, které koordinují, usměrňují, řídí a sledují funkce k měření kvality (polotovary a výrobky) a její vykazování [48]. V modelu funkcí se jedná o *zajištění kvality*, přičemž MOM jej pokrývá téměř celé.

Řízení skladových operací jsou aktivity sloužící k řízení a sledování surovin a produktů, inventarizaci, řídí, měří a vykazují přesuny materiálu mezi *pracovními centry* (viz kapitola 6.2), výrobou, kontrolou kvality a údržbou; směřují suroviny z a do skladů apod. [48]. V modelu funkcí je korelace s *řízením materiálu a energií* a s *řízením stavu zásob produktů*, avšak obojí pouze z určité části.

Každou tuto kategorii lze popsat univerzálním modelem aktivit. Lze jej vidět na obrázku 20, přičemž toky seshora představují komunikaci s úrovní 4. a toky vespod s úrovněmi 2 a 1. **Definice operací** říká, jak daný proces provést. **Kapacity operace** značí, jaké jsou dostupné zdroje podniku a jejich kapacity. **Požadavky operací** jsou vstupním bodem, který říká, co a v jakém množství má být provedeno. Výstupním bodem jsou **výstupy operací**, které podávají informace z celého procesu (co a jak bylo provedeno, problémy, aj.). Samotných aktivit v každé kategorii MOM je 8 a jsou následující:

- Detailní plánování
- Dispečerské řízení
- Řízení realizace

- Shromažďování dat
- Řízení zdrojů
- Správa definic
- Sledování
- Analýza výkonnosti

První čtyři aktivity v tomto modelu se dají označit za hlavní, přičemž poslední čtyři za pomocné, či vedlejší. Tok informací začíná požadavkem operace, na základě kterého se vytvoří harmonogram – **detailní plánování**. Na základě tohoto harmonogramu poté **dispečerské řízení** vydává pokyny k realizaci procesu (= kdy spouštět proces) – **řízení realizace**, ze kterého jsou shromažďována data – **shromažďování dat**. Do toho samozřejmě v průběhu vstupují pomocné aktivity k zajištění a sledování zdrojů (tj. zaměstnanců, zařízení, surovin a materiálu) – **řízení zdrojů** a **sledování** – nutných k realizaci procesu, spolu se způsoby, jak mají být dané produkty vyráběny (tj. jaké stroje použít, jaký je postup výroby, jaký je potřebný materiál aj.) – **správa definic**. Důležité je pak samotný proces analyzovat pro případné vylepšení – **analýza výkonnosti**.

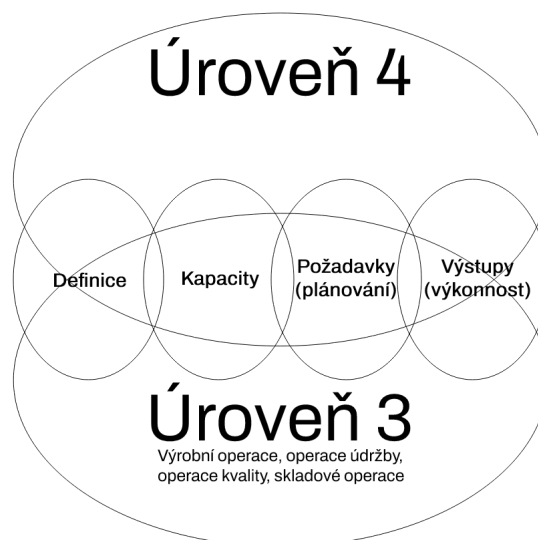
Je důležité zmínit, že nejsou vyobrazené všechny možné informační toky. Specifické procesy mohou vyžadovat výměnu informací mezi aktivitami, které nemají na obrázku 20 vyznačený informační tok. Dokonce samotné aktivity napříč kategoriemi mohou komunikovat mezi sebou (např. mezi *detailním plánováním výrobních operací* a *detailním plánováním skladových operací*).

6.5 Výměny informací a modely objektů

S95 definuje informace, které jsou vyměňovány mezi 3. a 4. úrovní, a pro tyto informace definuje i jejich přesnou strukturu (pomocí UML diagramů)^{184}. Vyměňované informace korespondují se 4 kategoriemi činností MOM, představenými v předchozí kapitole 6.4, a zároveň s modelem aktivit představeným v téže kapitole (resp. s jeho informačními toky v horní části – tj. z a do úrovně 4 – viz obrázek 20). Vyměňované informace jsou znázorněny na obrázku 21.

Na základě této výměny informací definuje S95 modely objektů těchto informací a popisuje je skrze UML diagramy tříd (diagramy ukážeme pouze ty nejdůležitější pro tuto diplomovou práci). Jedná se např. o modely definic, kapacit, žádostí či výstupů. Nicméně my se podíváme na 5 nezákladnějších modelů objektů, které jsou vyměňovány mezi úrovní 4 (ERP) a úrovní 3 (MOM). Ty totiž hrají určitou roli v pro nás relevantních modelech, které budeme představovat pro úroveň 3. Jsou následující:

- Personál
- Materiál



Obrázek 21: Informace předávané mezi úrovněmi 3 a 4
Zdroj: [46]

- Vybavení v kontextu rolí
- Fyzická aktiva
- Procesní segment

Jednotlivé modely níže krátce představíme, a pro některé z nich uvedeme i jejich diagram.

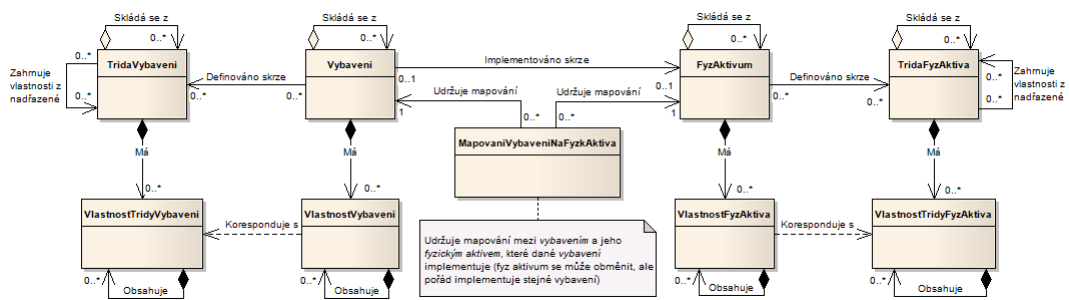
Model personálu definuje personál podniku (zaměstnance), jejich třídy (role v podniku) a kvalifikace. Sestává se ze dvou hlavních částí:

- Třída personálu – dá se chápat jako role, kterou v podniku zastává (operátor, kontrolor, ...)
- Osoba – přímo určitý zaměstnanec

Každá část má své vlastní vlastnosti (např. údaj o certifikaci).

Model materiálu slouží k definici materiálu v podniku. Informace o něm je třeba vyměňovat mezi jednotlivými úrovněmi, tudíž je potřeba mít jednotnou strukturu a názvosloví. Určuje právě 4 rozdělení, a to:

- Třída materiálu – slouží k organizaci materiálu do určitých tříd
- Definice materiálu – definuje určitý typ materiálu v dané třídě
- Šarže materiálu – Skupina určitého materiálu
- Podšarže materiálu – Podskupina materiálu v dané šarži (další rozdělení)



Obrázek 22: Modely *vybavení*, *fyzických aktiv* a vztah mezi nimi
Zdroj: [49]

Každá část má své vlastní vlastnosti (krom podšarže, která má vlastnosti shodné s vlastnostmi šarže, pod kterou patří).

Model vybavení v kontextu rolí je model pro obecný popis vybavení v podniku. Jedná se o určitou míru abstrakce, jelikož se nejedná přímo o fyzické instance vybavení (např. obecné vybavení *nádrž pro skladování benzínu*). Dvě části:

- Třída vybavení – slouží k organizaci a seskupení vybavení podle společných charakteristik
- Vybavení – reprezentuje určité vybavení

Každá část má své vlastní vlastnosti (např. kapacita).

Model fyzických aktiv navazuje na model předchozí, přičemž zde je řeč o fyzických instancích daného vybavení v podniku (např. *nádrž pro skladování benzínu #1*). Dvě části:

- Třída fyzického aktiva – slouží k organizaci a seskupení fyzických aktiv, podle společných charakteristik
- Fyzické aktivum – reprezentuje určité fyzické aktivum

Každá část má své vlastní vlastnosti (např. kapacita). Modely *vybavení v kontextu rolí* a *fyzických aktiv*, spolu s vztahy mezi nimi, lze vidět na obrázku 22.

Model procesního segmentu Proces segment je nejmenší jednotka ve výrobním procesu. Popisuje jednotlivý úkon, jež má být proveden. Zároveň slouží jako logické seskupení *materiálu*, *zařízení* a *personálu* potřebných k provedení onoho úkonu, včetně potřebného množství. Takové uskupení může být přímo potřebný materiál a personál, a to buď spolu s vybavením, nebo fyzickým aktivem.

6.5.1 Modely objektů informací v úrovni 3

V samotné 3. úrovni (MOM), S95 definuje velké množství jednotlivých modelů objektů, které lze rozčlenit do několika kategorií, mezi které patří například:

- vztahy mezi zdroji,
- definice práce,
- specifikace workflow,
- rozvrh práce,
- informace o výkonnosti práce,
- kapacity práce,
- ...

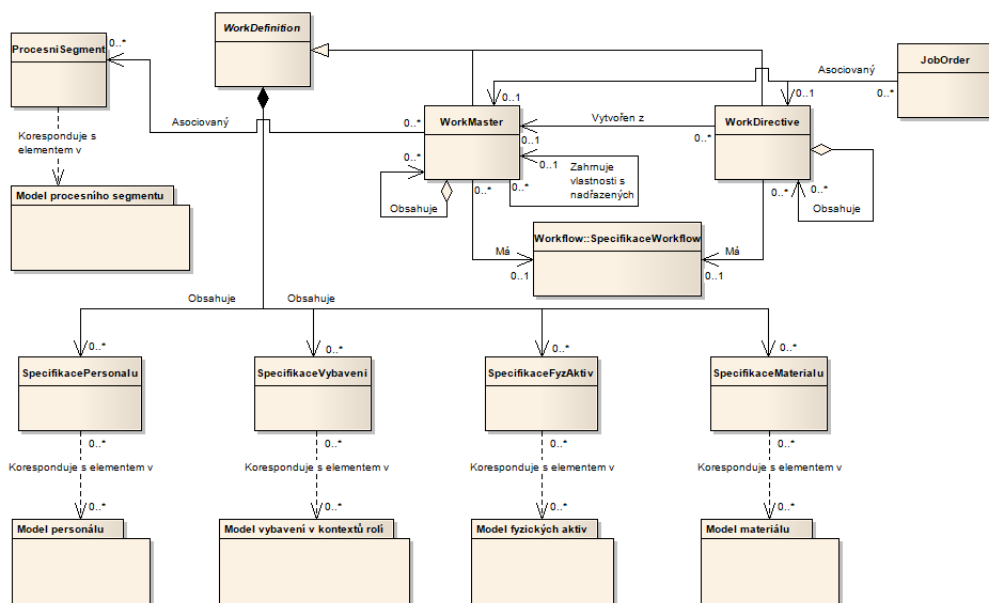
Nicméně my se podíváme jen na ty, jež jsou relevantní pro tuto diplomovou práci, a to na model definice práce (zejména work master a work directive), **workflow model** a lehce na model rozvrhu práce, pro usazení do kontextu výroby. Dopředu je třeba si uvědomit, že vykonávanou práci nemusí být pouze výroba samotná, nýbrž jedna ze 4 kategorií činností MOM, tedy *výroba*, *údržba*, *udržování kvality* či *skladové operace*.

Model definice práce ^{185} **Definice práce** je souhrnem zdrojů a workflow potřebných k vykonání určitého úkonu spadajícího do kontextu dané práce. **Work master** ^{186} je souhrnem zdrojů a instrukcí (nejedná se o kroky, nýbrž např. o recepty, SOPs apod.) potřebných k vykonání určitého úkonu, avšak nemá žádný odkaz na *job order*. Pakliže má odkaz na *job order*, pak se jedná o **Work directive** ^{187}, jež je kopií *work masteru* a řídí vykonávání *job order* (viz níže v *modelu rozvrhu práce*). Zjednodušený model lze vidět na obrázku 23.

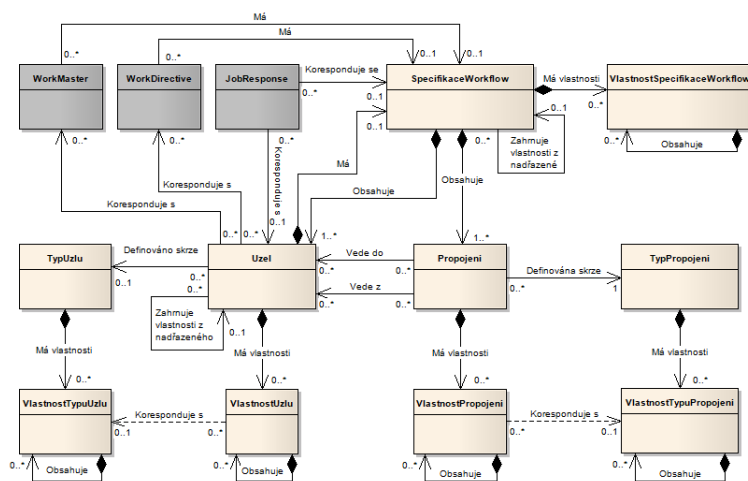
Workflow model ^{188} **workflow** ^{189} je souhrnem instrukcí pro vykonání dané práce. Tvoří jej uzly a hrany, jež reprezentují daný postup. Model je vyobrazen diagramem na obrázku 24. Jedním z příkladů workflow formátu je BPMN, jež je představen v kapitole 11.

Model rozvrhu práce ^{190} **Work schedule** ^{191} je chápán jako seskupení žádostí o vykonání práce, nazývaných *work request*. **Work request** ^{192} se skládá z *job orderů*. **Job order** ^{193} je část práce určená k vykonání v určitém *pracovním centru* a odkazuje na existující *work master* (tj. *job order* odkazuje na *work master*, který obsahuje workflow, jež definuje, jak daný úkon vykonat)

Nejpodstatnější pro tuto diplomovou práci je *model workflow*. Jelikož pojem *workflow* je více rozveden až v části o *workflow systémech* (viz část IV), je samotný *workflow model* dále rozveden a diskutován až v kapitole 13.



Obrázek 23: Model definice práce
Zdroj: podle [50]



Obrázek 24: Workflow model
Zdroj: [50]

Tabulka 4: Slovesa

Zprávy odesílatele	Odeslané informace
ACKNOWLEDGE	Potvrzení PROCESS žádosti
CANCEL	Smazání určitých existujících informací
CHANGE	Změna již existujících informací
CONFIRM	Potvrzení žádosti
GET	Žádost o informace
PROCESS	Nové informace
RESPOND	Odpověď na CHANGE žádost
SHOW	Odpověď s informacemi na GET žádost
SYNC ADD	Požadavek od vlastníka objektu na přidání informace
SYNC CHANGE	Požadavek od vlastníka objektu na změnu informace
SYNC DELETE	Požadavek od vlastníka objektu na smazání informace

6.6 Komunikace (mezi a napříč úrovněmi)

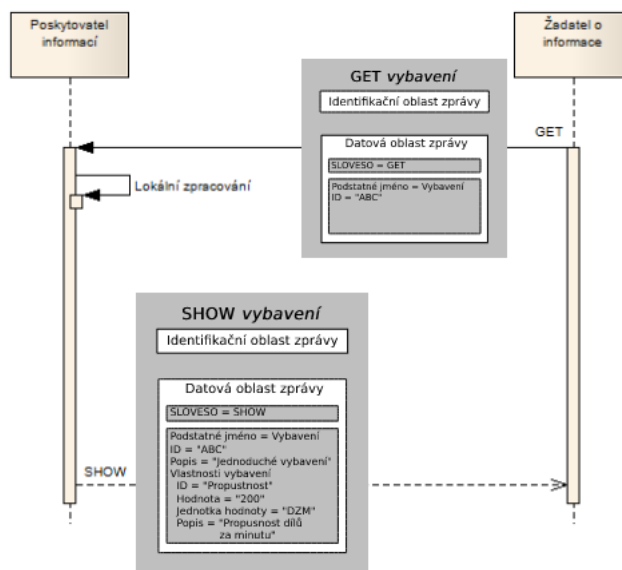
Důležitá je i samotná výměna informací, a to jak mezi úrovněmi 3 a 4, tak i uvnitř samotné 3. úrovně. Je zřejmé, že se jedná o jednu z podstatných složek normy, jelikož je nutné, aby si různorodé systémy vzájemně rozuměly. Nejedná se přímo o formát, v jakém jsou data posílána, nýbrž o způsob, jakým je výměna informací prováděna.

Norma S95 „zapouzdřuje“ komunikaci do **transakcí**, což je chápáno jako soubor souvisejících zpráv. **Zpráva** označuje strukturovaná data, jež jsou posílána z jedné aplikace do jiné. Má dvě hlavní složky, a to identifikační část a datovou část. Samotná datová část se skládá z dalších dvou částí, a to z podstatného jména a slovesa. **Podstatným jménem** (tedy datovou částí) je jeden z objektů (popsaných UML diagramy), které norma ve svých částech definuje (jsou to např. jednotlivé objekty vyskytující se v Hierarchii zařízení, viz kapitola 6.2; objekt Person, reprezentující osobu z personálu apod.). Mezi přípustná podstatná jména patří např. představený *workflow*, *work master*^{194}, *work directive*^{195} atd. **Sloveso** říká, co se má stát (akce), případně odpověď na žádost. Seznam sloves a jejich význam lze shlédnout v tabulce 4.

Příklad transakce spolu s představenou strukturou zprávy, jednotlivými podstatnými jmény a slovesy lze shlédnout na obrázku 25. Nicméně samotná výměna informací může probíhat v různých módech, definovaných jako transakční modely. Jsou definovány právě tři a jsou představeny níže.

PULL je typ transakce, kdy si strana A (žadatel) vyžádá určité informace od strany B (poskytovatel). Poskytovatel dat naslouchá zprávám typu *GET*, na základě kterých poté poskytuje informace ve zprávě typu *SHOW*. Analogií jsou query dotazy na databázové systémy.

PUSH transakce probíhají v obráceném sledu od *PULL* transakcí. Zde strana A (příjemce) naslouchá zprávám *PROCESS*, *CHANGE*, *CANCEL*, které posílá



Obrázek 25: Příklad transakce
Zdroj: [51]

Tabulka 5: PUSH zprávy

Zpráva odesílatele	Odeslané informace	Odpověď příjemce
PROCESS	Nové informace	ACKNOWLEDGE
CHANGE	Změna již existujících informací	RESPOND
CANCEL	Smazání určitých existujících informací	

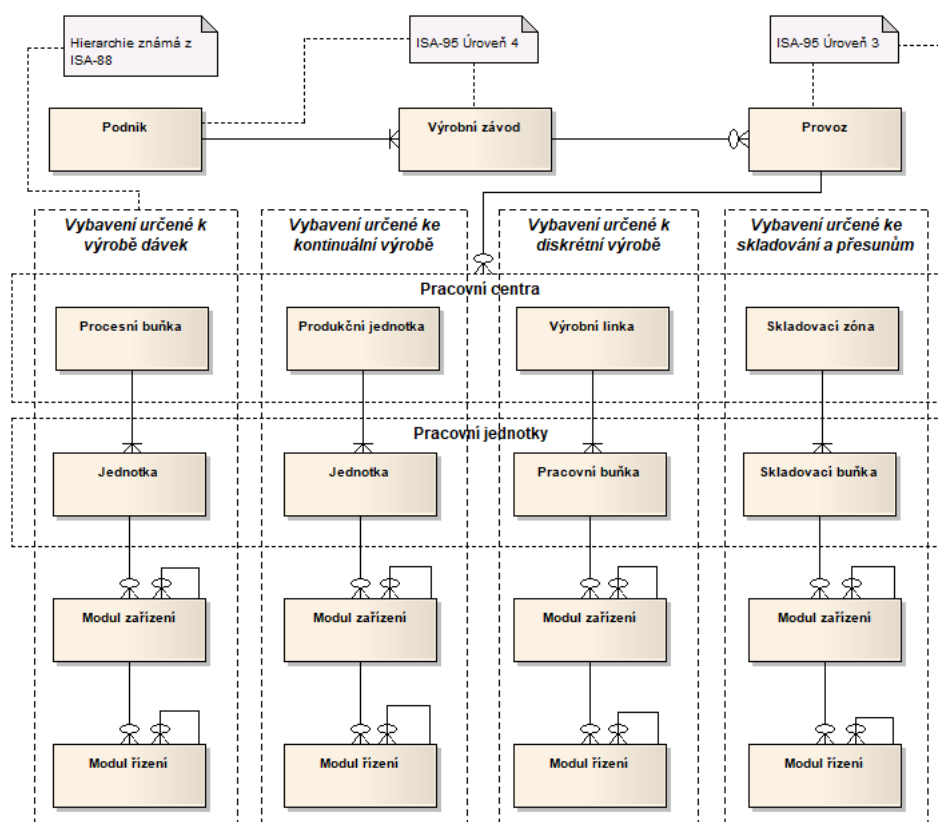
strana B (odesílatel). Příjemce odpovídá zprávami *ACKNOWLEDGE* a *RESPONSE*. Tento sled lze vidět v tabulce 5.

PUBLISH je poslední druh transakce, při kterém tzv. odběratelé naslouchají zprávám *SYNC* od vydavatele. Ti odesílají *SYNC* zprávy obsahující nové, změněné, či smazané informace.

Formát komunikace

Byl diskutován způsob komunikace mezi úrovněmi 4 a 3 (tedy v podst. mezi [ERP](#) a [MOM](#)) a vně samotné 3. úrovně. Nicméně samotný formát přenášených informací je mimo rozsah normy [S95](#). Za zmínku ovšem stojí pojem [Business-To-Manufacturing Markup Language \(B2MML\)](#)^{196} [52], který představila [MESA](#)²⁴. Jedná se o úplnou implementaci normy [S95](#) ve formátu [XML](#) v [XSD](#) schématu. Přesněji implementuje jednotlivé modely objektů (viz kapitola 6.5). [B2MML](#) je volně dostupný a je možné ho bezplatně používat, nicméně se na něj lze odkázat

²⁴Původně pojem představila organizace [World Batch Forum \(WBF\)](#), avšak ta se později spojila s [MESA](#) [53]



Obrázek 26: Korelace fyzického modelu (S88) a modelu hierarchie zařízení (S95)
Zdroj: podle [55]

i při tvorbě implementace vlastní. Součástí B2MML je dnes i BatchML^{197} [54] (viz kapitola 5.9.)

6.7 Korelace norem ANSI/ISA-88 a ANSI/ISA-95

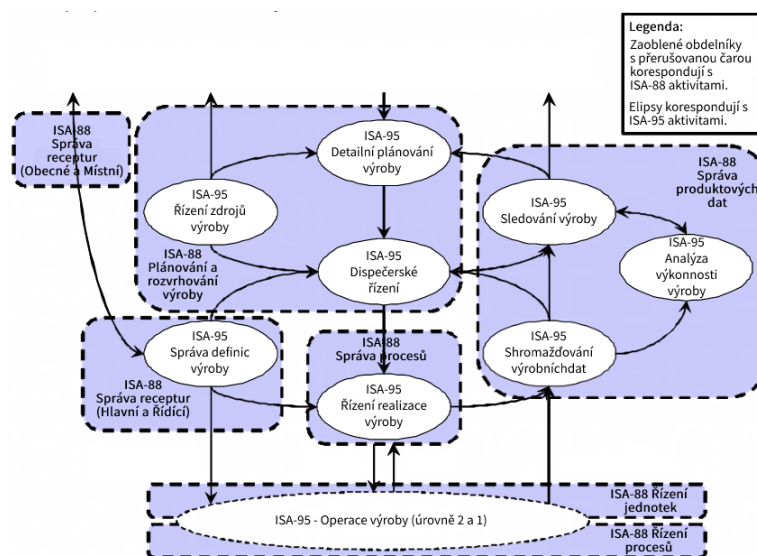
Jak již bylo párkrát v kapitole o normě S95 zmíněno, existuje vzájemná návaznost mezi normami S88 v S95. V této podkapitole vyzdihneme body zájmu, ve kterých se standardy potkávají a činnosti či konstrukty, jež společně definují.

Korelace fyzického modelu a modelu hierarchie zařízení

Jedny ze základních modelů v obou normách byly modely, jež adresovaly zařízení (vybavení) podniku. Tato korelace již byla adresována v kapitole 6.2 normy S95. Vzájemná korespondence je zvýrazněna na obrázku 26.

Korelace modelu řídicích aktivit a univerzálního modelu aktivit

V kapitole 5.7 normy S88, zabývající se řídicími funkcemi a aktivitami, byl představen model řídicích aktivit (viz obrázek 16). Ačkoliv to nemusí být zřejmé, tento



Obrázek 27: Korelace modelu řídicích aktivit (S88) a modelu aktivit řízení výrobních operací (S95)

Poznámka: jedná se univerzální model aktivit (viz obrázek 20) aplikovaný na řízení výrobních operací (viz kapitola 6.4)

Zdroj: [55]

model koresponduje s univerzálním modelem aktivit (viz obrázek 20) normy S95, představeným v kapitole 6.4. Vzájemná korespondence je vyjádřena a představena na obrázku 27.

Korelace modelů objektů

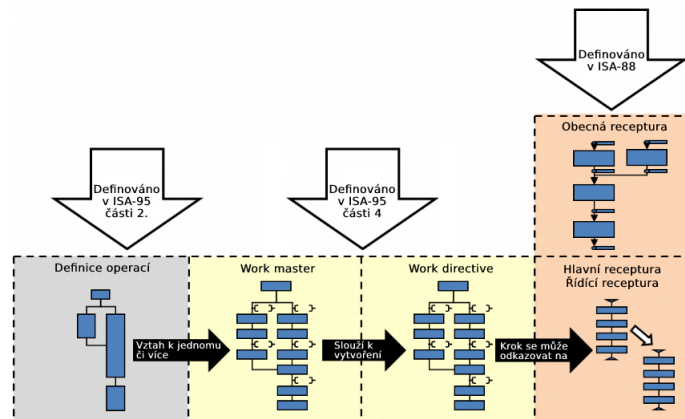
Obě zmíněné normy definují přesnou strukturu objektů a jejich modely. Jelikož jsme v kapitolách o normě S88 modely blíže nediskutovali, nebudeme tak činit ani zde.

Za zmínku ovšem stojí, že jednotlivé *modely receptur* z S88 mají blízký vztah s *modelem definice práce* z S95. Tento vztah je samozřejmě zřetelný i u více modelů, avšak pro potřeby této diplomové práce stačí zmínit pouze tento vztah. Zmíněná korelace je vidět na obrázku 28.

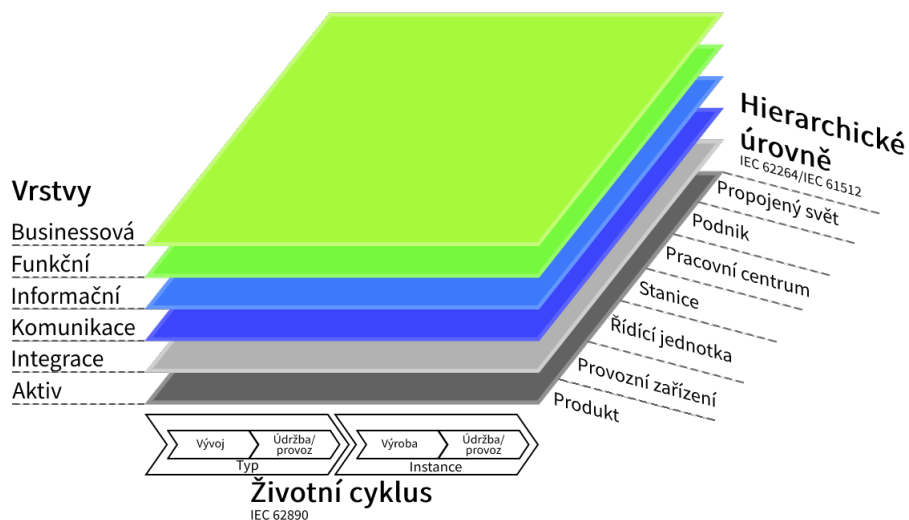
7 RAMI 4.0

V předchozích kapitolách jsme představili samotnou iniciativu Industry 4.0^[198] a normy S88^[199] a S95^[200]. V této kapitole se podíváme na formální a ucelený model, jež adresuje problematiku Industry 4.0 a bere v potaz i již zmíněné normy.

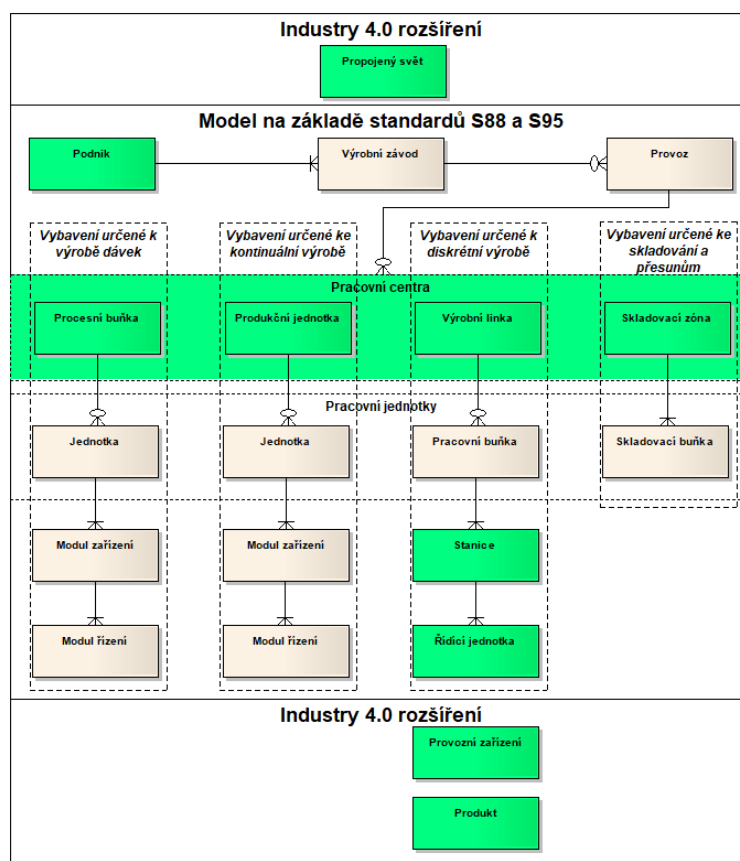
Reference Architecture Model for Industrie 4.0 (RAMI 4.0)^[201], česky Referenční architektonický model pro Industry 4.0, je třídídimenzionální mapa, jež adresuje nejdůležitější myšlenky Industry 4.0. Model se skládá ze tří jednotlivých os, které si popíšeme.



Obrázek 28: Vztahy mezi modely objektů S88 a S95
Zdroj: [50]



Obrázek 29: Reference Architecture Model for Industrie 4.0
Zdroj: podle RAMI 4.0



Obrázek 30: RAMI 4.0 hierarchie (pravá horizontální osa)

Poznámka: Zeleně označené části korespondují s pravou horizontální osou RAMI 4.0

Zdroj: Podle hierarchie RAMI 4.0

7.1 Pravá horizontální osa

Z obrázku 29 lze vidět, že se model odkazuje na normy IEC 62264^{202} a IEC 61512^{203}. Jedná se o Evropské verze norem S95 a S88. Tudíž si lze všimnout, že popis osy lehce koresponduje s hierarchií zařízení^{204} (resp. fyzickým modelem^{205}). Zde je vhodné odkázat se na kapitolu 6.7, jelikož obrázek 26 z oné kapitoly slouží jako základ pro tuto osu modelu. Jak je vyobrazeno na obrázku 30, RAMI 4.0 model tuto původní hierarchii lehce upravuje a rozšiřuje, právě o koncepty Industry 4.0. Ve vrchní části se jedná o napojení na vnější svět (IoT), jelikož normy S95 a S88 popisují struktury uvnitř podniku. Jelikož v Industry 4.0 se počítá s propojením podniků (napříč globálním hodnotovým řetězcem), bylo potřeba toto napojení přidat. V dolní části se jedná zejména o samotný produkt (viz *smart product*^{206} v kapitole 4.4.1) a tzv. *provozní zařízení*, který představuje funkční úroveň chytrého zařízení (např. chytrý senzor^{207}) [56]. Tato osa koresponduje s první charakteristikou Industry 4.0, představenou v kapitole 4.4.2.

7.2 Levá horizontální osa

Představuje ucelený model pro životní cyklus výrobních zařízení a produktů, a to na základě normy IEC 62890^{{208}25}. Osa má dvě části označované jako *typ* a *instance*. Za typ jednotlivých zařízení či produktů je považován jeho návrh a prototypování (tudíž sem patří i různá testování, validace aj.). Tudíž instance je výsledkem produkční výroby (na základě typu), jejímž výstupem jsou produkty, které míří přímo k zákazníkům. Každá část osy je ještě znovu rozdělena na další dvě části, kde první jsou vývoj a výroba a druhá část je shodná, tedy údržba a provoz (využívání). Na této ose se rovněž počítá s komunikací napříč, např. provozní data (z instance výrobku) mohou být využita ke zlepšení jeho vlastností \implies vedou k pozměnění návrhu. Zajímavostí je, že instance daného typu, jež dodavatel vyrobí a dodá zákazníkovi, je z počátku pro zákazníka opět typem a instancí se stává, až po nainstalování. Tato změna (z typu na instanci) může nastat několikrát [56]. Tato osa koresponduje zejména s druhou charakteristikou Industry 4.0, představenou v kapitole 4.4.2 a částečně také s charakteristikou třetí.

7.3 Vertikální osa

Rozkládá stroje a produkty (zde můžeme chápat jednotlivé složky pravé horizontální osy) na jednotlivé vrstvy a těm přiřazuje jasně danou funkcionalitu. Jednotlivé vrstvy si od nejspodnější popíšeme.

Vrstva aktiv zahrnuje fyzické části, jako jsou různé kovové části, ale i senzory^{209} a aktuátory^{210}, dokumenty, či dokonce lidé.

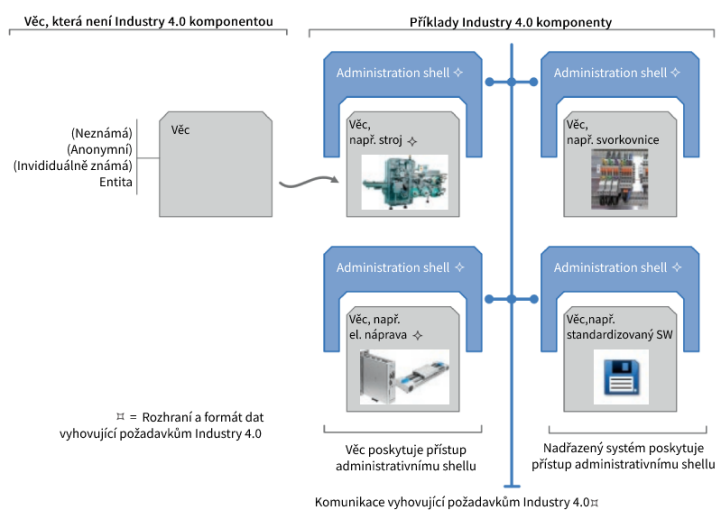
Integrační vrstva představuje rozhraní mezi skutečným světem a IT (digitálním) světem (tudíž i HMI ve vztahu k člověku). Obsahuje i fyzické prvky propojující skutečný a digitální svět (jako např. senzory či RFID čtečky). Dochází zde k propagování událostí mezi skutečným a digitálním světem (např. chyba systému vede k zastavení stroje apod.)

Komunikační vrstva standardizuje komunikaci směrem k informační vrstvě

Informační vrstva formálně popisuje pravidla, vykonává je, přiřazuje je k událostem, zajišťuje integritu dat a jejich sběr

Funkční vrstva formálně popisuje funkce, a připravuje půdu pro jejich horizontální integraci (**horizontální integrace může probíhat pouze uvnitř této vrstvy** [58]). Zároveň pouze zde může být realizován vzdálený přístup. Obsahuje prostředí pro podporu business procesů

²⁵IEC 62890 – „Měření, řízení a automatizace průmyslových procesů – Management životního cyklu systémů a částí“ [57]



Obrázek 31: Industry 4.0 komponenta
Zdroj: [56]

Businessová vrstva mapuje a propojuje business procesy podniku, obsahuje zákonné a regulační podmínky a zajišťuje integritu funkcí v hodnotovém toku

7.4 Model komponenty Industry 4.0

Na základě **RAMI 4.0** byl definován model komponent Industry 4.0^{211}. Každá komponenta musí nutně umět **komunikovat** svá data a funkce s okolím (dalšími Industry 4.0 komponenty). Zároveň musí umět svá data schraňovat napříč celým svým životním cyklem. Práci s daty model zapouzdřuje do tzv. **virtuální reprezentace**. Samotná data mohou být přímo v dané komponentě, nebo mohou být v nadřazeném **IT** systému (který opět musí být Industry 4.0 komponentou, aby je mohl komunikovat s ostatními komponentami) jejich správa je skrze **správu zdrojů**. V této virtuální reprezentaci jsou rovněž přístupná tzv. meta-data, označována jako **manifest**. Komponenta zároveň, na rozdíl od objektu, který zapouzdřuje, může mít i nějakou technickou funkcionalitu (např. dodatečný **SW**). Virtuální komponenta samotná leží v *informační vrstvě* modelu **RAMI 4.0**, přičemž její technická funkcionalita spadá do *vrstvy funkční*. Industry 4.0 komponentou se tedy může stát každý objekt ve výrobě (v podniku), avšak je potřeba zajistit, aby měl dříve zmíněné schopnosti. Pro takové účely slouží pojem **administration shell**. Administration shell^{212} slouží jako propojení fyzického a digitálního světa. Skrze něj se dají jakémukoliv objektu přidat zmíněné vlastnosti, čímž se z něj stane Industry 4.0 komponenta. Tato skutečnost je ilustrována na obrázku 31.

Existuje zde určitá návaznost a podobnost s pojmy **IoT**^{213} a **CPS**^{214}. Z podstaty definice a schopností, Industry 4.0 komponenta spadá do kategorie **IoT**. Zároveň taková komponenta zapadá i do definice **CPS** (viz definice 3). Stejně jako **CPS**, i tyto komponenty mohou být vnořené.

Část IV

Workflow systémy

V této části si přiblížíme workflow systémy, jejich podstatu a účel. Budou definovány pojmy vztahující se k nim, jako například samotný pojem workflow. Zároveň budou představeny standardy, jež se používají pro jeho vizualizaci, potažmo automatizaci.

8 Obecně

Podstata workflow systémů se vztahuje k business procesům^{215}, představeným v definici 2. Dá se říct, že jejich úkolem je přesně definovat daný business proces, a to rozdělením jej na jednotlivé malé úkoly, které budou jednoznačné. Mělo by být jasné, kdo takové úkoly bude vykonávat, ať lidé, či aplikace. Zároveň je důležité i řízení vykonávání těchto procesů, potažmo jejich automatizace.

8.1 Základní terminologie

Napříč touto částí bude používána jednotná a standardizovaná terminologie. Za její ucelenost vděčíme organizaci **WfMC**, o které bude řeč zejména v kapitole 9. Workflow se dá přeložit jako „tok práce“, avšak tento pojem se obecně nepřekládá. Dá se chápat jako určitý popis procesu (činnosti, práce) s cílem jej popsat v rámci **IT** a případně automatizovat. Z podnikového hlediska se procesem myslí přímo business proces (viz definice 2). Formální definice je následující:

Definice 9 (Workflow)

„*Workflow*^{216} je počítačové usnadnění či automatizace business procesů (ať už celého, či jeho částí)“ [59].

Definice fakticky říká, že pro daný *business proces* (např. vystavení faktury), je *workflow* jeho počítačovým ekvivalentem, rozdělený na jednotlivé kroky, s cílem takový proces automatizovat. Součástí *workflow* jsou i informace a dokumenty vyměňovány mezi jednotlivými *účastníky workflow*, za účelem provedení nějaké akce, a to v souladu s určitými procedurálními pravidly (pojem *workflow* v podst. koresponduje s **Workflow Management**^{217}) [60]. Systém (**SW**), pomocí kterého se workflow vytváří a řídí, se nazývá **Workflow Management System (WfMS)**^{218}.

Samotný počítačově reprezentovaný *business proces* spolu s jeho *workflow* společně, spadají pod pojem *definice procesu*^{219}.

Definice 10 (Definice procesu)

Je počítačovou reprezentací daného business procesu, který zahrnuje jak ma-

nuální proces, tak jeho workflow (tj. aktivity, vztahy, podmínky, ...). S touto reprezentací lze manipulovat pomocí **WfMS** a lze ji tudíž **automatizovat** [59] [60].

Každý proces se skládá z určitého počtu *aktivit*^{220}, přičemž mohou být manuální^{221}, či automatické^{222}.

Definice 11 (Aktivita)

Popis části práce, jež je logickým krokem v daném procesu [60].

Manuální aktivita, ačkoliv nemůže být automatizovaná, může být součástí *definice procesu*, avšak není součástí výsledného *workflow*^{223}.

Každý proces může mít instanci (**instance procesu**^{224}), jež představuje právě jedno provedení onoho procesu (včetně jeho dat). Uvnitř takové instance má i každá *aktivita* svou vlastní instanci (**instance aktivity**^{225}).

Každá *aktivita* generuje jednu či více *pracovních položek*^{226}. Ty následně představují úkon, jež má být proveden.

Definice 12 (Pracovní položka)

Práce, jež má být provedena, na základě aktivity [60].

Jednotlivé *pracovní položky* mohou být přidávány do tzv. *seznamů práce*^{227}.

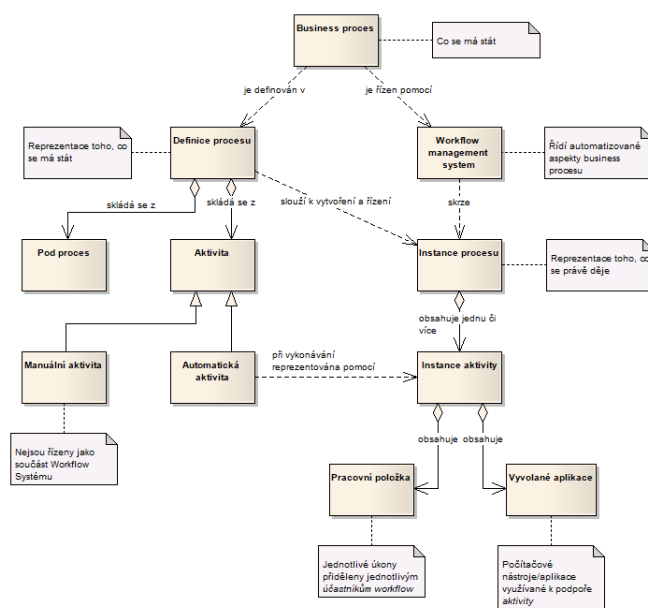
Definice 13 (Seznam prací)

Seznam *pracovních položek* přiřazených určitému *účastníkovi* (či účastníkům) *workflow* procesu (lze chápat např. člověka, popř. skupinu lidí, jež má vykonat danou *pracovní položku*)

Na základě *aktivity*^{228} může **WfMS**^{229} vyvolat i určitý **SW** (**vyvolaná aplikace**^{230}), jež slouží k automatizaci dané aktivity, či slouží jako podpurná aplikace při vykonávání dané *aktivity* určitým **účastníkem workflow**^{231} (např. zaměstnancem). Veškerá tato terminologie a její vztahy jsou jednotně vyobrazeny na obrázku 32.

8.2 Modelování

Každý *workflow*^{232}, potažmo *business proces*^{233}, musí být vymodelován, aby mohl být později použit, resp. vykonáván ve **WfMS**^{234}. Zpočátku se využívaly nástroje, které nebyly nutně přímo zamýšleny k modelování business procesů. Můžeme zmínit např. **UML**^{235}, přesněji se jednalo o diagram aktivit. Nicméně v průběhu let se začaly objevovat jazyky a standardy nové, které byly přímo určené k onomu modelování, ať už třeba Petriho sítě^{236} (či konstrukty z nich odvozené), notace od **WfMC** nebo **BPMN**^{237}. Některé z těchto standardů budou rozebrány v kapitolách níže.



Obrázek 32: Základní pojmy a jejich vztahy

Poznámka: nejedná se o standardní diagram v jazyku UML, avšak je použita notace vztahů z UML

Zdroj: podle [60]

8.3 Jazyky workflow systémů

Workflow systémy ke svému chodu potřebují a využívají určité jazyky, skrze které se definují jednotlivé *business procesy*^[238] do podoby, které systém rozumí. Rovněž jsou potřeba i jazyky, jež slouží pro vykonávání daného *workflow*^[239]. K tomu slouží právě dva druhy jazyků, a to:

- Definiční jazyky^[240]
- Exekutivní jazyky^[241]

Jak již bylo nastíněno, definiční jazyky slouží pro uložení *workflow* v určitém standardizovaném formátu, který lze přenášet v rámci systému *workflow*, ba dokonce mezi různými systémy. Na druhé straně jazyky exekutivní slouží k vykonávání daného *workflow* v daném systému workflow (SW). Tyto dvě množiny jazyků jsou různé, ačkoliv mnohé z nich využívají stejný formát, a to XML. Příklady obou druhů jazyků budou zmíněny napříč následujícími kapitolami.

8.4 Typy workflow systémů

Workflow systémy se dělí na 4 základní typy [61]. Existuje mezi nimi vzájemná provázanost a překrytí. Jednotlivé workflow standardy nutně nemusí rozlišovat mezi zmíněnými typy workflow (systémů). Výčet a jejich popis následuje.

Ad-Hoc workflow se vyskytují zejména u nestandardních procesů \implies určitá unikátnost každého workflow. V průběhu samotného procesu totiž může být *účastníkem workflow* „upraven“. např. jsou dopředu známy určité kroky, avšak jejich sekvenci určí *účastník*. Příkladem může být odpověď na dotaz zákazníka.

Administrativní workflow, jak již název napovídá, se zabývají rutinními administrativními činnostmi. Tudíž se vyznačují svou strukturovaností a jasnými definicemi. Zpravidla jsou spojovány s různými formuláři a dokumenty. Typicky se jedná například o práci s fakturami (vystavení apod.).

Kolaborativní workflow slouží k podpoře práce v týmu. Není příliš strukturovaný \implies připouští a očekává změny procesu za běhu. Typickým výstupem je dokument (přípravený daným týmem). Například se jedná o změnu designu produktu.

Produkční workflow je z pohledu této diplomové práce nejpodstatnější. Zabývá se hlavními business procesy²⁶ v podniku. Nejsou příliš pružné a jejich o definice se starají odborníci. Důležitá je zde integrace s dalšími výrobními systémy.

9 Workflow Reference Model

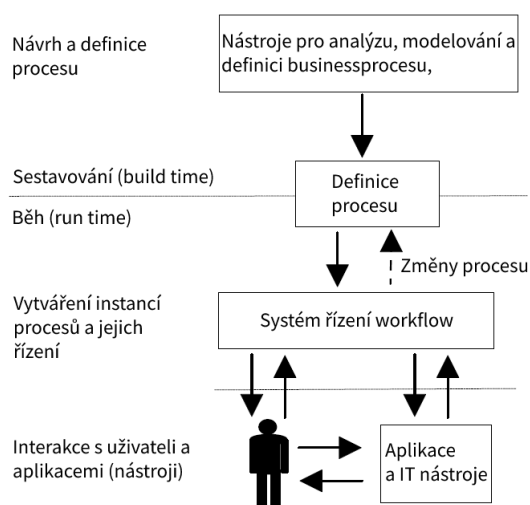
Stejně, jako tomu bylo se systémy **MES** a s nutností jejich standardizace, tak i s nástupem workflow systémů bylo potřeba uceleně definovat jednotlivé pojmy a správně zařadit funkcionalitu, aby mohly jednotlivé systémy vzájemně komunikovat bez větších potíží, a to i přes to, že řada produktů již na trhu existovala. Právě z toho důvodu v roce 1993 vznikla organizace **Workflow Management Coalition** (**WfMC**), jež sdružovala vývojáře, konzultanty, analytiky a vědecké pracovníky, jež se pohybovali v oblasti workflow a **BPM**, právě za účelem standardizace [62]. Jedním z hlavních přínosů organizace (mimo definice jednotlivých pojmů) je tzv. **Workflow Reference Model** (**WRM**)^{242}. Tento model rozděluje jednotlivou funkcionalitu workflow systémů a definuje rozhraní, která slouží pro integraci s jinými aplikacemi. V této kapitole si tento model rozebereme.

9.1 Obecná architektura workflow systémů

Abychom mohli představit samotný **WRM**, je potřeba si nejdříve rozebrat obecnou architekturu workflow systémů. **WfMS**^{243} poskytuje podporu pro tři základní oblasti, vyobrazené na obrázku 33, a to:

- Sestavovací funkce
- Řídící funkce procesu za běhu

²⁶Hlavní business proces – přidává k produktu určitou přidanou hodnotu [61]



Obrázek 33: Charakteristiky workflow systémů
Zdroj: [59]

- Interakce aktivit za běhu

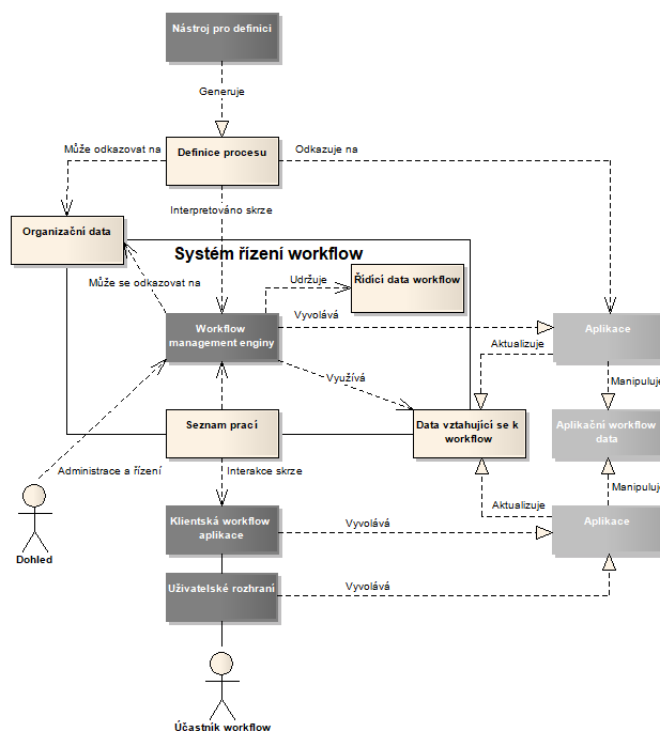
Sestavovací funkce slouží k převodu *business procesu*^{244} do jeho digitální interpretace, nazvané **definice procesu**^{245} (viz definice 10). Podoba může být textová, grafická či v nějakém formálním jazyku (definiční jazyky, viz. 8.3) Skládá se z jednotlivých aktivit, podmínek atd. (viz *definice procesu* v implementačním modelu, 9.2).

Řídící funkce procesu za běhu interpretují *definici procesu* v reálném světě \implies vytváří **instanci procesu**. Řídí jednotlivé *aktivity(instance aktivit)*^{246} uvnitř dané *instance procesu*^{247}, a to ať manuální (vykonávané lidmi), tak automatizované. Základním prvkem je tzv. workflow management **Software** (často označován jako **workflow engine**^{248}), jež se stará o vytváření a mazání procesu, řízení *aktivit* uvnitř procesu a zároveň o interakci s lidmi a jinými programy [59].

Interakce aktivit za běhu každá *aktivita*^{249} (viz definice 11) je buď manuální^{250}, či automatická^{251}. Je potřeba, aby dokázaly komunikovat se softwarem pro řízení workflow, aby bylo možné mezi nimi přecházet, *vyvolávat aplikace*^{252}, předávat potřebná data apod.

9.2 Generický model workflow systémů

Před samotným navržením **WRM** bylo potřeba nahlédnout na již existující workflow systémy a identifikovat jejich společné části. **WfMC** definovala tzv. implementační model, jež je aplikovatelný na naprostou většinu workflow systémů,



Obrázek 34: Implementační model

Poznámka: SW komponenty – tmavě šedé; různé systémové definice a data – béžové; aplikace a jejich databáze – světle šedé

Zdroj: podle [59]

a to i navzdory jejich rozdílům [59]. Tento model definuje jednotlivé komponenty těchto systémů a v zásadě se dají rozdělit do tří kategorií:

1. SW komponenty – zajišťování podpory jednotlivých funkcí ve workflow systémech
2. Různé systémové definice a řídicí data
3. Aplikace a jejich databáze, které nejsou součástí workflow systémů – mohou být vyvolány workflow systémem

Tento implementační model lze vidět na obrázku 34 (SW komponenty – tmavě šedé, různé systémové definice a data – béžové, aplikace a jejich databáze – světle šedé) a níže si popíšeme jeho jednotlivé komponenty.

Nástroj pro definici slouží k převodu *business procesu* do **definice procesu**, které rozumí workflow management **Software**.

Definice procesu^[253] se skládá z jednotlivých *aktivit*, podmínek (start, ukončení/dokončení, vykonávání, přechody), odkazů na *vyvolávané aplikace*, definicí

workflow dat atd. Zároveň odkazuje na **organizační data**, která uchovávají organizační strukturu v podniku.

Systém řízení workflow^{254} je složen z jednoho, či více *workflow engineů*^{255}, přičemž každý *engine* se stará o vykonávání určitých *instancí procesů*. *Systém řízení workflow* skrze jednotlivé *enginy* vykonává určité úkony a obsluhu (viz *workflow enginey* v kapitole 9.3.1).

Řídící data workflow zahrnují taková data, která vypovídají o vnitřní stavu jeho *instancí procesů* a *aktivit*, data potřebná pro restartování či zotavení procesu z chybového stavu. Mohou být obecně v *systému řízení workflow*, nebo být distribuovaná napříč jednotlivými *workflow enginey*.

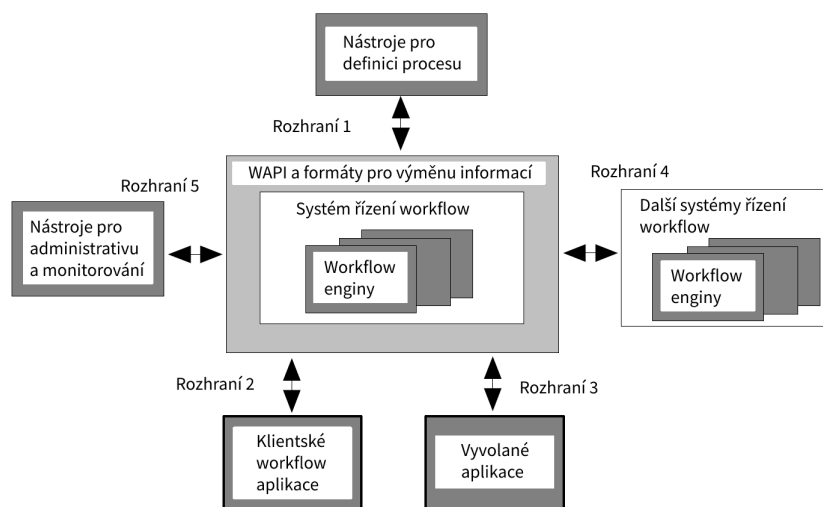
Data vztahující se k workflow jsou generována a aktualizována aplikacemi, které byly vyvolány *workflow enginey* (*vyvolané aplikace*). *Enginy* tato data využívají k navigaci napříč procesem, slouží tedy i k řízení.

Dohled nad jednotlivými pracovníky či pracovními stanicemi \implies nadřízený. Sledují mezní termíny, historii, pracovní výkony a jiné statistiky.

Seznam prací^{256} byl představen v definici 13. *Enginy* vkládají jednotlivé *pracovní položky* do těchto *seznamů*. Ty jsou poté *workflow systémem* prezentovány danému *účastníkovi workflow* (např. osobě), a to v různých podobách (může vidět všechny, nebo např. jen právě tu, kterou má vykonat).

Klientská workflow aplikace & uživatelské rozhraní představuje mezičlánek, mezi *Systémem řízení workflow* (tudíž i *enginy*) a *účastníky workflow*. **Uživatelské rozhraní** může být zakomponováno v *aplikaci*, či být odděleno. Skrze tuto kombinaci předává *účastníkovi* potřebné informace o práci, přičemž komplexita záleží na samotném *workflow systému*. *Aplikace* může umět pouze prezentovat aktuální *aktivitu*, nebo může umět komplexně přerozdělovat práci mezi více *účastníky*. Součástí funkcionality je vyvolávání klientských aplikací, jež jsou potřebné k výkonu určité činnosti (*klientská workflow aplikace* – automaticky vyvolaná, *uživatelské rozhraní* – manuálně vyvolaná *účastníkem*).

Aplikační workflow data jsou data, ke kterým mohou přistupovat pouze *vyvolané aplikace* (ať *enginy*, či *klientskou workflow aplikací* nebo *uživatelským rozhraním*). Nicméně *Workflow enginey* mohou mít za úkol tato data přemísťovat mezi jednotlivými *vyvolanými aplikacemi*.



Obrázek 35: [Workflow Reference Model](#)
Zdroj: [59]

9.3 Workflow Reference Model

Z generického modelu workflow dokázala organizace [WfMC](#) identifikovat základní oblasti, ve kterých je potřeba standardizovat výměnu informací. Bylo jich identifikováno právě 5, přičemž se nazývají „rozhraní“. Každé takové rozhraní představuje spoj mezi *systemem řízení workflow* a jednotlivými komponentami. Tento model^[257] lze vidět na obrázku 35. Jednotlivé komponenty (tj. oblasti, které jsou propojeny skrze ona rozhraní) již byly představeny v generickém modelu, tudíž zde nebudou znovu představeny. Výjimkou bude *system řízení workflow*, který zde bude více rozveden. Níže se na něj blíže podíváme a poté rozebereme jednotlivá rozhraní.

9.3.1 Systém řízení workflow

System řízení workflow^[258] je totožná funkcionalita, představená v generickém modelu workflow. Specifický je zde způsob komunikace s okolím. Ten je řešen skrze [Workflow Application Programming Interface \(WAPI\)](#). Součástí jsou i definice formátů, ve kterých jsou informace vyměňovány. V předchozí kapitole byl *system řízení workflow* představen jako jednotná složka, avšak v praxi tomu tak být nemusí. Může být rozdělen do více jednotlivých systémů, které spolu budou muset vzájemně komunikovat. např. každý *system řízení workflow* může zodpovídat za určitou skupinu procesů, a tím pádem komunikovat s určitou skupinou *účastníků workflow*^[259].

Každý *system řízení workflow* využívá a zapouzdřuje jeden či více *workflow engineů*^[260]. Tyto **Workflow engine** společně tvoří řídicí běhové prostředí uvnitř *systemu řízení workflow*. Výčet jejich funkcionality je následující [59]:

- interpretace *definice procesu*,

- řízení *instancí procesů*,
- řízení mezi *aktivitami* v procesu (sekvenční či paralelní), včetně *dat vztahujících se k workflow*,
- přihlašování a odhlašování *účastníků workflow*,
- přidávání *pracovních položek* do *pracovních seznamů* na základě *organizačních dat*,
- identifikaci *pracovních položek* pro jednotlivé *účastníky workflow*,
- správa *řídících dat* a *dat vztahujících se k workflow*, posílání *dat vztahujících se k workflow vyvolaným aplikacím* či *účastníkům workflow*,
- vyvolávání aplikací a nástrojů,
- dohledové činnosti pro řízení, administraci a auditní činnosti.

9.3.2 WAPI a výměna informací

Jednotlivé komponenty skrze svá rozhraní komunikují se *systémem řízení workflow* pomocí WAPI. Jedná se o sadu API volání s určitou sadou parametrů a sadou odpovědí. Zároveň jsou součástí i datové formáty.

9.3.3 1. rozhraní: import/export definice procesů

První rozhraní WRM leží mezi nástroji pro *definici procesů* a *systémem řízení workflow*. Fundament tohoto rozhraní je rozdělení sestavovacích funkcí a funkcí za běhu. Tím přináší i možnost exportu jednotné *definice procesu* do více různých workflow systémů. Přeposílaná data jsou nejen samotné *definice procesů*, nýbrž i různé podmnožiny, např. *aktivita* apod.

API volání ve WAPI pro toto rozhraní jsou členěny do tří kategorií [59]:

- Navázání relace
- Operace definice workflow
- Operace objektů v definicích workflow

9.3.4 2. rozhraní: klientské workflow aplikace

Druhé rozhraní spojuje *klientské workflow aplikace* se *systémem řízení workflow*. Jak již bylo naznačeno v generickém modelu (viz kapitola 9.2), součástí je i *uživatelské rozhraní*. Nicméně velmi důležitou částí jsou *seznamy prací*, které leží v *systému řízení workflow*. Toto rozhraní umožňuje *klientským workflow aplikacím* přístup do oněch seznamů, mohou z nich číst, označovat je za dokončené atd.

API volání ve WAPI pro toto rozhraní jsou členěny do následujících kategorií [59]:

- Navázání relace
- Operace definice workflow
- Funkce řízení procesů
- Funkce stavů procesů
- Funkce pro nakládání se seznamy prací a pracovními položkami
- Funkce dohledu nad procesy
- Funkce pro nakládání s daty
- Administrativní funkce
- Vyvolávání aplikací

9.3.5 3. rozhraní: vyvolané aplikace

Třetí rozhraní se zabývá komunikací mezi *vyvolanými aplikacemi* (jež jsou vyvolávány *workflow engine*) a *systemem řízení workflow*. Zde je potřeba si uvědomit, že existuje mnoho individuálních aplikací, které mohou být vyvolány, přičemž každá z nich má jiná specifika a potřeby. Vpodst. existují dva způsoby, jakými může *Workflow engine*^{261} vyvolávat aplikace, a to:

1. Zajistit podporu API volání přímo v dané *vyvolávané aplikaci*
2. Přidat mezičlánek (mezi rozhraní a *vyvolávanou aplikaci*), který bude rozumět API voláním a bude vyvolávat příslušné aplikace, označovaný jako **aplikační agent**

API volání ve **WAPI** pro toto rozhraní jsou členěny do tří kategorií [59]:

- Navázání relace
- Funkce řízení aktivit
- Funkce pro nakládání s daty

9.3.6 4. rozhraní: interoperabilita workflow systémů

Čtvrté rozhraní rozebírá interoperabilitu samotných workflow systémů, potažmo *systemů řízení workflow*. Interoperabilitou se myslí schopnost dvou rozdílných workflow systémů komunikovat a spolupracovat. Takové systémy by měly být schopné jednotně nahlížet na *definici procesu*, předávat si *řídící data* a *data vztahující se k workflow*.

Díky tomuto přístupu může nastat několik scénářů propojení „spolupráce“ mezi jednotlivými *systemy řízení workflow*:

1. Řetězení – v určitém bodu procesu A z jednoho *systemu řízení SRA* se aktivuje další proces B v jiném *systemu řízení SRB* (neexistuje žádná následná synchronizace mezi nimi).
2. Zanoření – v určitém bodu procesu A z jednoho *systemu řízení SRA* se aktivuje další proces B v jiném *systemu řízení SRB*, přičemž proces A je pozastaven a čeká na dokončení procesu B . Po dokončení B , pokračuje A dalším krokem (\implies jedná se o pod-proces, který běží v jiném *systemu řízení workflow*).
3. Přímé – proces A ze *systemu řízení SRA* a proces B ze *systemu řízení SRB*, jsou vzájemně provázány a vykonávány jako jeden kompozitní proces C , který má *aktivity* z A i B . Tudíž je proces vykonáván napříč SRA i SRB .
4. Paralelní synchronizace – jak proces A ze *systemu řízení SRA*, tak proces B ze *systemu řízení SRB*, mají definovanou určitou svou *aktivitu*, které když jsou v obou procesech dosaženy, tak je vyvolána nějaká společná událost.

Tato propojení jsou realizována buď skrze přímou komunikaci jednotlivých **WAPI** (které zapouzdřují *system řízení workflow* s ohledem na komunikaci). V tomto případě je nutné, aby měly stejný *definiční nástroj*, názvy, sdílely stejný typ dat apod. Nebo je druhou možností přidání mezičlánku v podobě **aplikační brány**. Skrze ni se docílí namapování jednotlivých (různých) *definic*, názvů a dat do jednotné podoby, pro následnou komunikaci.

API volání ve **WAPI** pro toto rozhraní jsou členěny do několika kategorií [59]:

- Navázání relace
- Operace nad *definicemi procesů* a jejich objekty
- Funkce řízení procesů
- Funkce stavů procesů
- Funkce řízení aktivit
- Funkce pro nakládání s daty

9.3.7 5. rozhraní: monitorování a administrace

Poslední, páté rozhraní se zabývá napojením monitorovacích a administrativních systémů a *systemů řízení workflow*. V podniku je běžné, že je potřeba monitorovat jednotlivé jeho procesy za účelem měření výkonu, optimalizace apod. Právě skrze toto rozhraní budou moci řídicí pracovníci vstupovat do workflow, vidět jeho stav a zároveň jej monitorovat.

API volání ve **WAPI** pro toto rozhraní jsou členěny do několika kategorií [59]:

- Navázání relace

- Operace řízení rolí
- Operace řízení auditů
- Operace řízení zdrojů
- Funkce dohledu nad procesy
- Funkce stavů procesů

9.4 Notace

Organizace [WfMC](#) rovněž představila grafickou notaci [63], jež měla sloužit k definování *business procesů*. Avšak notace jako samotná se neujala a byly používány jiné standardy (v pozdějších verzích specifikace jazyku [XPDL](#) již není ani zmíněna). Z toho důvodu zde nebude více rozebrána. Nicméně zajímavější to bylo na poli jazyků, které sloužily k onomu definování *business procesů*. Takové jazyky vytvořené organizací [WfMC](#) si představíme níže. Stojí za zmínku, že jejich struktura odpovídá struktuře grafické notace, jež [WfMC](#) představila.

9.5 Definiční jazyky

V 1. rozhraní byla definována výměna *definice procesů*. Nicméně pro spolehlivou výměnu bylo třeba definovat i formát. Je důležité říct, že definiční jazyky^{262} pouze nesou informace o samotné textové či grafické definici *workflow/business procesu*. Nejsou určeny a ani nemohou být použity pro vykonávání v systému workflow. K tomu slouží již zmíněné jazyky exekutivní^{263}.

Z počátku byl představen standardizovaný jazyk [Workflow Process Definition Language \(WPDL\)](#). Příklad modelování v tomto jazyku lze vidět v kódu 1 [63] (tento příklad je velmi jednoduchý a useknutý, slouží pouze ke znázornění syntaxe a způsobu definic). Bohužel se vydání tohoto jazyku sešlo s vydáním a šířením jazyku [XML](#). Tudíž organizace [WfMC](#) začala pracovat na jazyku novém, který již byl ve formátu [XML](#). Tento nový jazyk částečně vycházel z [WPDL](#) a nesl název [XML Process Definition Language \(XPDL\)](#)^{264} a na tento jazyk se podíváme blíže.

Jazyk [XPDL](#) se skládá z několika elementů. Ve verzi 1.0 se jednalo o elementy, které lze vidět v tabulce 6, jež vyobrazuje tyto elementy spolu s jejich českým ekvivalentem a významem. Samotný element *activity* má tři různé typy:

1. **Route** – jedná se o fiktivní aktivitu, která slouží pro vyjádření několika za-nořených podmínek (if, else if, else if, else, . . .), startovní/koncové symboly, pro vyjádření paralelismu apod.
2. **Block** – slouží k vykonání určité množiny aktivit (se svými přechody), po jejímž vykonání se pokračuje další aktivitou

```

1 WORKFLOW 'SALES_ORDER_PROCESSING'
2   CREATED 1998-07-15
3   NAME "Sales Order Processing "
4
5 // <Activity List>
6
7 ACTIVITY 'RECEIVE_ORDER'
8   DESCRIPTION "In the mailroom. When sales order comes in, send
9     copies to Sales, Finance, and Mfg."
10  IMPLEMENTATION NO
11  SPLIT AND
12 END_ACTIVITY
13
14 ACTIVITY 'NOTIFY_SALES'
15   DESCRIPTION "Track sale, keep customer happy"
16   IMPLEMENTATION NO
17 END_ACTIVITY
18   ...
19 // <Transition Information List>
20 TRANSITION 'T_1'
21   FROM 'RECEIVE_ORDER'
22   TO 'NOTIFY_SALES'
23 END_TRANSITION
24   ...

```

Zdrojový kód 1: Příklad procesu v jazyce [WPDL](#)

3. **Implementation** – jedná se aktivitu, jak ji známe z běžné definice, přičemž má následující druhy:

- **No** – jedná se buď o aktivitu manuální, či implicitní známou přímo *workflow engine*
- **Tool** – aktivita vykonávaná skrze jednu či více *vyvolaných aplikací*
- **Subflow** – jedná se o další pod proces (asynchronní i synchronní vykonávání)

Příklady zdrojových kódů lze vidět v příloze [B](#), konkrétně kódy [2](#), [3](#) a [4](#). Lze na nich vidět příklad zápisu *workflow* v jazyce [XPDL](#), tedy ve formátu [XML](#). Jednotlivé kódy na sebe „navazují“, resp. kódy [3](#) a [4](#) lze zanořit do kódu [2](#).

S představím [BPMN^{265}](#) se stal jazyk [XPDL](#) jedním z hlavních standardizovaných formátů, ve kterém mohly být graficky definované procesy v [BPMN](#) přenášeny. Tato skutečnost je více rozvedena v kapitole o [BPMN](#), viz [11.4](#).

9.6 Shrnutí

[WfMC](#) docílila určitého sjednocení a standardizace na poli workflow systémů. Z jednotné terminologie, a hlavně z modelu, který organizace představila, tě-

Tabulka 6: Elementy **XPDL** 1.0

Element	Český ekvivalent	Účel
Package	Balíček	slouží pro zapouzdření elementů níže
Application	Aplikace	koresponduje se představeným pojmem <i>vyvolávané aplikace</i> (viz pojem v kapitole 8.1)
WorkflowProcess	Definice procesu	koresponduje s <i>definicí procesu</i> (viz definice 10), resp. pod procesu
Activity	Aktivita	koresponduje s představeným pojmem <i>aktivita</i> (viz definice 11)
Transition	Přechod	označuje přechod mezi jednotlivými <i>aktivitami</i>
Participant	Účastník	koresponduje s představeným pojmem <i>účastník workflow</i>
DataField	Datové pole	koresponduje s pojmem <i>data vztahující se k workflow</i> (viz kapitola 9.2)
DataTypes	Typ dat	typ <i>dat vztahujících se k workflow</i>

žili jak samotní dodavatelé systémů, tak zejména zákazníci. Samotná organizace **WfMC** byla v roce 2019 rozpuštěna, jelikož považovala svou práci za hotovou [64].

Samotný **WRM**^{266} je dodnes skvělým přínosem do světa business procesů^{267} a workflow^{268}. Díky tomuto modelu mohou implementátoři workflow systémů jednotně vytvářet software pro řízení a správu workflow. Ovšem nejvíce z toho těží samotní uživatelé, resp. zákazníci, jelikož je mnohem jednodušší identifikovat, který systém splňuje požadavky, a který ne.

Model je pro implementaci systému workflow dnes již nepostradatelný. Dá se označit za jeden ze základních stavebních kamenů při implementaci. Díky jeho vymezení funkcionality lze vytvářet čisté implementace, jež budou umět vzájemně komunikovat s heterogenními systémy workflow (jejichž implementace bude také v souladu s **WRM**). Spolu s modelem byla představena i notace, která se neujala. Avšak s notací představený jazyk **XPDL** je používán dodnes (v průběhu života byl rozšiřován pro potřeby **BPMN**).

10 Petriho sítě

Petriho sítě^{269}, anglicky Petri Nets (PN), jsou nástroj představený Carlem Adamem Petrim, jež měl sloužit k modelování a analýze procesů [65]. Jedná se o formálně popsáný model, který má i své grafické vyjádření. Od jejich představení v roce 1962, byly Petriho sítě dále studovány a byla navržena a přijata různá rozšíření [66]. V následujících odstavcích a podkapitolách Petriho sítě představíme, rozebereme a ukážeme rozšíření, která jsou podstatná pro možnost modelovat

business procesy pomocí Petriho sítí. Zároveň bude představen konstrukt, jež byl přímo určen pro ono modelování business procesů.

Při definici Petriho sítí v této kapitole se vycházelo z [65] [66] [67].

Definice 14 (Petriho síť)

Petriho síť je uspořádaná čtveřice $PN = (P, T, F, M_0)$, kde

- P je konečná množina míst (*places*),
- T je konečná množina přechodů (*transitions*), přičemž platí $P \cap T = \emptyset$,
- $F \subseteq (P \times T) \cup (T \times P)$ je množina propojení (tok \rightarrow *flow*),
- M_0 je počáteční stav/značení (*marking*) – viz definice 15.²⁷

Na základě této definice zavedeme značení vstupních a výstupních míst resp. přechodů, pro jednotlivé přechody, resp. místa:

- Vstupní místa přechodu t značíme $\bullet t$. $p \in \bullet t \iff \exists a \in F: a = \langle p, t \rangle$ ²⁸.
- Výstupní místa přechodu t značíme $t \bullet$. $p \in t \bullet \iff \exists a \in F: a = \langle t, p \rangle$ ²⁹.
- Vstupní přechody místa p značíme $\bullet p$. $t \in \bullet p \iff \exists a \in F: a = \langle t, p \rangle$ ³⁰.
- Výstupní přechody místa p značíme $p \bullet$. $t \in p \bullet \iff \exists a \in F: a = \langle p, t \rangle$ ³¹.

Pojďme se na Petriho síť podívat trochu neformálně a představit jejich grafické znázornění. Petriho síť $PN = (P, T, F, M_0)$ lze vyjádřit biparitním grafem³² $G = (V, E)$, kde $V = P \cup T$ a $E = F$. **Místa** se v takovém grafu označují kruhem, přičemž se jedná o tzv. *pasivní* prvky sítě. Můžou být chápány jako podmínky, fáze či buffery. **Přechody** jsou označovány obdélníky a jedná se o *aktivní* prvky sítě, tudíž to zpravidla bývají nějaké činnosti, události apod. Mezi místy a přechody jsou hrany (propojení), přičemž jak vyplývá z definice, spojit lze pouze místo s přechodem a naopak.

Vykonávání v Petriho sítích je prováděno pomocí tzv. **tokenů**. Každé místo může obsahovat jeden či více tokenů (v grafu se značí tečkou uvnitř místa, popř. číslem, je-li tokenů více). Pokud vezmeme všechny místa z P a vezmeme v potaz počet tokenů v každém takovém $p \in P$, pak se jedná o **stav** Petriho sítě nebo **značení**.

²⁷Ve většině zdrojů je Petriho síť definována jako trojice (tj. bez počátečního stavu). V této práci si ji ovšem zavedeme jako čtveřici, jelikož pro účely modelování business procesů bude počáteční stav důležitou částí.

²⁸Jsou to místa p , pro která platí, že z p do t existuje propojení.

²⁹Jsou to místa p , pro které platí, že z t do p existuje propojení.

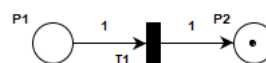
³⁰Jsou to přechody t , pro které platí, že z t do p existuje propojení.

³¹Jsou to přechody t , pro které platí, že z p do t existuje propojení.

³²Biparitní graf – takový $G = (V, E)$, kde $V = \{V_1 \cup V_2 \mid V_1 \subset V, V_2 \subset V, \text{přičemž } V_1 \cap V_2 = \emptyset\}$ a $E = \{\langle v_1, v_2 \rangle \mid v_1 \in V_1 \text{ a } v_2 \in V_2\}$



(a) Před provedením přechodu



(b) Po provedení přechodu

Obrázek 36: Příklad Petriho sítě

Definice 15 (Stav Petriho sítě)

Stav Petriho sítě $M: P \mapsto \mathbb{N}^{033}$

Počáteční stav se značí jako M_0 .

Stavy Petriho sítě můžeme značit jako formuli. Pro místa $p_1, p_2, p_3 \in P$ značíme $0p_1 + 2p_2 + 1p_3$, kde číslo před místem p_i značí, kolik tokenů je v daném místě. Další možností je značení pomocí vektorů. V tomto případě by odpovídající vektor byl $[0, 2, 1]$. Abychom mohli porovnávat jednotlivé stavy, je definováno následující částečné uspořádání: Pro libovolné M_1 a M_2 platí, že $M_1 \leq M_2 \iff \forall p \in P : M_1(p) \leq M_2(p)$, kde $M(p)$ značí počet tokenů v místě p , ve stavu M . Při vykonávání procesu v Petriho sítích se očividně jednotlivé stavy mění, tudíž se přesouvají i jednotlivé tokeny. Mějme jednoduchou $PN = (\{p_1, p_2\}, \{t_1\}, \{\langle p_1, t_1 \rangle, \langle t_1, p_2 \rangle\}, M_0 = [1, 0])$ (tj. počáteční stav s právě jedním tokenem v místě p_1). Přesunutí tohoto tokenu (**provedení přechodu**) z místa p_1 , do místa p_2 , přes přechod t_1 (příklad je vyobrazen na obrázku 36) je závislé na **proveditelnosti přechodu** t_1 :

Definice 16 (Proveditelnost přechodu)

Přechod t je ve stavu M *proveditelný*, pokud: $\forall p \in \bullet t : M(p) > 0^{34}$.

Definice 17 (Provedení přechodu)

Každý *proveditelný* přechod t může být proveden. Provedením t ve stavu M získáme nový stav M' takový, že:

$$\forall p \in P : M'(p) = \begin{cases} M(p) - 1 & \text{pokud } p \in \bullet t^{35} \\ M(p) + 1 & \text{pokud } p \in t \bullet^{36} \\ M(p) & \text{jinak} \end{cases}$$

V Petriho sítích se také může uvádět **ohodnocení jednotlivých propojení** (propojení z F). Pakliže máme místo $p \in P$ a přechod $t \in T$, jež jsou propojeny (tedy $\langle p, t \rangle \in F \implies p \in \bullet t$), tak ohodnocení tohoto propojení značí, kolik tokenů musí být v p , aby mohl být přechod t proveden. V grafu je to značeno

³³Jedná se o zobrazení z množiny míst P do množiny přirozených čísel a nuly – \mathbb{N}^0

³⁴Každé vstupní místo p pro přechod t má alespoň 1 token.

³⁵Zkonzumujeme jeden token z každého vstupního místa p přechodu t .

³⁶Vyprodukuje jeden token do každého výstupního místa p přechodu t .

číslem u příslušné hrany. Nicméně jelikož při modelování business procesů má smysl uvažovat pouze ohodnocení 1, tak se jím nebudeme explicitně zabývat.

Pro následující definice je potřeba si pro danou Petriho síť $PN = (P, T, F, M_0)$ a stav M_1 zavést následující notace:

- $M_1 \xrightarrow{t} M_2$ = přechod t je proveditelný ve stavu M_1 a provedením t získáme stav M_2 .
- $M_1 \rightarrow M_2$ = existuje přechod t takový, že $M_1 \xrightarrow{t} M_2$
- $M_1 \xrightarrow{\sigma} M_n$ = sekvence přechodů $\sigma = t_1 t_2 \dots t_{n-1}$ vede ze stavu M_1 do stavu M_n , tj. $M_1 \xrightarrow{t_1} M_2 \xrightarrow{t_2} \dots \xrightarrow{t_{n-1}} M_n$

Na základě těchto notací můžeme definovat **dosažitelnost stavu**:

Definice 18 (Dosažitelnost stavu)

Dosažitelnost stavu M_n ze stavu M_1 značíme $M_1 \xrightarrow{*} M_n$ a platí:
 $M_1 \xrightarrow{*} M_n \iff \exists \sigma: M_1 \xrightarrow{\sigma} M_n$

Tudíž lze říci, že určitý stav M je v $PN = (P, T, F, M_0)$ dosažitelný právě tehdy, platí-li $M_0 \xrightarrow{*} M$, tj. dokážeme nalézt takovou sekvenci přechodů $t_1 t_2 \dots t_{n-1}$, která nás dovede do stavu M .

Díky dosažitelnosti můžeme definovat tzv. **živost Petriho sítě**:

Definice 19 (Živá Petriho síť)

Petriho síť $PN = (P, T, F, M_0)$ se nazývá **Živou Petriho sítí**, pokud platí $\forall M$ a $\forall t$, kde $M_0 \xrightarrow{*} M$ a $t \in T: \exists M'$, kde $M \xrightarrow{*} M'$, jež dovoluje proveditelnost t ³⁷.

Definice živosti nám říká, že pokud je Petriho síť živá, pak lze každý její přechod t provést libovolně mnohokrát. Tedy můžeme se dostat do každého jejího přechodu t , nemůže nastat deadlock či zacyklení v nekonečné smyčce. Z principu business procesu (má nějaký začátek a nějaký konec), je potřeba při modelování využívat Petriho sítě, jež nejsou živé. Nicméně jednoduchou operací přidání jednoho přechodu lze takovou síť „oživit“.

Ještě si uvedeme dvě další vlastnosti, kterými lze Petriho síť označovat, a to **omezenost** a **bezpečnost**, která je speciálním případem omezenosti.

Definice 20 (Omezenost Petriho sítě)

Mějme Petriho síť $PN = (P, T, F, M_0)$. O libovolném $p \in P$ můžeme říct, že je **omezeným místem** právě když platí, že $\exists n \in \mathbb{N}$ tak, že $\forall M$, kde $M_0 \xrightarrow{*} M: M(p) \leq n$.

Pokud $\forall p \in P$ platí $M(p) \leq n$, pak můžeme říct, že daná PN je **omezená Petriho síť**.

³⁷Pokud pro každý dosažitelný stav M (z poč. stavu) a každý přechod t platí, že existuje nějaký dosažitelný stav M' (ze stavu M), který dovolí proveditelnost t , pak je Petriho síť živá.

Definice 21 (Bezpečnost Petriho sítě)

Mějme Petriho síť $PN = (P, T, F, M_0)$. O libovolném $p \in P$ můžeme říct, že je **bezpečným místem** právě když platí, že

$$\forall M, \text{ kde } M_0 \xrightarrow{*} M: M(p) \leq 1.$$

Pokud $\forall p \in P$ platí $M(p) \leq 1$, pak můžeme říct, že daná PN je **bezpečná Petriho síť**.

Jak později vyplyne, tak při modelování business procesů budeme využívat omezené Petriho sítě, dokonce se bude jednat o sítě bezpečné (tedy omezené číslem 1).

10.1 Rozšíření Petriho sítí

Představené základní Petriho sítě mají několik nedostatků, jež je činí nedostatečnými pro modelování business procesů. V následujících podkapitolách budou představena rozšíření Petriho sítí, jež zvýší jejich použitelnost pro ono modelování. Petriho sítě, které obsahují všechna následující tři rozšíření se obecně nazývají [High-Level Petri Nets \(HLPN\)](#)^{270}.

Je potřeba zmínit, že barevné rozšíření Petriho sítí je pro modelování business procesů stěžejní, avšak samotné je nedostatečné, což vyplyne při představení oněch dalších rozšíření. Tudíž přestože byl jedním z úkolů této diplomové práce pohled právě na Barevné Petriho sítě, tak bude představen takový konstrukt Petriho sítí, který je pro modelování business procesů dostatečný.

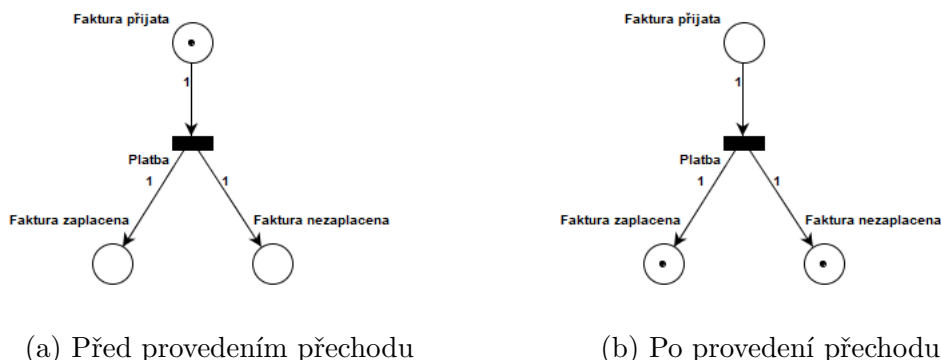
10.1.1 Barevné Petriho sítě

Jedním z velkých nedostatků je samotný **nedeterminismus** (který přechod je proveden dříve) a fakt, že **přechod t vyprodukuje token do všech $p \in t$** . Představíme-li si jednoduchou Petriho síť

PŘÍKLAD 22 (PŘÍKLAD PETRIHO SÍTĚ – FAKTURA)

$$\begin{aligned} PN &= (P, T, F, M_0), \text{ kde} \\ P &= \{Fakturapřijata, Fakturazaplacena, Fakturanezaplacena\}, \\ T &= \{Platba\}, \\ F &= \{\langle Fakturapřijata, Platba \rangle, \langle Platba, Fakturazaplacena \rangle, \\ &\quad \langle Platba, Fakturanezaplacena \rangle\}, \\ M_0 &= [1, 0, 0], \end{aligned}$$

pak provedení přechodu *Platba* vyprodukuje token do míst *Faktura zaplacena* i *Faktura nezaplacena*, Ilustrováno na obrázku 37. To je ovšem nežádoucí, jelikož téže faktura nemůže být, jak zaplacena, tak nezaplacena. Dalším problémem



Obrázek 37: Příklad Petriho sítě – faktura

je, že nemůžeme rozlišovat jednotlivé faktury, jelikož token sám o sobě nenes žádnou informaci.

Pokud Petriho sítě rozšíříme o tzv. obarvení tokenů, bavíme se o Barevných Petriho sítích (budeme je označovat jako **Coloured Petri Nets (CPN)**^{271}). Obarvením dosáhneme možnosti rozlišovat jednotlivé tokeny, a tudíž jim přiřazovat i různé vlastnosti. Tudíž token z předchozího příkladu může „být zabarven“ informací o odkazu na danou přijatou fakturu. Jednotlivé přechody t mohou s obarvením pracovat, a to jako:

1. Tokeny z $p \in \bullet t$ jsou konzumovány pouze pokud splňují nějakou **vstupní podmínku**.
2. Produkují tokeny pouze do některých $p \in t \bullet$ (na základě **výstupní podmínky**), a to v určitém počtu.
3. Obarvení mohou ovlivnit (pozměnit informaci).

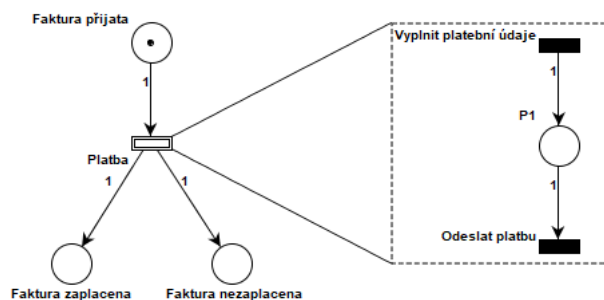
V našem příkladu 22 může být proveditelnost stavu *Platba* vázána na podmínku, že částka na faktuře (odkaz na ni nese zabarvený token) bude > 0 . Po provedení pak může vyprodukovat token pouze do jednoho místa, a to na základě informace v tokenu (tj. faktura zaplacená či nezaplacená).

10.1.2 Petriho sítě rozšířené o čas

Ačkoliv díky **CPN** už dokážeme adresovat nedeterminismus a adresaci tokenů, tak změny mezi stavy jsou vykonány ihned. Petriho sítě neumějí nijak pracovat s časem. Tudíž se Petriho sítě setkaly s dalším rozšířením, a to právě o čas. Toto rozšíření má několik obdob, nicméně my si představíme takové, jež tokenům přidává tzv. **time stamp** (neboli **časové razítko**) určující, kdy může být zkonsumován přechodem.

Definice 23 (Čas proveditelnosti přechodu)

Pro přechod t se jedná o nejdřívější okamžik, kdy všechny jeho $p \in \bullet t$ mají tokeny, jejichž časová značka je menší rovna aktuálnímu času.



Obrázek 38: Petriho sítě – hierarchické rozšíření

Máme-li dva *proveditelné* přechody, tak je *proveden* ten s nejnižším časem proveditelnosti. Po provedení jsou opět produkovány tokeny, které jsou opět opatřeny časovým razítkem. To může být stejné, jako při vstupu, nebo navýšené o určitou hodnotu.

10.1.3 Hierarchické Petriho sítě

Díky dvěma rozšířeními představeným výše, je již možné modelovat komplexní business procesy. Nicméně zkusili-li bychom si nějaký komplexní business proces vymodelovat, mohla by být korespondující Petriho síť (s patřičnými rozšířeními) velká a nepřehledná. Stejně, jako je tomu při programování, i při modelování je žádoucí „zabalit“ větší funkční prvky do určitých bloků (v naší analogii do jednotlivých funkcí), a ty pak zjednodušeně používat v hlavním modelu (v naší analogii v hlavní funkci). Z toho důvodu byly Petriho sítě rozšířeny o tzv. **pod-sítě**. Jedná o vnořenou síť PN' v síti PN , která se v síti PN vyobrazuje jako dvojitý obdélník (vychází z označení přechodů). Tato skutečnost je ilustrována na obrázku 38.

10.2 Workflow sítě

Na základě představených rozšíření Wil Van der Aalst definoval tzv. Workflow sítě^[272] [66] [67]. Mapují jednotlivé konstrukty workflow do rozšířených Petriho sítí. Jako základní prvek je brán samotný proces (business proces). V tomto případě se nabízí jasné mapování, a to proces \iff rozšířená Petriho síť. Nicméně je zde potřeba definovat určitá omezení na ony Petriho sítě, aby výsledný konstrukt správně popisoval daný business proces, a to zejména začátek a konec business procesu. Zároveň je nežádoucí, aby existovaly nějaké nedosažitelné stavy. Pro účely formální definice Workflow sítí si nejdříve definujeme cestu v Petriho síti.

Definice 24 (Cesta v Petriho síti)

Mějme Petriho síť $PN = (P, T, F, M_0)$. Cesta C v této PN z uzlu n_1 do uzlu n_k ($C(n_1, n_k)$) je sekvence $\langle n_1, n_2, \dots, n_k \rangle$, kde $n \in P \cup T$, $\langle n_i, n_{i+1} \rangle \in F$ a $1 \leq i \leq k - 1$.

Definice 25 (Workflow síť)

Petriho síť $PN = (P, T, F, M_0)$ je Workflow síť (WF-síť), právě když platí:

1. PN má místo $i \in P$ takové, že $\bullet i = \emptyset$, označováno jako **počáteční místo**.
2. PN má místo $o \in P$ takové, že $o \bullet = \emptyset$, označováno jako **koncové místo**.
3. Přidáme-li přechod t^* tak, že $\bullet t^* = \{o\}$ a $t^* \bullet = \{i\}$, pak $\forall k, j \in (P \cup T): \exists C(k, j)$ ³⁸.
4. $M_0(i) = 1 \wedge \forall p \in P \setminus \{i\}: M_0(p) = 0$ ³⁹.

Další mapování ve WF-sítích se nabízí na jednotlivé aktivity. Je zřejmé, že budou korespondovat s jednotlivými přechody, jelikož jak bylo řečeno pod definicí Petriho sítí, přechody se označují jako aktivní prvky sítě. Naopak místa tvoří pasivní prvky sítě a označují se jako conditions (podmínky), ale můžeme je chápat i jako stavy (např. zboží vyrobeno).

Samotné vykonávání procesu ve vymodelované síti bude reprezentováno tokeny. Nabízí se dvě možnosti, jak odlišit jednotlivé instance procesu. Buď lze využít **CPN**, tedy každou instanci obarvit. Druhou možností je pro každou instanci procesu vytvořit novou kopii vymodelované sítě. Druhá možnost se jeví jako přehlednější. Při vykonávání procesu může dojít k určitému větvení, kdy původně jeden startovní token, vyústí ve vícero tokenů. Pakliže dorazí token do **koncového stavu**, pak v dané síti **nesmí** existovat další token (stejného obarvení – pakliže využíváme rozlišení instancí pomocí **CPN**), jelikož by to znamenalo, že proces je špatně navržen. V následujících podkapitolách bude diskutováno přijetí rozšíření o barvy a čas. Hierarchické rozšíření je velmi intuitivní a není třeba jej rozvádět.

10.2.1 Konstrukty řízení toku ve Workflow sítích

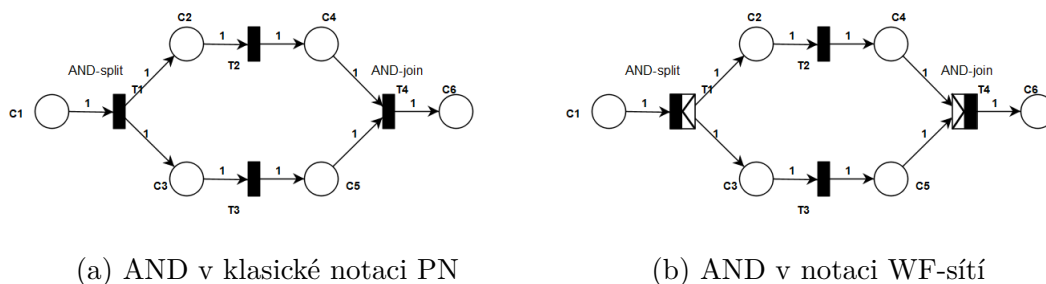
Wil Van der Aalst identifikoval 4 základní větvení ve Workflow, které promítl do Workflow sítí, spolu s novými notacemi [66] [67]. Jednotlivé konstrukty si krátce popíšeme a ukážeme i jejich nové přehlednější notace.

Sekvence je nejzákladnější konstrukt. Jedná o klasické po sobě jdoucí vykonávání, kdy aktivita B je vykonána až po vykonání aktivity A (tj. vykonání B je závislé na vykonání A).

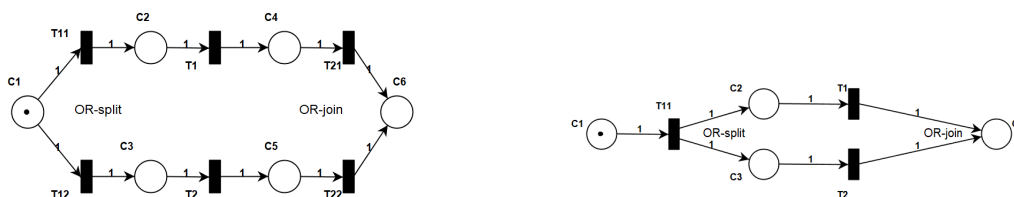
Paralelismus je mnohem zajímavější. Jedná se o vykonávání dvou (a více) různých větví zároveň. Jak je vidět z příkladu na obrázku 39a, tak $t1$ již z definice Petriho sítě, vyprodukuje tokeny do obou míst $c2$ a $c3$. V tomto větvení

³⁸Pokud spojíme koncové a počáteční místo, pak mezi každým přechodem a místem bude existovat cesta.

³⁹V počátečním stavu máme pouze jeden token, a to v počátečním místě – představuje jednu instanci procesu



Obrázek 39: Workflow sítě – AND a notace



(a) S vstupní podmínkou v aktivitě a dodatečnými route aktivitami (b) S výstupní podmínkou v aktivitě a bez dodatečných route aktivit

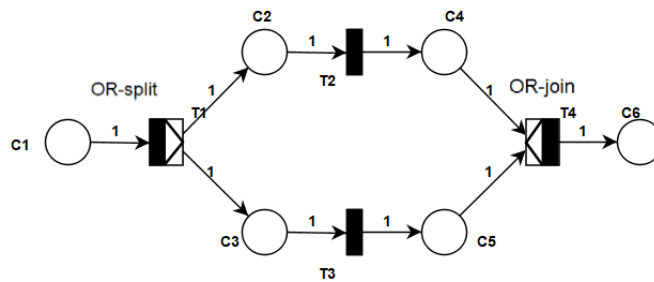
Obrázek 40: Workflow sítě – OR v klasické notaci

je přípustné, aby aktivity $t1$ a $t4$ byly buď přímo určitou aktivitou, nebo mohou být tzv. routovací aktivitou (koresponduje s *route activity* z jazyku XPD, viz 1. typ) pro účely modelování paralelismu. Aktivita $t1$ je označována jako **AND-split**^{273} a aktivita $t2$ jako **AND-join**^{274}. Toto větvení lze vidět na obrázku 39. Zároveň je pro takové větvení představena nová notace, a to pro lepší přehlednost sítě, viz obrázek 39b.

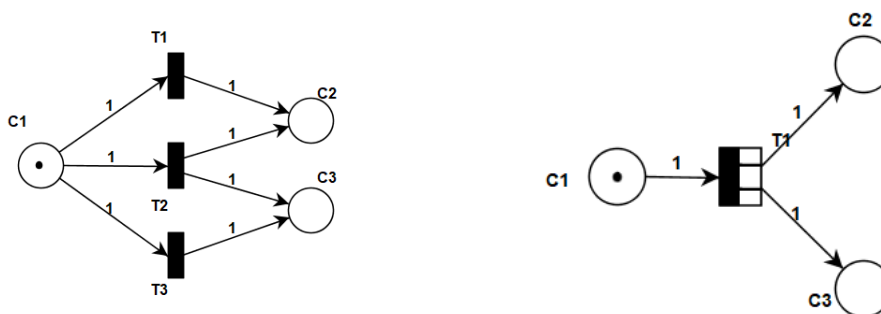
Podmíněnost ke svému modelování nutně vyžaduje akceptovat CPN rozšíření. Jak je vidět z příkladů na obrázku 40, tak bez CPN by nebylo možné vybrat pouze jednu cestu. Na zmíněných obrázcích jsou ilustrovány dvě možnosti, jak lze modelovat tzv. **OR-split**^{275} a **OR-join**^{276}. Obrázek 40a ukazuje modelování skrze vstupní podmínky přechodů $t11$ a $t12$. Naopak obrázek 40b využívá k modelování výstupní podmínku přechodu $t11$. V prvním případě je opět možné mít dodatečné routovací aktivity (jsou vyobrazeny), avšak modelovat lze i přímo do aktivit (tj. v 40a z $c1$ do $t1$ a $t2$). Ve Workflow sítích byla představena jednotná notace pro OR-split a OR-join, ta je vyobrazena na obrázku 41.

Kombinací paralelismu a podmíněnosti (tedy kombinací AND a OR), vznikl tzv. AND/OR konstrukt, který je vyobrazen na obrázku 42, spolu s jeho notací ve Workflow sítích.

Iterace je posledním konstruktem, přičemž název je vypovídající. V business procesu je někdy potřeba opakovat nějaký krok, dokud není splněna určitá pod-



Obrázek 41: Workflow sítě – OR v notaci



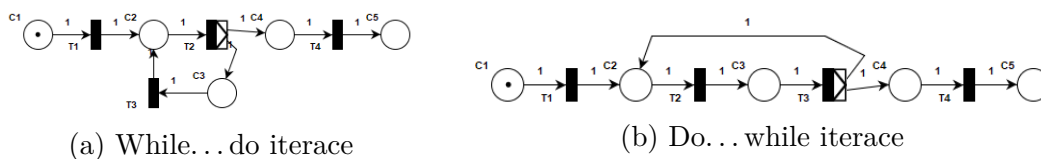
(a) AND/OR konstrukt v notaci PN (b) AND/OR konstrukt v notaci WF-sítí

Obrázek 42: Workflow sítě – AND/OR

mínka. Obecně můžeme modelovat dva různé cykly, a to takový, jež se před první iterací dotáže na podmínku (while...do cyklus), nebo takový který je vždy alespoň jednou vykonán a později se táže na podmínku (do...while cyklus). Cykly jsou společně vyobrazeny na obrázku 43. Jak lze vidět, tyto konstrukty využívají OR-splitu z *podmíněnosti*.

10.2.2 Provádění aktivit

Dosud bylo spouštění aktivity^{277} (přechodu) okamžité (tj. jakmile byl přítomny token na vstupních místech a zároveň byly splněny vstupní podmínky). Nicméně pro účely modelování business procesů^{278} bylo potřeba rozlišit a tedy rozšířit jednotlivé aktivity, resp. podmínky pro jejich spuštění. To může být okamžité, odložené, závislé na nějakém zdroji či události. V zásadě byly přijaty tři možnosti, jak může být aktivita spuštěna:



(a) While...do iterace

(b) Do...while iterace

Obrázek 43: Workflow sítě – Iterace



Obrázek 44: Workflow síť – Aktivity

1. **Zdrojem** – vyobrazován jako obdélník s tlustou šipkou směrem k aktivitě (např. zaměstnancem, který ji začne vykonávat)
2. **Externí událostí** – vyobrazován jako obdélník s obálkou (např. příjem dokumentu přes EDI rozhraní)
3. **Časovým signálem** – vyobrazován jako obdélník s hodinami (např. ve 12:00)

Jejich zmíněné odpovídající notace jsou vyobrazeny na obrázku 44.

Zároveň jsou samozřejmě zachovány aktivity, které se vykonávají okamžitě (tedy nemají žádný dodatečný symbol). Na obrázku jsou vyobrazeny tzv. AND-split aktivity, nicméně zmíněné tři možnosti je možné aplikovat na aktivity jakékoliv.

Je důležité zmínit, jak tato představená rozšíření aktivit (tedy přechodů) zapadají do Petriho sítí. Každý představený symbol může totiž korespondovat s dalším vstupním místem pro danou aktivitu, přičemž přítomnost tokenu v takovém místě značí přítomnost zdroje, či přijetí externí události. Tudíž aktivita nebude spuštěna (přechod nebude proveditelný), dokud nebude token na všech vstupních místech. Samotný časový signál můžeme modelovat díky zmíněnému rozšíření Petriho sítí o čas.

10.3 Shrnutí a závěr

Petriho síť^{279} stojí na velmi silném matematickém základu. Díky tomu je lze velmi dobře formalizovat. Ovšem pro modelování business procesů^{280} jsou základní Petriho síť nedostatečné. Jak bylo patřičně diskutováno, tak Petriho síť lze obohatit přípuštěním určitých rozšíření, která obohatí jejich syntaxi i sémantiku. Jedním z nich jsou tzv. CPN^{281}, zmíněné v zadání této diplomové práce. CPN se pro modelování business procesů bohužel ukázaly jako nedostatečné, avšak představují jedno z velmi důležitých rozšíření základních Petriho sítí, které je pro modelování business procesů potřeba.

Na základě představených HLPN^{282} byl identifikován a rozebrán matematický konstrukt, který byl přímo zamýšlen pro modelování business procesů. Jedná se o Workflow síť^{283}, představené Wil Van der Alstem. Díky zmíněným rozšířením, dodatečné definici, a upravené (zjednodušené) syntaxi, se Workflow síť zdály být vhodným adeptem k modelování business procesů. Avšak sám jejich tvůrce později přiznal, že jako takové jsou stále nedostatečné.

V [68] a [69] bylo identifikováno, že modelování určitých situací (založených na workflow patternech⁴⁰) je ve Workflow sítích velmi složité, či nemožné. Jedná se například o situace, kdy je potřeba odebrat tokeny z určitých míst v síti. Odebírat tokeny z míst mohou pouze přechody, neexistuje žádný univerzální způsob, jak při nějaké chybě zrušit vykonávání celého procesu, odebráním tokenů z míst, kde se mohou nacházet.

Na základě těchto informací lze soudit, že Petriho sítě (resp. Workflow sítě), poskytují velmi dobrý základ pro modelování business procesů, avšak k plnohodnotnému modelování a praktickému použití nedostačují.

Na závěr je vhodné zmínit, že na základě nedostatků Petriho sítí, vznikl konstrukt nový, a to **Yet Another Workflow Language**. Jedním z autorů je zmiňovaný Wil Van der Alst, tvůrce Workflow sítí. **YAWL** má pevný základ v Petriho sítích, avšak přidává určité konstrukty, jež adresují problémy modelování business procesů v původních Petriho sítích (resp. Workflow sítích). Bohužel, i přes zachování si určité formálnosti, nelze **YAWL** zpětně formalizovat do Petriho sítí, jako tomu bylo s Workflow sítěmi.

11 Business Process Model and Notation

Než si představíme další standard, jež se používá pro modelování business procesů, je potřeba lehce obohatit doposavad používanou terminologii, kterou jsme zavedli zejména v kapitole 8.1. V průběhu let byl pojem *workflow* používán jinak, než bylo zamýšleno a mnoho lidí si pod ním nedokáže představit jeho exaktní definici. Tím začal pojem *workflow*^[284] splývat s pojmem *business proces*^[285], avšak jejich definice jsou rozdílné (viz definice 2 a 9). Ona samotná definice *workflow* je poněkud svazující a nutně nemusí vyjadřovat to, co chceme. A jelikož fundamentální zaměření je na samotné procesy v podniku (businessu), začalo se více hovořit o samotném *business procesu*, jako takovém a disciplínách, které se k němu vztahují. Jednou takovou je **Business Process Management (BPM)**:

Definice 26 (**Business Process Management**)

Jedná se o řadu služeb a nástrojů pro modelování, řízení, analýzu, optimalizaci a automatizaci business procesů. Zahrnuje i podporu pro lidské účastníky a interakci s aplikacemi mimo **BPM** [70] [71].

Technologie, která implementuje **BPM**, se nazývá **Business Process Management System (BPMS)**. Zde již vzniká určitá paralela, a to **WfM** (*workflow*) vs. **BPM** a **WfMS** vs. **BPMS**. **BPMS** má již z podstaty definice mnohem větší záběr a můžeme jej chápat jako určitého „generačního nástupce“ a rozšíření dříve diskutovaných **WfMS**.

Standard **BPMN** byl původně vyvinut organizací **Business Process Management Initiative (BPMI)** v roce 2004, a to ve verzi 1.0. Účelem bylo poskyt-

⁴⁰viz <http://www.workflowpatterns.com/patterns/control/>

nout ucelenou reprezentaci business procesů, která bude snadno pochopitelná pro všechny, jež do business procesu vystupují (tj. od analytiků, přes vývojáře až po samotné uživatele) [72]. Fundamentem bylo tedy vytvořit pomyslný „most“ mezi návrhem business procesů a jejich implementacemi. Zároveň šlo o poskytnutí grafické notace k již existujícím implementacím (jež byly na bázi XML, zejména pro Business Process Execution Language for Web Services (BPEL4WS) (později WSBPEL).

Brzy po vydání se organizace BPMI sloučila s organizací Object Management Group (OMG), která za sebou již měla řadu úspěšných standardů, zejména velmi známé UML. Po tomto sloučení prošel standard v průběhu let několika revizemi. Jednalo se o samotnou revizi 1.0 a nové verze 1.1, 1.2 a 2.0. Nejnovější verze standardu BPMN nese označení 2.0.2 a je již plně pod záštitou OMG. Největší mezigenerační skok byl zaznamenán mezi verzemi 1.2 a 2.0, jelikož grafická notace byla značně rozšířena. V této diplomové práci budeme tedy uvádět názvy a notaci z verze nejnovější, tedy 2.0.2. Sluší se říct, že BPMN se v průběhu let stal velmi oblíbeným a používaným nástrojem pro modelování business procesů. Jedním z důvodů může být detailní a rozsáhlý popis ve specifikaci každé verze, např. viz [71] pro verzi nejnovější. Na základě informací v této specifikaci byla tato kapitola vytvořena.

Než si přiblížíme možnosti a způsoby modelování v BPMN, je potřeba zmínit, jak je v něm realizováno „provádění“, resp. simulace. Ta je obdobná, jako v Petriho sítích. Využívá se zde tzv. **tokenu**, který simuluje tok procesu. Avšak na rozdíl od Workflow sítí^[286], zde v průběhu simulace může existovat tokenů více v tom smyslu, dojde-li jeden z tokenů v určité větvi do koncové události, tak ostatní mohou stále dále probíhat. Tato skutečnost byla ve Workflow sítích nežádoucí.

11.1 Modely

Standard představuje tři základní modely, jež některé mají určité pod modely, mezi kterými se rozlišuje. Lze je chápat jako rozdílné diagramy, protože využívají rozdílné prvky standardu. Zároveň mají každý trochu jiný účel. Kategorizace je následující:

1. Procesy (orchestrace)
 - Soukromé procesy
 - Veřejné procesy
2. Spolupráce
 - Konverzace
3. Choreografie

Níže si jednotlivé modely lehce rozebereme a nastíníme jejich využití.

Procesy

Modely procesů, někdy označovány jako orchestrace, zachycují samotné business procesy jako takové, které zároveň rozdělují na základě toho, kde se proces odehrává. Dělí je tedy na soukromé a veřejné.

Soukromé procesy se odehrávají přímo uvnitř daného podniku. Obecně se jedná o *workflow/business procesy*. Mohou být **spustitelné**, či **nespustitelné**. Modelují se zejména z *aktivit, událostí, bran* a bez *bazénů*.

Veřejné procesy slouží k modelování interakcí mezi soukromými procesy a jinými procesy či účastníky (modelovány jako prázdné *bazény*). Modelovány jsou pouze *aktivity*, které se používají pro komunikaci s jiným procesem či účastníkem (zbylé jsou vynechány). Dá se tedy chápat jako nástroj pro vyobrazení *zpráv*, jež soukromé procesy komunikují z vnějškem.

Spolupráce

Modely spolupráce slouží k modelování interakcí mezi dvěma a více podnikovými subjekty (každý subjekt je modelován jako *bazén* – může být i prázdný). Například se může jednat o dva různé *veřejné procesy*, přičemž platí vlastnost, že jejich korespondující *soukromé procesy*, budou mít mnohem více *aktivit*.

Konverzace slouží k dodatečnému popisu modelů spolupráce. K modelování využívá *bazény* (pootočené do vertikální polohy) jako účastníky. Ti jsou mezi sebou propojeni šestiúhelníky, které značí, že mezi nimi probíhá určitá výměna informací (u šestiúhelníků je napsáno jaké informace jsou vyměňovány).

Choreografie

Modely choreografie definují chování mezi jednotlivými účastníky. Mají určitou podobnost se soukromými modely procesů (využívají stejné prvky). Rozdíl je ovšem v tom, co představují *aktivity*. V tomto případě představují určité interakce, ve kterých probíhá určitá výměna zpráv. Představíme-li si model *spolupráce*, pak model *choreografie* popisuje, co se děje mezi jednotlivými *účastníky* (ti jsou modelovány *bazény*).

11.2 Prvky a jejich notace

Standard má přibližně desítku základních prvků, na kterých staví. Patří mezi ně například *aktivita*, jak ji známe z definice 11. Nicméně prvků je více, přičemž každý má svou striktní notaci a zároveň určitá rozšíření. Tyto prvky spolu se svými rozšířeními tvoří tzv. **Business Process Diagram (BPD)**. Tyto prvky jsou kategorizovány do pěti následujících skupin:

- Tokové objekty

- Spojovací objekty
- Plavecké dráhy
- Datové objekty
- Artefakty

Níže si jednotlivé kategorie představíme, ukážeme jejich dělení, notaci atd.

11.2.1 Tokové objekty

Tokové objekty jsou hlavními prvky, které definují jednotlivé business procesy v [BPMN](#). Jedná se o tři následující prvky:

- Události
- Aktivity
- Brány

Tyto jednotlivé prvky si níže postupně představíme, uvedeme jejich rozšíření, vlastnosti a příslušné notace.

Události reprezentují nějakou událost, která se v *business procesu* stane. Je to základní řídicí prvek, jež může mít určitou příčinu (označováno jako **trigger** - spouštěč) či důsledek (označováno jako **result**). Existují právě tři typy událostí:

- Počáteční – označuje počátek business procesu
- Mezikrok – vyskytuje se v business procesu mezi počáteční a koncovou událostí
- Koncová – označuje konec business procesu

Tyto tři typy událostí jsou vyobrazeny na obrázku [45a](#).

Velmi zajímavou vlastností, jež [BPMN](#) pro události přidává, je možnost tyto jednotlivé typy událostí dále rozlišovat, a to následovně:

- Zpráva – událost závislá na zprávě (přímo adresována)
- Časovač – časová událost (např. každé dvě hodiny; každé pondělí v 8:00, ...)
- Error – slouží pro ošetření chyb
- Eskalace – obdoba erroru, která ovšem nenaruší chod samotného business procesu
- Zrušení – slouží ke zrušení transakcí podprocesů

- Kompenzační – slouží pro zrušení již vykonaných *aktivit*
- Podmíněná – slouží pro modelování podmínky, tj. dokud není podmínka splněna, business proces nepokračuje
- Odkaz – jedná se spíše o syntaktickou událost, která slouží pro jednodušší modelování (díky ní lze navázat na modelovaný proces na další straně apod.)
- Signál – obdoba *zprávy*, avšak není přímo adresován
- Ukončení – slouží k okamžitému ukončení veškerého vykonávání v procesu (vymaže všechny existující tokeny)
- Vícenásobná – slouží pro modelování vícero různých událostí skrze jednu událost, avšak ke spuštění stačí pouze jedna z událostí (např. místo zvlášť *zprávy* a *signálu*, lze použít pouze *vícenásobnou* událost)
- Vícenásobná paralelní – obdoba *vícenásobné události*, avšak ke spuštění je třeba všech událostí

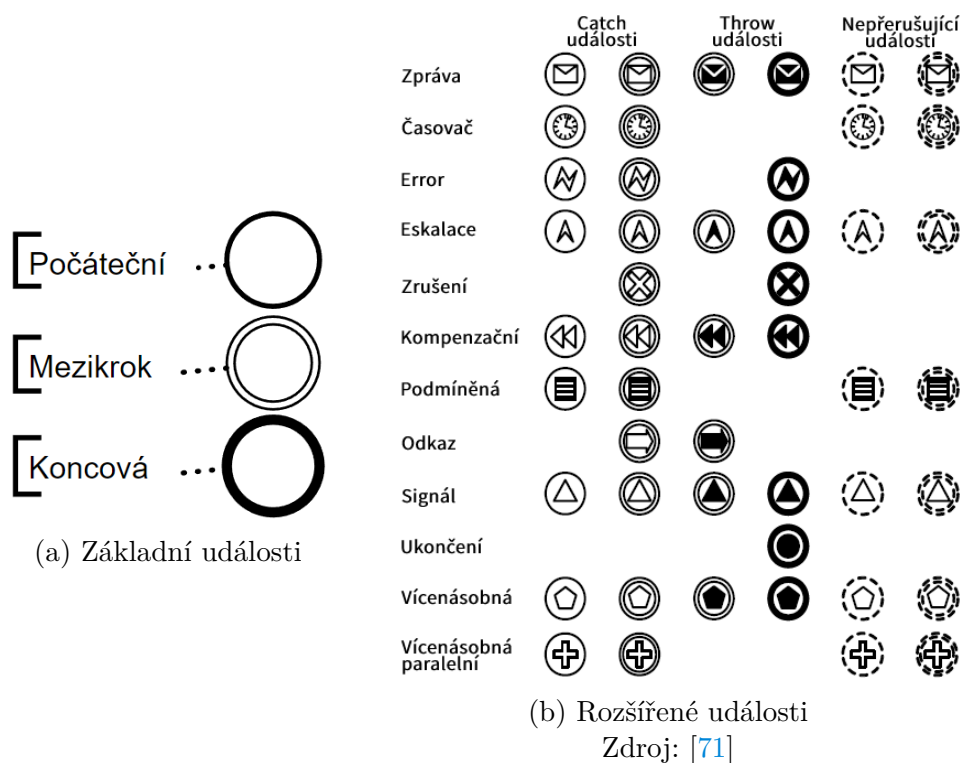
Tedy tyto jednotlivé rozšířené události mohou být buď *počáteční*, *mezikrokové* či *koncové*. Nicméně rozlišuje se u nich i fakt, jestli mají nějaký *trigger* (modelovány jako nevybarvené symboly – **catch události**), či produkují-li nějaký *result* (modelovány jako plně vybarvené symboly – **throw události**). *Catch události* povolí pokračování v procesu, až je daná *událost* splněna (např. přijetí zprávy). Naopak *throw události* vykonají danou činnost (např. odeslání zprávy) a proces pokračuje.

Obecně vzato, *počáteční události* mohou mít pouze nějaký *trigger*, kdežto *aktivity koncové* vždy produkují pouze *result*. *Události* typu *mezikrok* mohou mít jak *trigger*, tak produkovat *result*. Všechny jmenované možnosti jsou vždy modelovány s plnou kružnicí okolo symbolu. Avšak chceme-li říct, že se jedná o tzv. **nepřerušující události**, modelujeme je kružnicí přerušovanou. Jednotlivé typy *událostí*, jejich symboly a patřičná notace na základě jejich typu a vlastností jsou vyobrazeny na obrázku 45b.

Na závěr k *událostem* je třeba ještě zmínit, že existují tzv. **události na hraně aktivit**. Tyto *události* jsou aktivní pouze, je-li aktivní *aktivita*, na jejíž hraně leží. Nastane-li situace modelována onou *událostí*, pak může být aktivní *aktivita* buď přerušena (*přerušující událost*) nebo může nadále pokračovat a dojde k určitému větvení (*nepřerušující událost*). Příkladem může být modelování *kompenzace*, skrze *kompenzační událost* a asociovanou *kompenzační aktivitu*, viz obrázek 47b níže.

Aktivity představují danou část práce, jež má být vykonána. Základní dělení je následující:

- **Task** – představuje atomickou akci



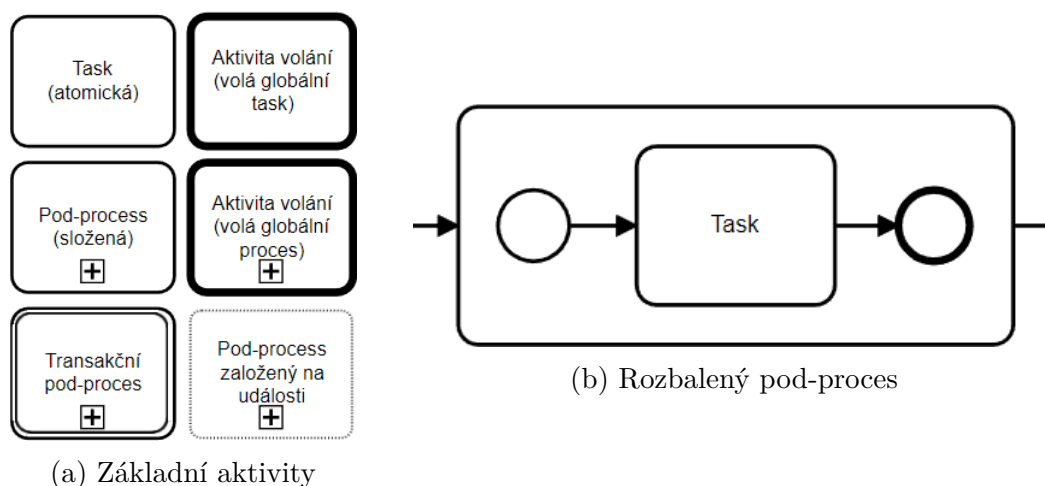
Obrázek 45: BPMN – události

- **Pod-proces** – představuje složenou aktivitu; slouží k modelování procesu uvnitř jiného procesu (zjednodušení diagramu)

Oba tyto druhy aktivit mohou být **aktivitou volání**. Je-li *aktivitou volání* nějaký *task*, pak se jedná o volání tzv. **globálního tasku**. Je-li *aktivitou volání* nějaký *pod-proces*, pak se jedná o volání tzv. **globálního procesu**. Tyto globální konstrukty je tedy možné definovat pouze jednou (tím se liší od klasického pod-procesu, který je potřeba definovat nově v každém procesu), a poté je volat z mnoha různých procesů bez nutnosti nové definice (tudíž se může jednat o často používané standardizované postupy).

Samotné *pod-procesy* mohou být modelovány buď jako sbalené (tj. vnitřní prvky nejsou specifikovány) či rozbalené (tj. vnitřní prvky jsou modelovány). Navíc se nadále dělí na dva speciální podtypy:

- Transakční pod-proces – jeho chování je řízeno skrze transakční protokol; před dokončením celého pod-procesu, musí všichni účastníci úspěšně dokončit svou část; může být zrušen skrze *událost zrušení*
- Pod-proces založený na události – představuje vnořený proces, z ani do kterého nevedou žádné *spojovací objekty*, jehož *počáteční událost* musí být rozšířená (tj. být např. *zprávou, errorem, ...*). Tudíž je proveden pouze, nastane-li daná definovaná událost

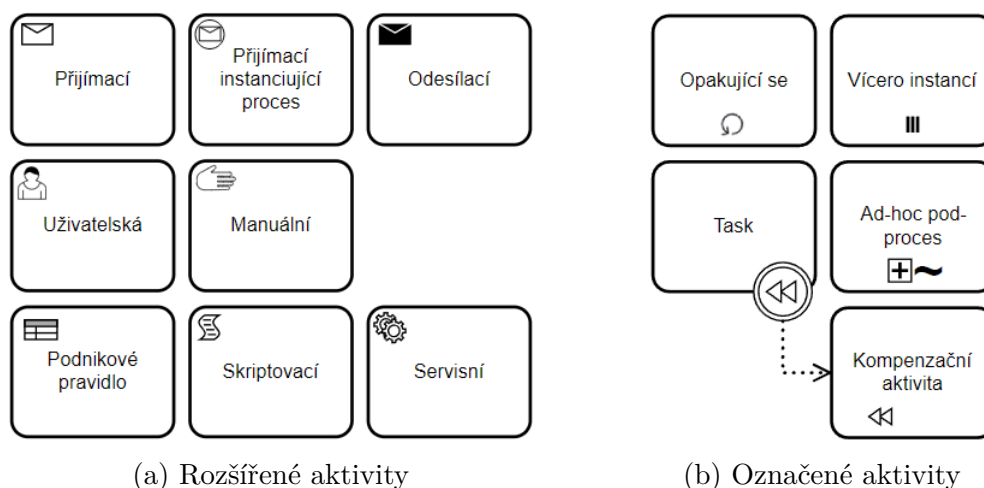


Obrázek 46: BPMN – základní aktivity

Jednotlivé sbalené verze pod-procesů se modelují jako tasky, avšak se znaménkem plus. Tato zmíněná notace *tasků* a *pod-procesů*, je společně vidět na obrázku 46a. Všechny sbalené verze pod-procesů, lze ovšem modelovat i jako jejich rozbalené ekvivalenty, tato notace je znázorněna na obrázku 46b (analogicky se musí dodržet styl a tloušťka čáry, jako u sbalených ekvivalentů).

Task aktivity mohou mít dodatečná rozšíření, obdobně jako události. Jednotlivé typy jsou následující:

- Přijímací – čeká na určitou *zprávu* od externího *účastníka*; po přijetí je dokončena
- Přijímací instanciující proces – speciální případ přijímací; slouží k vytvoření instance procesu (nahrazuje *počáteční událost* [typu *zpráva*])
- Odesílací – odesílá *zprávu* nějakému externímu *účastníkovi*; po odeslání se považuje za dokončenou
- Uživatelská – vykonávána pracovníkem za pomoci **SW** aplikace; po dokončení *engine*⁴¹ vyžaduje potvrzení
- Manuální – vykonávána pracovníkem, avšak bez jakékoliv pomoci **SW** aplikace; **BPMS** nesleduje její počátek či konec, jedná se pouze o určitý „koncept“
- Podnikové pravidlo – slouží k zapojení enginu podnikových pravidel, kterému může proces skrze tuto aktivitu předat nějaký vstup a obdržet od něj nějaký výstup (více rozvedeno v kapitole 11.3)
- Skriptovací – vykonávána čistě *engine*⁴¹, přičemž autor definuje skript v jazyce, který umí *engine*⁴¹ spustit
- Servisní – vykonávána nějakým **SW**, tudíž je automatizovaná



Obrázek 47: BPMN – rozšířené a označené aktivity

Jednotlivá rozšíření jsou znázorněna na obrázku 47a.

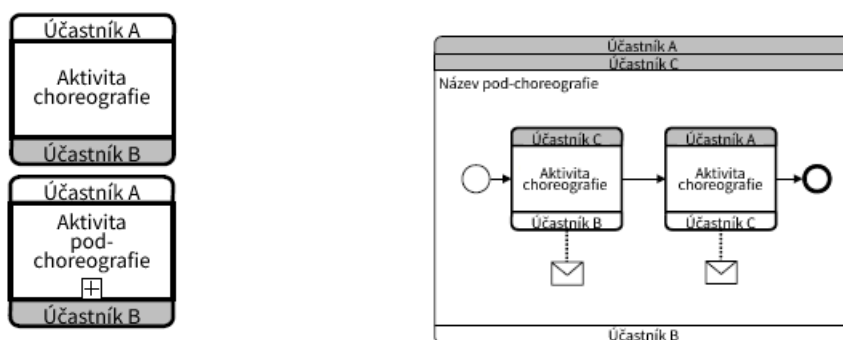
Nicméně *aktivity* mohou mít i dodatečné **označení** (marking), jež určuje jejich speciální funkcionalitu, existují právě čtyři následující:

- Opakující se – označuje aktivitu, jež se může opakovat
- Víceru instancí – označuje aktivitu, která může mít vícero instancí
- Kompenzační – přidružená aktivita k aktivitě s kompenzací; je vykonána, je-li aktivována přidružená *událost kompenzace*
- Ad-hoc – speciální pod-proces, ve kterém není předem určená sekvence vykonávání

První tři označení jsou aplikovatelná jak na aktivity typu *task*, tak i na *pod-proces*. Avšak poslední, ad-hoc označení, je aplikovatelné pouze na aktivity typu *pod-proces*. Notaci označení lze vidět na obrázku 47b.

Avšak při modelování *choreografie* se využívají ještě speciální typy *aktivit*, označovány jako **aktivity choreografie**. V jejich notaci na obrázku 48a lze vidět, že v horní a dolní části *aktivity choreografie* jsou jednotliví *účastníci procesu*. Jejich barevné označení symbolizuje iniciátora komunikace (tj. nevybarvený = iniciátor, šedý = ne iniciátor). Spolu s tím jsou spojeny jednotlivé zprávy, které indikují, co je vlastně komunikováno. To se zobrazuje skrze jednotlivé obálky, jejichž barevné označení je analogické a koreluje s obarvením *účastníků procesu*. Samotná *aktivita choreografie* má uvnitř svůj název značící, co se má vlastně vykonat. Stejně, jak tomu bylo s klasickými *aktivitami*, i zde je možné mít *aktivitu pod-choreografie*. Notace těchto aktivit lze vidět na obrázku 48.

⁴¹jedná se v podst. o *workflow engine* rozvedený v kapitole 9.3.1



(a) Aktivita choreografie a pod-choreografie (sbalená) (b) Aktivita pod-choreografie – rozbalená

Obrázek 48: BPMN – aktivity choreografie

Brány jsou základní větvičí⁴² prvky BPMN. Skrze ně lze simulovat např. paralelismus, podmíněnost či cykly, jak je známe z kapitoly o Workflow sítích (viz 10.2.1). Rozlišujeme mezi těmito branami:

- Paralelní – běžné modelování paralelismu; analogie AND^{{287}{288}} (viz 10.2.1)
- Inkluzní – na základě splnění podmínek (vyhodnocuje všechny) může být y z x větví spuštěno (doporučuje se modelovat tak, aby alespoň jedna vždy proběhla – např. defaultní větev); analogie OR^{{289}{290}} (viz 10.2.1)
- Vylučující – na základě podmínek pouze jedna z x větví bude spuštěna (obsahuje i defaultní větev); analogie XOR, může sloužit k modelování cyklů
- Založená na události – pouze větev, jejíž *událost* (případně *aktivita s událostí* – *příjímací*, *odesílací*) nastane jako první, bude spuštěna; může být i instanciující (nahrazuje počáteční *událost*)
- Založená na paralelních událostech – analogie s předchozí (využívá *události*), avšak po vykonání jedné větve na základě *události*, povoluje vykonávání i dalších větví (opět po vykonání *události*, jež podmiňuje onen chod); pouze instanciující
- Komplexní – umožňuje definovat „vlastní“ typ brány, a to ve vztahu k tokenům (tj. může produkovat x tokenů na základě přijetí y tokenů)

Jednotlivé *brány* a jejich notaci lze společně vidět na obrázku 49.

⁴²BPMN nutně nevyžaduje, aby se tok rozdělený určitou bránou, opět bránou stejného typu později spojil. Nicméně existuje nepsané pravidlo, které právě toto doporučuje, jelikož modelovaný proces je díky tomu přehlednější.



Obrázek 49: BPMN – brány

11.2.2 Spojovací objekty

Pro potřebu vyjádření směru toku (*tokenů*) v procesu, a tedy propojení jednotlivých *tokových objektů*, slouží objekty spojovací. Rovněž slouží k vyjádření toků dat a přiřazení prvků. Modelovány jsou jako šipky s plnou, přerušovanou, či tečkovanou čarou. Z pohledu diagramů se dělí do tří kategorií představených níže.

Sekvenční toky vyjadřují směr a pořadí vykonávání jednotlivých *tokových objektů*. Slouží i pro modelování toků podmíněných⁴³, přičemž obecně se dělí na následující:

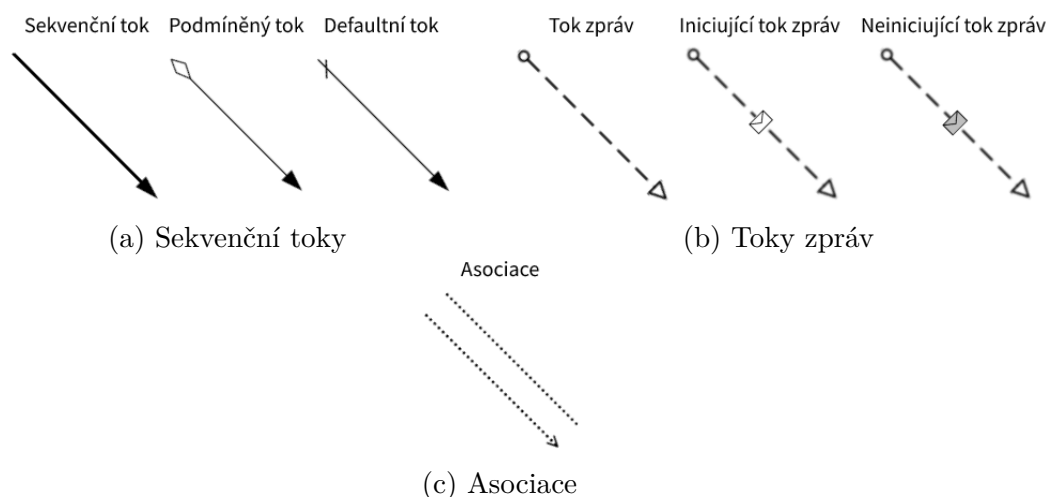
- Sekvenční tok – nejběžnější tok, určuje sekvenci vykonávání
- Podmíněný tok – modelování podmíněného toku *tokenu*; míří-li z aktivity, začíná malým kosočtvercem; míří-li z brány, kosočtverec přítomný není
- Defaultní tok – dá se chápat jako „else“ větev, tudíž nejsou-li splněny žádné podmíněné toky (např. *vylučující* či *inkluzní brány*), pak se aktivuje větev s tímto tokem

Notaci sekvenčních toků lze společně vidět na obrázku 50a.

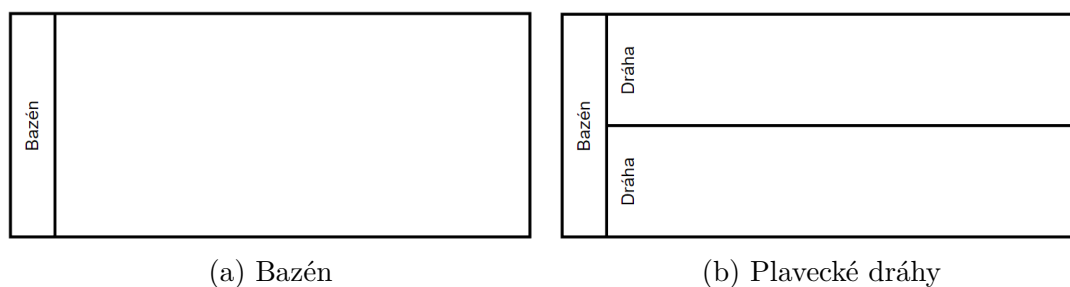
Toky zpráv slouží k modelování toků zpráv mezi dvěma *účastníky*⁴⁴ procesu (jsou reprezentovány *bazény*). Modelují se nevyplněnou šipkou s čárkovanou čarou, přičemž tento tok může být volitelně dekorován o obálku, která specifikuje, jaká zpráva je komunikována. V tomto případě rozlišujeme mezi iniciující zprávou (nevybarvená obálka) a neiniciující zprávou (šedá obálka). Notace je znázorněna na obrázku 50b.

Asociace přidružuje prvek k prvku jinému (např. *textovou anotaci* k prvku *aktivity*, *kompensační aktivitu* k *aktivitě*, ...). Modelují se tečkovanou čarou s/bez šipky, viz obrázek 50c.

⁴³BPMN povoluje modelování podmíněných toků přímo z aktivit (tj. větvení přímo z aktivity). Nicméně existuje nepsané pravidlo, které říká, že by se pro takové modelování měly



Obrázek 50: BPMN – toky



Obrázek 51: BPMN – plavecké dráhy

11.2.3 Plavecké dráhy

Bazény a **plavecké dráhy** slouží k modelování rozdílných účastníků procesů (např. v modelu *spolupráce* představují dva bazény dva rozdílné aktéry). Obecně *bazén* může být prázdný (bez aktivit), přičemž jako takový je poté chápán jako tzv. „black box“. V takovém případě není nutné modelovat, co daný *účastník* dělá. Jednotlivé *bazény* se dále mohou rozdělit na *plavecké dráhy*. V tomto případě se jedná spíše o samotnou organizaci *aktivit* a rozpínají se po celé délce *bazénu*. Jejich notaci lze vidět na obrázku 51, přičemž mohou být modelovány i vertikálně.

11.2.4 Datové objekty

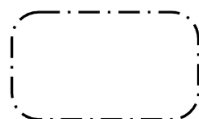
Datové objekty reprezentují data, jež jsou součástí business procesu, resp. skrze něj proudí (mohou to být různé dokumenty, jako např. faktura). Dělí se následovně:

použít brány, a to opět z důvodu přehlednosti modelovaného procesu.

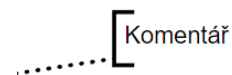
⁴⁴Není zde myšlen *účastník workflow*, jak byl zmíněn v závěru kapitoly 8.1. Jedná se o pouhou shodu pojmů napříč standardy.



Obrázek 52: BPMN – datové objekty



(a) Skupina



(b) Textová anotace

Obrázek 53: BPMN – artefakty

- Datový objekt – obecný objekt reprezentující určitá data (např. fakturu)
- Datový objekt kolekce – reprezentuje určitou kolekci datových objektů (např. seznam produktů z faktury)
- Vstupní data – reprezentují data, která jsou vstupem pro daný business proces; proces může vyžadovat vícero vstupních dat
- Výstupní data – reprezentují data, která jsou výstupem z daného business procesu; proces může produkovat vícero výstupních dat
- Úložiště dat – poskytuje *aktivitám* možnost ukládat či aktualizovat informace, které jsou mimo daný business proces (např. upravit nějaký záznam v databázi)

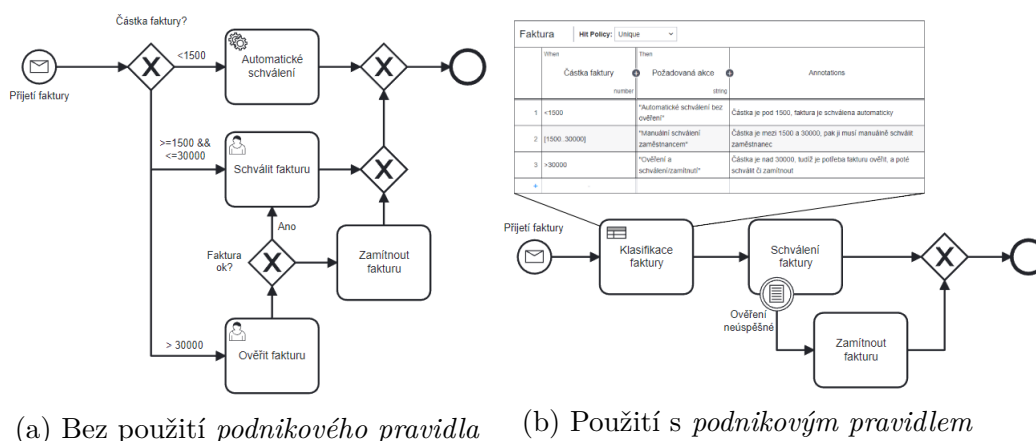
Notace jednotlivých datových objektů lze společně vidět na obrázku 52

11.2.5 Artefakty

Pakliže doposud představené prvky BPMN nedostačují pro danou modelovací situaci, jsou dostupné další dva dodatečné, označované jako artefakty. Ty nemají žádnou sémantickou funkci uvnitř daného modelovaného business procesu, nýbrž slouží pro zvýšení přehlednost či objasnění kontextu. Existují právě dva, a to:

- Skupina – slouží k určitému seskupení aktivit, jež patří k sobě; pouze pro účely dokumentace či analýzy
- Textová anotace – dodatečná textová informace k jakémukoliv prvku BPMN; k prvku je připojena *asociací* (spojovací objekt)

Dva výše zmíněné artefakty a jejich notaci lze vidět na obrázku 53. Tato množina artefaktů není uzavřená. Implementátoři BPMN a obecně návrháři business procesů, mohou přidávat dodatečné artefakty.



(a) Bez použití *podnikového pravidla* (b) Použití s *podnikovým pravidlem*

Obrázek 54: BPMN – příklad s DMN

Poznámka: rozhodovací tabulka z obrázku 54b nebývá vyobrazena. Zde slouží pro kontext a lepší pochopení.

11.3 Decision Model and Notation

Ačkoliv **Decision Model and Notation (DMN)** nepatří pod **BPMN**, je záhodno jej zmínit. Jedná se o další z mnoha standardů, jež vydala **OMG**. **DMN** bylo zamýšleno k použití právě spolu s **BPMN**, i když na sobě nejsou přímo závislé [73]. Tento standard je mimo oblast zájmu této diplomové práce, tudíž bude uveden jen velmi krátce, a to kvůli závislosti *aktivity* typu *podnikové pravidlo* v kapitole 11.2.1.

Jedná se o standard sloužící k popisu a modelování určitých rozhodovacích procesů \implies tvorba podnikových pravidel. Představuje určitou strukturalizaci a notaci při pohledu na rozhodování v podniku. Rozhoduje se skrze tabulky, které pro určitý vstup, definují určitý výstup. Budeme-li se dívat na vztah **DMN** a **BPMN**, pak v **BPMN** lze onu *aktivitu* typu *podnikové pravidlo* nahradit několika *branami*, jelikož v **DMN** jde „jen“ o určité rozhodování. Avšak takové větvení není znovu použitelné a může přispět k nepřehlednosti diagramu. Vzájemné použití **DMN** s **BPMN**, lze vidět na obrázku 54, přičemž 54a ukazuje proces bez použití *podnikového pravidla* a 54b s použitím *podnikového pravidla*, včetně rozhodovací tabulky (normálně nebývá vyobrazena, zde ji ukazujeme pro kontext).

11.4 XPD L a BPMN

Jak bylo zmíněno, **BPMN** od své první verze poskytovalo i mapování na *executivní jazyk*^[291] workflow (tedy **BP EL4WS**, později **WSBP EL**). Nicméně neposkytovalo žádné standardizované mapování do *jazyků definičních*^[292]. V tomto případě se velmi ujal jazyk **XPD L** od organizace **WfMC**. V průběhu let spolu organizace **BPMI**, později **OMG**, a **WfMC** spolupracovaly pro účely mapování **BPMN** právě do **XPD L**. Díky této spolupráci s každou novou verzí **BPMN** vycházela rovněž nová verze **XPD L**, která adresovala novinky přidané v **BPMN**. Tím se z jazyku **XPD L** stal de-facto standardní definiční jazyk pro **BPMN**.

Tabulka 7: Elementy [XPDL 2.2](#)

Element	Český ekvivalent	Účel
Package	Balíček	slouží pro zapouzdření elementů níže
Application	Aplikace	koresponduje se představeným pojmem <i>vyvolávané aplikace</i> (viz pojem v kapitole 8.1)
Swimlanes	Plavecké dráhy	korespondují s plaveckými drahami BPMN (viz kapitola 11.2.3)
Process definition	Definice procesu	koresponduje s <i>definicí procesu</i> (viz definice 10), resp. pod procesu
Activity	Aktivita	koresponduje s tokovými objekty BPMN (viz kapitola 11.2.1)
Transition	Přechod	koresponduje se sekvenčními toky (viz kapitola 11.2.2)
Partner links	Odkazy na partnery	představují komunikaci mezi dvěma účastníky
Messaging	Zprávy	koresponduje s <i>toky zpráv</i> (viz kapitola 11.2.2)
Association	Asociace	koresponduje s <i>asociací</i> (viz kapitola 11.2.2)
Participant	Účastník	koresponduje s představeným pojmem <i>účastník workflow</i> – nepředstavuje účastníka v kontextu BPMN (<i>tedy bazén</i>)
DataField/Property	Datové pole	koresponduje s pojmem <i>data vztahující se k workflow</i> (viz kapitola 9.2)
DataTypes	Typ dat	typ <i>dat vztahujících se k workflow</i>

Ve vydání [BPMN](#) ve verzi 2.0, byla mimo mnoho nových grafických prvků a revize modelů, součástí i nová, vlastní podoba definičního jazyka. Na základě této verze [BPMN](#) byl vydán jazyk [XPDL](#) ve verzi 2.2. Nicméně obsahoval podporu pouze pro modely procesů a modely spolupráce (tzn., že konverzace a choreografie nebyly podporovány) [\[74\]](#). Oproti verzi 1.0 byl jazyk více přizpůsoben pro [BPMN](#) prvky, tudíž se dočkal rozšíření o několik elementů. Ty můžeme vidět v tabulce [7](#).

11.5 Exekutivní [BPMN](#)

Ačkoliv [BPMN](#) patří pod *definičním nástroje* pro modelování *business procesů*, tak se lze setkat s pojmem **exekutivní [BPMN](#)**, tj. proces vymodelovaný v [BPMN](#), který lze spustit v *systému řízení workflow*. Nicméně takové označení může být lehce zavádějící, jelikož samotné [BPMN](#) spustitelné není. Avšak jeho specifikace uvádí jasné mapování na jazyk^{[\[293\]](#)}, jež exekutivní je, a to [Web Servi-](#)

ces Business Process Execution Language. Díky tomuto mapování lze jednotlivé *business procesy* vymodelované v BPMN převést do jazyku WSBPEL, přičemž taková reprezentace je následně spustitelná.

Omezením ovšem je, že toto mapování není dostupné pro všechny modely diskutované v kapitole 11.1, resp. pro všechny grafické prvky standardu. Spustitelné (tedy existuje pro ně mapování do WSBPEL) jsou pouze modely *procesů* a částečně modely *spolupráce*. Ostatní modely slouží pro dodatečný popis chování procesů, či pro náhled z jiného úhlu. Budeme-li se bavit o jednotlivých představených prvcích standardu, pak mezi spustitelné patří:

- veškeré *události*,
- *aktivity* vyjma *aktivit choreografie*,
- *brány*,
- *sekvenční toky* a *základní toky zpráv*,
- *plavecké dráhy*,
- a *datové objekty*.

Artefakty jsou z hlediska vykonávání procesu nepodstatné (popisují jen dodatečné informace, které pomáhají s přehledností), tudíž pro ně mapování neexistuje.

11.6 Shrnutí

Na rozdíl od dříve představených Petriho sítí a konceptů a standardů od WfMC, se BPMN zabývá *business procesy*, nežli jen *workflow*. Tím pádem pokrývá i mnohem větší oblast, jež je v dnešním rychle rozvíjejícím se světě nezanedbatelná. Tomu odpovídají i představené modely, potažmo notace. Na rozdíl od WRM, kde se hovořilo o modelování jednotlivých *workflow*, BPMN dnes rozlišuje mezi vícero diagramy, přičemž ty slouží k lepšímu pochopení celého *business procesu* a jeho spojitostí.

Pro tyto účely BPMN rozlišuje mezi modely procesů, spolupráce a choreografie. Nejenže pro tyto modely představuje až vyčerpávající množinu jednotlivých grafických prvků s pevně danou notací, jež se společně nazývají *Business Process Diagram*, nýbrž poskytuje i mapování do *exekutivního jazyka*^{294} WSBPEL. Bylo diskutováno, že zejména pro modely choreografie toto mapování ovšem neexistuje. Nicméně pro BPMN existuje i mapování do *jazyku definičního*^{295}, a to do XPDL. Nejen díky těmto faktům je standard BPMN velmi vhodným adeptem k modelování *business procesů*.

12 Doporučení standardů

Na základě informací a představení jednotlivých standardů napříč částí [IV](#), bude v této kapitole doporučen jeden či více standardů, které jsou vhodné pro implementaci workflow systémů. Jmenovitě půjde o standard [WRM](#) představený v kapitole [9](#), Petriho sítě (včetně [CPN](#) a Workflow sítí) představené v kapitole [10](#) a standard [BPMN](#) představený v kapitole [11](#).

12.1 Standard pro obecnou implementaci

Při implementaci *workflow systému*, resp. jakéhokoliv systému, je velmi důležitý jeho samotný návrh. Je potřeba si uvědomit, že tyto systémy v podniku nemusí být jediné a často budou muset komunikovat s jinými heterogenními *systémy workflow*. Je tedy možné, že by odběratel takového [SW](#) musel dodatečně řešit vzájemnou komunikaci heterogenních systémů, což by ho mohlo od použití odradit. Z toho důvodu je důležité, aby takové systémy na daný problém nahlížely podobně a hlavně, aby mezi sebou uměly komunikovat. Takové požadavky se dají splnit, bude-li implementace v souladu s nějakým standardem, jež takový problém adresuje. V případě *workflow systémů* se jedná o představený standard [Workflow Reference Model](#)^{296} od [WfMC](#). Nejenže jasně vymezuje pojmy, nýbrž díky němu lze jednoznačně určit a strukturalizovat funkcionalitu v *systému workflow*, spolu se způsoby komunikace s jinými heterogenními (ale i homogenními) systémy. Z toho důvodu můžeme standard [Workflow Reference Model](#) označit za nezbytný při návrhu (a následné implementaci) workflow systémů.

12.2 Standard pro definici a modelování

Jednou ze základních ideí standardu [WRM](#) zmíněného výše, je určité oddělení definiční a exekutivní částí workflow. Definiční část souvisí se samotným modelováním *business procesu* (viz kapitola [8.2](#)), přičemž následně vytvořený model je reprezentován (a tudíž i komunikován) v určitém formátu, a to například v jazyku [XPDL](#) (viz kapitola [8.3](#)). Právě toto oddělení je adresováno 1. rozhraním standardu [WRM](#) (viz kapitola [9.3](#)), které leží mezi *systémy řízení workflow*^{297} a *nástroji pro definici business procesů* (terminologie je konzistentní s terminologií v kapitole [9.2](#)).

Business proces^{298} musí být řádně definován a vymodelován, aby jej bylo možné realizovat v *systému řízení workflow* (exekutivní část workflow systému). Ačkoliv by bylo možné *business proces* vymodelovat v textové podobě, a to např. pomocí zmíněného jazyku [XPDL](#), tak takový postup by byl značně uživatelsky nepřívětivý. Nehledě na fakt, že může být potřeba daný *business proces* konzultovat s klientem, tudíž by nakonec byla potřeba jeho přehledné grafické reprezentace. Ačkoliv [XPDL](#) bylo představeno společně s přidruženou grafickou notací, tak ta se neujala a bylo od ní opuštěno. Z toho důvodu byla potřeba jiných standardů a metod modelování *business procesů*.

V této diplomové práci byly pro tento účel představeny Petriho sítě^{299} (včetně *Coloured Petri Nets*^{300} a Workflow sítí^{301}) a standard *BPMN*^{302}. Zmíněná notace přidružená jazyku *XPDL* nebude brána v potaz z důvodu uvedených v kapitole 9.4. V práci bylo diskutováno, že Petriho sítě jsou pro plnohodnotné modelování *business procesů* nedostatečné, přičemž ani jejich rozšíření o barvu (tj. konstrukt *CPN*) nesplňuje všechny požadavky (jako například přehlednost výsledného modelu procesu). Na základě těchto poznatků byl diskutován konstrukt Workflow sítí, který se bohužel rovněž ukázal být nedostatečným. Tato fakta jsou prezentována napříč kapitolou 10, přičemž jsou shrnuty v podkapitole 10.3.

Naproti tomu zmíněný standard *BPMN* obsahuje velkou množinu grafických prvků, díky nimž lze velmi detailně a přehledně vymodelovat daný *business proces* (nejen výrobního charakteru). Nicméně se standard neomezuje pouze na modelování samotného průběhu procesů, nýbrž poskytuje i nadstandardní modelovací prvky (modely *konverzace* a *choreografie*), skrze které lze na *business proces* nahlédnout z jiného úhlu pohledu. Díky těmto faktům umožňuje *BPMN* modelovat ty nejzákladnější konstrukty (větvení, jako *AND-split*^{303}, *OR-split*^{304} apod.), ale i konstrukty mnohem složitější (např. zrušení celé instance procesu – *BPMN* toho dosahuje skrze diskutované *transakce*). Složitější konstrukty nebyly v této diplomové práci blíže diskutovány, nicméně jejich přehled a podporu v *BPMN*, lze shlédnout v [75].

Další velkou výhodou je již připravené mapování prvků *BPMN* (nejedná se o všechny prvky standardu, viz kapitola 11.5) do *jazyku exekutivního*^{305}, a to do *WSBPEL*. Co víc, pro *BPMN* rovněž existuje mapování (viz kapitola 11.4) do *jazyku definičního*^{306}, kterým je *XPDL*. Tato samotná mapování velmi usnadní implementaci *definičního nástroje*, potažmo celého workflow systému. Z důvodů uvedených v této kapitole lze *Business Process Model and Notation* doporučit jako standard pro definici a modelování uvnitř systému workflow.

13 Usazení do kontextu Industry 4.0 a norem ANSI/ISA-88 a ANSI/ISA-95

V této kapitole bude spolu s usazením do širšího kontextu, rovněž prezentována analýza v *UML*. Veškeré modely a diagramy, které jsou součástí této analýzy, lze nalézt v příloze A. V názvech jednotlivých tříd prvků *BPMN* se používá slovo „objekt“, které nesouvisí s „objektem“ v *OOP* paradigmatu! Toto slovo bylo použito pro účely zachování konzistence pojmenovávání prvků *BPMN* standardu.

Nejdříve se podívejme, co vlastně budeme usazovat a kam. V našem případě se budeme snažit usadit následující bloky:

- workflow systém (zde můžeme říct, že *BPMS*),
- standard *BPMN*,
- a standard *WRM*.

Ty budeme chtít usadit do následujících kontextů:

- Industry 4.0, a to skrze:
 - RAMI 4.0
 - Model Industry 4.0 komponenty
- Norem [ANSI/ISA-88](#) a [ANSI/ISA-95](#)

Pro lepší přehlednost a navigaci mezi jednotlivými kapitoly, si zde uvedeme určitou osnovu napojení korespondující s podkapitoly níže. Ona osnova je následující:

1. Ukážeme propojení kontextů Industry 4.0 a norem [S88](#) a [S95](#), přičemž pro účely tohoto propojení bude prezentován i korespondující diagram. Díky tomuto propojení bude následně jednoduší usazovat naše jednotlivé „bloky“. Diskutováno v podkapitole [13.1](#).
2. Usadíme [BPMS](#) do společného kontextu Industry 4.0 a norem [S88](#) a [S95](#). Rovněž zde více rozvedeme *workflow model* normy [S95](#), čímž si připravíme půdu pro další krok. Diskutováno v podkapitole [13.2](#).
3. Ukážeme, jak standard [BPMN](#) pasuje do Industry 4.0 (skrze [BPMS](#)), a jak může korespondovat a splňovat požadavky normy [S95](#). Toto bude prezentováno skrze vymodelované doménové modely (diagramy) standardu [BPMN](#), které následně napojíme na diagram *workflow modelu*, známého z [S95](#). Diskutováno v podkapitole [13.3](#).
4. Jako poslední bude do předchozích bodů usazen i standard [WRM](#). Diskutováno v podkapitole [13.4](#).

13.1 Propojení Industry 4.0 a norem [S88](#) a [S95](#)

Nejdříve se krátce podíváme na určitou standardizaci Industry 4.0^{307}, kterou se zabývá [RAMI 4.0](#)^{308}. Tento fakt byl již adresován v kapitole [7](#). Jak bylo diskutováno, tento model zachycuje jednotlivé charakteristiky Industry 4.0, jak byly jmenovány v kapitole [4.4.2](#). Rovněž je součástí jeden z nejdůležitějších konceptů Industry 4.0, a to [CPS](#), v [RAMI 4.0](#) reprezentován Industry 4.0 komponentou^{309}. Ta požaduje určitou konektivitu, přičemž ta je zajišťována skrze *administrative shell*^{310}, přičemž samotná konektivita je realizována skrze koncept [IoT](#)^{311} a technologie, které jsme zmiňovali v kapitole [3](#). Pro připomenutí, Industry 4.0 komponentami v našem případě nebudou pouze výrobní stroje (tzn. nějaká fyzická aktiva), nýbrž i [SW](#), přesněji [BPMS](#) (potažmo [WfMS](#)⁴⁵). Pro tyto elementy [RAMI 4.0](#) definuje, kde a jakým způsobem probíhá komunikace, její vrstvení apod.

⁴⁵Od této chvíle budeme již uvádět pouze [BPMS](#), jelikož jak bylo diskutováno dříve, [WfMS](#) se dá považovat za jeho podmnožinu

Následuje napojení norem S88^{312} a S95^{313} na RAMI 4.0^{314}. Toho lze docílit skrze jednu z os RAMI 4.0. Jak bylo diskutováno v kapitole 7.1, jedná se o pravou horizontální osu, která bere zmíněné normy v potaz a adresuje určité jejich části. Přesněji řečeno její jednotlivé složky korespondují s hierarchií zařízení^{315} normy S95, potažmo s fyzickým modelem^{316} normy S88 (tuto skutečnost lze vidět na dříve představeném obrázku 30).

Tudíž samotné RAMI 4.0 do určité míry propojuje zmíněné normy s konceptem Industry 4.0. Nicméně my můžeme jít dál a na základě těchto informací vytvořit doménový model, který bude tyto vztahy vyjadřovat i ve vztahu k Industry 4.0 komponentě. Onen diagram je vidět na obrázku 55. V horní části diagramu lze vidět vybrané části RAMI 4.0 hierarchie (tedy právě horizontální osy, jak byly představeny v kapitole 7.1). Jedná se o STANICI, RIDICIJEDNOTKU a PROVOZNIŽARIZENI, jelikož o těchto lze uvažovat jako o potenciálních Industry 4.0 komponentách (potažmo CPS^{317}). Tyto vztahy jsou vyjádřeny kompozicí, jelikož při zániku zařízení (spadajícího pod jednu z položek hierarchie), zaniká i samotná Industry 4.0 komponenta. Ta je realizována skrze samotnou VEC, kterou reprezentuje a ADMINISTRATIONSHELL^{318}, který poskytuje dané věci komunikační schopnosti, schopnost práce s daty apod. Samozřejmostí je zachování možnosti zanoření Industry 4.0 komponent (tj. Industry 4.0 komponenta se může skládat z více různých Industry 4.0 komponent).

13.2 BPMS v Industry 4.0 a v souladu s normou S95

Nyní je třeba uvážit, jak do tohoto kontextu usadit samotný Business Process Management System. Na základě propojení realizovaného v předchozí kapitole můžeme říct, že BPMS lze realizovat jako Industry 4.0 komponentu^{319}, čímž zaručíme jeho určitou kompatibilitu v rámci RAMI 4.0^{320}, a tedy i konceptu Industry 4.0^{321}. Tím jsme samozřejmě vedeni i k normě S95^{322}, potažmo S88^{323}, které jsou v RAMI 4.0 zohledněné. Je třeba si totiž uvědomit, v jaké úrovni (viz kapitola 6.1) bude BPMS pracovat. Lze vidět, že se bude jednat o úroveň 3, a tudíž se jedná o součást Manufacturing Operations Management. Bude-li BPMS pracovat na této úrovni, musí nutně dodržovat způsoby a formáty komunikace normy S95. Například podíváme-li se zpět do kapitoly 6.4 na obrázek 20, tak na základě vykonávání určitého vymodelovaného *business procesu*, může BPMS vydat příkaz k výrobě (požadavek operace), na základě kterého bude naplánována výroba.

Nicméně v našem kontextu se podíváme zejména na výměnu jednotlivých definic *business procesů*, popř. i jeho „exekutivních verzí“. Ta se totiž rovněž bude dít vně 3. úrovně. Zde je potřeba jít menší oklikou, a to zpět k normě S95, přesněji k jejímu *modelu workflow*^{324}. Navážeme totiž na kapitolu 6.5.1, přesněji na zmíněný *model workflow* vyobrazený na obrázku 24. Norma totiž počítá s výměnou workflow informací. Avšak dosud nebyl zmíněn fakt, že tento model počítá jak s přenosem definic workflow, tak i s jejich instancemi. Tudíž vyměňované informace nejsou pouze definičního, nýbrž i exekutivního charakteru. Formálněji

řečeno, třída *specifikace workflow* může nabývat dvou různých typů, a to [46]:

- Vzor
- Instance

Použijeme-li analogii z OOP, pak *vzorem* rozumíme třídu a *instancí* rozumíme objekt vytvořený z dané třídy. Jasně tedy vyplývá, že *specifikaci workflow* typu *vzor* nelze spustit, avšak typ *instance* spustit lze.

Pro připomenutí je vhodné zmínit, že ačkoliv S95 hovoří o *workflow* a my diskutujeme *business proces*, tak lze tuto záměnu pojmů přehlédnout. Jak bylo totiž diskutováno v úvodu kapitoly 11, tak můžeme tyto pojmy chápat identicky (i když se jedná o chybnou původní interpretaci pojmu *workflow*). Na základě této nové informace ohledně *workflow modelu* lze tedy jasně říct, že lze vyměňovat jak definice business procesů, tak i jejich exekutivní verze.

13.3 BPMN v Industry 4.0 a v souladu s normou S95

Díky předchozí kapitole lze nyní hovořit o usazení standardu BPMN^{325} v normě S95^{326}. Víme, že BPMN se týká *definičního nástroje*, a tedy je součástí BPMS. Skrze diskutované usazení BPMS do kontextu Industry 4.0, je tranzitivně usazeno i BPMN. Diskutovali jsme rovněž, že BPMS běží ve 3. úrovni (tj. je součástí MOM) definované normou S95. Je tedy potřeba ukázat, jestli a jak může standard BPMN korespondovat s *modelem workflow*^{327} zmíněné normy, čímž by splnil její požadavky.

BPMN jsme představili v kapitole 11, nicméně ve zkratce můžeme říct, že se jedná o určitou množinu grafických prvků s pevně danou sémantikou, skrze kterou se modelují a definují *business procesy*. Na základě informací představených ve zmíněné kapitole si v následujících podkapitolách rozebereme prvky standardu BPMN do jednotlivých doménových modelů.

Pro začátek je vhodné ukázat vztahy jednotlivých balíčků, které budeme představovat. Ty lze vidět na obrázku 56. Jednotlivé balíčky jsou následující:

- Základní balíček diagramů DIAGRAM, rozebíraný v podkapitole 13.3.1.
- Balíček *tokových objektů* FLOW, rozebíraný v podkapitole 13.3.2 a následné balíčky *událostí*, *aktivit* a *bran*, pojmenované jako EVENTS, ACTIVITIES a GATES.
- Balíček *spojovacích objektů* CONNECTING, rozebíraný v podkapitole 13.3.3.
- Balíček *plaveckých drah* SWIMLANES, rozebíraný v podkapitole 13.3.4.
- Balíček *datových objektů* DATA, rozebíraný v podkapitole 13.3.5.
- Balíček *artefaktů* ARTIFACTS, rozebíraný v podkapitole 13.3.6.

13.3.1 Diagramy

Základem je samotný diagram, do kterého se *business proces* modeluje skrze jednotlivé grafické prvky BPMN. Vymodelovaný doménový model lze vidět na obrázku 57. Jak lze vidět, adresovány jsou všechny tři druhy diagramů (reprezentující jednotlivé modely BPMN z kapitoly 11.1). Nicméně jelikož modely *choreografie* nejsou z pohledu exekutivního BPMN zajímavé, není na ně, a jim přidružené grafické prvky, brán zřetel ani ve vymodelovaném diagramu (viz kapitola 11.5). Jednotlivé třídy DIAGRAMPROCESU a DIAGRAMSPOLUPRACE tedy obsahují své jednotlivé grafické prvky, reprezentované třídou OBJEKTDIAGRAMU, čímž se myslí právě jeden z grafických prvků standardu BPMN.

13.3.2 Tokové objekty

Nejpodstatnější grafické prvky jsou zajisté tokové objekty. V doménovém diagramu na obrázku 58 si lze všimnout, že obecně TOKOVYOBJEKT dědí z OBJEKTUDIAGRAMU, a tudíž přebírá i jeho atributy. Zajímavější jsou ovšem jednotlivé doménové diagramy jednotlivých prvků, jež spadají pod TOKOVYOBJEKT.

Události jsou první z nich, vyobrazené na obrázku 59. U nich je potřeba rozlišovat, zda se jedná o *počáteční*, *mezikrok*, či *koncovou událost*. Toho je docíleno skrze základní dělení na *Catch* a *Throw události*, a následného dělení na ony tři typy. *Throw události* jsou poměrně přímočaré, zajímavější jsou *události Catch*. U těch má smysl rozlišovat, zda jsou přerušující, či nikoliv. U CATCHUDALOSTMEZIKROK je vymodelován i speciální případ této události, a to tzv. *hraniční události*. Toho je docíleno skrze odkaz na případnou aktivitu a odvozený atribut JEHRANICNI skrze OCL pravidlo: `derive: self.odkazNaAktivitu implies true`. Samotné typy rozšířených událostí jsou modelovány skrze třídu ROZSIRENYTYPUDALOSTI, spojenou s OBJEKTUDALOST skrze kompozici.

Aktivity jsou další a jejich doménový model lze vidět na obrázku 60. Samotná abstraktní třída OBJEKTAKTIVITA obsahuje atribut JEKOMPENZACNI, který stanovuje, zda-li se jedná o *kompensační aktivitu*. Atribut OZNACENI přidává *aktivitám* možnost být *opakující se* či mít *vícero instancí*, a to skrze výčtový typ. Základní dělení je na *tasky* a *pod-procesy*. Samotné *task aktivity* mohou mít mnoho různých typů, což je ilustrováno skrze dědění z třídy TASK. U té je rovněž třeba rozlišovat, zda-li se jedná o *klasickou aktivitu*, *globální aktivitu*, či *aktivitu volání*. Toho je docíleno skrze příslušné atributy a jim přidružené invarianty, jelikož *aktivita volání* nemůže rovněž být *aktivitou globální*.

- Invariant pro atribut JEAKTIVITAVOLANI: `self.jeAktivitaVolani = true implies self.jeGlobalni = false`
- Invariant pro atribut JEGLOBALNI: `self.jeGlobalni = true implies self.jeAktivitaVolani = false`

Třída TASK rovněž implementuje rozhraní AKTIVITAVOLANI, jež předepisuje atribut pro *volanou (globální) aktivitu*. Zajímavější je ovšem třída POD-PROCESAKTIVITA, přičemž její atribut JESBALENA vyjadřuje, zda-li je tento *pod-proces* sbalený, či rozbalený. Jedná se o třídu abstraktní, jelikož je nutné rozlišovat mezi samotným *pod-procesem* a *pod-procesem*, který je *aktivitou volání*, jelikož druhý zmíněný nemůže být *ad-hoc* ani *globální*. Samotná třída POD-PROCESAKTIVITAVOLANI opět implementuje již zmíněné rozhraní. Avšak největším rozdílem je fakt, že POD-PROCES má kompoziční vztah k samotným *tokovým objektům*, jelikož je-li rozbalený, tak se skládá z dalších grafických prvků. Dále je vyobrazeno jeho samotné rozdělení na zbývající dva typy *pod-procesů*, a to *transakční* a *událostní*.

Brány představují poslední z *tokových objektů* a jejich doménový model lze vidět na obrázku 61. Oproti předchozím není tak komplexní, jelikož vyjadřuje prosté rozdělení jednotlivých typů *bran* do tříd. Za zmínku stojí atribut JEINSTANCIUJICI třídy ZALOZENANAUDALOSTI, jelikož *brány založené na událostech* mohou sloužit k instanciování daného *business procesu*.

13.3.3 Spojovací objekty

Vztahy, resp. propojení mezi jednotlivými *tokovými objekty*, jsou v BPMN diagramech modelovány skrze *spojovací objekty*. Jejich doménový model lze vidět na obrázku 62. Jedná se tedy opět o jeden z objektů diagramu, z jehož třídy také dědí. Pro úplnost jsou modelovány veškeré jednotlivé typy *spojovací objektů*, jak byly představeny. Avšak pouze třídy SEKVENCNI TOK (včetně tříd, jež z ní dědí) a TOKZPRAV (bez tříd, jež z ní dědí) mají smysl z pohledu exekutivního BPMN. Jednotlivé asociace VEDE Z a VEDE DO vyjadřují, z a do kterých grafických prvků mohou jednotlivé toky vést.

13.3.4 Plavecké dráhy

Bazény, resp. *plavecké dráhy* představují účastníky procesu. Na modelu z obrázku 63 lze vidět, že OBJEKTPLAVECKEDRAHY je abstraktní třída. Instanciovat lze tedy pouze BAZÉN a PLAVECKADRAHA, přičemž ta může být pouze uvnitř nějakého *bazénu*, při jehož zániku zaniká i samotná *plavecká dráha*.

13.3.5 Datové objekty

Datové objekty BPMN jsou předposlední modelovanou kategorií. Doménový model lze vidět na obrázku 64, přičemž zachycuje všechny jednotlivé typy *dat*. Nejzajímavější na diagramu je vztah mezi abstraktní třídou OBJEKTDAT a APLIKACNI OBJEKT, jež vyjadřuje určitou referenci, kterou si *objekty dat* nesou na určitý objekt v systému, jako například faktura. Pochopitelně KOLEKCE má vícero odkazů na takové objekty.

13.3.6 Artefakty

Pro úplnost jsou uvedeny a vymodelovány i *artefakty*. Jak bylo patřičně zmíněno, tak pro ně neexistuje mapování do *exekutivního jazyka*^{328}, tudíž nejsou z pohledu exekutivního BPMN zajímavé. Nicméně jejich doménový model lze vidět na obrázku 65 a netřeba jej blíže popisovat.

13.3.7 Napojení na S95

S tímto vědomím můžeme jednotlivé prvky BPMN napojit na *model workflow* z normy S95. Způsob tohoto napojení je ilustrován na obrázku 66. V *modelu workflow* je základní třídou třída SPECIFIKACEWORKFLOW. Ta může přímo korespondovat s naší třídou BPMNDIAGRAM, jež představuje určité „zapouzdření“ celého business procesu, tj. jednotlivých uzlů a propojení. Jednotlivé *tokové objekty* BPMN, reprezentované třídami OBJEKTUDALOST, OBJEKTAKTIVITA a OBJEKTBRANA, lze v kontextu S95 chápat jako jednotlivé uzly, proto ona asociace s třídou UZEL. Stejně je tomu s plaveckými drahami a datovými objekty, reprezentovanými třídami OBJEKTPLAVECKEDRAHY a DATOVYOBJEKT. Zbývá nám už pouze třída SPOJOVACIOBJEKT, jež koresponduje s třídou PROPOJENI. Lze si všimnout, že součástí mapování nejsou *artefakty* BPMN, a to z již zmíněného důvodu jejich neexistujícího mapování na *exekutivní jazyk*.

V závěru můžeme ještě poukázat na nezmíněnou výhodu BPMN, v kontextu Industry 4.0^{329}, a to přítomnost *modelu spolupráce*. Jedním z rysů Industry 4.0 je totiž spolupráce (ať už se jedná o vertikální či horizontální integraci), ať už uvnitř podniku mezi odděleními či s vnějším světem. Nicméně řeč je zejména i o samotné komunikaci a spolupráci různých Industry 4.0 komponent^{330}. Tudíž BPMN poskytuje modelovací aparát právě i pro tyto případy. Nyní jsme tedy propojili BPMN s Industry 4.0 a S95⁴⁶.

13.4 WRM

Zbývá nám už pouze usazení standardu WRM^{331}. Jak bylo patřičně diskutováno, tak WRM standardizuje přístup k BPMS, k jeho návrhu a tedy i implementaci. Lze tedy říci, že nám strukturalizuje samotný BPMS (jeho vnitřní strukturu), což ovšem nijak nekoliduje s konceptem Industry 4.0^{332} nebo normami S95^{333} a S88^{334}. V souladu s předchozími kapitolami nám ovšem nic nebrání v představení doménového modelu i pro samotný standard WRM. Ten lze vidět na obrázku 67. Z podstaty standardu tento model rozsekává funkcionalitu BPMS na menší oblasti, které propojuje skrze jednotlivá komunikační rozhraní. Z pohledu této práce je nejzajímavější třída DEFINICNINASTROJ, spolu s ROZHRANI1 (a potažmo WAPI). Diskutovali jsme totiž zejména standardy pro definici a modelování *business procesů*, načež byl doporučen standard BPMN^{335}. Právě ten je nedílnou součástí onoho *definičního nástroje*, skrze který je vytvořena *definice*

⁴⁶Jelikož lze normu S88 určitým způsobem usadit do normy S95, tak je zmíněné propojení realizováno i s požadavky normy S88.

procesu, která je skrze ROZHRANI1 a WAPI komunikována až k SYSTEMŘIZENIWORKFLOW, potažmo k příslušnému WORKFLOWENGINE. Formát komunikace může být např. v jazyku XPDL, jelikož jeho mapování na BPMN nám tranzitivně umožňuje navázat jej na normu S95 (a tedy splnit její požadavky). Samotný *workflow engine* má následně za úkol, na základě dané *definice workflow*, proces interpretovat, tedy vytvářet jeho *instance*, které mohou být spouštěny. Jelikož pro BPMN existuje mapování do *exekutivního jazyku*^{336} WSBPEL, tak pomocí následující přeměny: $XPDL \rightarrow BPMN \rightarrow WSBPEL$, lze z definice procesu vytvořit jeho exekutivní verzi, která může být instanciována.

Ačkoliv byl WRM představen v kapitole 9.3, tak pro úplnost můžeme krátce zmínit i funkcionalitu ostatních rozhraní v tomto modelu na obrázku 67. Skrze ROZHRANI2 proudí jednotlivé *pracovní položky* (tzn. jednotlivé úkoly) k *účastníkům workflow* (např. zaměstnancům podniku). ROZHRANI3 slouží pro komunikaci a určitou obsluhu *vyvolaných aplikací*. Pro kooperaci s dalšími *systemy řízení workflow*, slouží ROZHRANI4. A konečně ROZHRANI5 slouží pro administrativní obsluhu celého workflow systémů (např. měnění stavu určitých *instancí procesů*) a samozřejmě jsou i nástroje pro monitorování (sledování výkonnosti, problémů apod.).

Závěr

V úvodu práce byly pro uvedení do problematiky představeny (podnikové) informační systémy spolu s jejich stručnou historií. Dále byla patřičně rozebrána iniciativa Industry 4.0 včetně technologií a konceptů, na kterých stojí. Z toho důvodu byla věnována pozornost [Internet of Things](#), [Cyber-physical system](#), či [Digital Twin](#). Na základě těchto informací byl uveden standard [Reference Architecture Model for Industrie 4.0](#), který myšlenky Industry 4.0 formalizoval. Rovněž byl kladen důraz na představení norem [ANSI/ISA-88](#) a [ANSI/ISA-95](#), ve snaze ukázat jejich fundamentální myšlenky. Byly identifikovány společné oblasti obou norem, a tedy i jejich určité propojení.

V poslední části práce byly popsány samotné workflow systémy a zejména zmíněné vybrané standardy pro automatizaci řízení procesu (workflow) výroby. Jednalo se o [Workflow Reference Model](#), Petriho sítě a [Business Process Model and Notation](#). Standard [WRM](#) se ukázal být velmi podstatným, a to zejména při samotném návrhu workflow systému, jelikož standardizuje ucelený pohled na takové systémy. Při představení Petriho sítí byly uvedeny i [Coloured Petri Nets](#), které jsou zmíněné v zadání této práce. Bohužel se ukázalo, že samotné [CPN](#) jsou pro modelování business procesů nedostatečné, tudíž byl uveden konstrukt, který na nich částečně závisí. Jednalo se o Workflow sítě. Rovněž byl představen standard [Business Process Model and Notation](#) s jeho až vyčerpávající grafickou notací.

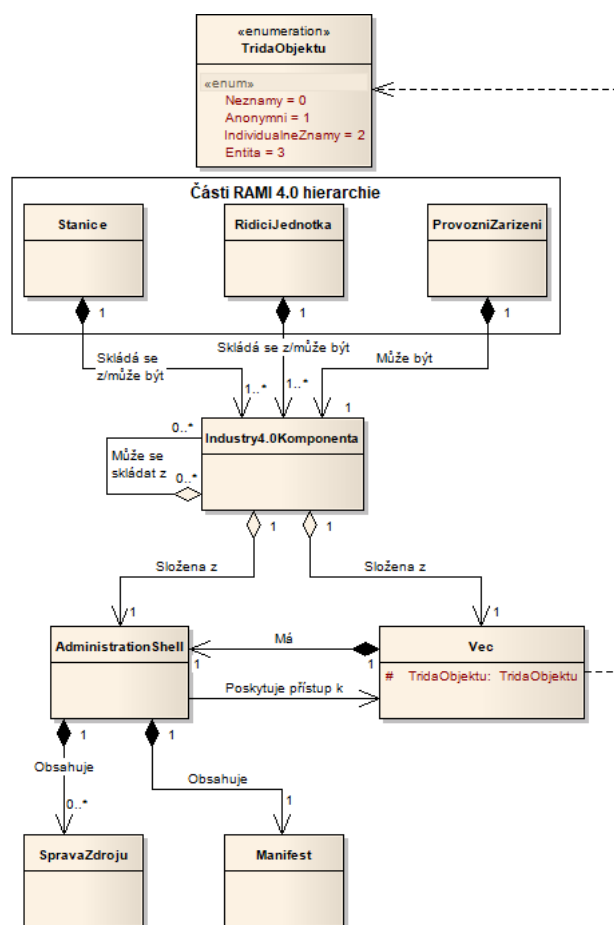
V samotném závěru byly, na základě informací prezentovaných v této práci, doporučeny standardy [WRM](#) spolu s [BPMN](#). Rovněž byly workflow systémy spolu s těmito standardy usazeny do kontextu Industry 4.0, přičemž bylo ukázáno, jak zapadají do požadavků norem [ANSI/ISA-88](#) a [ANSI/ISA-95](#). Součástí je i analýza systému propojení workflow systémů a těchto standardů se zmíněnými normami a iniciativou Industry 4.0, a to v [UML](#).

Conclusions

In the initial chapters of this thesis, enterprise information systems together with their brief history are introduced. Next, the Industry 4.0 initiative, including the technologies and concepts, which enable it are examined. For this reason, [Internet of Things](#), [Cyber-physical system](#) and [Digital Twin](#) are discussed as well. Based on this information a [Reference Architecture Model for Industrie 4.0](#) is introduced, which formalizes ideas of Industry 4.0. Also, the main parts of standards [ANSI/ISA-88](#) and [ANSI/ISA-95](#) are summarized and the interconnection between these two standards is identified.

Workflow systems and the above-mentioned automation and manufacturing control standards are discussed in the latter part of this thesis, namely [Workflow Reference Model](#), Petri nets and [Business Process Model and Notation](#). It was identified, that the [WRM](#) is quite essential because it standardizes the whole concept of workflow systems, which makes its designing much more robust. While introducing Petri nets [Coloured Petri Nets](#), mentioned in the assignment of this thesis, are discussed as well. However, it is deemed, that [CPN](#) are insufficient for business process modelling. Thus, a new construct partially based on [CPN](#) named Workflow nets is introduced. [BPMN](#) is also described, including its exhaustive graphical notation.

Based on the information presented, [WRM](#) and [BPMN](#) standards are recommended. Workflow systems together with both standards are put together with Industry 4.0 as well as with [ANSI/ISA-88](#) and [ANSI/ISA-95](#). The analysis of such interconnection between workflow systems and different standards is created in [UML](#) and is included as well.

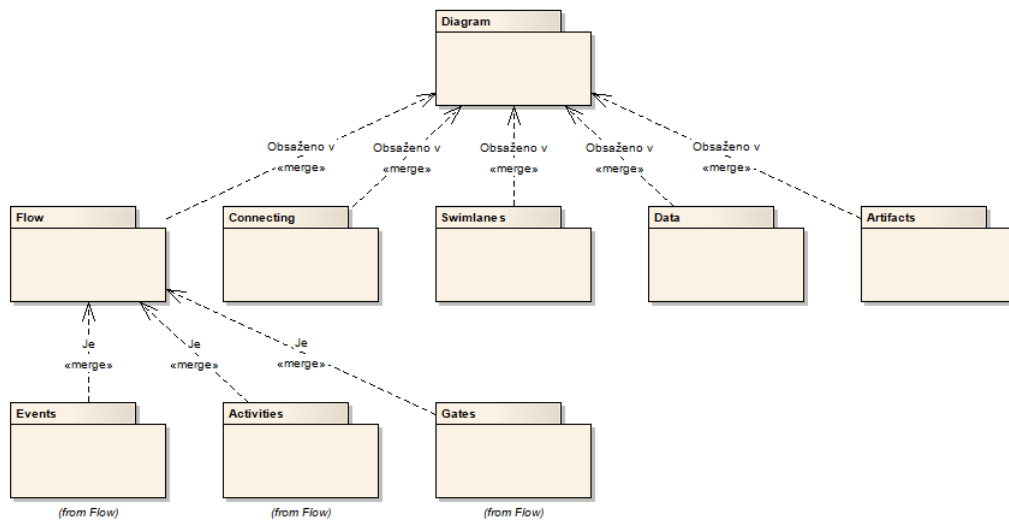


Obrázek 55: Doménový model – propojení Industry 4.0 komponenty, RAMI 4.0 a S95

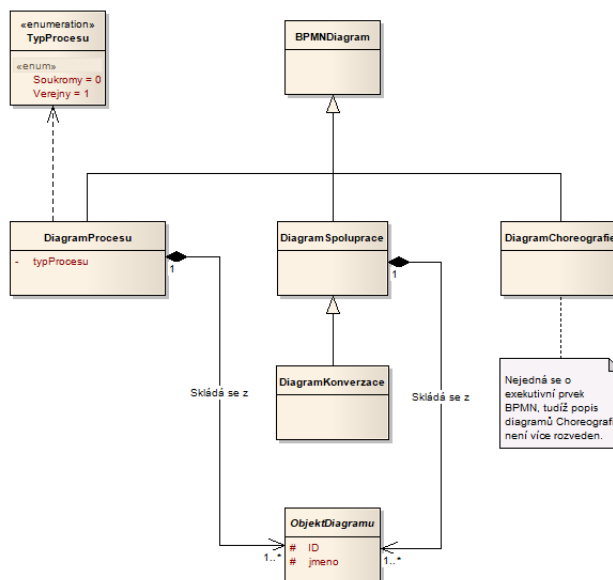
A Analýza systému v UML

V této příloze jsou jednotlivé doménové diagramy v jazyce UML, pro potřeby této diplomové práce. Jednotlivé diagramy jsou vysvětleny, usazeny do kontextu a je na ně odkazováno napříč kapitolou 13.

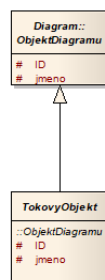
Je důležité zmínit, že v názvech jednotlivých tříd prvků BPMN se používá slovo „objekt“, které nesouvisí s „objektem“ z paradigmatu OOP! Pojmenování bylo zachováno kvůli jednotlivým třídám prvků BPMN, jak byly vyjmenovány v kapitole 11.2.



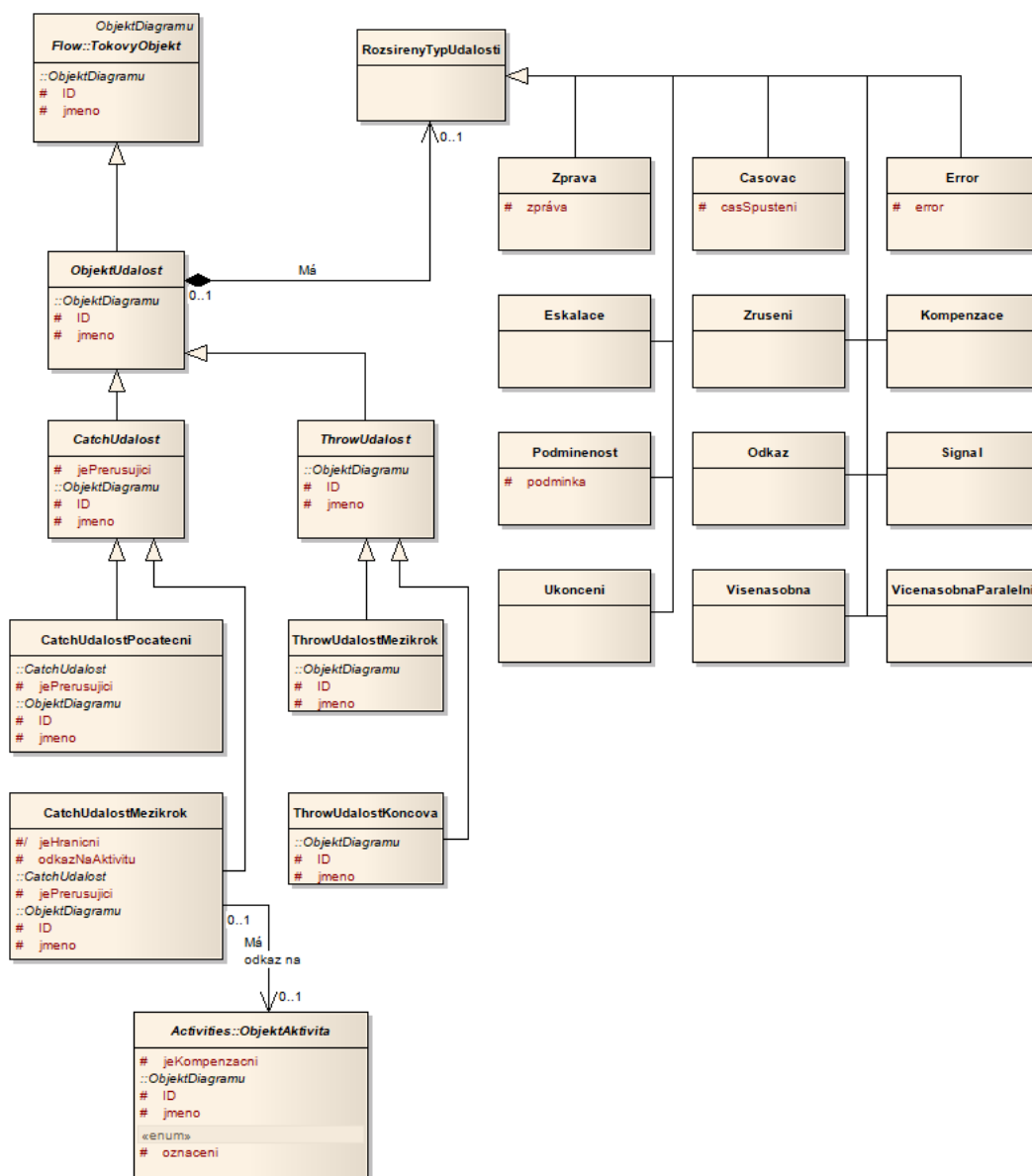
Obrázek 56: Vztahy mezi jednotlivými balíčky



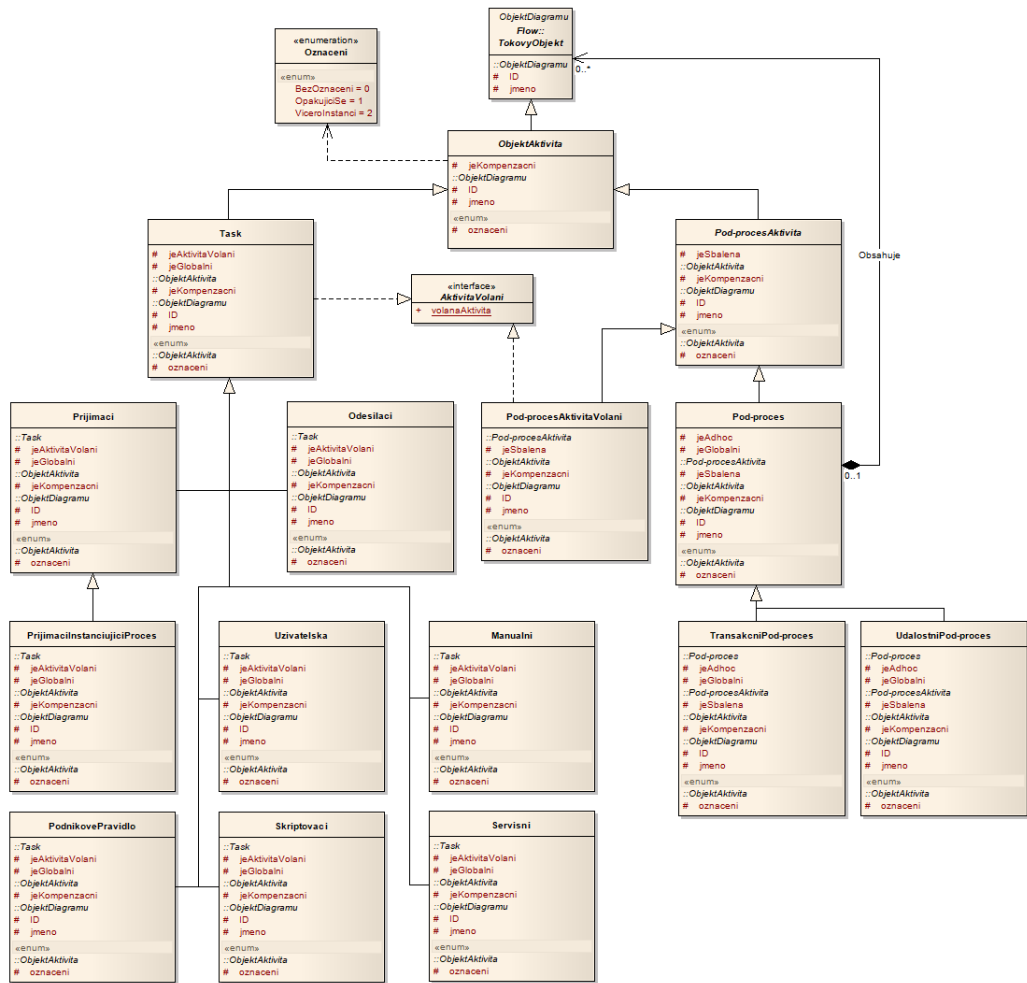
Obrázek 57: Doménový model – BPMN Diagram



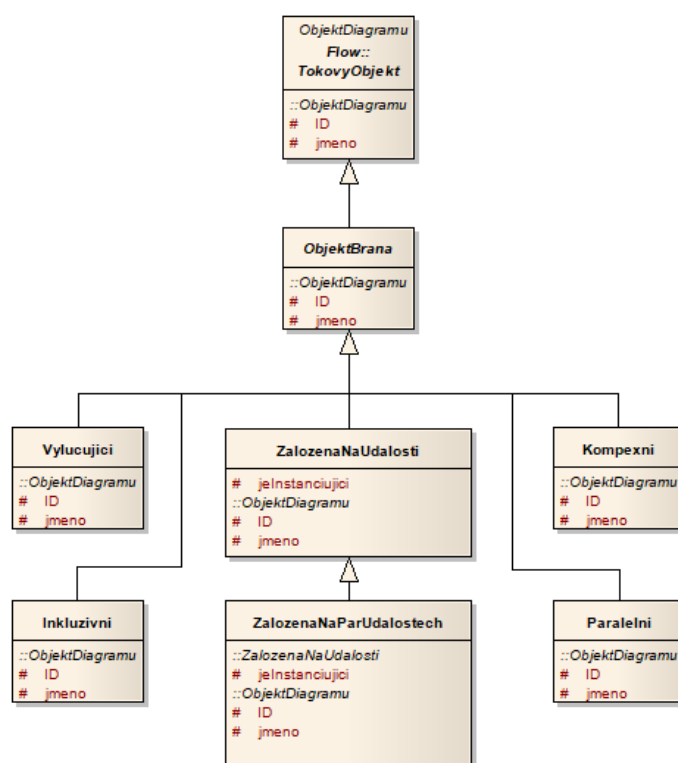
Obrázek 58: Doménový model – Tokové objekty



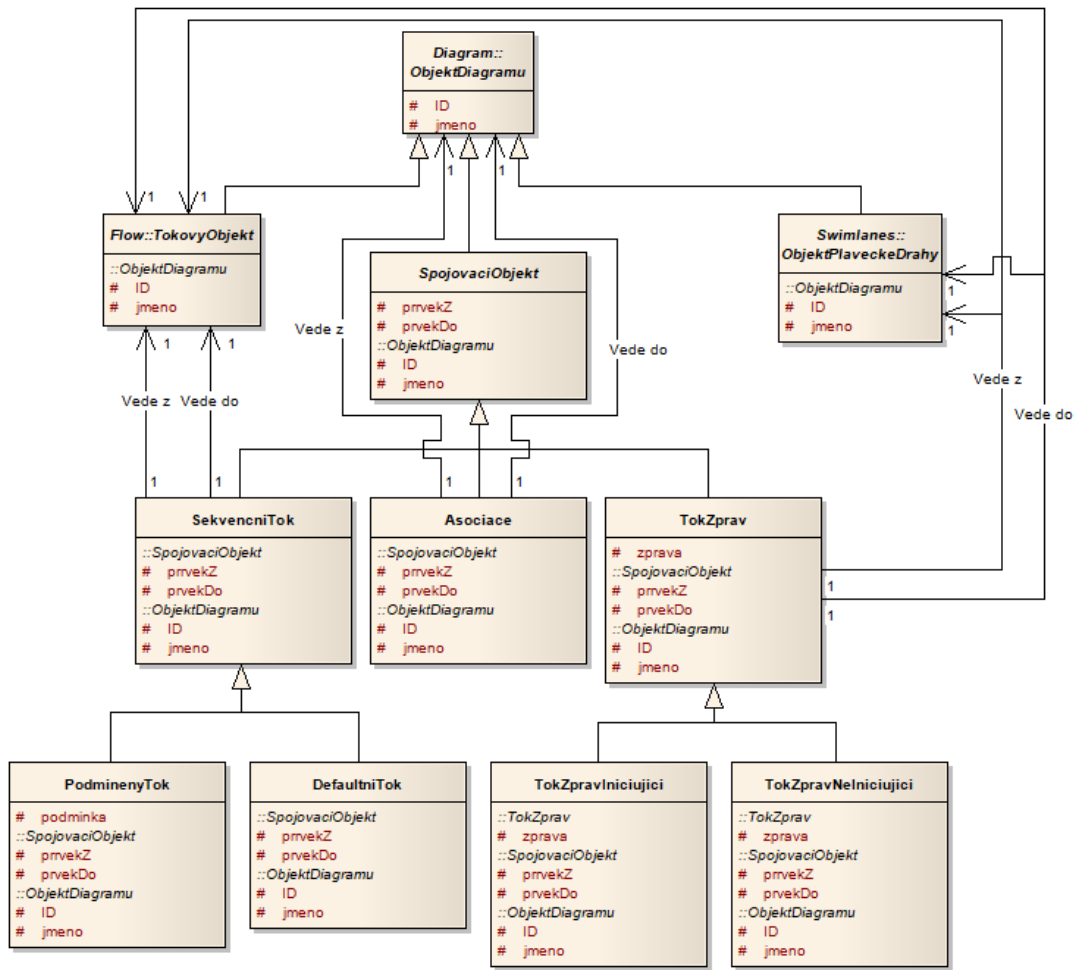
Obrázek 59: Doménový model – Události



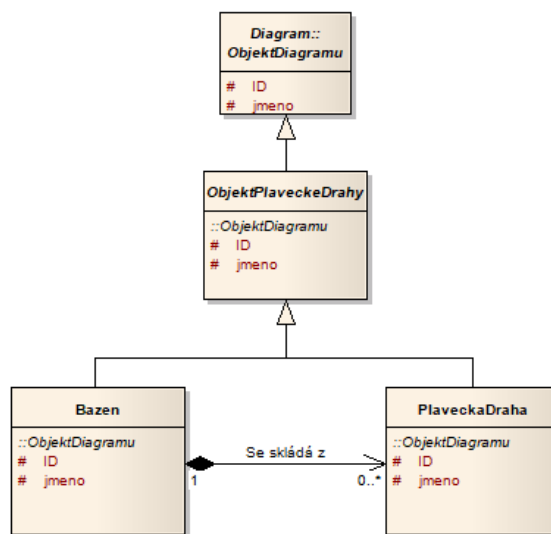
Obrázek 60: Doménový model – Aktivita



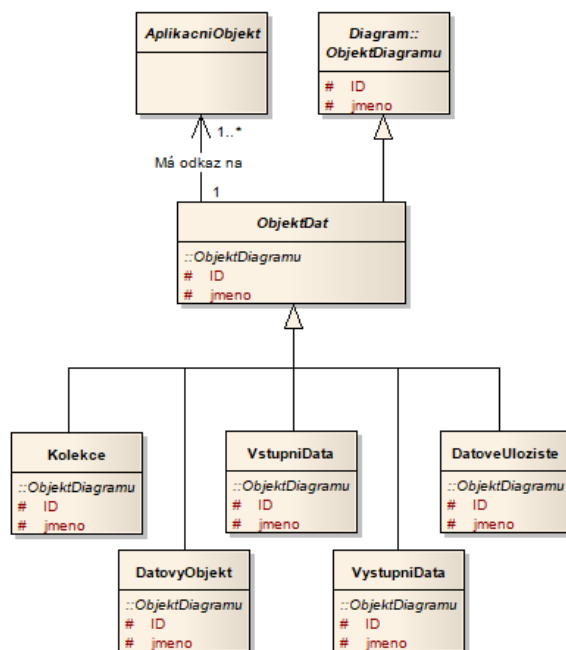
Obrázek 61: Doménový model – Brány



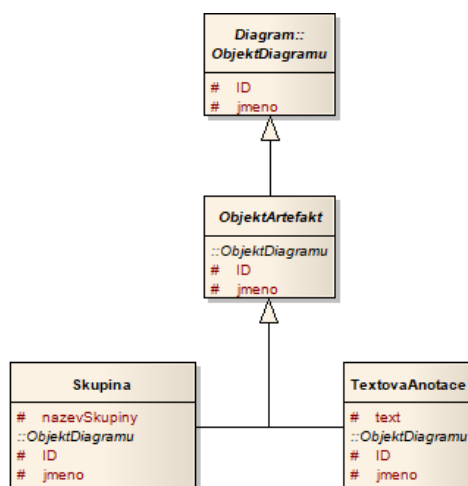
Obrázek 62: Doménový model – Propojení



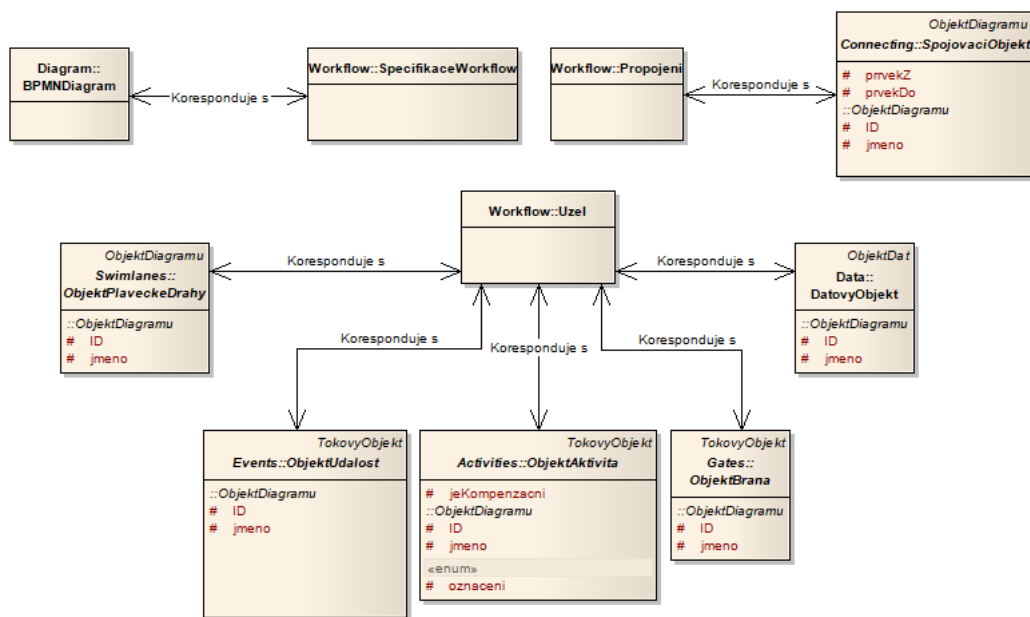
Obrázek 63: Doménový model – Plavecké dráhy



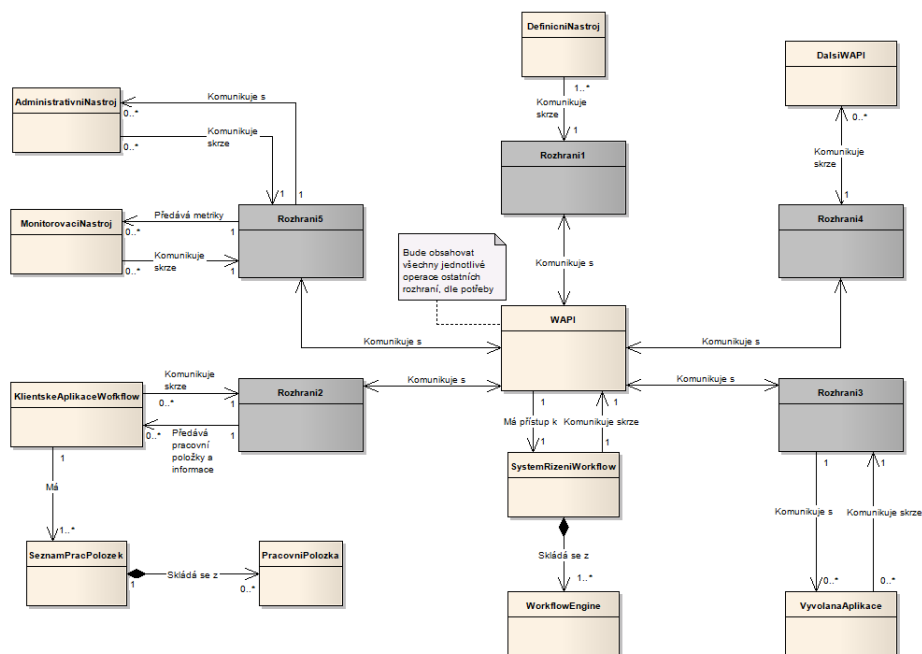
Obrázek 64: Doménový model – Data



Obrázek 65: Doménový model – Artefakty



Obrázek 66: Doménový model – napojení BPMN na S95



Obrázek 67: Doménový model – WRM

B Příklad zdrojového kódu XPDL

Příloha obsahuje příklady zdrojových kódů v jazyce XPDL vycházející z [76]. Je na ně odkazováno z kapitoly 9.5.

```
1 <?xml version="1.0" encoding="us-ascii"?>
2 <Package xmlns="http://www.wfmc.org/2002/XPDL1.0" xmlns:xpdl="http
   ://www.wfmc.org/2002/XPDL1.0" xmlns:xsi="http://www.w3.org/2001/
   XMLSchema-instance" xmlns:xyz="http://www.xyzeorder.com/workflow
   " xsi:schemaLocation="http://www.wfmc.org/2002/XPDL1.0
3 http://www.wfmc.org/standards/docs/xpdl.xsd" Id="0" Name="sample
   workflow process">
4   <PackageHeader>
5     ...
6   </PackageHeader>
7     ...
8   <TypeDeclarations>
9     <TypeDeclaration Id="Order" Name="Order">
10      <ExternalReference location="C:\wfmc\sample\orderschema.xsd"/>
11    </TypeDeclaration>
12    ...
13  </TypeDeclarations>
14  <Participants>
15    <Participant Id="DBConnection">
16      <ParticipantType Type="SYSTEM"/>
17      <Description>Reference to Database Resource</Description>
18    </Participant>
19  </Participants>
20  <Applications/>
21  <DataFields/>
22
23  <WorkflowProcesses>
24    <WorkflowProcess Id="1" Name="EOrder" AccessLevel="PUBLIC">
25      <ProcessHeader/>
26      <FormalParameters>
27        ...
28      </FormalParameters>
29      <DataFields>
30        ...
31      </DataFields>
32      <Applications>
33        ...
34      </Applications>
35      <Activities>
36        ...
37      </Activities>
38      <Transitions>
39        ...
40      </Transitions>
41    </WorkflowProcess>
42    <WorkflowProcess Id="2" Name="FillOrder" AccessLevel="PRIVATE">
43      ...
44  </WorkflowProcess>
```

```

45     ...
46 </WorkflowProcess>
47 </Package>

```

Zdrojový kód 2: Příklad v jazyce XPDL

```

1     ...
2 <Activities>
3   <!-- Implementation activity typu Tool -->
4   <Activity Id="1" Name="Check Data">
5     <Implementation>
6       <Tool Id="checkData" Type="APPLICATION">
7         ...
8       </Tool>
9     </Implementation>
10    <TransitionRestrictions>
11      <TransitionRestriction>
12        <Split Type="XOR">
13          <TransitionRefs>
14            <TransitionRef Id="22"/>
15            <TransitionRef Id="23"/>
16          </TransitionRefs>
17        </Split>
18      </TransitionRestriction>
19    </TransitionRestrictions>
20    <ExtendedAttributes>
21      <ExtendedAttribute Name="Coordinates">
22        <xyz:Coordinates xpos="183" ypos="389"/>
23      </ExtendedAttribute>
24    </ExtendedAttributes>
25  </Activity>
26  <!-- Implementation activity typu No -->
27  <Activity Id="58" Name="Raise Alarm">
28    <Implementation>
29      <No/>
30    </Implementation>
31    <ExtendedAttributes>
32      ...
33      <ExtendedAttribute Name="SystemActivity" Value="Alarm"/>
34    </ExtendedAttributes>
35  </Activity>
36  <!-- Implementation activity typu Subflow -->
37  <Activity Id="11" Name="Fill Order Subprocess">
38    <Implementation>
39      <SubFlow Id="FillOrder" Execution="ASYNCHR">
40        <ActualParameters>
41          <ActualParameter>orderNumber</ActualParameter>
42          <ActualParameter>orderInfo.orderType</ActualParameter>
43          <ActualParameter>orderInfo.emailAddress</ActualParameter>
44        </ActualParameters>
45      </SubFlow>
46    </Implementation>

```

```

47     ...
48     </Activity>
49     <!-- Route activity -->
50     <Activity Id="5">
51         <Route/>
52         ...
53     </Activity>
54     ...
55 </Activities>
56 ...

```

Zdrojový kód 3: Příklad aktivit v jazyce [XPDL](#)

```

1     ...
2     <Transitions>
3         <Transition Id="1" From="9" To="8"/>
4         <Transition Id="17" From="8" To="33">
5             <Condition Type="OTHERWISE"/>
6         </Transition>
7         <Transition Id="22" From="1" To="12">
8             <Condition>status == "Valid Data"</Condition>
9         </Transition>
10        ...
11    </Transitions>
12    ...

```

Zdrojový kód 4: Příklad přechodů v jazyce [XPDL](#)

C Formátování a slovníky přeložený pojmů

Tato příloha slouží jako slovník mezi původními anglickými termíny a překlady použitými v této diplomové práci. Jelikož zejména pro jednotlivé normy a standardy neexistují oficiální české překlady a termíny v těchto normách a standardech mohou být důležité (např. pro korektní dohledání v dané normě/standardu), je v následujících tabulkách možné dohledat původní název v anglickém jazyce oproti protikladu, přeloženému pro účely této diplomové práce.

C.1 Formátování pojmů

Při znovupoužití jednotlivých pojmů, které byly definovány a představeny, bylo v textu použito formátování *kurzívou*, za účelem jejich jednodušší identifikace a rozlišení.

C.2 Tabulky s pojmy

Pojmy jsou členěny do jednotlivých tabulek dle normy/standardu. Uvnitř každé tabulky jsou navíc dále kategorizovány. Seřazeny jsou chronologicky od jejich použití. Seznam tabulek je následující:

- Pro termíny z normy [ANSI/ISA-88^{337}](#) slouží tabulka 8.
- Pro termíny z normy [ANSI/ISA-95^{338}](#) slouží tabulka 9.
- Pro termíny ze standardu [Reference Architecture Model for Industrie 4.0^{339}](#) slouží tabulka 10.
- Pro termíny ze standardu [Workflow Reference Model^{340}](#) slouží tabulka 11.
- Pro termíny ze standardu [Business Process Model and Notation^{341}](#) slouží tabulka 12.

Tabulka 8: Slovník přeložený pojmů S88

Původní anglický název	Český překlad
Part 1. – Models and terminology	Část 1. – Modely a terminologie
Part 2. – Data structures and guidelines for languages	Část 2. – Datové struktury a metodické pokyny pro jazyky
Part 3. – General and Site recipe models and representation	Část 3. – Modely a reprezentace obecných a místních receptur
Part 4. – Batch production records	Část 4. – Záznamy dávkové výroby
Continual process	Kontinuální výroba
Discrete parts manufacturing process	Diskrétní výroba

Původní anglický název	Český překlad
Batch process	Dávková výroba
Batch	Dávka
Lot	Šarže
Model dávkového procesu	
Process model	Model dávkového procesu
Process stage	Fáze procesu
Process operation	Operace procesu
Process action	Akce procesu
Fyzický model	
Physical model	Fyzický model
Enterprise	Podnik
Site	Výrobní závod
Area	Provoz
Process cell	Procesní buňka
Unit	Jednotka
Equipment module	Modul zařízení
Control module	Modul řízení
Equipment entities	Subjekty vybavení
Model procedurálního řízení	
Procedural control model	Model procedurálního řízení
Batch control	Dávkové řízení
Basic control	Základní řízení
Procedural control	Procedurální řízení
Coordination control	Koordinační řízení
Procedure	Procedura
Unit procedure	Jednotková procedura
Operation	Operace
Phase	Fáze
Receptury	
Recipe	Receptura
General recipe	Obecná receptura
Site recipe	Místní receptura
Master recipe	Hlavní receptura
Control recipe	Řídící receptura
Procedural entity	Procedurální prvky
Stavy a módy	
Procedural state model	Stavy a módy
Idle	Nečinný
Running	Běžící
Complete	Dokončený
Pausing	Pozastavující se
Paused	Pozastaven

Původní anglický název	Český překlad
Holding	Přidrží se
Held	Přidržen
Restarting	Restartující se
Stopping	Zastavující se
Stopped	Zastaven
Aborting	Rušící se
Aborted	Zrušen
Aktivity a funkce řízení dávkové výroby	
Control activity model	Aktivity a funkce řízení dávkové výroby
Recipe management	Správa receptur
Production planning and scheduling	Plánování a rozvrhování výroby
Production information management	Správa produktových dat
Process cell management	Správa procesů
Unit supervision	Řízení jednotek
Process control	Řízení procesů
Personnel and environmental protection	Ochrana zaměstnanců a životního prostředí

Tabulka 9: Slovník přeložený pojmů S95

Původní anglický název	Český překlad
Part 1. – Models and terminology	Část 1. – Modely a terminologie
Part 2. – Objects and attributes for Enterprise-Control system integration	Část 2. – Objekty a atributy pro integrované systémy podnikového řízení
Part 3. – Activity models of Manufacturing Operations Management	Část 3. – Modely činnosti Manufacturing Operations Management
Part 4. – Objects and attributes for Manufacturing Operations Management integration	Část 4. – Objekty a atributy pro integraci Manufacturing Operations Management
Part 5. – Business-to-Manufacturing transactions	Část 5. – Transakce mezi obchodem a výrobou
Part 6. – Messaging service model	Část 6. – Model služby zasílání zpráv
Part 7. – Alias service model	Část 7. – Model služby alias
Part 8. – Information exchange profiles	Část 8. – Profily výměny informací
Functional hierarchy	Hierarchie řízení
Business planning & logistics	Podnikové plánování a logistika
Batch control	Dávkové řízení

Původní anglický název	Český překlad
Continuous control	Kontinuální řízení
Discrete control	Diskrétní řízení
Funkce 3. úrovně	
Resource allocation and control	Řízení a přidělování zdrojů
Dispatching production	Dispečerské řízení výroby
Data collection and acquisition	Shromažďování a získávání dat
Quality operations management	Řízení kvality
Process management	Řízení procesů
Production tracking	Sledování výroby
Performance analysis	Analýza výkonnosti
Operations and detailed scheduling	Provoz a podrobné plánování
Document control	Správa dokumentace
Labor management	Řízení pracovních sil
Maintenance operations management	Řízení údržby
Movement, storage and tracking of materials	Přesun, uskladnění a sledování materiálu
Hierarchie zařízení	
Equipment hierarchy	Hierarchie zařízení
Enterprise	Podnik
Site	Výrobní závod
Area	Provoz
Work centers	Pracovní centra
Process cell	Procesní buňka
Production unit	Produkční jednotka
Production line	Výrobní linka
Storage zone	Skladovací zóna
Work units	Pracovní jednotky
Unit	Jednotka
Work cell	Pracovní buňka
Storage unit	Skladovací buňka
Model funkcí	
Functional model	Model funkcí
Order processing	Zpracování objednávek
Production scheduling	Plánování výroby
Production control	Řízení výroby
Material and energy control	Řízení materiálu a energií
Procurement	Zásobování
Quality assurance	Zajištění kvality
Product inventory control	Řízení stavu zásob a produktů
Product cost accounting	Nákladové účetnictví produktu
Product shipping admin	Správa přepravy produktu

Původní anglický název	Český překlad
Maintenance management	Správa údržby
Marketing & sales	Prodej a marketing
R&D and engineering	Výzkum, vývoj a inženýrství
MOM	
Production operations management	Řízení výrobních operací
Maintenance operations management	Řízení operací údržby
Quality operations management	Řízení operací kvality
Inventory operations management	Řízení skladových operací
Operations definition	Definice operací
Operations capability	Kapacity operací
Operations request	Požadavky operací
Operations response	Výstupy operací
Detailed scheduling	Detailní plánování
Dispatching	Dispečerské řízení
Execution management	Řízení realizace
Data collection	Shromažďování dat
Resource management	Řízení zdrojů
Definition management	Správa definic
Tracking	Sledování výroby
Performance analysis	Analýza výkonnosti
Modely	
Personnel model	Model personálu
Material model	Model materiálu
Lot	Šarže
Sublot	Podšarže
Role based equipment model	Model vybavení v kontextu rolí
Physical asset model	Model fyzický aktiv
Process segment model	Model procesního segmentu
Work definition model	Model definice práce
Work schedule model	Model rozvrhu práce
Resource relationship network information	Vztahy mezi zdroji
Work definition information	Definice práce
Workflow specification information	Specifikace workflow
Work schedule information	Rozvrh práce
Work performance information	Informace o výkonnosti práce
Work capability information	Kapacity práce
Komunikace	
Noun	Podstatné jméno
Verb	Sloveso

Tabulka 10: Slovník přeložený pojmů [RAMI 4.0](#)

Původní anglický název	Český překlad
Pravá horizontální osa	
Connected world	Propojený svět
Enterprise	Podnik
Work centers	Pracovní centra
Station	Stanice
Control device	Řídící jednotka
Field device	Provozní zařízení
Product	Produkt
Levá horizontální osa	
Development	Vývoj
Maintenance/Usage	Údržba/provoz
Production	Výroba
Type	Typ
Instance	Instance
Vertikální osa (vrstvy)	
Business	Businessová vrstva
Functional	Funkční vrstva
Information	Informační vrstva
Communication	Komunikační vrstva
Integration	Integrační vrstva
Asset	Vrstva aktiv
Industry 4.0 komponenta	
Resource manager	Správa zdrojů

Tabulka 11: Slovník přeložený pojmů [WRM](#)

Původní anglický název	Český překlad
Process definition	Definice procesu
Activity	Aktivita
Process instance	Instance procesu
Activity instance	Instance aktivity
Work item	Pracovní položka
Work list	Seznam prací
Invoked application	Vyvolaná aplikace
Implementační model	
Definition tool	Nástroj pro definici
Organisation/Role model data	Organizační data
Workflow enactment service	Systém řízení workflow
WFM Engine(s)	Workflow management enginy
Workflow control data	Řídící data workflow

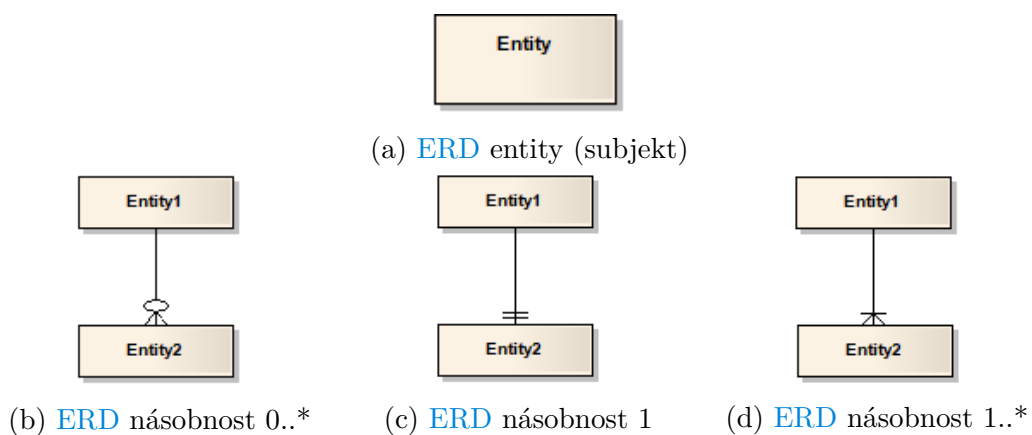
Původní anglický název	Český překlad
Workflow relevant data	Data vztahující se k workflow
Worklist handler/Workflow client application	Klientská workflow aplikace
User interface	Uživatelské rozhraní
Application(s)	Aplikace
Workflow application data	Aplikační workflow data
Supervisor	Dohled
Workflow participant	Účastník workflow
WRM	
Process definition tools	Nástroje pro definici procesu
Workflow API and Interchange formats	WAPI a formáty pro výměnu informací
Workflow client applications	Klientské workflow aplikace
Invoked applications	Vyvolané aplikace
Administration & monitoring tools	Nástroje pro administrativu a monitorování

Tabulka 12: Slovník přeložený pojmů **BPMN**

Původní anglický název	Český překlad
Orchestration	Orchestrace
Collaboration	Spolupráce
Conversation	Konverzace
Choreographies	Choreografie
Flow objects	Tokové objekty
Connecting objects	Spojovací objekty
Swimlanes	Plavecké dráhy
Data	Datové objekty
Artifacts	Artefakty
Events	Události
Activities	Aktivity
Gateways	Brány
Události	
Start	Počáteční
Intermediate	Mezikrok
End	Koncová
Message	Zpráva
Timer	Časovač
Error	Error
Escalation	Eskalace
Cancel	Zrušení

Původní anglický název	Český překlad
Compensation	Kompenzace
Conditional	Podmíněná
Link	Odkaz
Signal	Signál
Terminate	Ukončení
Multiple	Vícenásobná
Parallel multiple	Vícenásobná paralelní
Catching events	Catch události
Throwing events	Throw události
Non-interrupting events	Nepřerušující události
Boundary events	Události na hraně aktivit/Hraniční události
Aktivita	
Atomic activity (task)	Task
Sub-process	Pod-proces
Collapsed sub-process	Sbalený pod-proces
Expanded sub-process	Rozbalený pod-proces
Call activity	Aktivita volání
Transaction	Transakční pod-proces
Receive task	Přijímací
Receive task (instantiating)	Přijímací instancující proces
Send task	Odesílací
User task	Uživatelská
Manual	Manuální
Business rule	Podnikové pravidlo
Business rules engine	Engine podnikových pravidel
Script task	Skriptovací
Service task	Servisní
Marking (marked activities)	Označení (označené aktivity)
Loop task	Opakující se
Multiple instance	Vícero instancí
Compensation	Kompenzační
Ad-hoc	Ad-hoc
Choreography task	Aktivita choreografie
Collapsed sub-choreography	Sbalená aktivita pod-choreografie
Brány	
Exclusive	Vylučující
Event-based	Založená na události
Parallel event-based	Založená na paralelních událostech
Inclusive	Inkluzivní
Complex	Komplexní
Parallel	Paralelní

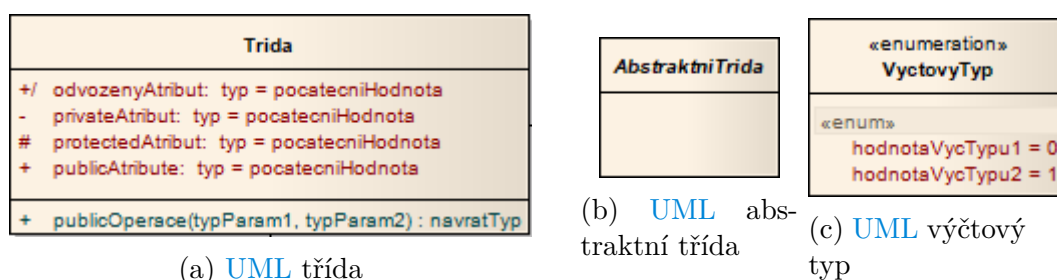
Původní anglický název	Český překlad
Spojovací objekty	
Sequence flow	Sekvenční toky
Conditional flow	Podmíněný tok
Default flow	Defaultní tok
Message flow	Toky zpráv
Association	Asociace
Plavecké dráhy	
Pool	Bazén
Swimlane	Plavecká dráha
Datové objekty	
Data object	Datový objekt
Data object (collection)	Datový objekt kolekce
Data input	Vstupní data
Data output	Výstupní data
Data store	Úložiště dat
Artefakty	
Group	Skupina
Text annotation	Textová anotace



Obrázek 68: Notace ERD diagramů

D Vysvětlení ERD diagramů

Tato příloha slouží k vysvětlení notace [Entity-relationship diagramů](#). Tyto diagramy slouží k modelování vztahů mezi různými entitami (subjekty). V této diplomové práci byla použita notace vyobrazená na obrázku 68, kde je vidět i notace násobností.



Obrázek 69: Notace UML diagramů – třídy

E Vysvětlení diagramů UML

Tato příloha slouží k vysvětlení notace diagramů UML ve verzi 2.1. Byly použity **diagramy tříd** a zejména **doménové diagramy**, které mají v UML jednotnou notaci.

E.1 Diagramy tříd

Tyto diagramy slouží k modelování datových struktur, jejich atributů, operací a vztahů mezi nimi. V této diplomové práci byla notace popsána níže. Nejedná se o popis notace kompletní, nýbrž pouze její částí, které byly použity.

E.1.1 Třídy

Třída je modelována jako čtverec s tučným názvem. Obsahuje dva základní prvky:

1. Atributy – vlastnosti třídy; proměnné
2. Operace – funkcionalita třídy; její metody

U obou prvků se rozlišuje jejich viditelnost, a to následovně:

- + – Public,
- # – Protected,
- - – Private,

přičemž atribut může obsahovat ještě speciální lomítko za viditelností „/“, které říká, že se jedná o atribut odvozený. Pakliže třídu instanciujeme, vzniká **objekt třídy**. Samotná třída může být také **abstraktní**, což je třída, ze které nemůže objekt vzniknout. Pakliže je v diagramu použita třída z jiného balíčku, pak je její název ve formátu <NázevBalíčku>::<JménoTřídy>. Notaci tříd lze vidět na obrázcích 69a a 69b.

E.1.1.1 Výčtový typ (enumeration) Výčtový typ představuje nový typ, kterého mohou atributy a parametry operací nabývat. Součástí mohou být i jeho možné hodnoty. Notaci lze vidět na obrázku [69c](#).

E.1.2 Vztahy mezi třídami

Mezi jednotlivými třídami (obecně prvky) vyjadřujeme následující vztahy:

- Asociace – vyjadřuje vztah mezi dvěma třídami
- Agregace – vztah mezi dvěma třídami říkájící, že daná třída se skládá z prvků jiné třídy
- Kompozice – silnější agregace, kde při zániku celku, zanikají i jednotlivé prvky, ze kterých se celek skládá
- Dědičnost – představuje vztah, kdy jedna třída dědí z třídy jiné (tzn. je jejím typem), pravidlo „is a“
- Závislost – slabší vztah mezi třídami, který značí, že třída nějakým způsobem závisí na třídě jiné

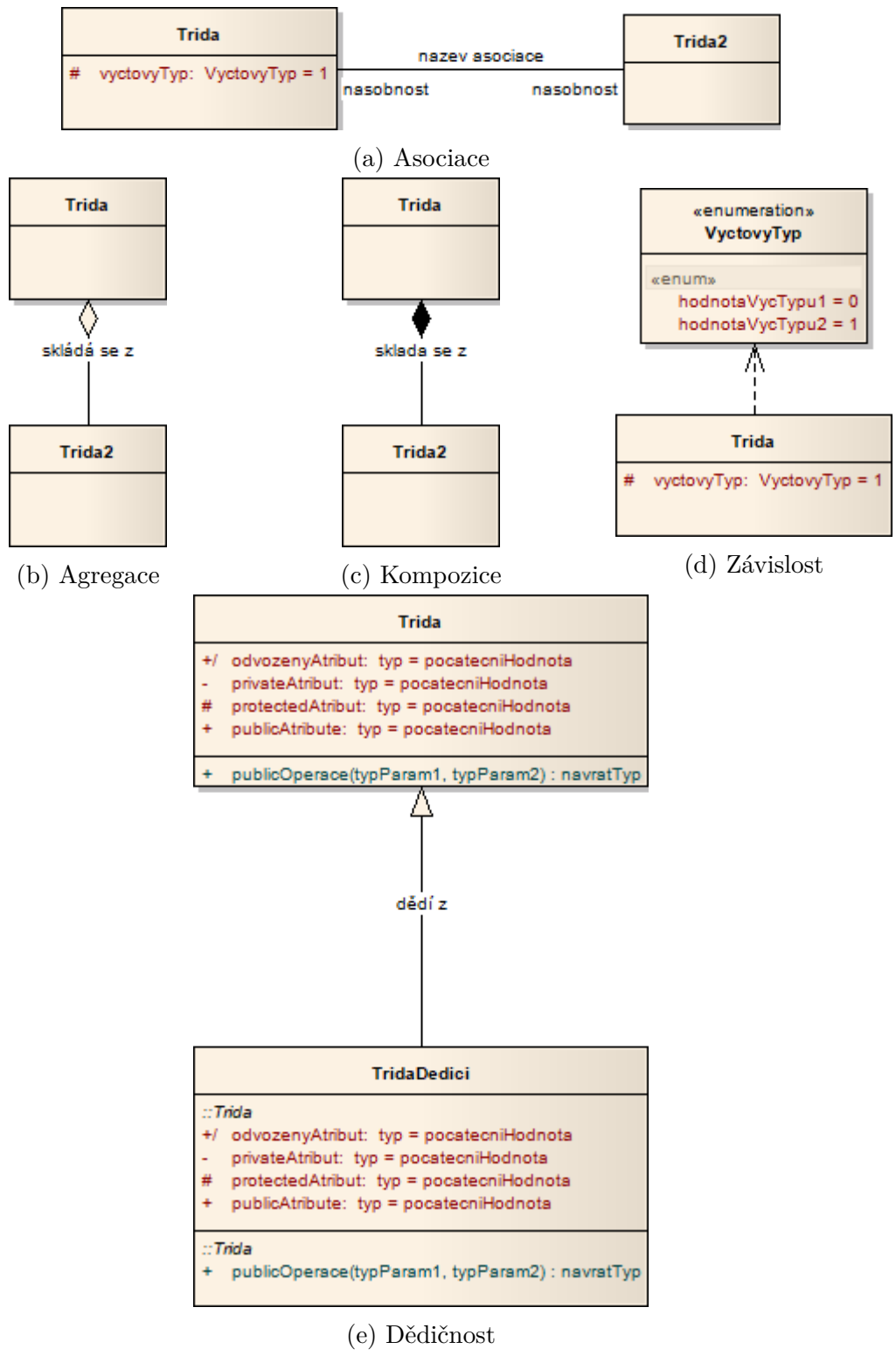
U prvních tří se může uvádět navigace (šipka ukazující, jak jsou třídy vzájemně dostupné) a zároveň násobnost (jež říká, s kolika objekty může být daný objekt propojen). Dědičnost má určitou „navigaci“ pevně danou v notaci. U závislosti je vhodné uvádět navigaci také (odkud kam). Samotnou násobnost můžeme vyjadřovat následovně:

- celým číslem (např. „0“, „1“, ...),
- intervalem (např. „[0..5], [1..3]“, ...),
- neomezeným počtem, skrze symbol „*“.

Notaci jednotlivých vztahů lze vidět na obrázku [70](#).

E.2 Doménové diagramy

Tyto diagramy se dají označit za určitou podmnožinu diagramů tříd, avšak tyto diagramy jsou do určité míry zjednodušené. Zásadní rozdíl je absence metod, přičemž jsou zachovány atributy, které avšak nemusejí mít datové typy. Tento typ diagramu slouží k zachycení základních vztahů a potenciálních konceptů tříd. Na základě něj jsou zpracovávány detailnější diagramy tříd [\[77\]](#)[\[78\]](#). Notace je totožná s tou představenou pro diagramy tříd, tudíž zde nebude explicitně uvedena.



Obrázek 70: Notace UML diagramů – vztahy

F Použité nástroje

Tato příloha slouží pro výčet nástrojů použitých při tvorbě této diplomové práce. Pro tvorbu a úpravu obrázků byl použit grafický nástroj Photopea⁴⁷. Pro modelování Petriho sítí byl použit nástroj PIPE 2 ve verzi 4.3⁴⁸. Pro tvorbu UML diagramů byl použit Enterprise architect ve verzi 7.1⁴⁹. Pro BPMN a DMN modelování byly použity nástroje:

- BPMN.io⁵⁰,
- a Camunda modeler⁵¹.

⁴⁷www.photopea.com

⁴⁸<http://pipe2.sourceforge.net/>

⁴⁹<https://sparxsystems.com/>

⁵⁰<https://bpmn.io/>

⁵¹<https://modeler.cloud.camunda.io/>

G Obsah příloženého CD/DVD

Tato příloha slouží pro popis obsahu příloženého CD/DVD. Níže je popsána adresářová struktura a obsah.

doc/

Text práce ve formátu PDF a veškeré soubory potřebné pro vygenerování PDF dokumentu, a to v ZIP archivu.

src/

Obsahuje kompletní projekt vytvořený v programu Enterprise Architect (verze 7.1), ve formátu EAP. Součástí jsou veškeré diagramy vytvořené a použité v této diplomové práci. Package ANALYSIS obsahuje analytickou část práce z přílohy [A](#).

Seznam zkratek

AI Artificial intelligence

AM Additive manufacturing (aditivní výroba)

API Application Programming Interface

AR Augmented Reality

AWS Amazon Web Services

B2MML Business-To-Manufacturing Markup Language

BatchML Batch Markup Language

BOM Bill Of Material

BPD Business Process Diagram

BPEL4WS Business Process Execution Language for Web Services

BPM Business Process Management

BPMI Business Process Management Initiative

BPMN Business Process Model and Notation

BPMS Business Process Management System

CAN Controller Area Network

CPN Coloured Petri Nets

CPS Cyber-physical system

CRM Customer relationship management

DMN Decision Model and Notation

DT Digital Twin

ERD Entity-relationship diagram

ERP Enterprise Resource Planning

FBT Field Bus Technology

GUI Graphical User Interface

GUID Global Unique Identifier

HLPN High-Level Petri Nets

HMI Human Machine Interface

HW Hardware

IaaS Infrastructure as a Service

IIoT Industrial Internet of Things

IoT Internet of Things

IP Internet Protocol

ISA International Society of Automation

IT Informační Technologie

M2M Machine-to-Machine

MES Manufacturing Execution System

MESA Manufacturing Enterprise Solutions Association

MOM Manufacturing Operations Management

MRP Material Requirements planning

MRP II Manufacturing resource planning

MVC Model-View-Controller

NFC Near Field Communication

OCL Objekt Constraint Language

OMG Object Management Group

OOP Objektově orientované programování

PaaS Platform as a Service

PIS Podnikový informační systém

PLC Programmable Logic Controller

PN Petri Net

RAMI 4.0 Reference Architecture Model for Industrie 4.0

RFID Radio Frequency Identification

S88 ANSI/ISA-88

S95 ANSI/ISA-95

SaaS Software as a Service

SF Smart Factory (chytrá továrna)

SOP Standard Operating Procedure

SW Software

UI User Interface

UML Unified Modeling Language

W3C World Wide Web Consortium

WAPI Workflow Application Programming Interface

WBF World Batch Forum

WfM Workflow Management

WfMC Workflow Management Coalition

WfMS Workflow Management System

WPDL Workflow Process Definition Language

WRM Workflow Reference Model

WSBPEL Web Services Business Process Execution Language

XML Extensible Markup Language

XPDL XML Process Definition Language

XSD XML Schema Definition

YAWL Yet Another Workflow Language

Literatura

- [1] VALACICH, Joseph; SCHNEIDER, Christoph. *Information Systems Today: Managing the Digital World*. Eighth. 2018. 560 s. ISBN 978-0-13-463520-0.
- [2] RAINER, R. Kelly; CASEY, G. Ciagelski. *Introduction to Information Systems: Supporting and Transforming Business*. Fourth. 2012. 480 s. ISBN 978-1-118-06334-7.
- [3] KIRCHMER, Mathias. *High Performance Through Business Process Management: Strategy Execution in a Digital World*. Third. 2017. 231 s. ISBN 978-3-319-51259-4.
- [4] APICS. *APICS Dictionary: The essential supply chain reference*. Fourteenth. 2013. 198 s. ISBN 978-0-9882146-1-3.
- [5] KURBEL, E. Karl. *Enterprise Resource Planning and Supply Chain Management:: Functions, Business Processes and Software for Manufacturing Companies*. First. 2013. 365 s. ISBN 978-3-642-31573-2.
- [6] WALLACE, F. Thomas; KREMZAR, H. Michael. *ERP: Making It Happen: The Implementers' Guide to Success with Enterprise Resource Planning*. Third. 2001. 385 s. ISBN 0-471-39201-4.
- [7] ORACLE. *Enterprise Resource Planning (ERP) | Oracle* [online]. [cit. 2022-1-17]. Dostupný z: <https://www.oracle.com/erp/>.
- [8] MICROSOFT. *What is ERP and why do you need it? | Microsoft Dynamics 365* [online]. [cit. 2022-1-17]. Dostupný z: <https://dynamics.microsoft.com/en-us/erp/what-is-erp/>.
- [9] ASSECO SOLUTIONS, a. s. *Helios* [online]. [cit. 2022-2-8]. Dostupný z: <https://www.helios.eu/>.
- [10] A.S, ABRA Software. *Informační systém a ERP pro každou firmu | ABRA Software* [online]. [cit. 2022-2-8]. Dostupný z: <https://www.abra.eu/>.
- [11] AMINI, Mohammad; ABUKARI, Arnold Mashud. *ERP SYSTEMS ARCHITECTURE FOR THE MODERN AGE: A REVIEW OF THE STATE OF THE ART TECHNOLOGIES* [online]. [cit. 2022-1-18]. Dostupný z: https://www.researchgate.net/publication/343450571_ERP_SYSTEMS_ARCHITECTURE_FOR_THE_MODERN_AGE_A_REVIEW_OF_THE_STATE_OF_THE_ART_TECHNOLOGIES
- [12] GILLIS, Alexander S. *What is IoT (Internet of Things) and How Does it Work?* [online]. [cit. 2022-2-7]. Dostupný z: <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>.
- [13] GILCHRIST, Alasdair. *Industry 4.0: The Industrial Internet of Things*. First. Thailand: Apress, 2016. 259 s. ISBN 978-1-4842-2046-7.
- [14] ELAHI, Ata; GSCHWENDER, Adam. *Zigbee Wireless Sensor and Control Network*. 2009. 289 s. ISBN 978-0-137-13485-4.

- [15] ZIEGLER, Sébastien; CRETTEZ, Cédric; ADDITIONAL CONTRIBUTIONS FROM IOT6 PARTNERS AND THE INDUSTRY ADVISORY BOARD. *Universal Integration of the Internet of Things through an IPv6-based Service Oriented Architecture enabling heterogeneous components interoperability* [online]. [cit. 2022-1-25].
Dostupný z: <https://iot6.eu/sites/default/files/IoT6%20-%20D7.3.pdf>.
- [16] MICROSOFT. *What is Azure–Microsoft Cloud Services / Microsoft Azure* [online]. [cit. 2022-1-23].
Dostupný z: <https://azure.microsoft.com/en-us/overview/what-is-azure/>.
- [17] AMAZON. *What is AWS* [online]. [cit. 2022-1-23].
Dostupný z: <https://aws.amazon.com/>.
- [18] GOOGLE. *Cloud Computing Services / Google Cloud* [online]. [cit. 2022-1-23].
Dostupný z: <https://cloud.google.com/>.
- [19] MICROSOFT. *Personal Cloud Storage – Microsoft OneDrive* [online]. [cit. 2022-1-23]. Dostupný z: <https://www.microsoft.com/en-us/microsoft-365/onedrive/online-cloud-storage>.
- [20] DROPBOX. *Dropbox* [online]. [cit. 2022-1-23].
Dostupný z: <https://www.dropbox.com/>.
- [21] MICROSOFT. *What is IaaS? Infrastructure as a Service / Microsoft Azure* [online]. [cit. 2022-1-19].
Dostupný z: <https://azure.microsoft.com/en-us/overview/what-is-iaas/>.
- [22] MICROSOFT. *What is PaaS? Platform as a Service / Microsoft Azure* [online]. [cit. 2022-1-19]. Dostupný z: <https://azure.microsoft.com/en-us/overview/what-is-paas/>.
- [23] MICROSOFT. *What is SaaS? Software as a Service / Microsoft Azure* [online]. [cit. 2022-1-19]. Dostupný z: <https://azure.microsoft.com/en-us/overview/what-is-saas/>.
- [24] SABRE, www.sabre.cz. *Od 1. průmyslové revoluce ke 4. / Technický týdeník* [online]. [cit. 2022-4-1].
Dostupný z: https://www.technickytydenik.cz/rubriky/ekonomika-byznys/od-1-prumyslove-revoluce-ke-4_31001.html.
- [25] DESOUTTER TOOLS. *Průmyslová revoluce – Od Průmyslu 1.0 k Průmyslu 4.0 - Desoutter Industrial Tools* [online]. [cit. 2022-4-1].
Dostupný z: <https://www.desouttertools.cz/prumysl-4-0/novinky/563/prumyslova-revoluce-od-prumyslu-1-0-k-prumyslu-4-0#>.
- [26] HANDL, JAN. *Průmyslová revoluce stručně a přehledně / HistorieBlog.cz* [online]. [cit. 2022-4-1].
Dostupný z: <https://www.historieblog.cz/2018/09/prumyslova-revoluce-strucne-a-prehledne/>.

- [27] NĚMEC, Václav. *dějepis.com* [online]. [cit. 2022-4-1].
Dostupný z: <https://www.dejepis.com/ucebnice/anglicka-prumyslova-revoluce-a-jeji-rozsireni/>.
- [28] BUSINESSINFO.CZ. *Čtyři průmyslové revoluce* | *BusinessInfo.cz* [online]. [cit. 2022-4-1].
Dostupný z: <https://www.businessinfo.cz/clanky/ctyri-prumyslove-revoluce/>.
- [29] WIKIPEDIA. *Incandescent light bulb* - *Wikipedia* [online]. [cit. 2022-4-1].
Dostupný z: https://en.wikipedia.org/wiki/Incandescent_light_bulb.
- [30] WIKIPEDIA. *Tesla coil* - *Wikipedia* [online]. [cit. 2022-4-1].
Dostupný z: https://en.wikipedia.org/wiki/Tesla_coil.
- [31] WIKIPEDIA. *Programmable logic controller* - *Wikipedia* [online]. [cit. 2022-4-1].
Dostupný z: https://en.wikipedia.org/wiki/Programmable_logic_controller.
- [32] LEE, Edward A. The Past, Present and Future of Cyber-Physical Systems: A Focus on Models. *Sensors*. 2015, roč. 15, č. 3, s. 4837–4869. Dostupný také z: <https://www.mdpi.com/1424-8220/15/3/4837>. ISSN 1424-8220.
- [33] WIKIPEDIA. *Sensor* – *Wikipedia*.
Dostupný z: <https://en.wikipedia.org/wiki/Sensor>.
- [34] WIKIPEDIA. *Actuator* - *Wikipedia*.
Dostupný z: <https://en.wikipedia.org/wiki/Actuator>.
- [35] SUBRAMANIAN, Balakrishnan. *Cyber Physical Systems*.
Dostupný z: <https://datascience.foundation/datatalk/cyber-physical-systems>.
- [36] FOUNDATION, National Science. *nsf21551 Cyber-Physical Systems (CPS) / NSF - National Science Foundation*.
Dostupný z: https://www.nsf.gov/publications/pub_summ.jsp?ods_key=nsf21551.
- [37] NEGRI, Elisa; FUMAGALLI, Luca; MACCHI, Marco. A Review of the Roles of Digital Twin in CPS-based Production Systems. *Procedia Manufacturing*. 2017, roč. 11, s. 939–948. Dostupný také z: <https://www.sciencedirect.com/science/article/pii/S2351978917304067>.
ISSN 2351-9789.
- [38] MINISTERSTVO PRŮMYSLU A OBCHODU ČESKÉ REPUBLIKY. *Iniciativa Průmysl 4.0*. 2017.
Dostupný také z: <https://www.mpo.cz/assets/dokumenty/53723/64358/658713/priloha001.pdf>
- [39] CHMELÁŘ, Aleš; VOLČÍK, Stanislav; NECHUTA, Aleš; HOLUB, Ondřej. *Dopady digitalizace na trh práce v ČR a EU*. 2015.
Dostupný také z: <https://www.vlada.cz/assets/evropske-zalezitosti/analyzy-EU/Dopady-digitalizace-na-trh-prace-CR-a-EU.pdf>.
- [40] MAGRETTA, Joan. *Understanding Michael Porter: The Essential Guide to Competition and Strategy*. 2011. ISBN 1422160599,9781422160596.
- [41] JOHNSON, Charlotta. *S88 for Beginners*. 2004.

- [42] TECHNOR. *Hledání* [online]. [cit. 2022-4-25].
Dostupný z: <https://www.technicke-normy-csn.cz/hledani/?search=61512>.
- [43] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-88.01-1995: Batch Control Part 1: Models and Terminology*. 1995. ISBN 1-55617-562-0.
- [44] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-88.00.01-2010: Batch Control Part 1: Models and Terminology*. 2010. ISBN 978-1-936007-75-2.
- [45] TECHNOR. *Hledání* [online]. [cit. 2022-4-25].
Dostupný z: <https://www.technicke-normy-csn.cz/hledani/?search=62264>.
- [46] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-95.01-2010: Enterprise-Control System Integration Part 1: Models and Terminology*. 2010. ISBN 978-1-936007-47-9.
- [47] MESA INTERNATIONAL. *MES Functional model* [online]. [cit. 2022-4-25].
Dostupný z: <https://growthzonesitesprod.azureedge.net/wp-content/uploads/sites/2220/2021/06/MESA-11-Functional-Model-1997.jpg>.
- [48] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-95.03-2013: Enterprise-Control System Integration Part 3: Activity Models of Manufacturing Operations Management*. 2013. ISBN 978-0-876640-33-3.
- [49] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-95.02-2018: Enterprise-Control System Integration Part 2: Object and attributes for Enterprise-control System Integration*. 2018. ISBN 978-1-63331-033-6.
- [50] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-95.04-2018: Enterprise-Control System Integration Part 4: Object and Attributes for Manufacturing Operations Management Integration*. 2018. ISBN 978-1-64331-026-8.
- [51] INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-95.05-2013: Enterprise-Control System Integration Part 5: Business-To-Manufacturing Transactions*. 2013. ISBN 978-0-876640-35-7.
- [52] MESA INTERNATIONAL. *B2MML* [online]. [cit. 2022-5-2].
Dostupný z: <https://mesa.org/topics-resources/b2mml/>.
- [53] MINTCHELL, Gary. *MESA International and WBF to Merge Organizations* [online]. [cit. 2022-5-2].
Dostupný z: <https://www.automationworld.com/products/software/news/13306899/mesa-international-and-wbf-to-merge-organizations>.
- [54] AUTOMATION.COM. *MESA Announces Version 7 Release of B2MML-BatchML Specifications* [online]. [cit. 2022-5-2].
Dostupný z: <https://www.automation.com/en-us/products/november-2020/mesa-version-7-b2mml-batchml-specifications>.
- [55] INTERNATIONAL SOCIETY OF AUTOMATION. *ISA-TR-88.95.1: Using ISA-88 and ISA-95 Together*. 2008. ISBN 978-1-934394-78-6.

- [56] VDI & ZVEI. *Status report: Reference Architecture Model Industrie 4.0 (RAMI4.0)* [online]. [cit. 2022-5-12].
Dostupný z: https://www.zvei.org/fileadmin/user_upload/Themen/Industrie_4.0/Das_Referenz_Architekturmodell/5305_Publikation_GMA_Status_Report_ZVEI_Reference_Architecture_Model_Industrie_4.0.pdf.
- [57] TECHNOR. *ČSN EN IEC 62890 (180410)* [online]. [cit. 2022-6-10]. Dostupný z: <https://www.technicke-normy-csn.cz/csn-en-iec-62890-180410-168588.html>.
- [58] BITKOM E.V.; VDMA E.V.; ZVEI E.V. *Implementation Strategy Industrie 4.0: Report on the results of the Industrie 4.0 Platform* [online]. [cit. 2022-5-12].
Dostupný z: https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2022/05/Implementation_Strategy-Industrie-40-ENG.pdf.
- [59] WORKFLOW MANAGEMENT COALITION. *The Workflow Reference Model*. 1995.
- [60] WORKFLOW MANAGEMENT COALITION. *Terminology & Glossary*. 1999.
- [61] CARDA, Antonín; KUNSTOVÁ, Renata. *Workflow : nástroj manažera pro řízení podnikových procesů*. 2005. Management v informační společnosti. ISBN 8024706660.
- [62] WORKFLOW MANAGEMENT COALITION. *Workflow Management Coalition* [online]. [cit. 2022-5-16].
Dostupný z: <http://wfmc.org/>.
- [63] WORKFLOW MANAGEMENT COALITION. *Interface 1: Process Definition Interchange Q&A and Examples*. 1999.
- [64] FUTURE STRATEGIES. *BT Awards News Updates* [online]. [cit. 2022-6-5].
Dostupný z: <http://businesstransformationawards.org/news.html>.
- [65] PROF. ING. IVO VONDRÁK, CSc. *METODY BYZNYS MODELOVÁNÍ*. 2004. Dostupný také z: http://vondrak.cs.vsb.cz/download/Metody_byznys_modelovani.pdf.
- [66] AALST, Wil van der. The Application of Petri Nets to Workflow Management. *Journal of Circuits, Systems, and Computers*. 1998, roč. 8, s. 21–66. Dostupný také z: <http://dx.doi.org/10.1142/S0218126698000043>.
- [67] AALST, Wil van der; HEE, Kees van. *Workflow Management: Models, Methods, and Systems*. 1st. 2002. Cooperative Information Systems. ISBN 9780262011891; 0262011891.
- [68] AALST, Wil van der; ADAMS, Michael; HOFSTEDE, Arthur ter; RUSSELL, Nick. *Modern Business Process Automation: YAWL and its Support Environment*. První vyd. 2010. ISBN 9783642031205; 364203120X; 9783642031212; 3642031218; 3642031226; 9783642031229.
- [69] AALST, Wil van der; HOFSTEDE, Arthur ter. *YAWL: Yet Another Workflow Language (Revised version)*.
Dostupný také z: <https://yawlfoundation.github.io/assets/files/yawlrevtech.pdf>.

- [70] CENTER OF EXCELLENCE. *Business Process Management (BPM): Glossary*. 2009.
- [71] OBJECT MANAGEMENT GROUP. *Business Process Model and Notation (BPMN): Version 2.0.2*. 2013. Dostupný také z: <https://www.omg.org/spec/BPMN/2.0/>.
- [72] OBJECT MANAGEMENT GROUP. *Business Process Modeling Notation Specification: Version 1.1*. 2006. Dostupný také z: <https://www.omg.org/spec/BPMN/1.0/>.
- [73] OBJECT MANAGEMENT GROUP. *Decision Model and Notation: Version 1.4*. 2021. Dostupný také z: <https://www.omg.org/spec/DMN/>.
- [74] WORKFLOW MANAGEMENT COALITION. *Workflow Process Definition Interface – XML Process Definition Language*. 2012.
- [75] WORKFLOW PATTERNS INITIATIVE. *Standard Evaluations* [online]. [cit. 2022-6-10]. Dostupný z: <http://www.workflowpatterns.com/evaluations/standard/index.php>.
- [76] WORKFLOW MANAGEMENT COALITION. *Workflow Process Definition Interface – XML Process Definition Language*. 2002.
- [77] SYSTEMS, Sparx. *Create a Domain Model* [online]. [cit. 2022-7-12]. Dostupný z: https://sparxsystems.com/enterprise_architect_user_guide/15.2/model_domain.
- [78] MONSALVE, Mauricio. *Object-Oriented Software Development: The Domain Model* [online]. [cit. 2022-7-12]. Dostupný z: <https://homepage.cs.uiowa.edu/~tinelli/classes/022/Spring15/Notes/chap9.pdf>.
- [79] LAUDON, C. Kenneth; LAUDON, P. Jane. *Essentials of Management Information Systems*. Tenth. 2013. 504 s. ISBN 978-0-13-266855-2.
- [80] DOSTÁL, Petr; RAIS, Karel; SOJKA, Zdeněk. *Pokročilé metody manažerského rozhodování*. 2005. 168 s. ISBN 978-80-247-6320-0.
- [81] GANESH, K.; MOHAPATRA, Sanjay; ANBUUDAYASANKAR, P. S.; SIVAKUMAR, P. *Enterprise Resource Planning: Fundamentals of Design and Implementation*. 2014. 174 s. ISBN 978-3-319-05927-3.
- [82] POPKOVA, G. Elena; RAGULINA, V. Yulia; BOGOVIZ, V. Aleksei. *Industry 4.0: Industrial Revolution of the 21st Century*. 1st ed. 2019. 293 s. Studies in Systems, Decision and Control 169. ISBN 978-3-319-94309-1,978-3-319-94310-7.
- [83] ABDELMAJIED, FathyElsayed Youssef. *Industry 4.0 and Its Implications: Concept, Opportunities, and Future Directions* [online]. [cit. 2022-5-11]. Dostupný z: <https://www.intechopen.com/online-first/80514>.
- [84] INTERNATIONAL SOCIETY OF AUTOMATION. *View ISA Standards: A Member Benefit- ISA* [online]. [cit. 2022-4-25]. Dostupný z: <https://www.isa.org/standards-and-publications/isa-standards/member-access-to-standards/>.

- [85] BRANDL, Dennis. *Batch Control Using the ANSI/ISA-88 Standard*. 2004.
- [86] CENTER OF EXCELLENCE. *Business Process Management (BPM) Glossary*. 2009.
- [87] BPM-OFFENSIVE. *BPMN 2.0 Poster* [online]. [cit. 2022-6-10].
Dostupný z: http://www.bpmb.de/images/BPMN2_0_Poster_EN.pdf.

Rejstřík

- 3D tisk, 33
- additive manufacturing (aditivní výroba), 33
- administration shell, 70, 113
- administrative shell, 112
- akce procesu, 39
- aktivita, 72, 72, 75, 93
 - automatická, 72, 75
 - manuální, 72, 75
- aktuátor, 27, 27, 41, 50, 69
- and-join, 92, 103
- and-split, 92, 103, 111
- ansi/isa-88, 37, 50, 66, 113, 113, 117, 132
- ansi/isa-95, 49, 66, 113, 113, 114, 117, 132
- architektura ERP, 17
- augmented reality (rozšířená realita), 32
- autonomizace, 31
- batch, 38
- batch control, 42
- batch markup language, 49, 65
- batch process, 38
- big data, 24, 29, 33
- bluetooth, 20
- bluetooth low energy, 20
- business proces, 13, 15, 71, 72, 73, 75, 84, 93, 94, 95, 110
- business process model and notation, 72, 83, 111, 114, 117, 132
- business-to-manufacturing markup language, 64
- cloud, 22, 33
- cloudové nasazení, 18
- coloured petri nets, 89, 94, 111
- cyber-physical system, 26, 27, 29, 30, 32, 70, 113
- definice procesu, 71, 75, 76
- definiční jazyk, 73, 82, 107, 109, 111
- digital twin, 26, 29, 30, 31, 32, 32, 32
- digitální dvojče, 29
- diskrétní výroba, 38, 54
- dodatečné informace receptury, 44
- druhá průmyslová revoluce, 26
- dávka, 38
- dávková výroba, 38, 54
- dávkové řízení, 42
- dávkový proces, 38
- enterprise resource planning, 14, 51
- exekutivní jazyk, 73, 82, 107, 108, 109, 111, 117, 118
- formule, 43
- fyzický model, 39, 53, 68, 113
- fáze, 43, 48
- fáze procesu, 39
- hierarchie zařízení, 53, 68, 113
- hierarchie řízení, 50
- high-level petri nets, 88, 94
- hlavička receptury, 43
- hlavní receptura, 44, 47
- hodnotový řetězec, 35
- horizontální integrace, 35
- IEC 61512, 37, 68
- IEC 62264, 49, 68
- IEC 62890, 69
- industrial internet of things, 24, 24
- industry 4.0, 26, 29, 29, 34, 66, 112, 113, 117, 117
- industry 4.0 komponenta, 70, 112, 113, 117
- informační systém, 12
- infrastructure as a service, 23, 23, 24
- instance aktivity, 72, 75
- instance procesu, 72, 75
- internet of things, 19, 26, 30, 33, 70, 112
- jednotka, 41, 48, 48
- jednotková procedura, 43
- job order, 61
- kontinuální výroba, 38, 54
- koordinační řízení, 43

kybernetická bezpečnost, 33
 machine-to-machine, 24
 manufacturing execution system, 51
 manufacturing operation management, 51, 56
 manufacturing resource planning, 14
 material requirements planning, 13
 model definice práce, 61
 model dávkového procesu, 39
 model funkcí, 54
 model procedurálního řízení, 41
 model rozvrhu práce, 61
 model řídicích aktivit, 47
 modely objektů, 48, 58
 modul zařízení, 41, 48
 modul řízení, 41, 48
 místní receptura, 44, 47
 near field communication, 22
 obecná receptura, 44, 47
 on-premise nasazení, 17
 operace, 43
 operace procesu, 39
 or-join, 92, 103
 or-split, 92, 103, 111
 petriho síť, 72, 84, 94, 111
 platform as a service, 23, 24
 plánování a rozvrhování výroby, 48
 podnik, 39, 53
 podnikový informační systém, 12
 podnikový proces, 13
 požadavky na vybavení, 43
 pracovní centrum, 53
 pracovní jednotka, 53
 pracovní položka, 72
 procedura, 43
 procedura receptury, 43
 procedurální prvky, 42, 46, 48
 procedurální řízení, 42, 48
 procesní buňka, 41, 48
 programmable logic controller, 26
 programovatelný logický automat, 26
 provoz, 41, 53
 první průmyslová revoluce, 25
 průběžné inženýrství, 36
 průmysl 4.0, 30
 průmyslová revoluce, 25
 radio frequency identification, 20, 22
 receptura, 43
 reference architecture model for industrie 4.0, 66, 112, 113, 113, 132
 rfid tag, 20, 31
 senzor, 19, 24, 24, 27, 27, 29, 41, 50, 68, 69
 seznam prací, 72, 77
 simulace, 32
 smart factory, 34
 smart product, 26, 31, 68
 software as a service, 24
 sp88, 37
 sp95, 50
 správa procesů, 48
 správa produktových dat, 48
 správa receptur, 47
 standard operating procedure, 52
 subjekty vybavení, 41, 43, 45, 46
 systém řízení workflow, 77, 78, 110
 třetí průmyslová revoluce, 26
 unified modeling language, 72
 vertikální integrace, 34
 vyvolaná aplikace, 72, 75
 výrobní proces, 37
 výrobní závod, 39, 53
 wi-fi, 19
 wi-fi backscatter, 19
 work directive, 61, 63
 work master, 61, 63
 work request, 61
 work schedule, 61
 workflow, 61, 71, 72, 72, 73, 84, 95
 workflow engine, 75, 77, 78, 80
 workflow management, 71
 workflow management system, 71, 72, 72, 74
 workflow model, 61, 113, 114
 workflow reference model, 74, 78, 84, 110, 117, 132

workflow síť, [90](#), [94](#), [96](#), [111](#)
xml process definition language, [82](#)
zigbee, [20](#)
zigbee IP, [20](#)
základní řízení, [42](#), [48](#), [48](#)
účastník workflow, [72](#), [78](#)
čtvrtá průmyslová revoluce, [26](#)
řídící receptura, [45](#), [47](#)
řízení jednotek, [48](#)