



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



UNIVERSITAS
OSTRAVIENSIS

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

REPREZENTACE ZNALOSTÍ

URČENO PRO VZDĚLÁVÁNÍ V AKREDITOVANÝCH
STUDIJNÍCH PROGRAMECH

MARTIN ŽÁČEK

ČÍSLO OPERAČNÍHO PROGRAMU: CZ.1.07

NÁZEV OPERAČNÍHO PROGRAMU:

VZDĚLÁVÁNÍ PRO KONKURENCESCHOPNOST

OPATŘENÍ: 7.2

ČÍSLO OBLASTI PODPORY: 7.2.2

**INOVACE VÝUKY INFORMATICKÝCH PŘEDMĚTŮ VE
STUDIJNÍCH PROGRAMECH OSTRAVSKÉ UNIVERZITY**

REGISTRAČNÍ ČÍSLO PROJEKTU: CZ.1.07/2.2.00/28.0245

OSTRAVA 2013

Tento projekt je spolufinancován Evropským sociálním fondem a státním rozpočtem České republiky

Recenzent: Ing. Radek Sztefek

Název: Reprezentace znalostí
Autor: Martin Žáček
Vydání: první, 2013
Počet stran: 190

Jazyková korektura nebyla provedena, za jazykovou stránku odpovídá autor.

© Martin Žáček
© Ostravská univerzita v Ostravě

OBSAH

1	REPREZENTACE ZNALOSTÍ	9
1.1	POČÁTKY A ROZVOJ REPREZENTACE DEKLARATIVNÍCH TVRZENÍ	10
1.1.1	<i>Znalosti a jejich formální reprezentace</i>	<i>10</i>
1.1.2	<i>Znalosti, datové modely a ontologie</i>	<i>13</i>
1.1.3	<i>Reprezentace znalostí a znalostní inženýrství</i>	<i>14</i>
1.2	KONCEPTOVĚ ORIENTOVANÉ MODELOVÁNÍ JAKO PODMÍNKA MOŽNÉ REPREZENTACE ZNALOSTÍ	17
1.2.1	<i>Konceptově orientované paradigma</i>	<i>17</i>
1.2.2	<i>Principy konceptově orientovaných jazyků reprezentace znalostí</i>	<i>18</i>
1.2.3	<i>Konceptově orientované jazyky a deskripční logika</i>	<i>20</i>
2	ASOCIATIVNÍ SÍTĚ.....	22
2.1	OMEZENÍ ASOCIATIVNÍCH SÍTÍ.....	25
2.2	JAZYK.....	25
2.3	ODVOZOVACÍ PRINCIPY	28
2.4	SÉMANTICKÁ KOREKTNOST A ÚPLNOST.....	30
2.4.1	<i>Interpretace asociativní sítě a její struktura</i>	<i>30</i>
2.4.2	<i>Pravdivost báзовých vektorů a sítí.....</i>	<i>30</i>
2.4.3	<i>Splnitelnost a platnost univerzálních sítí</i>	<i>31</i>
2.4.4	<i>Negace v asociativní síti.....</i>	<i>32</i>
2.4.5	<i>Sémantická korektnost odvozování v asociativních sítích.....</i>	<i>33</i>
3	SOWOVY KONCEPTUÁLNÍ GRAFY	35
3.1	ZNALOSTNÍ BÁZE.....	39
3.1.1	<i>Ontologie jako základní princip konceptuálních grafů.....</i>	<i>39</i>
3.1.2	<i>Relace subtypu.....</i>	<i>40</i>
3.1.3	<i>Entita a absurdita.....</i>	<i>41</i>
3.1.4	<i>Lambda výrazy pro definování typů konceptů a vztahů</i>	<i>41</i>
3.2	ONTOLOGIE V KONCEPTUÁLNÍCH GRAFECH	42
3.2.1	<i>Typy konceptů.....</i>	<i>42</i>
3.2.2	<i>Typy vztahů.....</i>	<i>43</i>
3.2.3	<i>Referenty.....</i>	<i>43</i>
3.2.4	<i>Zahnížděné grafy.....</i>	<i>45</i>
3.2.5	<i>Kontext</i>	<i>45</i>

Reprezentace znalostí

3.3	KONCEPTUÁLNÍ GRAF A LOGIKA	45
3.3.1	<i>Konjunkce v konceptuálním grafu</i>	46
3.3.2	<i>Disjunkce v konceptuálním grafu</i>	46
4	DESKRIPČNÍ LOGIKA.....	48
4.1	KONCEPTOVĚ ORIENTOVANÉ JAZYKY A DESKRIPČNÍ LOGIKA.....	48
4.1.1	<i>Asociativních sítě, rámce a konceptové jazyky.....</i>	49
4.1.2	<i>Deskripční logika a logika prvního řádu.....</i>	49
4.1.3	<i>Některé implementace deskripční logiky</i>	50
4.2	JAZYK DESKRIPČNÍ LOGIKY.....	51
4.2.1	<i>Jazyky LDL a LDL1 deskripční logiky.....</i>	51
4.2.2	<i>Syntax jazyka deskripční logiky.....</i>	51
4.2.3	<i>Sémantika jazyka deskripční logiky.....</i>	53
4.2.4	<i>Subsumpce konceptů a taxonomie</i>	54
4.3	ZNALOSTNÍ BÁZE V DESKRIPČNÍ LOGICE	55
4.3.1	<i>Terminologické znalostní systémy</i>	55
4.3.2	<i>Znalostní báze.....</i>	56
4.3.3	<i>TBoxy znalostních bází</i>	56
4.3.4	<i>ABoxy znalostních bází.....</i>	57
4.4	ROZHODOVÁNÍ V DESKRIPČNÍ LOGICE	60
4.4.1	<i>Splnitelnost konceptu a znalostní báze</i>	60
4.4.2	<i>Rozhodování splnitelnosti a subsumpce konceptů.....</i>	61
4.5	LOGICKÝ DŮSLEDEK ZNALOSTNÍ BÁZE	62
4.5.1	<i>Sémantické tablo v deskripční logice DL.....</i>	63
4.6	ČÁSTEČNĚ ROZHODNUTELNÁ DESKRIPČNÍ LOGIKA DL1	66
4.6.1	<i>Očekávané vlastnosti sémantiky jazyka LDL1.....</i>	66
4.6.2	<i>Modifikovaná sémantika $\forall R.C$ v deskripční logice DL1.....</i>	67
4.6.3	<i>Přepisovací pravidla negací konceptů $\exists R.C$ a $\forall R.C$ v DL1</i>	69
4.6.4	<i>Tablový rozhodovací algoritmus v DL1</i>	70
4.6.5	<i>Tablové rozhodování splnitelnosti znalostní báze v DL1</i>	72
4.6.6	<i>Tablové důkazy logické platnosti a logického důsledku v LDL1 ..</i>	73
4.6.7	<i>Strukturální subsumpce</i>	76
4.6.8	<i>Odvozování z ABoxu.....</i>	77
4.6.9	<i>Rozšíření znalostní báze podle spouštěcích pravidel.....</i>	77
5	MODÁLNÍ K-LOGIKA.....	79
5.1	SYNTAX JAZYKA LK.....	79

Reprezentace znalostí

5.2	SÉMANTIKA JAZYKA LK.....	80
5.2.1	<i>Sémantika možných světů.....</i>	80
5.2.2	<i>Pravdivost a splnitelnost formulí modální logiky K.....</i>	81
5.3	FORMÁLNÍ SYSTÉM VÝROKOVÉ MODÁLNÍ LOGIKY	83
5.3.1	<i>Axiomy modálních logik.....</i>	83
5.3.2	<i>Odvozovací pravidla modální logiky.....</i>	85
5.3.3	<i>Sémantická korektnost a úplnost modální logiky K</i>	86
5.4	SÉMANTICKÉ TABLO MODÁLNÍ K-LOGIKY	87
5.4.1	<i>Indexování možných světů a tablová odvozovací pravidla</i>	87
5.4.2	<i>Příklady modálních tablových důkazů</i>	88
6	NEMONOTÓNŇÍ LOGIKY	94
6.1	NEMONOTÓNŇÍ BUDOVÁNÍ TEORIE NAD ZNALOSTNÍ BÁZÍ	94
6.1.1	<i>Monotónnost budování teorie v systémech logiky prvního řádu...95</i>	
6.1.2	<i>Charakteristické vlastnosti monotónních formálních systémů.....97</i>	
6.1.3	<i>Nemonotónnost teorií.....99</i>	
6.2	VÝVOJ NEMONOTÓNŇÍCH PŘÍSTUPŮ	100
6.2.1	<i>Dva způsoby nemonotónní modifikace logiky prvního řádu</i>	100
6.2.2	<i>Reiterova modifikace default odvozovacími pravidly</i>	100
6.2.3	<i>Modifikace rozšířením jazyka v autoepistemické logice</i>	103
6.2.4	<i>Další možnosti modifikace rozšířením jazyka</i>	104
6.2.5	<i>Nemonotónnost založená na dědičnosti a specifičnosti předpokladů.....106</i>	
6.3	FORMALIZACE TEORIE V NEMONOTÓNŇÍCH SYSTÉMECH.....	108
6.3.1	<i>Význam konsistence v nemonotónních systémech</i>	108
6.3.2	<i>Konsistence z formálního hlediska.....108</i>	
6.3.3	<i>Budování teorií v nemonotónních systémech</i>	111
6.3.4	<i>Formalizace default teorie</i>	112
6.3.5	<i>Požadované vlastnosti nemonotónních teorií.....113</i>	
6.3.6	<i>Reiterova default-negace.....114</i>	
6.3.7	<i>Extenze default negace podle Reitera</i>	115
6.4	DEFAULT DŮKAZY JAKO ARGUMENTY	116
7	SÉMANTICKÝ WEB	120
7.1	UNIVERZÁLNÍ INFORMAČNÍ PROSTOR WEBU	120
7.2	OD SYNTAKTICKÉHO K SÉMANTICKÉMU WEBU.....	121
7.2.1	<i>Konsorcium W3C a jeho cíle.....121</i>	

Reprezentace znalostí

7.2.2	<i>Web jako aplikace nad internetem.....</i>	122
7.2.3	<i>Idea sémantického webu.....</i>	122
7.2.4	<i>Vývoj popisu významu dat</i>	124
7.2.5	<i>Strojově srozumitelné informace, všeobecně sdílená terminologie 124</i>	
7.2.6	<i>Ontologie</i>	126
8	FORMÁLNÍ ONTOLOGIE	129
8.1	ONTOLOGIE A VIZE SÉMANTICKÉHO WEBU	129
8.1.1	<i>Strojová čitelnost metadat webových dokumentů.....</i>	130
8.2	ZNALOSTI A ONTOLOGIE	131
8.2.1	<i>Problémová nezávislost znalostí.....</i>	131
8.2.2	<i>Konceptuální úroveň modelování zájmové domény</i>	131
8.2.3	<i>Zamýšlená interpretace a možné světy</i>	132
8.2.4	<i>Od konceptualizace k implementaci báze znalostí.....</i>	133
8.2.5	<i>Ontologická formalizace.....</i>	135
8.3	ONTOLOGICKÁ KONCEPTUALIZACE	135
8.3.1	<i>Paradigma, metamodel a jazyk.....</i>	135
8.3.2	<i>Od datového modelu ke konceptuálnímu modelu</i>	137
8.3.3	<i>Syntax jazyka konceptuálního modelování, (meta)model.....</i>	137
8.3.4	<i>Vlastnosti jazyka konceptuálního modelování domény</i>	138
8.3.5	<i>Ontologie, metamodel a konceptualizace</i>	139
8.4	VÝVOJ PŘÍSTUPŮ FORMÁLNÍ ONTOLOGIE	142
8.4.1	<i>Předchozí vývoj.....</i>	142
8.4.2	<i>Reprezentace struktury domény třídami, stromy a svazy</i>	143
8.5	SOUČASNÉ POJETÍ A PŘÍSTUPY FORMÁLNÍ ONTOLOGIE.....	144
8.5.1	<i>Trojí význam pojmu ontologie</i>	144
8.5.2	<i>Typy a hierarchie ontologií</i>	144
8.5.3	<i>Přístupy k doménové formální ontologii.....</i>	145
8.6	FORMÁLNÍ KONCEPTOVÁ ANALÝZA A FORMÁLNÍ ONTOLOGIE	147
8.6.1	<i>Formální konceptová analýza dat</i>	147
8.6.2	<i>Konceptový svaz.....</i>	147
8.6.3	<i>Formální koncept v daném kontextu.....</i>	148
8.6.4	<i>Uzávěrové vlastnosti konceptů</i>	150
8.6.5	<i>Konceptové grafy FCA</i>	151
8.6.6	<i>Dolování znalostí z konceptových svazů.....</i>	153

Reprezentace znalostí

8.7	ONTOLOGIE VYŠŠÍCH ÚROVNÍ.....	154
8.7.1	<i>Obecná ontologie vyšší úrovně Russela a Norviga.....</i>	<i>154</i>
8.7.2	<i>Obecný ontologický jazyk GOL (General Ontological Language)...</i>	<i>155</i>
8.7.3	<i>Formát výměny znalostí KIF (Knowledge Interchange Format)</i>	<i>155</i>
8.7.4	<i>Sowův návrh ontologie nejvyšší úrovně.....</i>	<i>155</i>
8.7.5	<i>CYC ontologie.....</i>	<i>157</i>
8.7.6	<i>WordNet.....</i>	<i>158</i>
8.7.7	<i>SUMO ontologie.....</i>	<i>159</i>
8.8	NĚKTERÉ SYSTÉMY NA PODPORU TVORBY ONTOLOGIÍ	159
8.8.1	<i>OntoClean.....</i>	<i>159</i>
8.8.2	<i>Ontolingua.....</i>	<i>160</i>
8.8.3	<i>ONIONS, OnTopic.....</i>	<i>161</i>
8.8.4	<i>OCML.....</i>	<i>161</i>
8.8.5	<i>OKBC a XOL.....</i>	<i>162</i>
9	MODELÝ A JAZYKY PRO SÉMANTICKÝ WEB	164
9.1	OBEČNÝ DATOVÝ MODEL RDF	164
9.1.1	<i>Vlastnosti (atributy) zdrojů.....</i>	<i>165</i>
9.1.2	<i>Idea a formy reprezentace v RDF.....</i>	<i>165</i>
9.2	ABSTRAKTNÍ SYNTAX MODELU RDF.....	166
9.2.1	<i>Uniformní identifikace zdrojů.....</i>	<i>166</i>
9.2.2	<i>RDF - trojice a graf.....</i>	<i>167</i>
9.2.3	<i>Způsoby reprezentace RDF grafu.....</i>	<i>168</i>
9.3	SYSTÉM KONCEPTŮ V RDF MODELU	170
9.4	SPECIFIKACE TŘÍD A VLASTNOSTÍ V RDFS	170
9.5	FORMÁLNÍ SÉMANTIKA RDF.....	171
9.5.1	<i>RDF jazyk prvního řádu.....</i>	<i>171</i>
9.5.2	<i>Interpretace slovníku RDF(S).....</i>	<i>172</i>
9.5.3	<i>Prázdné uzly jako existenční proměnné.....</i>	<i>174</i>
9.5.4	<i>Interpretace bazových grafů.....</i>	<i>175</i>
9.5.5	<i>Modely a logické důsledky RDF grafů.....</i>	<i>175</i>
9.6	ONTOLOGICKÝ JAZYK WEBU OWL	176
9.6.1	<i>RDF a OWL.....</i>	<i>176</i>
9.6.2	<i>Historie vývoje OWL.....</i>	<i>177</i>
9.6.3	<i>Podjazyky OWL.....</i>	<i>178</i>

Reprezentace znalostí

9.6.4	<i>Reprezentace ontologie v jazyce OWL</i>	178
9.6.5	<i>Ontologie v OWL jako XML dokument systému Protégé</i>	179
9.7	ONTOLOGICKÉ JAZYKY A DATOVÉ MODELOVÁNÍ.....	181
9.7.1	<i>Datový model a ontologie</i>	181
9.7.2	<i>Prostředky RDF(S), OWL a E-R datového modelování</i>	181
9.8	ZHODNOCENÍ PŘÍNOSU RDF/OWL MODELOVÁNÍ.....	181
9.8.1	<i>RDF datový model a jazyk OWL</i>	181
9.8.2	<i>Typické vlastnosti modelování v RDF/OWL</i>	181
9.8.3	<i>Výhody a nevýhody modelování RDF/OWL</i>	182
9.8.4	<i>Ontologie, XML Schema a objektový princip UML</i>	183
	LITERATURA	185
	VYSVĚTLIVKY K POUŽÍVANÝM SYMBOLŮM	190

1 Reprezentace znalostí

V této kapitole se dozvíte:

- počátky a rozvoj reprezentace deklarativních tvrzení,
- co je to sémantika a ontologie,
- co jsou informace a co jsou data,
- směry znalostního inženýrství,
- o konceptově orientovaném modelování,
- jaké jsou konceptově orientované jazyky a deskripční logika.

Po jejím prostudování byste měli být schopni:

- definovat klíčové role reprezentace znalostí,
- rozlišit směry umělé inteligence na modelovaný svět a reprezentaci,
- definovat, co je to znalost, datový model a ontologie,
- vytvářet znalostní bázi,
- tvořit konceptuální model prostřednictvím určitého paradigmatu,
- definovat koncept, konceptové paradigma, reprezentaci a jazyk.

Klíčová slova této kapitoly:

Znalost, reprezentace znalostí, ontologie, modelovaný svět, datový model, metadata, znalostní báze, paradigma, extenze, intenze, deskripční logika.

Doba potřebná ke studiu: 7 hodin

Průvodce studiem

Studium této kapitoly Vás zavede do problematiky reprezentace znalostí. V kapitole se zaměříme na počátky a rozvoj deklarativních tvrzení. Vysvětlíme si ty nezákladnější pojmy, se kterými se můžete setkat, a vysvětlíme si, co to jsou znalosti.

Proto, prosím, věnujte této kapitole co nejvíce času, protože pak snáze pochopíte problematiku reprezentace znalostí. Následně si dejte hodinu pauzu a zkuste se prokousat touto kapitolou ještě jednou.

Znalosti o modelovaném světě musejí být reprezentovány takovou formou, aby z nich bylo možno odvozovat další znalosti. Aby systém mohl být považován za uměle inteligentní, musí disponovat strojově využitelnými znalostmi, aby bylo možno pracovat se znalostmi v podobě vhodné pro počítačovou implementaci. Proto je potřeba zavést určitý formalismus – reprezentační jazyk



Reprezentace znalostí

schopný odrážet vztahy mezi znalostmi o modelovaném světě uložených v lidských myslích a formálními prostředky zapsaných znalostí.

Inspirací tématu této disertační práce byla skutečnost, že predikátová logika prvního řádu, včetně svých lehce zjednodušených variant, jako např. deskripční logika, představuje zpravidla pro amatérského uživatele nezvládnutelný formální prostředek reprezentace. Na druhé straně se ale modelování světa formálními prostředky stává stále nutnějším požadavkem vzhledem ke vzrůstající snaze o převedení inteligentních lidských činností na počítače.

Modelování v rámci umělé inteligence probíhá nejméně na dvou úrovních abstrakce, a to na úrovni konceptualizace a na úrovni implementační. Na první z nich, tedy na vyšší úrovni abstrakce, probíhá konceptualizace, jejímž výstupem je konceptuální model v jazyce. Tento jazyk má sice přísná formální pravidla, ale zároveň disponuje takovými jazykovými prostředky, aby jim mohl porozumět i laický uživatel a podílet se pak na tvorbě konceptuálního modelu vymezeného referenčního systému. Druhá úroveň používá pro implementaci modelu zpravidla jazyk, který zpravidla bývá modifikací jazyka logiky prvního řádu a vyžaduje tak hlubší zvládnutí formálních jazykových prostředků.

Jedním z prostředků konceptuálního modelování je model a jazyk RDF, který byl původně určen pro modelování nad webovými zdroji, ale získává stále více obliby i v takových oblastech, které s webovými zdroji přímo nesouvisí. Zásahu na tom má především konceptová orientace modelu, umožňující následnou implementaci v konceptovém jazyku OWL, který vychází z deskripční logiky a který je modifikací logiky prvního řádu.

Jazyk RDF, vzhledem ke svému původnímu určení, nedisponuje prostředky k vyjádření univerzálních nebo existenčních tvrzení a k vyjádření negace tvrzení. S tím souvisí i nemožnost provádění formální dedukce v RDF, tj. již na konceptuální úrovni.

1.1 Počátky a rozvoj reprezentace deklarativních tvrzení

Počátkem druhé poloviny 20. století vznikl významný mezník ve vědeckém směru informatiky, kdy se tento vědecký směr rozdělil na dva disjunktní směry – teorie programování a umělá inteligence. Cílem umělé inteligence je vytváření strojů a systémů, který by modelovaly inteligentní činnosti lidských bytostí s využitím softwarových agentů, kteří jsou naprogramováni tak, aby jejich činnost byla chápána jako projev inteligence. Modelování inteligentních činností je realizováno pomocí dvou částí, ne zcela disjunktních částí, a to epistemologické a heuristické části. Epistemologie jako oblast filosofie, která se zabývá otázkami „co je to znalost?“ a „jak jsou znalosti získané?“. Naproti tomu heuristika je zaměřena spíše na hledání vhodných řešení konkrétních problémů za daných podmínek. Zde řešené problémy zpravidla spadají do oblasti epistemologie.

1.1.1 Znalosti a jejich formální reprezentace

Ne všechny informace, které jsou potřeba ke zpracování textu, jsou zakódovány ve struktuře jazyka. Často jsou ke správnému porozumění obsahu textu nutné znalosti o světě – buď obecné (např. že ptáci umí létat nebo že k

Reprezentace znalostí

otevření zamčených dveří je zapotřebí klíč), nebo i velmi specifické či odborné, které lze u čtenáře daného textu předpokládat (např. v matematickém časopise, že sudé číslo větší než 2 nemůže být prvočíslem). Jak se ukazuje, největší výzvou v tomto směru není tyto znalosti nashromáždit, ale vhodně je reprezentovat a strukturovat, efektivně v nich vyhledávat a používat je k vyvozování znalostí nových. Tyto cíle ve své podstatě odpovídají úkolu zkonstruovat umělou inteligenci, která jistě patří mezi největší a nejzajímavější témata moderní vědy.

Reprezentace znalostí je jednou z nejdůležitějších a zároveň nejobtížnějších oblastí umělé inteligence. Klád se důraz na to, co má být znalostní aplikací řešeno a až pak, jak to má být řešeno.

Znalost se tak stává klíčovým prvkem znalostně orientovaných aplikací, např. expertních systémů. Expertní systém lze chápat jako inteligentní počítačový program, který užívá znalosti a inferenční procedury k řešení problému, které jsou natolik obtížné, že pro své řešení vyžadují významnou lidskou expertízu.

Na tvůrcích znalostního systému tedy závisí volba jak zvolit vhodný způsob zviditelnění (zachycení, reprezentace) znalostí, tzv. reprezentační schéma. Takové schéma si lze představit jako soubor pravidel a postupů, které mají být dodržovány pro zachycení znalostí. Schéma by mělo být pro člověka dostatečně srozumitelné, univerzální, resp. schopné zachytit širokou škálu znalostí – od obecnějších po specifičtější. Dále by mělo podporovat snadnou modularitu (rozšiřování) znalostí.

Klíčová role reprezentace znalostí

Jak bylo výše zmíněno, tak v rámci epistemologického přístupu jsou znalosti o modelovaném světě (doméně) reprezentovány takovou formou, aby z nich bylo možno „inteligentně“ odvodit další znalosti, resp. řešení problémů. Aby mohl být formální systém inteligentní, je potřeba, aby disponoval strojově využitelnými znalostmi. Jedinou cestou, jak poskytnout systému takové znalosti, je vtělit je do určitých formálních struktur a to je problém reprezentace znalostí.

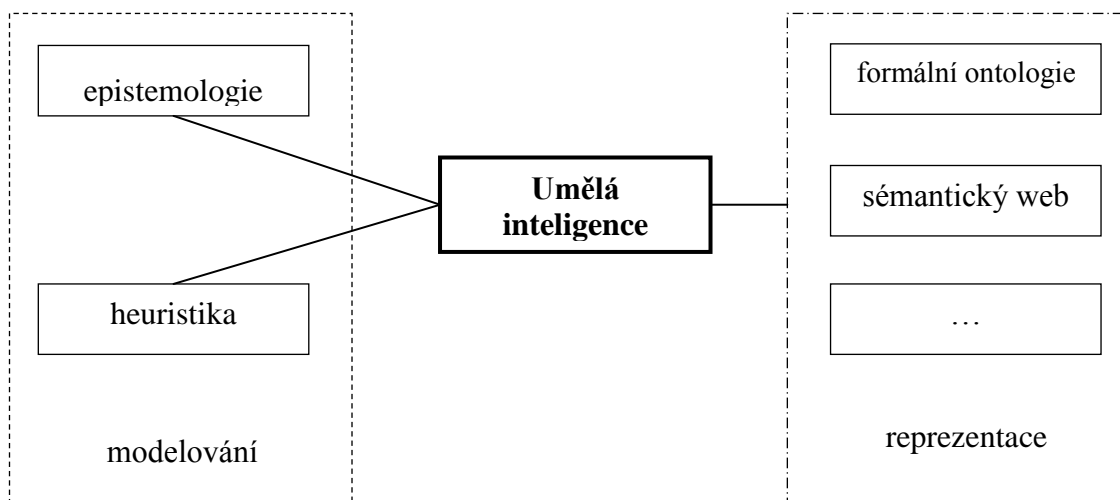
Ať už je zvolen jakýkoliv formalismus reprezentace znalostí, vždy musí mít sémantiku vycházející z pojmu pravdivosti, má-li reprezentovat skutečné znalosti o modelovaném světě reprezentované zvolenými formálními prostředky. Bez systematické závislosti mezi formou reprezentace a významem znalostí výše formulovaný cíl nelze realizovat.

Musí tedy existovat korespondence mezi výrazy formálního jazyka a modelovaným světem. Znalost zároveň neznamená, že svět je takový, jaký je právě nyní a nic jiného nemůže nastat. Nelze se tedy obejít bez toho, že by každá modelovaná skutečnost (svět) byla uvažována v jisté množině možných světů, zachycující právě tuto možnou variabilitu modelovaného světa.



Reprezentace znalostí

Znalosti o znalostech – ontologie



Obrázek 1.1: Směry umělé inteligence v oblasti reprezentace znalostí.

Jak je uvedeno na obrázku (Obrázek 1.1), pojmy formální ontologie a sémantický web představují zhruba v posledním desetiletí významnou část umělé inteligence. Jedná se o takový způsob reprezentace znalostí, aby se v něm dokázal na základě významů (sémantiky) znalostí „vyznat“ softwarový agent. K tomu je ale třeba opatřit znalosti novými znalostmi o nich samých – meta- znalostmi.

Pojmy, pomocí kterých se meta- znalosti reprezentují, jsou shromažďovány podle příslušnosti k významovým doménám do terminologických slovníků – ontologií. Inspirací a zároveň cílem tohoto nově orientovaného úsilí v reprezentaci znalostí se stala idea sémantického webu, v němž by se softwarový agent orientoval natolik, že by dokázal na základě významově vymezeného dotazu nalézt nejen odpověď, ale též další souvislosti a odvozené důsledky, které si tazatel ani nemusí umět představit.



Sémantika a ontologie

Oba pojmy, sémantiky jazyka a ontologie, se poprvé objevily již za doby řecké filosofie, kde sémantika jazyka je definována jako oblast filosofie zabývající se studiem povahy významu prvků jazyka, který je určen syntaktickými pravidly v rámci své gramatiky.

Naproti tomu ontologie v řecké filozofii zaujímal směr, který se zabýval identifikací významu nejobecnějších termínů pojmenovávajících věci a jevy, které reálně (konkrétně nebo abstraktně) existují.

V podstatě filosofická ontologie nám říká, co existuje, kdežto smyslem sémantiky reprezentujícího jazyka nám sděluje jak výstižně a co nejpřesněji popsat povahu toho, co existuje.

V případě formální ontologie jde o to, jak v rámci formálně reprezentované domény popsat formálními prostředky to, co v této doméně existuje.

Reprezentace znalostí

1.1.2 Znalosti, datové modely a ontologie

Někteří lidé si myslí, že by bylo nejlepší reprezentovat znalosti stejně tak, jak jsou reprezentovány v lidském mozku (lidských myslích), nebo reprezentovat znalosti ve formě lidského jazyka. Bohužel zatím nevíme, jak jsou znalosti reprezentovány v lidských myslích, nebo jak manipulovat s lidskými jazyky stejně jako lidskou myslí.

Abychom pochopili tak složitý lidský orgán, jako je mozek, a jeho inteligentní projev, musíme se nejdříve seznámit se základními pojmy, kterými jsou již několikrát zmiňované znalosti, následně datové modely a ontologie [10].

Znalosti

Pokud budeme v epistemologické oblasti umělé inteligenci, tak se jedná o formální manipulaci se znalostmi. Základní slovem reprezentací znalostí je znalost neboli znalosti, které vycházejí z informací a ty zase vycházejí z dat.

Pojmem dat se obecně rozumí všechny znakové řetězce vstupující do výpočetního procesu. Do výpočetního procesu však zřídka vstupují data, která nemají žádný význam, která tedy nejsou nějakým způsobem interpretována. Interpretací dat se přitom rozumí smysluplné přiřazení významu (sémantiky) datům.

Informaci tvoří data spolu se svou interpretací, pojem informace je tedy neoddělitelný od významu dat, která jsou jejími nositeli.

Znalost je navíc informace, která je použitelná a začlenitelná, resp. odvoditelná v souvislosti s jinými informacemi.

Abychom si více přiblížili rozdíl mezi daty a informacemi, podívejme se na tabulku (Tabulka 1.1). Levá tabulka obsahuje nic neříkající čísla a řetězce. Pokud ale tabulce doplníme záhlaví (pravá tabulka), čili číslům a řetězcům přidělíme význam v záhlavích sloupců, potom interpretovaná data se stávají informacemi, se kterými lze již pracovat např. jako s tabulkou údajů o osobách v relační databázi. Jestliže je navíc těmto informacím přiřazena další interpretace pomocí metadat, která data s daným významem přesněji specifikují a začleňují do širších souvislostí – zde např. tvrzením, že se jedná o údaje o zaměstnancích jistého podniku, kterým bude na míru ušit podnikový úbor – jedná se již o znalosti.

Tabulka 1.1: Rozdíl mezi daty a informacemi

<i>data</i>			<i>informace</i>		
Alena	164	68	Jméno	Výška	Váha
Miroslav	170	77	Alena	164	68
			Miroslav	170	77



Reprezentace znalostí

Tatáž data je možno interpretovat různými způsoby. Zpravidla jsou ale data již vytvářena s představou, jaká bude jejich interpretace. Jedná se o „zamýšlenou interpretaci“ dat.

K vytváření znalostí z dat je známá celá řada způsobů kódování těchto dat a jejich interpretace. V problematice reprezentace znalostí se jedná o volbu vhodného formalismu, který svými prostředky dokáže uchopit interpretovaná data i s jejich metadaty pro další automatizované zpracování.

Počítače byly v počátcích svého nasazení využívány především k realizaci pracných výpočtových postupů. Takový druh modelování lidské intelektuální činnosti vychází ze znalostí určitých pracovních postupů, tedy z určitých procedurálních znalostí.

Deklarativní znalosti o objektech (entitách) modelované domény spočívají v konstatování jejich stavů, vlastností nebo vzájemných vztahů. Počítačovou reprezentací deklarativních znalostí jsou pak spolu s odvozovacími pravidly vhodně strukturovaná data. Pozornost se tím obrací k tvorbě znalostníchází a datovému modelování.

1.1.3 Reprezentace znalostí a znalostní inženýrství

Abychom mohli pracovat se znalostmi v podobě vhodné pro počítačovou implementaci, je potřeba zavést určitý formalismus. Jedná se o zavedení reprezentačního jazyka, který je schopný odrážet vztahy mezi znalostmi o světě uložených v lidských myslích. K tomu aby mohl naprogramovaný agent řešit konkrétní úlohy a odvozovat další znalosti, je zapotřebí mít kromě reprezentačního jazyka taky formální prostředky zapsaných znalostí.

Vytvářením znalostníchází jako souborů formálně zapsaných znalostí o určité vymezené zájmové doméně neboli modelovaném světě se zabývá znalostní inženýrství, které za účelem získání znalostí v určitém oboru zpravidla využívá pohovorů s experty. Tomuto procesu se říká znalostní akvizice. Tyto znalosti jsou vkládány do znalostníchází ve formě formulí zvoleného reprezentačního jazyka.

Než znalostní inženýr dospěje k této výsledné formě znalostí formalizovaných pomocí formálního jazyka, měl by splňovat základní podmínky kompetentnosti přístupu k dané problémové doméně i jazykovým prostředkům zvolené formalizace a měl by si ujasnit řadu výchozích předpokladů. Těmi jsou především

- vymezení způsobu vidění modelovaného světa – paradigma,
- stanovení pojetí pravdivosti deklarativních tvrzení znalostníáže,
- vymezení pojetí principů odvozování dalších znalostí ze znalostníáže,
- specifikace formálního systému a jazyka v závislosti na předcházejících uvedených předpokladech.

Na základě těchto předpokladů a podmínek znalostní inženýr sestojí konceptuální datový model znalostníáže zájmové domény, ke které se znalostníáže vztahuje a která je jejím referenčním systémem. Konceptuální

Reprezentace znalostí

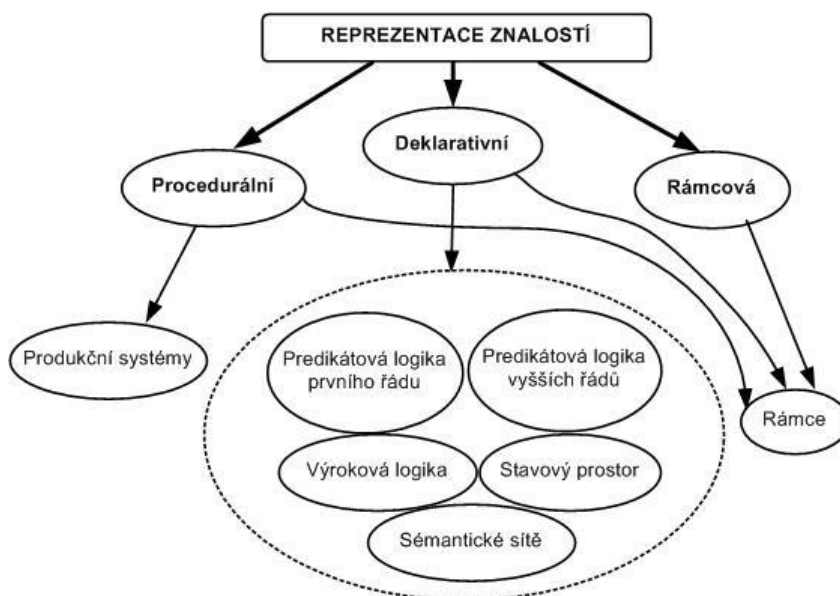
modelování dané domény úzce souvisí s pojmem formální ontologie, známým (bez přívlastku "formální") především z tradiční filosofie jako teorie bytí.

Zatímco formální logika jako jazyk reprezentace se zabývá formálně logickými strukturami tvrzení, jako jsou pravdivost, platnost, konsistence, a to nezávisle na pravdomluvnosti autorů těchto tvrzení, formální ontologie se zabývá formálně ontologickými strukturami, které bývají předmětem teorie částí, celků, typů a jejich instancí, identity, závislosti, jedinečnosti, tj. formálními aspekty příslušnosti objektů bez ohledu na jejich partikulární povahu. Tyto ontologické struktury umožňují formální zachycení hierarchií pojmů – konceptů, na nichž jsou znalosti postaveny.

Vytváření znalostníchází

Znalosti jsou vkládány do znalostní báze

- ve formě formulí zvoleného reprezentačního jazyka,
- nebo ve formě určitých komplexů grafových symbolů, které vyjadřují zamýšlený význam formalizovaných tvrzení způsobem srozumitelným i laikům.



Obrázek 1.2: Reprezentace znalostí

Výběr formálního jazyka má pro reprezentaci znalostí zásadní význam (Obrázek 1.2). Základem, z něhož by měly formálně reprezentační prostředky vycházet, je logika prvního řádu, a to pro její vysokou expresivitu a celou řadu již vyvinutých prostředků formální dedukce. Jako perspektivní se ukázala i deskripční logika jako její poněkud zjednodušující modifikace, zachovávající korektnost a poněkud zlepšující jasnost a srozumitelnost. Je zřejmé, že samotný obsah znalostí o zájmové doméně je třeba modelově zjednodušovat při zachování jeho korektnosti.

Reprezentace znalostí

Podle Russela a Norviga [viz literatura] je třeba si při zapisování tvrzení formou formulí znalostní báze vždy položit tyto otázky:

- Proč je to pravda? Nebylo by lepší zapsat místo tohoto tvrzení podmínky, kdy je to pravda?
- Jak obecně je to aplikovatelné? Nebylo by možné zobecnění pro širší třídu objektů?
- Potřebuji skutečně k zápisu tohoto tvrzení nové predikáty?
- V jakém vztahu je uvedené tvrzení o třídě objektů k jiným třídám objektů již popsanych ve znalostí bázi?
- Jak je to s podtřídou a nadtřídou popisovaných objektů?

Znalostní inženýr musí proto být

- do hloubky seznámen s doménou (její ontologií), které se znalosti mají týkat, aby mohl reprezentovat důležité vlastnosti a vztahy v této doméně,
- musí znát dobře možnosti reprezentačního jazyka,
- musí mít představu i o možnostech implementace odvozovacích procedur, aby byla zajištěna možnost odpovědí v reálném čase na dotazy kladené znalostní bázi.

K tomu všemu je třeba držet se metodologie, inspirované, v následujících krocích:

1. Rozhodnutí, co je předmětem zájmu – o které objekty a vzájemné vztahy v rámci zájmové domény se jedná, které naopak mohou být ignorovány.
2. Na základě stanoveného paradigmatu, tj. způsobu vidění modelovaného světa, vymezení základních primitivů modelování (např. koncepty, jejich vlastnosti a vzájemné vztahy).
3. Stanovení slovníku potřebných predikátů, funkcí a konstant odpovídajících vymezeným primitivním prvkům modelování v dané doméně.
4. Uvedení do souladu stanoveného slovníku s termíny hierarchicky uspořádaného slovníku dané domény (pokud již existuje), který představuje ontologii dané domény, tj. určitou teorii existence této domény.
5. Zapsání znalostí o doméně spolu se znalostmi obsaženými v příslušné ontologii formulami zvoleného jazyka do znalostní báze.

Logické formule jsou pak speciálními axiomy, vypovídajícími základní skutečnosti o doméně, které mají dvojí účel:

- zpřesnění jejich interpretace, a to pomocí příslušné ontologie uvažované domény,
- možnost odvozování dalších tvrzení o objektech dané domény.

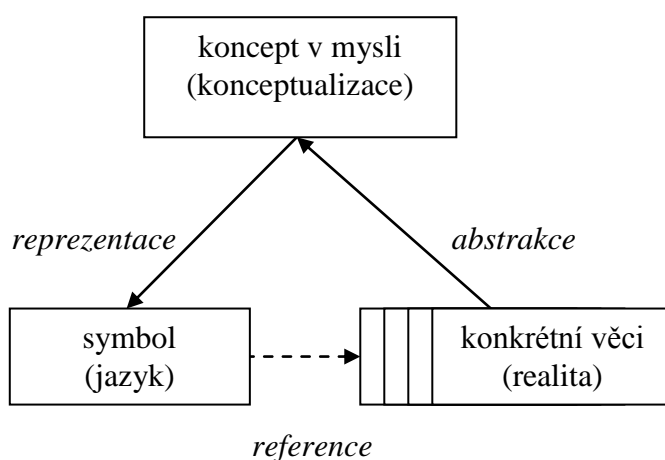
Reprezentace znalostí

Jsou-li stanoveny axiomy, je základ báze znalostí vytvořen. Je zřejmé, že tato báze musí být doplněna fakty a dále doladěována a rozšiřována na základě nově získaných znalostí o dané oblasti.

1.2 Konceptově orientované modelování jako podmínka možné reprezentace znalostí

Modelování v rámci umělé inteligence probíhá již standardně nejméně ve dvou úrovních abstrakce, a to na úrovni konceptualizace a na úrovni implementační. Na první z nich, tj. na vyšší úrovni abstrakce, probíhá konceptualizace, jejímž výstupem je konceptuální model v jazyce. Tento jazyk má již sice přísná formální pravidla, ale zároveň disponuje takovými jazykovými prostředky, aby jim mohl porozumět i laický uživatel a podílet se tak na tvorbě konceptuálního modelu vymezeného referenčního systému. Druhá úroveň používá pro implementaci modelu zpravidla jazyk, který je modifikací jazyka logiky prvního řádu a vyžaduje tak hlubší zvládnutí jazykových prostředků.

V lingvistice je dobře znám tzv. Ullmannův trojúhelník znázorňující vztahy mezi realitou, abstrahovanými pojmy a jejich jazykovou reprezentací (Obrázek 1.3).



Obrázek 1.3: Ullmannův trojúhelník

Čerchovaná šipka s názvem reference mezi jazykem a realitou zachycuje skutečnost, že vztah mezi jazykovým výrazem a reálnou věcí je vždy podložen určitou konceptualizací. Jazykovému výrazu koleje mohou podle jedné konceptualizace odpovídat v realitě železniční trať, v jiné konceptualizaci zase ubytovací budovy pro studenty.

1.2.1 Konceptově orientované paradigma

Při tvorbě konceptuálního modelu tvůrce vidí modelovaný svět prostřednictvím určitého paradigmatu, tj. souboru vhodných pojmů s danou sémantikou. Paradigma bývá obvykle vázáno na již formalizovaný modelovací jazyk

Reprezentace znalostí

definovaný zpravidla na nejvyšší úrovni abstrakce tzv. metamodelem. Prvním krokem, který musí tedy znalostní inženýr učinit, je podle výše uvedené metodologie abstrakce modelovaného světa, která vyústí v rozhodnutí, co je předmětem zájmu, tj. o které objekty a jejich vlastnosti a vzájemné vztahy se jedná a které naopak mohou být ignorovány. Toto rozhodnutí učiní v souladu se zvoleným paradigmatem, což umožní používat na nejvyšší úrovni abstrakce konkrétní nástroj modelování – určitý formalizovatelný metamodel.

1.2.2 Principy konceptově orientovaných jazyků reprezentace znalostí

Konceptově orientovaný způsob vidění modelovaného světa v posledních letech v reprezentaci znalostí převažuje. Základním stavebním prvkem modelu je vlastnost (atribut). Koncept (na koncept se můžeme dívat jako na množinu objektů sdílejících určité vlastnosti) je pomocí svých vlastností významově vymezen. Důvodem tohoto přístupu je jednak větší možnost formální manipulace s jednotlivými vlastnostmi a jednak možnost formálního vyjádření konceptů na základě vlastností všeobecně sdíleného významu, opírajícího se o ontologie příslušných doménových oblastí. V neposlední řadě je důvodem konceptového přístupu omezení libovůle volby reprezentačních predikátů na výhradně unární predikáty pro reprezentaci konceptů a výhradně binární predikáty pro reprezentaci vztahů (rolí).

V konceptově orientovaných (konceptových) jazycích jsou koncepty, reprezentovány unárními predikáty, jako např. predikát „být_hmyz“ nebo „být_černý“, a jsou používány k reprezentaci tříd, které jsou zde komponovanými koncepty, reprezentovanými logickými kompozicemi unárních predikátů. Pro účely specifikace jejich vzájemných vztahů neboli rolí jsou určeny binární relace a jim odpovídající binární predikáty. Koncepty mají tu vlastnost, že je můžeme strukturovat do hierarchií.



Definice 1.1 (konceptu)

Koncept je (event. neukončený) soubor entit sdílejících nějaké základní vlastnosti, vzájemně se lišících v jiných vlastnostech.



Definice 1.2 (konceptového paradigmatu, reprezentace a jazyka)

Konceptové paradigma vidí svět jako množinu pojmů – konceptů jednoduchých nebo komponovaných) a jejich vzájemných vztahů (rolí).

Konceptová reprezentace rozlišuje:

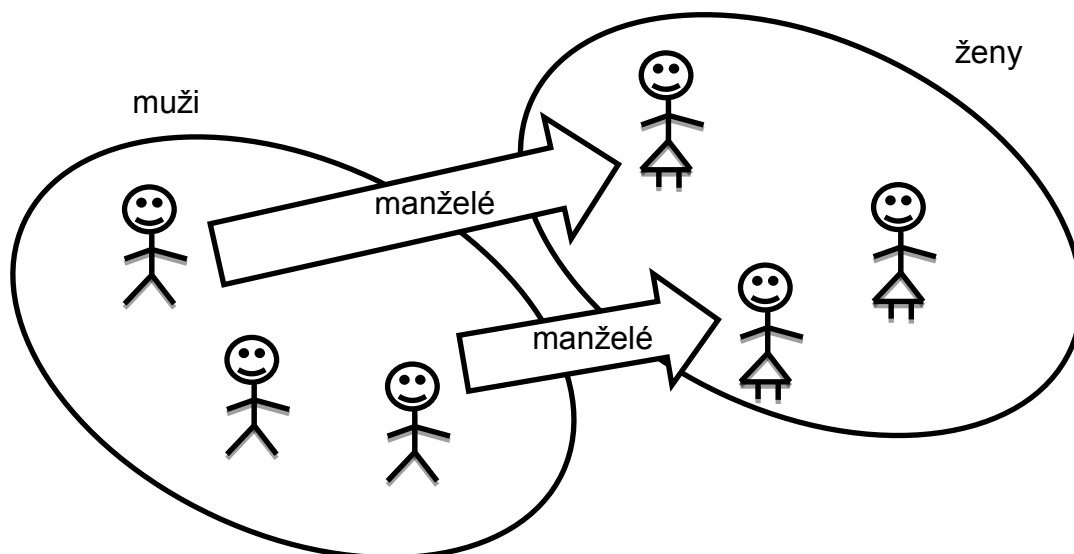
- terminologické znalosti (Tbox znalostní báze) – týkají se konceptů a jejich vzájemných vztahů,
- tvrzení o jejich instancích (Abox znalostní báze) – týkají se příslušnosti objektů konceptům a jejich vztahům (rolím).

V některých literaturách se konceptová reprezentace rozděluje na tři části a to Tbox, Abox a Rbox. Kde Abox obsahuje informace o jedincích (individuích, objektů) a jejich konceptu a rolí ve vztahu k jedinci. Tbox se týká informací

Reprezentace znalostí

ohledně konceptů a jejich vzájemných závislostí. Naproti tomu Rbox obsahuje pouze informace ohledně rolí a jejich závislostí.

Příklad 1.1



Obrázek 1.4: Konceptová reprezentace

Pro náš příklad z obrázku (Obrázek 1.4) by to vypadalo následovně:

- jména individuí: marie, jan, martin, anna, ...,
 - v logice prvního řádu zastávají pozici konstant,
- jména konceptu: Žena, Muž,
 - v logice prvního řádu zastávají pozici unárních predikátů,
- jména rolí: manželé,
 - v logice prvního řádu zastávají pozici binárních predikátů.

Formální konceptové jazyky tedy disponují predikáty:

- unárními pro reprezentaci konceptů,
- binárními pro reprezentaci jejich vztahů (rolí).

Jak je uvedeno výše, koncept je ve formální reprezentace znalostí zpravidla reprezentován unárním predikátem logiky prvního řádu, vztah mezi koncepty binárním predikátem logiky prvního řádu. Význam (sémantiku) konceptu (vztahu) lze pak definovat pomocí:

- *extenzi* – výčtem prvků příslušné interpretující relace (denotátem příslušného predikátu) v rámci dané interpretační struktury – zde jde o denotační sémantiku,
- *intenzi* – formální specifikací jeho vlastností v reprezentujícím jazyce (např. v jazyce predikátové logiky nebo v jazyce daného graficky podporovaného modelu) – jde o axiomatickou sémantiku definovanou pomocí logické formule nebo jejich množiny, neboť význam

Reprezentace znalostí

konceptu/vztahu je definován prostřednictvím speciálních axiomů odpovídající logické teorie,

- jako zvláštní případ intenzionální formální specifikace je možno uvažovat specifikaci prostřednictvím vzájemných souvislostí daného konceptu/vztahu s jinými koncepty/vztahy v rámci ontologické sémantiky, kterou lze též přepsat do logických formulí, která však vyžaduje existenci formální ontologie odpovídající zájmové domény.



Příklad 1.2

Koncept "malé_přirozené_číslo" je reprezentovaný unárním predikátem

malé_přirozené_číslo(x)

a může mít přiřazen svůj význam jako denotát

$$D(\text{malé_přirozené_číslo}(x)) = \{1,2,3\},$$

tj. unární relaci v rámci nějaké interpretační struktury. Tento denotát pak určuje *extenzi* predikátu **malé_přirozené_číslo(x)**.

Formální specifikací uvedeného konceptu pak může být predikátová formule

$$\forall x (\text{integer}(x) \ \& \ 1 \leq x \ \& \ x \leq 3),$$

určující jeho *intenzi*.

1.2.3 Konceptově orientované jazyky a deskripční logika

Konceptově orientované (konceptové) jazyky se vyznačují tím, že vycházejí z popisu (deskripce) pojmů – konceptů, jejich vlastností a vzájemných vztahů. Tento přístup je charakteristický i v jazycích a systémech založených na rámcích nebo sémantických sítích i dalších navazujících grafových systémech.

Společnou myšlenkou asociativních sítí, rámců a konceptových jazyků je nejen reprezentovat znalosti ve formě konceptů a vztahů mezi nimi, ale včetně množinové inkluze extenzí konceptů umožňující zachycení hierarchických struktur konceptů.

Jako sjednocující formalismus reprezentace znalostí prostřednictvím konceptově orientovaných jazyků bývá označována deskripční logika. Reprezentace znalostí o modelované doméně pomocí deskripční logiky spočívá v definování relevantních konceptů a jejich rolí v uvažovaném světě a následném využití definovaných konceptů k formálnímu popisu modelovaného světa. Znalosti reprezentované ve znalostních bázích deskripční logiky jsou formulovány jako sémanticky strukturované informace, umožňující následné odvozování logickými pravidly.

Deskripční logika, stejně jako jiné formální přístupy reprezentace znalostí představuje formalismus vztahujících se vždy k určité doméně – reprezentovanému světu (referenčnímu systému). Reprezentace spočívá v definování relevantních konceptů v uvažovaném světě, zpravidla prostřednictvím rolí, a následném využití definovaných konceptů k formálnímu

Reprezentace znalostí

popisu uvažovaného světa. Znalosti reprezentované ve znalostních bázích deskripční logiky jsou formulovány jako ontologicky strukturované informace, umožňující následné odvozování logickými pravidly. Z hlediska logiky prvního řádu je deskripční logika její strukturovaný fragment disponující pouze unárními predikáty pro reprezentaci konceptů a binárními predikáty pro reprezentaci rolí.

Formální systém deskripční logiky je určen jazykem, znalostní bází, která představuje soubor speciálních axiomů pro dedukci a pravidly tuto dedukci umožňující.

Kontrolní otázky

1. Co je to znalost?
2. Jak jsou znalosti získané?
3. Jaké jsou klíčové role reprezentace znalostí?
4. Jak vytváříme znalostní bázi a z čeho se skládá?



Shrnutí

Kapitola se zabývá počátkem a rozvojem reprezentace deklarativních znalostí. Reprezentace znalostí je jednou z nejdůležitějších a zároveň nejobtížnějších oblastí umělé inteligence. Znalost se tak stává klíčovým prvkem znalostně orientovaných aplikací, např. expertních systémů. v rámci epistemologického přístupu jsou znalosti o modelovaném světě (doméně) reprezentovány takovou formou, aby z nich bylo možno „inteligentně“ odvodit další znalosti, resp. řešení problémů. Aby mohl být formální systém inteligentní, je potřeba, aby disponoval strojově využitelnými znalostmi. Jedinou cestou, jak poskytnout systému takové znalosti, je vtělit je do určitých formálních struktur a to je problém reprezentace znalostí. Ať už je zvolen jakýkoliv formalismus reprezentace znalostí, vždy musí mít sémantiku vycházející z pojmu pravdivosti, má-li reprezentovat skutečné znalosti o modelovaném světě reprezentované zvolenými formálními prostředky.



Pojmy, pomocí kterých se meta-znalosti reprezentují, jsou shromažďovány podle příslušnosti k významovým doménám do terminologických slovníků – ontologií. Oba pojmy, sémantiky jazyka a ontologie, se poprvé objevily již za doby řecké filosofie, kde sémantika jazyka je definována jako oblast filosofie zabývající se studiem povahy významu prvků jazyka, který je určen syntaktickými pravidly v rámci své gramatiky.

2 Asociativní síť

V této kapitole se dozvíte:

- o vývojové řadě grafových formálních systémů,
- o vývoji a historii asociativních (sémantických) sítí,
- základní vlastnosti o asociativních sítí – syntax a sémantika.

Po jejím prostudování byste měli být schopni:

- definovat základní vektor asociativní sítě,
- rozlišit podmíněnou a nepodmíněnou síť,
- tvořit podmínky.

Klíčová slova této kapitoly:

Asociativní síť, sémantická síť, Sowovy konceptuální grafy, RDF graf, grafový formální systém, odvozovací pravidla, sémantika, syntax, klauzulární logika, podmíněná a nepodmíněná síť, negace.

Doba potřebná ke studiu: 8 hodin

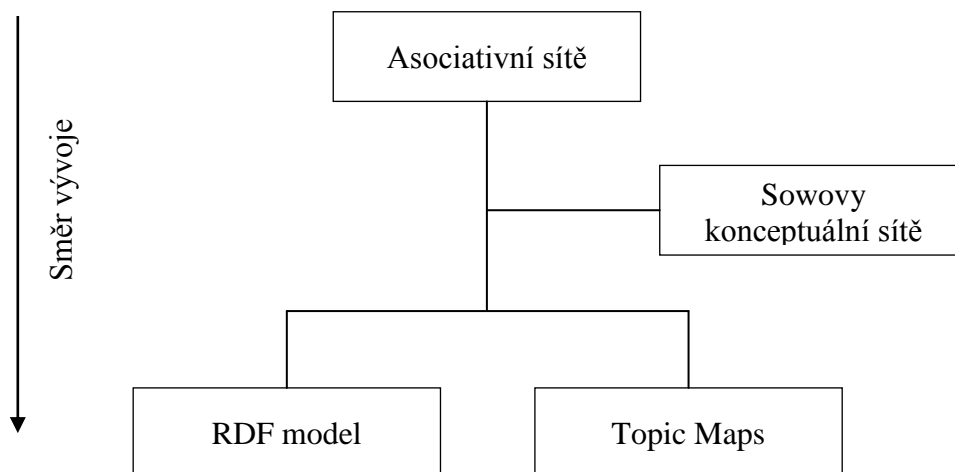


Průvodce studiem

Na reprezentaci znalostí jsou kladeny často protichůdné požadavky a to, aby znalosti byly reprezentovány modulárně, a v rozporu s modularitou stojí požadavek na sémantické sdružování znalostí. Myšlenku sdružování podobných znalostí využívá reprezentace pomocí sémantických sítí, Sowových konceptuálních grafů či rámců (framů). Všechny tyto nástroje sjednocujeme pod společný název – grafové systémy reprezentace znalostí – grafová reprezentace.

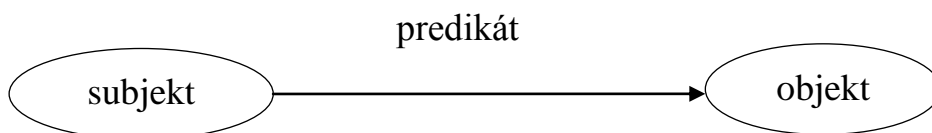
Ačkoliv sledované grafové systémy reprezentace znalostí (sémantické síť, Sowovy konceptuální síť, RDF model a mapy témat) do jisté míry tvoří vývojovou řadu (Obrázek 2.1), nelze hovořit o návaznosti jejich vývoje. Je zřejmé, že jejich nejvýraznější vlastnosti ukazují na motivaci, k jakému účelu byl původně každý z nich vyvinut. Tuto vývojovou řadu znázorňuje následující obrázek (Obrázek 2.1). Následující kapitola s názvem Asociativní síť jako formální systém bude rozebrána dopodrobna, jelikož asociativní síť stály na vrcholu vývoje.

Reprezentace znalostí



Obrázek 2.1: Vývoj grafových systémů reprezentace znalostí.

Sémantické síť, později známé jako asociativní síť, se poprvé objevily v roce 1968. Jako prvním zakladatelem sémantických sítí můžeme považovat M. R. Quilliana [viz literatura], který síť zavedl pro účel modelování sémantiky anglických vět. Reprezentace sémantických sítí vychází stejně jako predikátová logika z atomů, reprezentujících pomocí vhodně zvolených predikátů základní (atomická) tvrzení síť. Tyto tvrzení mají charakter vektoru ($\langle \text{subjekt} \langle \text{predikát/má_vlastnost} \rangle \langle \text{objekt} \rangle$). Sémantické síť na rozdíl od logiky prvního řádu disponují pouze dvoumístným predikátem, protože pouze takové relace lze reprezentovat jako síť, kde každá hrana spojuje dva uzly (Obrázek 2.2).



Obrázek 2.2: Základní tvrzení

Slovům, vyskytujícím se ve tvrzeních, je v asociativních sítích přiřazován význam pomocí jejich příslušnosti k hierarchickým strukturám jim odpovídajících pojmů. Grafové podoba tohoto objasňování významů slov v širších souvislostech činí asociativní síť jednoduchým a snadno pochopitelným prostředkem formální reprezentace znalostí.

Sémantické síť zpravidla nedisponují prostředky pro reprezentaci univerzální a existenční kvantifikace. Formule logiky prvního řádu je však potřeba pro možnost reprezentace sítí upravit do speciálního klauzulárního tvaru. To je problém, který asociativní síť sdílí s klauzulární logikou.



Kolektivu vědeckých pracovníků a doktorandů zabývajících se danou problematikou se jevílo jako vhodné, vypořádat se s problémem reprezentace kvantifikátory vázaných proměnných obdobně jako je tomu v případě klauzulární logiky: všechny proměnné, označované zde symboly začínajícími velkým písmenem, jsou univerzálně vázány, existenčně vázané proměnné lze skolemizací upravit na existenční termy. To následně umožnilo zavedení

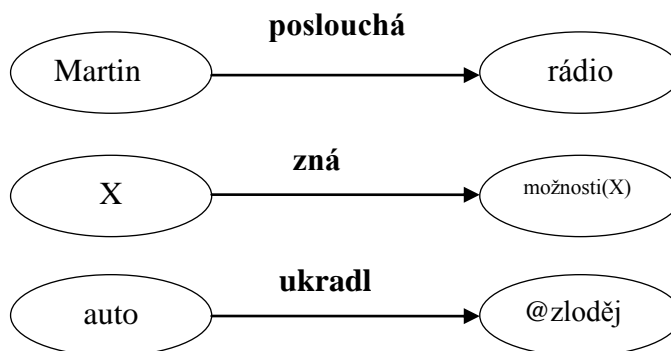
Reprezentace znalostí

formálního systému asociativních sítí se všemi požadovanými vlastnostmi, včetně možnosti dedukce odpovídající dedukci v predikátové logice prvního řádu.



Příklad 2.1

Nechť máme atomické výroky: „Martin poslouchá rádio.“, „Každý zná své možnosti.“, „Nějaký zloděj ukradl auto.“ Tyto atomické výroky můžeme znázornit pomocí sémantické sítě (Obrázek 2.3).



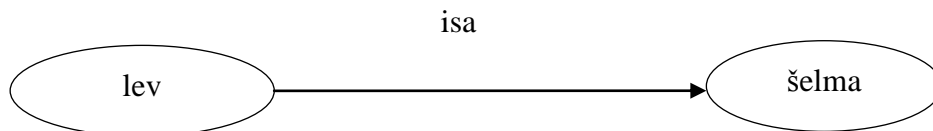
Obrázek 2.3: Grafová notace příkladu 2.1

V sémantických sítích lze výhodně vyjadřovat vztahy množinovou inkluzí a příslušností v množině, lze zde reprezentovat jedinečné i obecné pojmy. Vztahy množinové inkluze a příslušnosti ke třídě umožňují efektivně reprezentovat taxonomické (hierarchické) uspořádání objektů. Pak lze přímočaře provádět odvozování specializací či generalizací. Při specializaci se informace v taxonomii přenášejí od obecnějších typů ke speciálnějším, při generalizaci je tomu naopak. Velmi často se v této souvislosti hovoří o *isa* hierarchiích (z angl. „... is a ...“) a o *ako* hierarchiích (z angl. „... a kind of ...“). V jedné sémantické síti mohou být znalosti asociovány z mnoha různých hledisek.

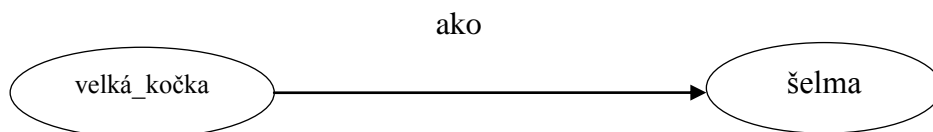
Rozdíl mezi *isa* a *ako*:



- *isa* – vyjadřuje vztah konceptu a jeho instance (Obrázek 2.4),
- *ako* – vyjadřuje vztah konceptu a subkonceptu (Obrázek 2.5).



Obrázek 2.4: isa hierarchie



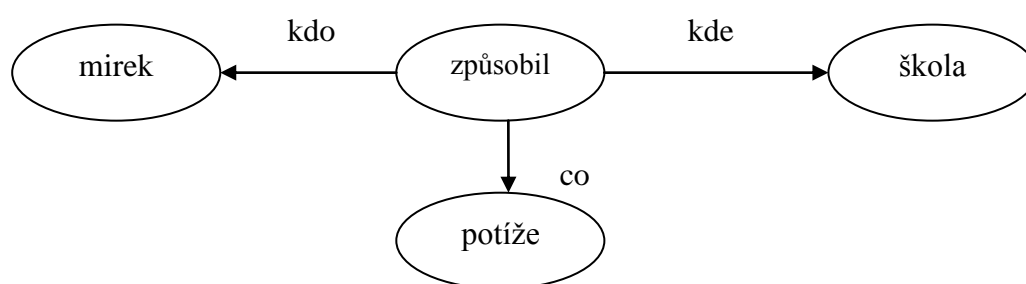
Obrázek 2.5: ako hierarchie

Reprezentace znalostí

2.1 Omezení asociativních sítí

Omezení konvenčních sémantických sítí byla intenzivně studována řadou pracovníků v umělé inteligenci. Mnozí věří, že základním pojmem je síla sémantických sítí a může být doplněna, například logikou, která zlepšuje sémantické síť pomocí její expresivní síly a robustnosti. Ostatní zase věří, že pojem sémantické sítě může být zlepšen o včlenění principů dedukce.

Sémantické síť jsou postavené na tzv. trojici, která na rozdíl od logiky prvního řádu disponuje pouze dvojmístným predikátem (pouze takové relace lze reprezentovat jako síť). Tyto binární vztahy jsou obvykle jednoduše reprezentovatelné, ale někdy je to obtížné. Příkladem může být věta: „Mirek způsobil problémy ve škole.“ Viz Obrázek 2.6.



Obrázek 2.6: Tvrzení sémantické sítě

Mezi další problémová tvrzení můžeme zařadit negaci – „Mirek nechodí do školy“. Nebo disjunkci – „Mirek jí pizzu nebo brambory a rybu.“

Podstatným omezením pro sémantické síť je kvantifikace. Sémantické síť nedisponují prostředky pro reprezentaci univerzální a existenční kvantifikace. Například: „Každý pes už někdy pokousal pošťáka.“ „Každý pes už někdy pokousal každého pošťáka.“ „Existuje pes, který už někdy pokousal pošťáka.“ Řešením kvantifikace jsou segmentové sémantické síť, které mohou reprezentovat kvantifikované síť. Druhým způsobem je převod formule logiky prvního řádu do speciálního klauzulárního tvaru. To je problém, který sémantické síť sdílejí s klauzulární logikou. Princip převodu spočívá v tom, že všechny proměnné, označené zde symboly začínající velkým písmenem, jsou univerzálně vázané, existenčně vázané proměnné lze skolemizací upravit na existenční termy.

2.2 Jazyk

Aby systém asociativních sítí byl formálním (axiomatickým) systémem, je třeba

1. definovat syntax jeho jazyka,
2. stanovit, jakým způsobem probíhá formální odvozování v asociativní síti, tj.
 - a. definovat, co je výchozí znalostní báze, sestávající ze speciálních axiomů teorie,

Reprezentace znalostí

- b. charakterizovat odvozovací pravidla, jimiž se ze znalostní báze generují věty teorie,
3. definovat jeho sémantiku a dokázat jeho sémantickou korektnost.



Definice 2.1

Asociativních (sémantická) síť je ohodnocený graf sestávající z uzlů, ohodnocených termy, a hran, ohodnocených binárními predikátovými symboly, přičemž hrany spojují některé dvojice uzlů.

Jazyk asociativní sítě disponuje těmito typy symbolů:

1) Uzly

- a) grafové symboly označující uzly sítě
- b) termy v návěštích uzlů - znakové řetězce
 - i) pro označení proměnných (začínající velkým písmenem): X, Y, \dots
 - ii) pro označení konstant (začínající malým písmenem nebo číslicí):
dům, kočka, 12,...
 - iii) pro funkční symboly : $\sin(X), matka(X), \dots$
- c) pro existenční termy (začínající symbolem @): @někdo, @známý(X),
...

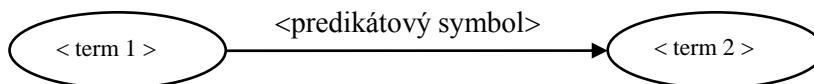
<návěští uzlu>

2) Hrany

- a) grafové symboly označující hrany sítě
- b) binární predikátové symboly v návěštích hran: $isa(X, Y), bydlí(X, Y), \dots$
(atributy v závorkách se v grafu neuvádějí).

<návěští hrany>

Atom asociativní sítě tvoří *vektor* sestávající ze dvou uzlů, jejichž návěštím jsou termy, a jedné spojující hrany, jejímž návěštím je predikátový symbol (Obrázek 2.7).



Obrázek 2.7: Vektor

Asociativní síť tvoří množina (ne nutně) vzájemně propojených atomických vektorů.

V rámci znalostníchází asociativních sítí jsou tvrzení reprezentována sítěmi těchto typů:

- nepodmíněné síť
 - bázové (fakta),
 - univerzální;
- podmíněné síť
 - bázové,
 - univerzální.

Reprezentace znalostí

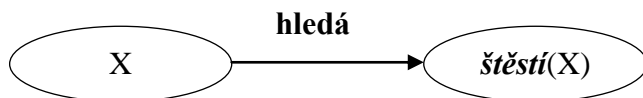
Nepodmíněné bázev sítě reprezentují fakta. Podmínky vyjadřují, co z čeho vyplývá. Podmínky se zakreslují tak, že seatomy antecedentu označují čárkovanými hranami, konsekventu plnou čarou. Podmínky v asociativních sítích představují, podobně jako je tomu v predikátové nebo klauzulární logice, pravidla s antecedentem a konsekventem. Na skutečnosti, zdali se v návěštích jejich uzlů vyskytují nebo nevyskytují proměnné, závisí jejich zařazení mezi *univerzální* nebo *bázové podmínky*. Obecný tvar podmínky je v sémantických sítích

$$q \text{ if } p_1, p_2, \dots, p_n$$

Podmínky v sítích mohou mít více než jeden atom v antecedentu, ale pouze jeden (na rozdíl od klauzulární logiky) v konsekventu.

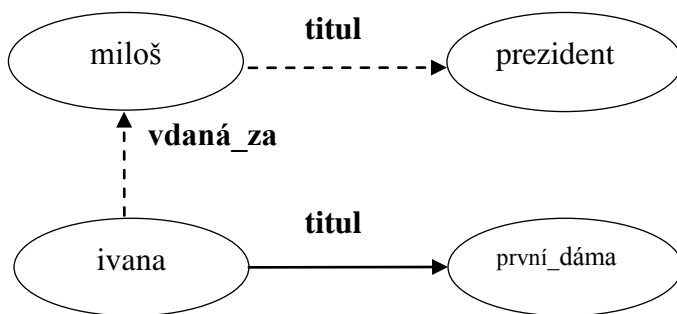
Bázová nepodmíněná síť, obsahuje v návěštích svých uzlů výhradně konstanty, např. reprezentace tvrzení „Martin poslouchá rádio.“ na Obrázku 2.3.

Příkladem *univerzální nepodmíněné sítě* reprezentující tvrzení „Každý hledá své štěstí.“ je vektor:



Obrázek 2.8: Univerzální nepodmíněná síť

Bázová podmínka reprezentující tvrzení „Ivana je první dáma, jestliže se vdala za prezidenta Miloše.“:



Obrázek 2.9: Bázová podmínka reprezentující tvrzení

Úkol

1. Vytvořte asociativní síť příbuzenských vztahů ve Vaší rodině.
2. Zavedením proměnných tuto síť zobecněte.
3. Vytvořte podmínku, která vyjadřuje stav být babičkou (dědečkem).



2.3 Odvozovací principy

Odvozováním v asociativních sítích se rozumí doplňování dalších vektorů nebo termů v návěštích jejich uzlů do sítě, která je pro toto odvozování hlavní sítí, pomocí jiné podmíněné (zvláštní) sítě některým z těchto odvozovacích pravidel:

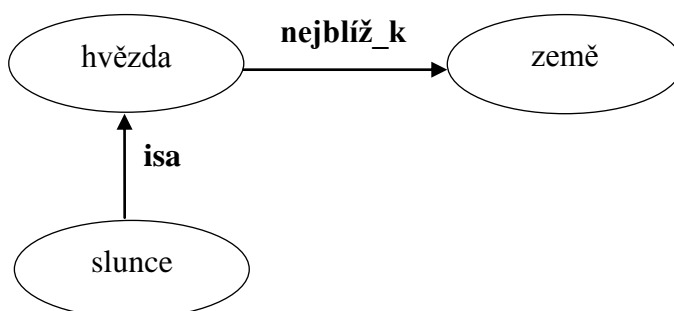


- *Pravidlo uniformní substituce* – Jestliže se všechny ohodnocené hrany podmíněné sítě objeví v hlavní sítí, obsahující proměnné jako návěští jimi spojených uzlů, pak lze návěští z uzlů podmíněné sítě substituuovat uniformně za návěští odpovídajících uzlů sítě hlavní, označených proměnnými.
- *Pravidlo transferu* – Jestliže se všechny čárkované hrany antecedentu podmíněné sítě objeví (jako plné) v hlavní sítí, přičemž návěští odpovídajících si uzlů se buď shodují, nebo jsou v některé ze sítí proměnnými, pak lze plný vektor (event. po aplikaci pravidla uniformní substituce) konsekventu z podmíněné sítě doplnit do sítě hlavní.

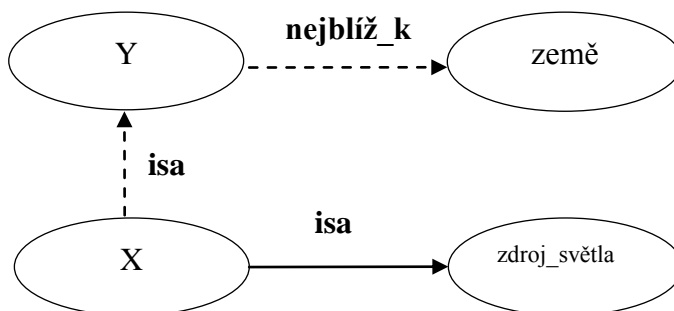


Příklad 2.2 pravidlo transferu

Hlavní síť reprezentující některé vztahy v naší sluneční soustavě:



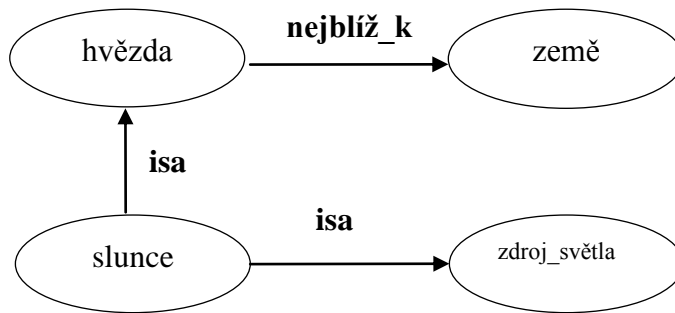
Obrázek 2.10: Hlavní síť sluneční soustavy.



Obrázek 2.11: Vedlejší síť s univerzálním kvantifikátorem.

Doplnění vektoru konsekventu univerzální sítě (Obrázek 2.11) (zvláštní/vedlejší sítě) do hlavní sítě, vznikne podle pravidla transferu nová síť:

Reprezentace znalostí



Obrázek 2.12: Hlavní síť po pravidlu transferu.

Pravidlo transferu zde zjevně odpovídá známému pravidlu Modus Ponens (MP) logiky prvního řádu:

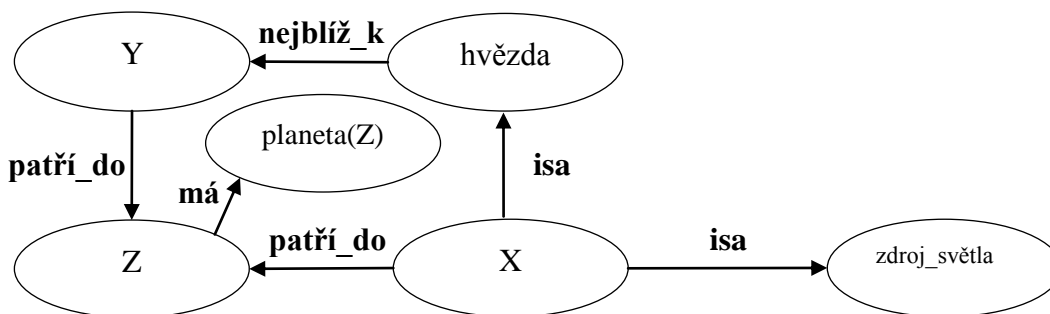
$$z A \text{ a } A \rightarrow B \text{ odvodit } B,$$

kde graf na Obrázek 2.10 představuje tvrzení A, graf na Obrázek 2.11 implikaci $A \rightarrow B$ a graf na Obrázek 2.12 obsahuje B jako výsledek aplikace pravidla MP.

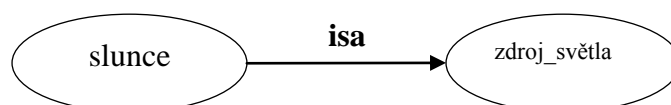
Příklad 2.3 pravidlo uniformní substituce



Je-li hlavní síť univerzální síť jako na obrázku dole (Obrázek 2.13), která mimo jiné tvrdí, že jakákoliv soustava v galaxii má své planety, a zvláštní síť na Obrázek 2.14, je možno podle pravidla uniformní substituce do této hlavní dítě doplnit za X term Slunce a získat tak novou univerzální síť (Obrázek 2.15). Tato nová univerzální síť reprezentuje tvrzení, že kterákoliv soustava, která bude patřit Slunce, má své planety. Požadavek uniformní substituce zaručí přiřazení každé soustavě správně jeho planety. Tzn., že v našem případě existuje pouze jedna jediná soustava v galaxii, kde je Slunce a to je Sluneční soustava. Navíc tato Sluneční soustava má své planety (Merkur, Venuše, Země, atd.).

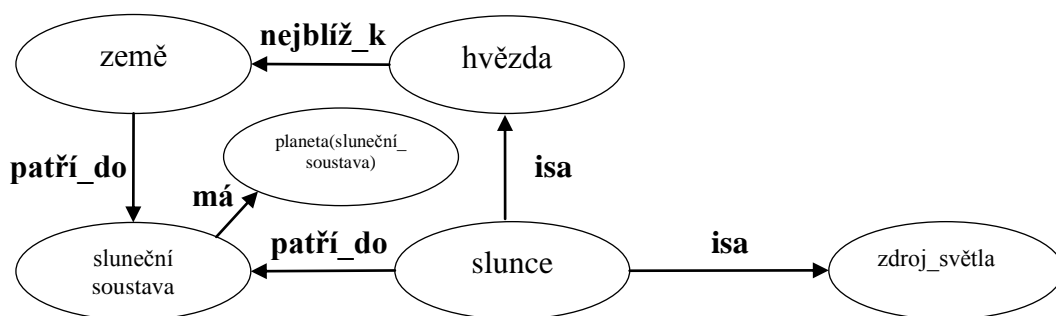


Obrázek 2.13: Hlavní síť pro odvozování.



Obrázek 2.14: Zvláštní síť pro odvozování.

Reprezentace znalostí



Obrázek 2.15: Odvozená síť.

2.4 Sémantická korektnost a úplnost

2.4.1 Interpretace asociativní sítě a její struktura

Pro přiřazení struktury dané síti pro možnost její interpretace platí podobně jako v klauzulární logice určitá pravidla.



Definice 2.2

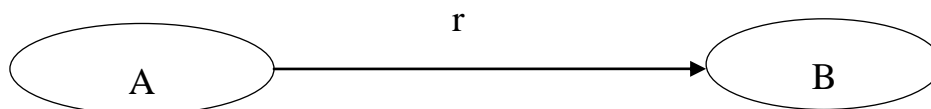
Strukturu interpretace asociativní sítě tvoří trojice množin $\{W, F, R\}$, kde

1. W je množina objektů světa, který má být asociativní sítí reprezentován.
2. Každé konstantě sítě je přiřazen nějaký objekt universa diskursu W daného světa - její denotát.
3. Každé existenční konstantě sítě je přiřazeno jako množina možných denotátů universum diskursu W .
4. Každé funkci je přiřazen jako její denotát předpis z množiny F pro její vyhodnocení.
5. Každému n -árnímu existenčnímu funktoru sítě je přiřazena jako množina možných denotátů množina všech n -árních funkcí nad universem diskursu W .
6. Každému predikátu sítě je přiřazen jeho denotát – relace z množiny R definovaná na daném universu diskursu W .

2.4.2 Pravdivost bázevých vektorů a sítí



Definice 2.3



Obrázek 2.16: Bázevých vektor

Reprezentace znalostí

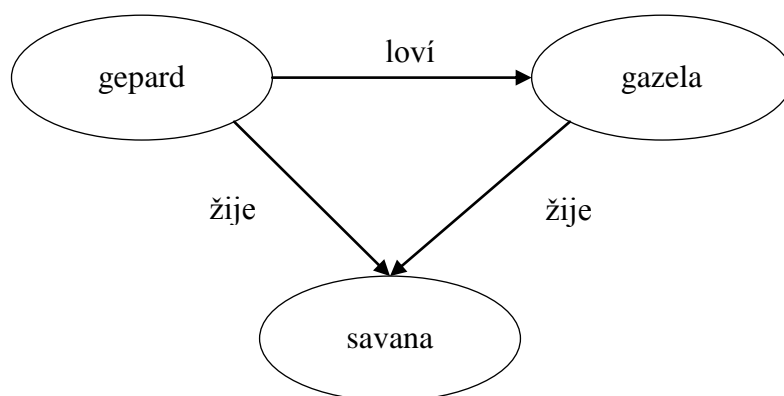
Bázový vektor (Obrázek 2.16) je pravdivý ve struktuře S dané interpretace, jestliže denotát objektu A a denotát objektu B jsou v relaci R , která je denotátem predikátu r ve struktuře S . V opačném případě je nepravdivý.

Bázová nepodmíněná síť je pravdivá ve struktuře S dané interpretace, jsou-li v ní pravdivé všechny její bázové vektory.

Bázová podmínka je nepravdivá ve struktuře S , jestliže všechny její vektory antecedentu jsou pravdivé a vektor konsekventu je nepravdivý v S . Jinak je tato podmínka pravdivá.

Příklad 2.4

Svět reprezentovaný sítí se týká těchto tří objektů: gepard, gazela a savana. V tomto světě platí, že gepard i gazela žijí v savaně, navíc gepard loví gazelu.



Obrázek 2.17: Znalostní báze

Universum diskursu: $W = \{\text{gepard, gazela, savana}\}$

Denotáty konstant: $D(\text{gepard}) = \text{Gepard}$, $D(\text{gazela}) = \text{Gazela}$, $D(\text{savana}) = \text{Savana}$.

Denotáty predikátů: $D(\text{žije}) = \{(\text{Gepard, Savana}), (\text{Gazela, Savana})\}$

$D(\text{loví}) = \{(\text{Gepard, Gazela})\}$

Ve struktuře interpretace jsou všechny bázové vektory pravdivé. Totéž pak platí o síti jako celku.



2.4.3 Splnitelnost a platnost univerzálních sítí

V interpretaci s danou strukturou mohou být proměnným sítě přiřazeny jejich valuaci (ohodnocením) nějaké hodnoty z universa diskursu W . Z univerzálního tvrzení se touto valuací stává bázové tvrzení, jehož pravdivost lze v dané struktuře interpretace určit.

Reprezentace znalostí



Definice 2.4

Nechť A je univerzální síť, S je struktura její interpretace a v nějaká valuace všech proměnných z A v S . Nechť $A[v]$ je výsledná síť valuace. Potom v *splňuje* A v S , právě když $A[v]$ je pravdivá.



Definice 2.5

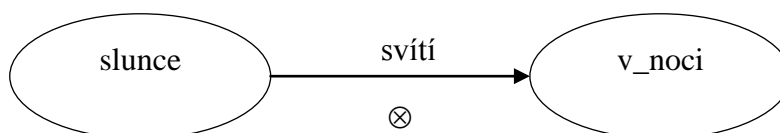
Univerzální asociativní síť (nepodmíněná síť nebo podmínka) C je *platná* ve struktuře S , právě když každá valuace proměnných z C splňuje C v S . Jinak je C nepravdivá v S . Struktura S potom síť C *splňuje*, je jejím *modelem*.

Struktura příkladu 2.2 je splňující strukturou, tj. modelem dané sítě, neboť síť, která nemá proměnné je též podle definice 2.5 platná v S .

2.4.4 Negace v asociativní síti

K popření bázevého vektoru V se používá zvláštního obratu, tj. podmínky, která říká, že je-li V pravdivý, pak následuje nepravdivý konsekvent. Problémem je, jaké nepravdy lze použít pro účel popření - jde totiž o logickou nepravdu, tj. takovou, která je false ve všech interpretacích. Pro vyjádření, že neplatí tvrzení „Slunce svítí v noci.“ není možno použít např. obratu „Svítili slunce v noci, pak $1 = 0$.“, neboť při jiné interpretaci predikátu $=$ nemusí konsekvent podmínku popírat.

Protože neexistují vektory typu kontradikce, je třeba pro účel popření zavést speciální prvek sítě zvaný falsum a označovaný \otimes , který je false v každé interpretaci (Obrázek 2.18).



Obrázek 2.18: Negativní tvrzení



Věta 2.1

Nechť V je bázevý vektor. Pak v libovolné struktuře interpretace má podmínka s V jako antecedentem a \otimes jako konsekventem opačnou pravdivostní hodnotu k V .

Důkaz

Předpokládejme, že podmínka V je true v nějaké struktuře S . Protože její konsekvent je v S false, musí její antecedent podle definice být též false v S . Předpokládejme naopak, že podmínka V je false v S . Pak protože její konsekvent je false v S , její antecedent V musí být true v S podle definice pravdivosti bázevých podmínek.

Reprezentace znalostí

Definice 2.6 (negativní tvrzení)

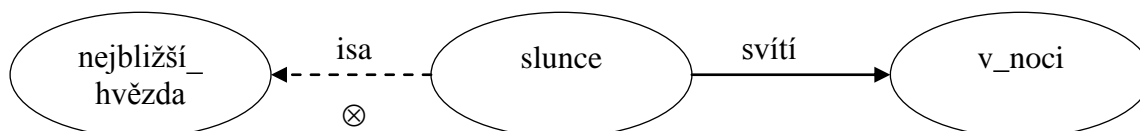
Podmínky, jejichž konsekventem je falsum, se nazývají *negativní tvrzení*.

Odvozovací metody se v asociativních sítích používají k dokazování tvrzení. Sémanticky to znamená, že tvrzení dokázané s využitím těchto dvou pravidel je pravdivé ve struktuře S, jsou-li tvrzení, z nichž se odvozovalo, pravdivá v S.



Příklad 2.5

Pokud slunce svítí v noci, není to nejbližší hvězda k zemi.



Obrázek 2.19: Negativní tvrzení v konsekventu.

2.4.5 Sémantická korektnost odvozování v asociativních sítích

Odvozování v asociativních sítích je postupem, jehož cílem je doplnit univerzální síť, která je zde sítí hlavní, při splnění nějakého pravidla reprezentovaného sítí zvláštní. Aby odvozování bylo korektním postupem, je třeba zaručit, že každá struktura splňující jak hlavní síť, tak i síť zvláštní, splňovala i síť odvozenou. Je proto třeba pro každé z odvozovacích pravidel dokázat jeho sémantickou korektnost.

Věta 2.2 (o sémantické korektnosti pravidla univerzální substituce)

Jestliže struktura S splňuje zvláštní síť C a existuje taková vazba vektorů z C, která vede ke shodě s vektory univerzální hlavní sítě, pak instance, která se odvodí z hlavní sítě uniformními substitucemi bazových termů sítě C za proměnné hlavní sítě, je rovněž strukturou S splněna.

Důkaz

Bázové termy na místě proměnných jsou prvky universa diskursu W struktury S a představují valuaci těchto proměnných univerzální sítě platné v S. Protože všechny proměnné hlavní sítě jsou univerzálního charakteru, je síť i touto valuací splněna.

Věta 2.3 (o sémantické korektnosti pravidla transferu)

Jestliže struktura S splňuje pravidlo C zvláštní sítě a každý z vektorů jeho antecedentu se objeví v hlavní síti jako pravdivý bazový vektor ve struktuře S, resp. stane se jím z univerzálního vektoru po aplikaci pravidla uniformní substituce, potom též konsekvent podmínky C, který byl pravidlem transferu přidán do sítě, je pravdivý ve struktuře S.

Důkaz vyplývá přímo z definice pravdivosti bazových podmínek.

Reprezentace znalostí



Kontrolní otázky

1. Definujte syntax jazyka asociativních sítí.
2. Jaký je rozdíl mezi nepodmíněnou sítí a podmínkou?
3. Čím se liší báze sítí od univerzální sítě?
4. Kdy je báze vektor pravdivý?
5. Kdy je pravdivá báze nepodmíněná sítí?
6. Kdy je pravdivá báze podmínka?
7. Kdy je splnitelná a kdy je platná univerzální nepodmíněná asociativní sítí?
8. Kdy je splnitelná a kdy je platná univerzální podmínka?



Korespondenční úkol

Pomocí asociativní (sémantické) sítě zaznamenejte svůj rodokmen. Zachyť všechny vztahy v rodině.



Úkoly k zamyšlení

Do asociativní sítě reprezentující Vaše rodinné vztahy zařaďte tvrzení, že jste otcem/matkou, synem/dcerou. Na základě příslušné podmínky pak v sítí odvoďte, že vaše matka/otec je babičkou/dědečkem.



Shrnutí

Reprezentace sémantických sítí vychází stejně jako predikátová logika z atomů, reprezentujících pomocí vhodně zvolených predikátů základní (atomická) tvrzení sítě. Tyto tvrzení mají charakter vektoru ($\langle \text{subjekt} \rangle \langle \text{predikát/má_vlastnost} \rangle \langle \text{objekt} \rangle$). Sémantické sítě na rozdíl od logiky prvního řádu disponují pouze dvoumístným predikátem, protože pouze takové relace lze reprezentovat jako sítí, kde každá hrana spojuje dva uzly.

Sémantické sítě zpravidla nedisponují prostředky pro reprezentaci univerzální a existenční kvantifikace. Formule logiky prvního řádu je však potřeba pro možnost reprezentace sítí upravit do speciálního klauzulárního tvaru. To je problém, který asociativní sítě sdílejí s klauzulární logikou.

V rámci znalostníchází asociativních sítí jsou tvrzení reprezentována sítěmi těchto typů – nepodmíněné sítě a podmíněné sítě. Navíc tyto sítě ještě dělíme na báze a univerzální sítě.

Odvozováním v asociativních sítích se rozumí doplňování dalších vektorů nebo termů v návěštích jejich uzlů do sítě, která je pro toto odvozování hlavní sítí, pomocí jiné podmíněné (zvláštní) sítě některým z těchto dvou odvozovacích pravidel – pravidlo uniformní substituce a pravidlo transferu.

Protože neexistují vektory typu kontradikce, je třeba pro účel popření zavést speciální prvek sítě zvaný falsum a označovaný \otimes , který je false v každé interpretaci.

3 Sowy konceptuální grafy

V této kapitole se dozvíte:

- základní vlastnosti konceptuálních grafů,
- co je to koncept a jaký má vztah v grafu,
- z čeho se skládá znalostní báze,
- o ontologii jako základnímu principu konceptuálních grafů.

Po jejím prostudování byste měli být schopni:

- definovat konceptuální graf,
- určit vztah, valenci a signaturu u grafu,
- definovat typ konceptů a typy vztahů.

Klíčová slova této kapitoly:

Konceptuální graf, koncept, vztah, valence, signatura, typ, subtyp, supertyp, referent.

Doba potřebná ke studiu: 3 hodiny

Průvodce studiem

Pojem sémantických sítí má sice svůj původ v lingvistice 60. let minulého století, brzy však konceptově orientovaný prostředek reprezentace znalostí zdomácněl nejen v lingvistice, ale i v informatice, neboť představuje snadný a srozumitelný prostředek modelování především na konceptuální úrovni.

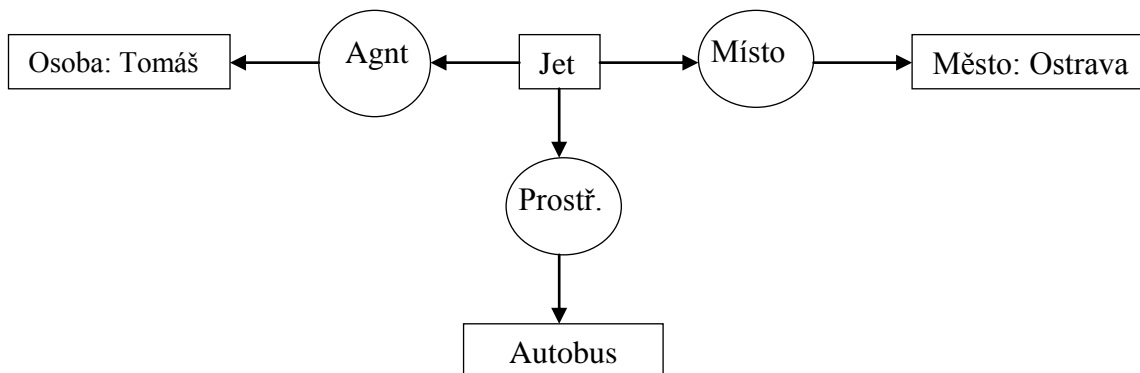
V této kapitole se seznámíme s dalším prostředkem pro reprezentaci znalostí. Cílem této kapitoly je popis konceptuálních grafů a tak, prosím, věnujte této kapitole minimálně 3 hodiny.



Další z forem reprezentace znalostí je konceptuální graf, který má na rozdíl od sémantické sítě větší expresivitu. Konceptuální grafy byly vytvořeny na základě existenciálních grafů od Charles Sanders Peirce (1909) a autorem byl J. F. Sowa, jehož práce započala roku 1976. Konceptuální grafy byly původně vyvíjeny jako grafy v textové podobě (lineární notace) pro snadné používání počítačových programů. Konceptuální grafy jsou schopné reprezentovat širokou škálu znalostních tříd – od predikátové logiky k přirozeným jazykům.

Jazyk konceptuálních grafů je v podstatě založen na logice prvního řádu s tradiční deklarativní sémantikou s řadou výjimek, které z rámce logiky prvního řádu vybočují. Naším cílem bude zachování všech požadavků na logiku prvního řádu.

Reprezentace znalostí



Obrázek 3.1: „Tomáš hodlá jet autobusem do Ostravy.“

Konceptuální graf uvedený na Obrázek 3.1 reprezentuje napsanou nebo setříděnou verzi logiky. Každý ze čtyř konceptů má hodnotu typu, která reprezentuje typ entity konceptu odkazující se na: Osobu, Jet, Ostrava a Autobus. Dva z konceptů mají jméno, které identifikuje referenta: Tomáš nebo Ostrava. Každý ze tří konceptuálních relací má hodnotu typu, které reprezentuje typ relace: agent (Agnt), místo (Místo) nebo prostředek (Prostř.). Konceptuální graf jako celek znamená, že osoba Tomáš je agentem instance „hodlá jet“, město Ostrava je místem a autobus je nástrojem. Obrázek (Obrázek 3.1) může být přeložen do následující formule:



Příklad 3.1

$$(\exists x)(\exists y)(\text{Jet}(x) \wedge \text{Osoba}(\text{Tomáš}) \wedge \text{Město}(\text{Ostrava}) \wedge \text{Autobus}(y) \wedge \\ \text{Agnt}(x, \text{Tomáš}) \wedge \text{Místo}(x, \text{Ostrava}) \wedge \text{Prostř.}(x, y))$$

Jak ukazuje tento příklad, tak jediné logické operátory, které se používají na obrázku (Obrázek 3.1), jsou konjunkce a existenční kvantifikátor. Tyto dva operátory jsou nejvíce běžné v překladech z přirozených jazyků.



Definice 3.1 (konceptuálního grafu)

Konceptuální graf je síť uzlů dvou druhů pro reprezentaci

1. konceptů
2. jejich vzájemných vztahů

Hrany mezi těmito uzly jsou orientované. Konceptuální graf je bipartitní graf. Hrana nikdy nespojuje dva koncepty ani dva vztahy, vždy spojuje koncept se vztahem nebo naopak.

Konceptuální graf je bipartitní graf, který využívá dva typy notace – grafovou a lineární.

Reprezentace znalostí

Definice 3.2 (jazykových symbolů)

Jazykovými symboly grafické notace konceptuálních grafů jsou:

- obdélníky reprezentující v grafu koncepty a jejich individuové instance,
- elipsy reprezentující vztahy mezi nimi (role).

Lineárními jazykovými prostředky pro reprezentaci konceptuálních grafů jsou:

- koncepty uváděné v hranatých závorkách,
- vztahy uváděné v okrouhlých závorkách.

Koncepty, které reprezentují podgrafy, jsou nástrojem vyjádření kontextu.



Příklad 3.2

Lineární notace (Obrázek 3.1)

[osoba: Tomáš] ← (Agnt) ← [Jet] → (Místo) → [Město: Ostrava]
→ (Prostř.) → [Autobus]



Definice 3.3 (konceptu v konceptuálním grafu)

Koncept sestává ze dvou složek:

- typ konceptu
- referent

označení v lineární notaci: [typ: referent].

Např. výraz [osoba: Jan] reprezentuje tvrzení „existuje osoba jménem Jan“. Referent může být i prázdný – např. výraz [kniha] reprezentuje tvrzení „existuje kniha“



Definice 3.4 (vztahu v konceptuálním grafu)

Se vztahy souvisejí tři pojmy: *typ vztahu*, *valence* a *signatura*.

- *Typem vztahu* je jméno vztahu. Každý vztah má *typ vztahu* (nemá referenta).
- *Valence* vztahu je číslo hrany, která k němu patří.
- *Signatura* vztahu je seznam typů konceptů patřících ke vztahu.

Typ vztahu určuje valenci a signaturu, tj. počet hran k němu příslušných a typy konceptů, které se k němu vztahují. Valence vztahu je konstanta označující počet hran patřících k danému vztahu. Vztah valence n se nazývá n -adický vztah, speciálně monadický ($n = 1$), dyadický ($n = 2$), triadický ($n = 3$).



Reprezentace znalostí



Příklad 3.3 pokračování k Př. 3.2

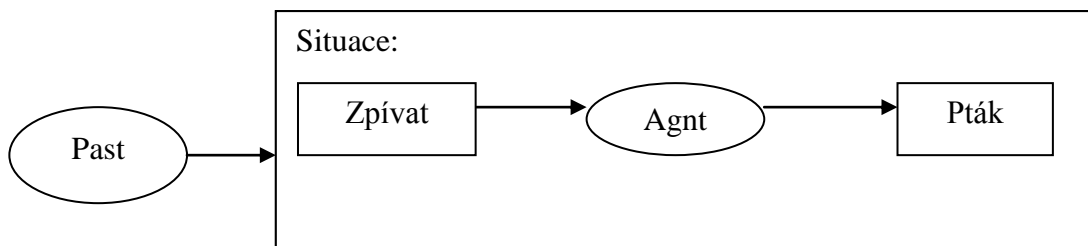
Typ vztahu Agnt je Agnt, valence je 1 a znamená vztah mezi činností a osobou (živou bytostí).



Příklad 3.4

Monadický vztah je např. (Past) – pro minulý čas, (Psbl) – pro možná.

(Past) \longrightarrow [Situace: [Zpívat] \longrightarrow (Agnt) \longrightarrow [Pták]]



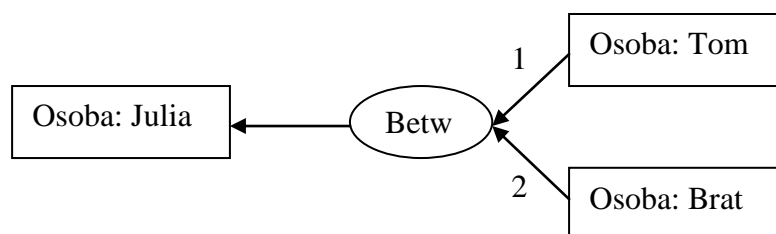
Obrázek 3.2: Monadický vztah

Příklad *triadického vztahu*:

[Osoba: Julia] \longleftarrow (Betw) \longleftarrow 1 \longleftarrow [Osoba: Tom]

\longleftarrow 2 \longleftarrow [Osoba: Brad]

„Julie je mezi Tomem a Janem.“



Obrázek 3.3: Triadický vztah

Signatura vztahu se zapisuje jako seznam $\langle t_1, \dots, t_n \rangle$ typů konceptů připojených ke vztahu.

Např. Agnt má signaturu $\langle \text{akce, živočich} \rangle$



Směr hran

Konvence: Jestliže vztahu náleží n hran, směřuje prvních $n-1$ hran ke vztahu, zbývající n -tá hrana od vztahu. Toto pořadí je dáno signaturou.

Reprezentace znalostí

3.1 Znalostní báze

Znalostní báze sestává z těchto složek:

- Soubor konceptuálních grafů vyjadřujících znalosti o modelovaném světě – referenčním systému.
- Hierarchie typů vyjadřující typy konceptů, potřebných k reprezentaci daného referenčního systému.
- Hierarchie typů vztahů, vyjadřující typy vztahů, potřebných k reprezentaci daného referenčního systému.
- Katalog objektů, který obsahuje identity všech objektů, které se objeví ve znalostní bázi.

3.1.1 Ontologie jako základní princip konceptuálních grafů

Sowa [viz literatura]: „Předmětem ontologie je studium kategorií věcí, které existují nebo mohou existovat v nějaké oblasti. Produkt takového studia, zvaný ontologie, je katalog typů věcí považované za existující v oblasti zájmu D z pohledu osoby, která používá jazyk L pro účely popisu oblasti D.“

Základním pojmem ontologie jako základního principu konceptuálních grafů je pojem typ. V konceptuálních grafech vystupují typy konceptů a typy vztahů. Typ je návěští (jméno), které je přiřazeno množině objektů podobného pojetí.

Těmito entitami mohou být např.

- Individuály (jako osoba, kočka, bus,...)
- Vlastnosti (jako krásný, inteligentní, smrtelný, modrý,...)
- Akce (jako políbit, bliknout, jít,...)
- Abstraktní entity (jako PřirozenéČíslo, Množina,...)

Přestože typ je jméno skupiny entit, vždy existuje pouze jako abstraktní. Obráceně neplatí, že cokoliv abstraktního je typem. Např. 2 je abstraktní, ale není typem, nýbrž instancí typu PřirozenéČíslo.

Definice 3.5 (způsobů definování typů)

Typy mohou být definovány následujícími způsoby:

- Extenzí – např. typ RodinaKovářů je dána výčtem <Daniela, Jiří, Jana, Jan>
- Intenzí – seznamem vlastností individuí náležících typu, resp. obecným pravidlem řazení individuí k tomuto typu (typ LichéČíslo je číslo nedělitelné dvěma).
- Axiómem – typy, které jsou matematickými entitami. Např. Peanovy axiomy definující typ PřirozenéČíslo:
 - 0 je přirozené číslo



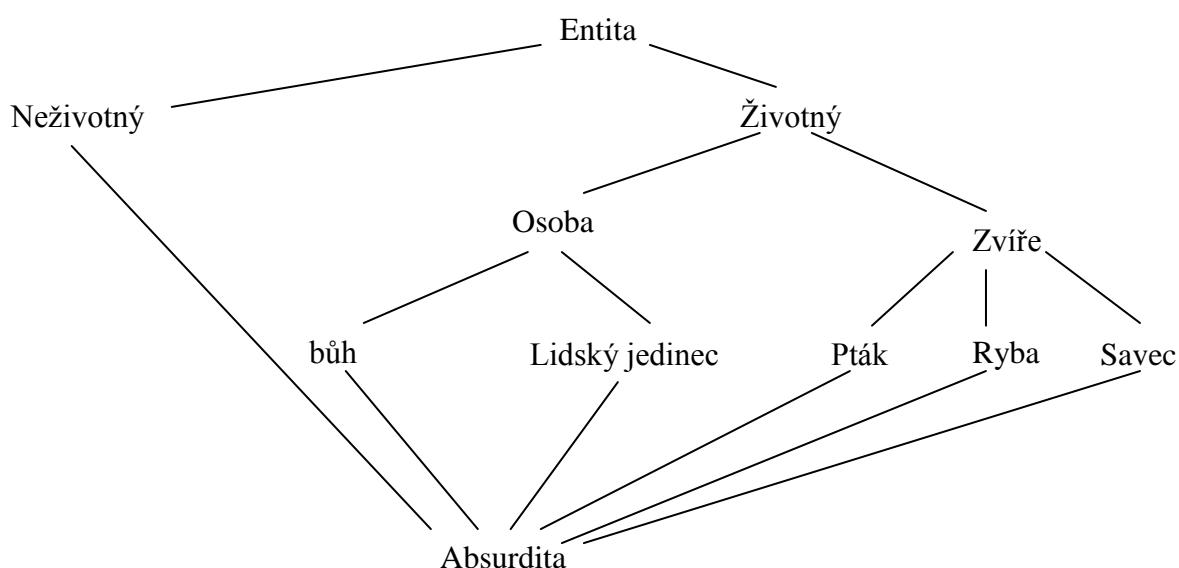
Reprezentace znalostí

- Je-li a přirozené číslo, je $a+1$ rovněž přirozené číslo.
- Jestliže z dokazatelnosti nějaké vlastnosti pro a vyplývá její dokazatelnost pro $a+1$ a jestliže je tato vlastnost dokazatelná pro 0 , pak tato vlastnost platí pro všechna přirozená čísla.
- Jestliže $a+1 = b+1$, potom $a = b$.
- Přičtením 1 k přirozenému číslu nelze nikdy získat číslo 0 .

Čísla, která splňují axiomy, od prvního až posledního bodu jsou právě přirozená čísla.

- Odkazem (referencí) na jiný typ s diferenciací – StálýZaměstnanec je Zaměstnanec se stálou smlouvou.

Hierarchie typů



Typy lze organizovat do hierarchií. Hierarchie typů tvoří svaz, v němž je částečné uspořádání dáno relací subtypu (označuje se $A = B$, je-li A subtypem B – např. pták = živočich).

Obrázek 3.4: Hierarchie typů

Zápis hierarchie typů v lineární formě (PROLOG + CG)

Entita > Životný, Neživotný

Životný > Osoba, Zvíře

....

3.1.2 Relace subtypu

Jestliže $A = B$ (A je subtypem B), pak

- buď A je B
- nebo A je specializací B

Reprezentace znalostí

Např. typ A (Pták) je specializací typu B (Zvíře).

Specializace A jiného typu B zahrnuje všechny vlastnosti B s přidáním některých dalších omezení – např. „má křídla“, „klade vejce“,...

Aby se odlišila identita (Zvíře = Zvíře) od vlastního subtypu, zapisuje se $A < B$.

Pro relaci subtypu platí, že je tranzitivní.

3.1.3 Entita a absurdita

Entita je univerzální typ – typ, který je supertypem všech ostatních typů v hierarchii. Typ Entita reprezentuje libovolnou entitu.

Absurdita je absurdní typ – typ, který je subtypem všech ostatních typů v hierarchii.

Dědičnost

Je-li $A = B$, A dědí všechny vlastnosti B. Savci dědí všechny vlastnosti Zvířete.

Částečné uspořádání subtypu je reflexivní, antisymetrické a tranzitivní.

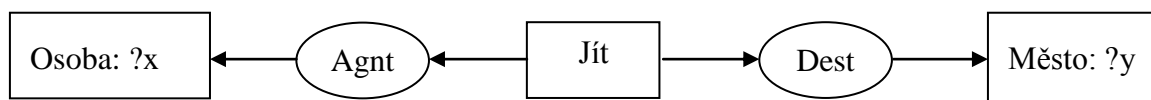
3.1.4 Lambda výrazy pro definování typů konceptů a vztahů

Jde o grafy, které umožňují formulovat obecná tvrzení.

Příklad 3.5

„Někdo jde do města.“

$[Osoba: ?x] \leftarrow (Agnt) \leftarrow [Jet] \longrightarrow (Dest) \longrightarrow [Město: ?y]$



Definice 3.6 (návěští)

Definice návěští má dvě části.

- Část pojmenovávací („vztah $JetDoAalborgu(*x)$ je“).
- Část definiční (konceptuální graf).



Příklad 3.6

$[Profesor: Alfred] \leftarrow (JetDo Aalborgu)$

Koncept $[Profesor: Alfred]$ je aktuálním parametrem lambda výrazu

$[Profesor: Alfred] \leftarrow ([Osoba: ?x] \leftarrow (Agnt) \leftarrow [Jet] \longrightarrow (Dest) \longrightarrow [Město: Aalborg])$



Reprezentace znalostí

[Profesor: Alfred] \longleftarrow (Agnt) \longleftarrow [Jet] \longrightarrow (Dest) \longrightarrow [Město: Aalborg]

Profesor je subtypem Osoba.



Definice 3.7 (signaturey)

Jestliže v n-adickém lambda výrazu e typem c_1 je t_1, \dots , typem c_n je t_n , potom signaturu e tvoří $\langle t_1, t_2, \dots, t_n \rangle$



Příklad 3.7

Vztah $\text{JetDo}(*x, *y)$ je

[Osoba: ?x] \longleftarrow (Agnt) \longleftarrow [Jet] \longrightarrow (Dest) \longrightarrow [Město: ?y] diadický
lambda výraz se signaturou $\langle \text{Osoba}, \text{Město} \rangle$

Vztah $\text{JetDoAalborgu}(*x)$ je

[Osoba: ?x] \longleftarrow (Agnt) \longleftarrow [Jet] \longrightarrow (Dest) \longrightarrow [Město: Aalborg]
monadický lambda výraz se signaturou $\langle \text{Osoba} \rangle$

Druhé lze vyjádřit pomocí prvního:

[Osoba: ?x] \longrightarrow [JetDo] \longrightarrow [Město: Aalborg]

3.2 Ontologie v konceptuálních grafech

Každá znalostní báze musí mít hierarchii konceptů T a každý její konceptuální graf se musí vztahovat k T .

3.2.1 Typy konceptů



Definice 3.8 (hierarchie typů konceptů)

Hierarchie typů konceptů T je síť (svaz) návěští typů, která mohou být dvojího typu:

- Primitivní, která jsou pouhými jmény zařaditelnými do hierarchie T
- Defínovaná pomocí monadických lambda výrazů

Každá hierarchie typů obsahuje dvě primitivní typová návěští – entitu a absurditu.

Všechny koncepty mají typ konceptu. Pro každý koncept je typem konceptu buď návěští z T nebo monadický lambda výraz.

To znamená, že v hierarchii typů není třeba všech možných typů. Ty se dodefinují monadickým lambda výrazem.

Typ konceptu nemůže zůstat na rozdíl od referenta prázdný, např. [: Jan].

Typem konceptu může být

- Návěští typu z T

Reprezentace znalostí

- Monadický lambda výraz.

3.2.2 Typy vztahů

Definice 3.9 (hierarchie typů vztahů)

Hierarchie typů vztahů R je síť (svaz) návěští typů, která mohou být dvojího typu:

- Primitivní, která jsou pouhými jmény zařaditelnými do hierarchie R
- Definovaná pomocí n-adických lambda výrazů

Primitivní návěští typu vztahu jsou zařaditelná do R.

Definovaná návěští typů vztahů sestávají z

- Jména
- N-adického lambda výrazu – definice např. vztah JetDo

Vztah nemusí být návěštím vztahu, lze využít přímo definice (lambda výrazu).



Příklad 3.8

Vztah JeModrý:

[Hračka: Míč]) ← ([Předmět: ?x] → (Attr) → [Modrý])

n-adické návěští vztahu může být subtypem pouze opět n-adického návěští vztahu (stejná valence).



3.2.3 Referenty

Nejdůležitější druhy referentů:

- **Prázdný referent:** např. [Auto] - „existuje auto“, [Auto: Felicia] „existuje auto jménem Felicia“.
- **Lokátor:** individuální označení = pojmenovaný referent
Lokátor je referent, který říká, kde lze nalézt individuum, k němuž směřuje reference – fyzický svět, katalog individuí.
Existují dva druhy lokátorů

1. individuální označovače
2. indexace

1. Individuálními označovači jsou např. jména osob, míst, organizací.
Např.

[Osoba: Jan] ← (Expr) ← [Žít] → (Loc) → [Město: Aalborg]
“Jan žije v Aalborgu“. Zde jsou lokátory Jan a Aalborg, které jsou individuálními označeními.

2. Indexační referenty jsou používány, je-li třeba referovat na individua, jejichž identita závisí na kontextu. Zde # označuje „určitý“. Např.

Reprezentace znalostí

[Mléko: #] ← (Thme) ← [Pít] → (Agnt) → [Kočka: #].
„Určitá kočka pije určité mléko“.

- **Množina věcí:**

[Host: {Tom, Julia, Brad}] ← (Agnt) ← [Zpívat] → (Thme)
→ [Píseň: HappyBerthdayToYou] → (To) → [Osoba: Alfred]
Hosté Tom, Julia a Brad zpívají Alfrédovi píseň Happy Birthday To You.

Speciální množinou je {*} - např. [Pták: *]

[Zpívat] → (Agnt) → [Pták: *] → (In) → [Třešeň] – „Ptáci zpívající na třešni“.

- **Deskriptory:** Celé grafy mohou být referenty.

(Past) → [Situace: [Osoba: Jan] ← (Agnt) ← [Políbit] → (Benf) → [Osoba: Mary]].

Zde konceptuální graf [Osoba: Jan] ← (Agnt) ← [Políbit] → (Benf) → [Osoba: Mary] je referentem konceptu typu Situace. Zde graf vyplňuje slot referenta.

- **Počty:** Pomocí symbolu @ lze stanovit, kolika individuí se referent týká.

[Osoba] → (Has) → [Noha: @2]

- **Univerzální kvantifikace** – Př. [ŽijícíRyba: ∀] → (Attr) →

[Mokrý]

“Všechny žijící ryby jsou mokré“.



Definice 3.10 (referenta)

Koncept má typ a referenta: [typ: referent]

Referent sestává z

- Jeho kvantifikátoru
- Jeho designátoru

Kvantifikátory mohou být

- **Existenční kvantifikátor:** je reprezentován buď prázdným referentem, nebo symbolem \exists . Např. [Kočka] → (On) → [Matrace]. Někjaká kočka je na nějaké matraci.
- **Definovaný kvantifikátor:** viz např. univerzální kvantifikaci nebo množinu výčtem prvků nebo množinu s * (viz výše).

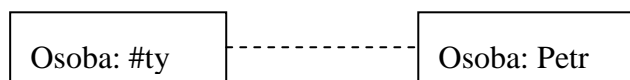
Designátory mohou být

- **Literál** je syntaktický reprezentant formy referentu. Jsou tři typy literálů:
 1. číslo - [Číslo: 18] (rozlišit od kvantifikátoru @18, který udává počet)
 2. string - [String: abcd],
 3. kódovaný literál - [Míra: <18, cm>].
- **Lokátor** indikuje způsob vyhledání referenta - [Kočka : Micka], [Student: #ty] - (viz výše)
- **Deskriptor** – konceptuální graf je deskriptorem referenta (viz výše)

Reprezentace znalostí

Koreferenti – dva nebo více konceptů v grafu mohou referovat totéž individuum. V grafu čárkovaná čára – koreferenční spojení.

[Osoba: #ty] ---- [Osoba: Petr] „Ty jsi Petr“.



3.2.4 Zahnížděné grafy

Definice 3.11 (zahnížděného grafu)

Graf G1 je zahnížděn v konceptu C, jestliže buď G1 je přímou součástí referenta C nebo G1 je přímou částí referenta konceptu C2, který je zahnížděn v C.



Příklad 3.9

[Předpoklad: [Profesor: Alfred] → (Attr) → [Dobrý]] „předpokládá se, že Alfréd je dobrý profesor“.



3.2.5 Kontext

Definice 3.12 (kontextu)

Kontext C je koncept, jehož designátorem je neprázdný konceptuální graf G.

V předcházejícím příkladu tvoří Předpoklad kontext, v němž se vyskytuje subgraf.



3.3 Konceptuální graf a Logika

Každý konceptuální graf lze převést do formule predikátové logiky.

V konceptuálním grafu se symbol negace umísťuje výhradně před kontext (Situační, Předpoklad,...).

Příklad 3.10

¬[Situační: [Slunce: #] ← (Agnt) ← [Svítit]]

„Slunce nesvítit“.

¬[Předpoklad: [Muž] ← (Expr) ← [Rozumět] → (Thme) → [Žena: {*}]]

„Není pravda, že existuje muž, který rozumí ženám“.



Reprezentace znalostí

3.3.1 Konjunkce v konceptuálním grafu

V konceptuálním grafu se konjunkce vyjadřuje výhradně v rámci téhož kontextu, a to bez spojovacích vztahů nebo linií.



Příklad 3.11

[Předpoklad: [Žena: *x] \longrightarrow (Attr) \longrightarrow [Krásná]

[Žena: *x] \longrightarrow (Attr) \longrightarrow [Nebezpečná]]

„Existuje žena, která je krásná a nebezpečná“.

3.3.2 Disjunkce v konceptuálním grafu

V konceptuálním grafu se disjunkce $G1 \vee G2$ řeší pomocí konjunkce a negace – DeMorganovým pravidlem:

1. Negace kontextu přes G1
2. Negace kontextu přes G2
3. Zařazení 1. a 2. do společného kontextu
4. Negace 3.



Příklad 3.12

\neg [Situace:

\neg [Situace:

[Osoba: Jan]- - [Blázen]

]

\neg [Situace:

[Osoba: Jan] \longrightarrow (Attr) \longrightarrow [Chytrý] \longrightarrow (Meas) \longrightarrow [Stupeň:
#velmi]

]

]



Kontrolní otázky a úkoly

1. Co to znamená, když se řekne, že graf je bipartitní?
2. Navazuje konceptuální graf na asociativní síť?
- 3.

Reprezentace znalostí

Shrnutí

Další z forem reprezentace znalostí je konceptuální graf, který má na rozdíl od sémantické sítě větší expresivitu. Konceptuální grafy byly vytvořeny na základě existenciálních grafů od Charles Sanders Peirce (1909) a autorem byl J. F. Sowa, jehož práce započala roku 1976. Konceptuální grafy byly původně vyvíjeny jako grafy v textové podobě (lineární notace) pro snadné používání počítačových programů. Konceptuální grafy jsou schopné reprezentovat širokou škálu znalostních tříd – od predikátové logiky k přirozeným jazykům.

Jazyk konceptuálních grafů je v podstatě založen na logice prvního řádu s tradiční deklarativní sémantikou s řadou výjimek, které z rámce logiky prvního řádu vybočují. Naším cílem bylo zachování všech požadavků na logiku prvního řádu.



4 Deskripční logika

Cílem kapitoly je:

- vysvětlení pojmu konceptově orientované jazyky a deskripční logika,
- určení vztahu deskripční logiky k informatice,
- vysvětlení vztahu deskripční logiky k logice prvního řádu,
- definice jazyku deskripční logiky,
- definice znalostní báze v deskripční logice.

Po jejím prostudování byste měli být schopni:

- definovat konceptově orientované jazyky a deskripční logiku,
- rozlišovat rozdíl mezi logikou prvního řádu a deskripční logiky,
- definovat jazyk LDL a LDL1,
- dokazovat logický důsledek pomocí sémantických tabel.

Klíčová slova:

Deskripční logika, asociativní sítě, rámce, logika prvního řádu, ABox, TBox, syntax a sémantika jazyka, znalostní báze, logický důsledek, sémantické tablo.

Čas potřebný ke studiu: 10 hodin



Průvodcem studia

Tradiční formálně logické systémy jsou založeny na metodách přesného usuzování, s tím že závěry jsou přesné (mají být přesné). Není zde možnost pochybení v pravidlech, která jsou dána přesně. Dedukce vyžaduje, aby informace, na niž je prováděna, byly kompletní, přesné a konzistentní. Naopak skutečný svět vyžaduje usuzování „selským rozumem“ („zdravým rozumem“) navzdory nekompletní, nepřesné, potencionálně nekonzistentní informaci.

4.1 Konceptově orientované jazyky a deskripční logika

Deskripční logika jako logický reprezentant konceptových jazyků bývá charakterizována jako sjednocující formalismus pro reprezentaci v systémech založených na rámcích, asociativních sítích, objektově orientované reprezentaci znalostí, sémantických datových modelech, relačních databázích a v neposlední řadě na snaze o vytvoření ontologického jazyka nejvyšší úrovně.

Z hlediska logiky prvního řádu je deskripční logika její strukturovaný fragment. Formální systém deskripční logiky je určen jazykem, znalostní bází, která představuje soubor speciálních axiomů pro dedukci, a pravidly (mechanismem) tuto dedukci umožňujícími.

Reprezentace znalostí

4.1.1 Asociativních sítě, rámce a konceptové jazyky

Společnou myšlenkou asociativních sítí, rámců a konceptových jazyků je reprezentovat znalosti ve formě tříd a vztahů mezi nimi, včetně množinové inkluze, umožňující zachycení hierarchických struktur.



Problémy s asociativními sítěmi, jimž bude dále věnována samostatná kapitola, souvisí s jejich nejasnou sémantikou, způsobenou především zařazováním tvrzení o obecných třídách (kategoriích) s tvrzeními, týkajícími se instancí tříd, do společných sítí. Hrany v sítích reprezentují různé druhy vztahů mezi uzly – jsou zde vedle sebe ty, které reprezentují znalosti obecného charakteru, spolu s těmi, které reprezentují konkrétní instance těchto obecných znalostí.

Rámec zpravidla reprezentuje koncept (třidu) definovaný identifikátorem a elementy zvanými sloty – každý slot odpovídá atributu, který má mít příslušný koncept. Hodnoty atributů jsou buď prvky konkrétní domény, nebo identifikátory dalších rámců. Rámec může též reprezentovat samostatné individuum s atributem *isa* („is a“) ve vztahu ke třídě, o jejíž instanci jde. Obecně rámce nedisponují dostatečně vyhovujícím odvozovacím systémem. Snaha napravit tento nedostatek vedla k vytvoření systému KL (1985), KL-ONE a řady dalších. Rámce se podobně jako asociativní sítě vyznačují nevyjasněnou sémantikou.

4.1.2 Deskripční logika a logika prvního řádu

V konceptově orientovaných jazycích jsou koncepty používány k reprezentaci tříd jako množin prvků a pro účely specifikace jejich vlastností nebo vztahů jsou určeny binární relace reprezentující role. Koncepty jsou uspořadatelné do hierarchických struktur, které jsou terminologickými slovníky, nazývanými též ontologiemi.



Deskripční logiku je formální systém disponující speciálním konceptově orientovaným jazykem pro reprezentaci znalostí, vyvinutým jako reakce na některé problémy související s jinými systémy reprezentace znalostí. Na jedné straně zde stojí vysoce expresivní logika prvního řádu, nedisponující dostatečně efektivními prostředky vyjádření znalostí v jejich souvislostech a hierarchiích, na druhé straně jsou to rámce a asociativní sítě, které sice prostředky vyjádření vzájemných souvislostí znalostí a hierarchií pojmů disponují, avšak nejsou vybaveny dostatečně vyhovujícími odvozovacími prostředky.

Myšlenkou deskripční logiky, vycházející ze zjednodušené verze logiky prvního řádu, připouštějící pouze unární a binární predikáty, je reprezentovat znalosti ve formě pojmů (konceptů) a vztahů mezi nimi (rolí), včetně inkluze množin, umožňující zachycení hierarchických struktur.

Hlavním charakteristickým rysem systémů deskripční logiky je to, že znalostní báze je tvořena dvěma komponentami: První komponentou je tzv. TBox (terminologický box), tj. obecné schéma týkající se tříd individuí, která mají být reprezentována, jejich obecných vlastností a vzájemných vztahů. Na formulích TBoxu lze formálně rozhodovat jejich splnitelnost, resp. jejich vzájemnou subsumpci, tj. zdali jedna z nich je obecnější než druhá. Druhou



Reprezentace znalostí

komponentou znalostní báze je ABox (assertional) instancí tohoto schématu, obsahující tvrzení týkající se vzájemných vztahů tříd a individuí. Rozhodování o ABoxu se týká především jeho splnitelnosti, tj. zdali existuje jeho model. Na základě tvrzení z ABoxu lze dále odvozovat, zdali jsou určitá individua instancemi některých tvrzení z TBoxu.

Ve srovnání s logikou prvního řádu se deskripční logika vyznačuje nižší expresivitou, způsobenou omezením pouze na unární a binární predikáty, na druhé straně však disponuje efektivními prostředky snadného vyjádření a manipulace s koncepty v jejich vzájemných strukturovaných souvislostech, včetně hierarchií.

Deskripční logika, stejně jako jiné formální přístupy reprezentace znalostí představuje formalismus vztahujících se vždy k určité doméně – reprezentovanému světu. Reprezentace spočívá v definování relevantních konceptů v uvažovaném světě, zpravidla prostřednictvím rolí, a následném využití definovaných konceptů k formálnímu popisu uvažovaného světa. Znalosti reprezentované ve znalostních bázích deskripční logiky jsou formulovány jako ontologicky strukturované informace, umožňující následné odvozování. Z hlediska logiky prvního řádu je deskripční logika její strukturovaný fragment.

4.1.3 Některé implementace deskripční logiky

První implementovanou deskripční logikou byl Brachmanův (1977) systém KL-ONE, ovlivněný *rámčovými* (frame) systémy. Jak popisuje McGregor v roce 1991 v jedné z prvních prací, hodnotících implementace DL, systémy vyvinuté v té době byly BACK [viz literatura], který se měl stát informační bází určité obchodní společnosti, CLASSIC jako báze určité komerčně zaměřené aplikace, K-REP, který tvořil informační bázi komerční aplikace v lékařství, LOOM pro využití k celé řadě vládních výzkumných programů.

Charakteristické pro toto první období implementací DL bylo zaměření na tzv. strukturální *subsumpční* algoritmy, které spočívají v normalizaci komponovaných konceptů a následném rozhodování na základě porovnávání, zdali je některý koncept subsumován konceptem jiným. Tyto algoritmy mají v případě deskripčních logik s nízkou expresivitou polynommickou složitost. Typický pro toto období vývoje byl též omezený epistemologický slovník *epistemologických primitivů*, který bylo potřeba udržovat a doplňovat.

Je nesporné, že platí přímá úměra mezi expresivitou jazyka (vybaveností pokročilými konstruktory) a výpočtovou složitostí základních rozhodovacích algoritmů. Pozdější implementace DL, tj. DLP, FACT a RACE se vyznačovaly vyšší expresivitou a zároveň přiměřenými výpočtovými možnostmi. V těchto systémech již nebyl limitován počet *epistemologických primitivů* ve slovnících. Pro rozhodování a odvozování ze znalostních bází byly využívány tablové algoritmy.

Zjištění, že DL se svým vědecky podloženým formálním základem je právě tou bází, na které mohou stavět jazyky XML, RDFS pro vývojově vyšší generaci reprezentace znalostí na webu, vedlo k vývoji ontologického jazyka OIL

Reprezentace znalostí

(Fensel et al, 2001) a jazyka DAML – Darpa Agent Markup Language (Hendler-McGuinness, 2000). V současné době je OWL, vyvinutý konsorciem W3C jako následník DAML+OIL a SHOQ, přijímán jako standardní ontologický jazyk pro *sémantický web*.

4.2 Jazyk deskripční logiky

4.2.1 Jazyky LDL a LDL1 deskripční logiky

V této knize budou uvedeny dva jazyky deskripční logiky L_{DL} a L_{DL1} , lišící se především v sémantice univerzálně kvantifikovaného konstrukturu. Jazyk L_{DL} „rozhodnutelné“ deskripční logiky DL, zavedený původními autory, se vyznačuje, jak bude ukázáno dále, tak závažnými nedostatky, že se autoři rozhodli definovat jeho modifikovanou verzi L_{DL1} v deskripční logice DL1 s vlastnostmi odpovídajícími logice prvního řádu.

4.2.2 Syntax jazyka deskripční logiky

Gramatiky různých modifikací jazyka deskripční logiky se od sebe liší výběrem konstrukturu. FL - jazyky [3] zahrnují konstruktory konjunkce, univerzální kvantifikace a nekvalifikované existenční kvantifikace. AL - jazyk je pak rozšířením jazyka FL o konstruktor negace (komplementu) a speciální konstruktory \perp a \top . Superjazyky jazyka FL (podobně AL) jsou vždy určeny řetězcem např. $FL[E][U][C][R][N][O]$.

Jazyk deskripční logiky L_{DL} (L_{DL1}), který zde bude zaveden, je speciální konceptově orientovaný jazyk pro reprezentaci *pojmu*, tj. *konceptů* (tříd, kategorií), *rolí*, tj. vlastností a vztahů mezi koncepty a jejich *individuálními objekty*, vygenerovaný nad danou *abecedou* množinou *konstrukturu*.

Syntax jazyka je v DL i DL1 definována stejně, tj. touž abecedou a týmiž syntaktickými pravidly.



Definice 4.1 (abecedy jazyka deskripční logiky)

Abecedu jazyka deskripční logiky tvoří

- množina N_C symbolů pro jména konceptů - unárních predikátových symbolů, které reprezentují třídy objektů
- množina N_R symbolů pro jména rolí - binárních predikátových symbolů, které reprezentují vztahy (relace) mezi dvojicemi objektů
- množina N_O symbolů pro jména individuálních objektů
- pomocné symboly - závorky.

Pro jména individuálních objektů platí v jazyce deskripční logiky L_{DL} definovaném v „*předpoklad jedinečného jména*“, tj. předpoklad, že neexistuje dvojice navzájem různých jmen pojmenovávajících též objekt. V jazyce L_{DL1} tento předpoklad neplatí.



Reprezentace znalostí

Bázi konceptového jazyka deskripční logiky tvoří *atomické koncepty* – unární predikátové symboly z množiny N_C a *atomické role* – binární predikátové symboly z množiny N_R , z nichž se vytvářejí komponované koncepty a role pomocí gramatických pravidel.

Gramatická pravidla tvoří množina konstruktorů pro vytváření formulí definujících na bázi atomických formulí další, zpravidla komponované konceptové formule (*Cformule*) a komponované formule rolí (*Rformule*), včetně formulí reprezentujících tvrzení o příslušnosti individuálních objektů k uvažovaným konceptům nebo rolím.

V následující definici jsou uvedeny konstruktory, které se zpravidla vyskytují v *AL-jazycích*, z nichž nejužívanější je *ALC* s konstruktory negace, konjunkce, disjunkce, omezením hodnoty a existenčním omezením. Deskripční logika označovaná jako *ALCN* disponuje navíc konstruktorem omezení počtem.



Definice 4.2 (gramatiky jazyků LDL a LDL1 Backus–Naurovou formou)

$\langle \text{atomická role} \rangle ::= R / S / \dots$
 $\langle Rformule \rangle ::= \langle \text{atomická role} \rangle$
 $::= (\langle Rformule \rangle) \sqcap (\langle Rformule \rangle)$ (konjunkce rolí - *R*)
 $::= (\langle Rformule \rangle)^{\bar{}}$ (inverze role - *I*)
 $::= \neg(\langle Rformule \rangle)$ (negace role)
 $\langle \text{instance role} \rangle ::= \langle Rformule \rangle(\langle \text{individuum} \rangle, \langle \text{individuum} \rangle)$
 $\langle \text{individuum} \rangle ::= a / b /, a_1 /, \dots$
 $\langle \text{atomický koncept} \rangle ::= A / B / \dots$
 $\langle Cformule \rangle ::= \langle \text{atomický koncept} \rangle$
 $::= \top$
 $::= \perp$
 $::= \neg(\langle Cformule \rangle)$ (negace konceptu - *C*)
 $::= (\langle Cformule \rangle) \sqcap (\langle Cformule \rangle)$ (konjunkce konceptů)
 $::= (\langle Cformule \rangle) \sqcup (\langle Cformule \rangle)$ (disjunkce konceptů - *U*)
 $::= (\langle Rformule \rangle)$ (nekvalifikované existenční omezení - *E*)
 $::= \exists(\langle Rformule \rangle).(\langle Cformule \rangle)$ (existenční omezení hodnoty - *E*)
 $::= \forall(\langle Rformule \rangle).(\langle Cformule \rangle)$ (univerzální omezení hodnoty)
 $::= \exists^{>n}(\langle Rformule \rangle)$ (omezení počtem - *N*)
 $::= \exists^{\leq n}(\langle Rformule \rangle)$ - “-”
 $\langle \text{instance konceptu} \rangle ::= \langle Cformule \rangle(\langle \text{individuum} \rangle)$

Výrazy vytvořené podle uvedených gramatických pravidel nevyžadují explicitní zápis četnosti (arity) příslušných (predikátových) symbolů určených k reprezentaci konceptů nebo rolí, neboť podle pozice v jednotlivých konstruktech jsou rozlišitelné unární koncepty od binárních rolí.

Konstruktory nemusí být nutně vzájemně nezávislé, jedny mohou simulovat jiné. Využívá se především následujících přepisů podle deMorganových pravidel:



$$\neg(C \sqcup D) \equiv \neg C \sqcap \neg D$$

$$\neg(C \sqcap D) \equiv \neg C \sqcup \neg D$$

Reprezentace znalostí

Příklad 4.1

Příklad reprezentace konceptu "Člověk, který si vzal lékařku a má nejméně 5 dětí, které jsou pedagogy." s využitím konstruktorů existenčního a univerzálního omezení a omezení počtem:

$$\text{člověk} \sqcap \neg \text{žena} \sqcap \exists \text{manželství} . (\text{žena} \sqcap \text{lékař} \sqcap \exists^{>5} \text{má_dítě} \sqcap \forall \text{má_dítě} . \text{pedagog})$$



4.2.3 Sémantika jazyka deskripční logiky

Obdobně jako je tomu v logice prvního řádu, je třeba v deskripční logice uvažovat pojmy pravdivosti, splnitelnosti a platnosti konstruktů jazyka v dané interpretaci, určené svou strukturou.

Definice 4.3 (struktury a pravidel interpretace)

Interpretace jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) deskripční logiky je určena svou strukturou a interpretačními pravidly. Přitom strukturu interpretace $I = (\Delta^I, \bullet^I)$ tvoří neprázdná množina objektů Δ^I (universum diskursu) a funkce \bullet^I (denotační funkce interpretace I), která přiřazuje každému konceptu C extenzi konceptu, tj. podmnožinu $C^I \subseteq \Delta^I$ objektů, pro něž koncept C platí (je pravdivý), a každé roli R extenzi role, tj. podmnožinu $R^I \subseteq \Delta^I \times \Delta^I$ dvojic objektů, pro něž role R platí (je pravdivá).

Interpretační pravidla stanoví extenze jednotlivých typů konceptů a rolí vytvořených konstruktory jazyka:



$$\begin{aligned} \top^I &= \Delta^I = \Delta \\ \perp^I &= \emptyset \\ (\neg C)^I &= \Delta^I \setminus C^I \\ (C \sqcap D)^I &= C^I \cap D^I \\ (C \sqcup D)^I &= C^I \cup D^I \\ (\forall R . C)^I &= \{x \in \Delta^I \mid \forall y (R(x,y) \rightarrow C(y))\} && \text{pro jazyk } \mathbf{L}_{DL} \\ (\forall R . C)^I &= \{x \in \Delta^I \mid \exists t R(x,t) \ \& \ \forall y (R(x,y) \rightarrow C(y))\} && \text{pro jazyk } \mathbf{L}_{DL1} \\ (\exists R . C)^I &= \{x \in \Delta^I \mid \exists y (R(x,y) \ \& \ C(y))\} \\ (\exists^{>n} R)^I &= \{x \in \Delta^I \mid \# \{y \mid R(x,y)\} > n\} \\ (\exists^{\leq n} R)^I &= \{x \in \Delta^I \mid \# \{y \mid R(x,y)\} \leq n\} \\ (R \sqcap S)^I &= R^I \cap S^I \\ (R^-)^I &= \{(y,x) \mid R(x,y)\} \end{aligned}$$

Reprezentace znalostí

Poznámky:

Některé varianty jazyka disponují pouze jednodušší podobou výše uvedené definice (kvalifikovaného) existenčního omezení ($\exists R.C$) a jeho sémantiky.

Konstruktor *nekvalifikovaného existenčního omezení* má tvar $\exists R$, resp. $\exists RT$ a význam $(\exists R)^I = \{x \in \Delta^I \mid \exists y R(x,y)\}$.

Charakteristickým rysem deskripční logiky jsou konstruktory pro stanovení vztahů mezi koncepty a rolemi. Konstruktory $\forall R.C$, $\exists R.C$ provází hodnotové omezení vyjadřující typy objektů, které mohou vyplňovat role (pro rozlišení funkcí atributů v relacích individuální objekt tvořící druhý atribut je zpravidla nazýván výplní role). Dalším typickým omezením je omezení (N) kardinality množin výplní rolí.



Definice 4.4 (splnitelnosti konceptu /role)

Koncept (role) je v dané interpretaci *splnitelný* (-á), je-li jeho (její) extenze neprázdná.

4.2.4 Subsumpce konceptů a taxonomie



Definice 4.5 (subsumpce konceptů)

Nechť C a D jsou koncepty. C je *subsumován* D , tj. $C \sqsubseteq D$, právě když platí $C^I \subseteq D^I$ pro každou interpretaci I , C je *ekvivalentní* D , tj. $C \equiv D$, právě když platí $C^I \equiv D^I$ pro každou interpretaci I .



Příklad 4.2

Příklady subsumpcí konceptů :

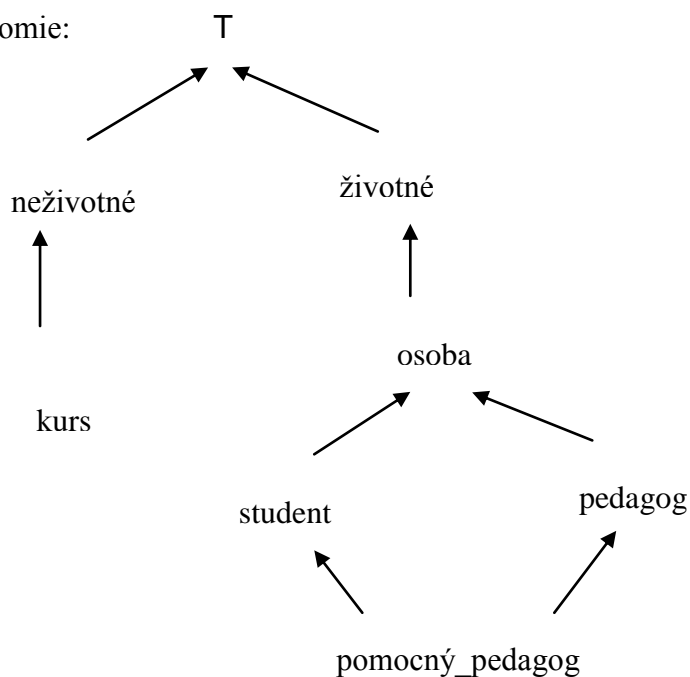
1. $\text{dospělý} \sqcap \text{muž} \sqsubseteq \text{dospělý}$
2. $\text{dospělý} \sqcap \text{bohatý} \sqcap \text{muž} \sqsubseteq \text{dospělý} \sqcap \text{muž}$
3. $\forall \text{ má_dítě} . (\text{dospělý} \sqcap \text{muž}) \sqsubseteq \forall \text{ má_dítě} . \text{dospělý}$

Subsumpce představuje v prostoru konceptů relaci *částečného uspořádání*. Základem hierarchické klasifikace je pak pojem *taxonomie*, představující minimální subsumpci, která je reflexivně-transitivním uzávěrem v prostoru konceptů. Klasifikace představuje zařazení konceptu do taxonomie.

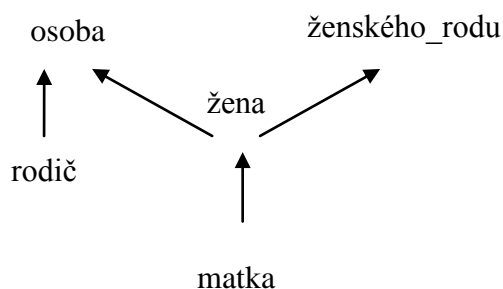
Reprezentace znalostí

Příklad 4.3

Příklady taxonomie:



Obrázek 4.1: Příklad taxonomie



Obrázek 4.2: Příklad taxonomie

4.3 Znalostní báze v deskripční logice

4.3.1 Terminologické znalostní systémy

V rámci umělé inteligence jsou systémy založené na znalostech takové, v nichž jsou znalosti reprezentovány explicitně pomocí vhodného jazyka a umožňují tak odvozování dalších implicitně obsažených znalostí pomocí odvozovacího mechanismu. Báze znalostí spolu s odvozovacím mechanismem se zpravidla nazývá *systém reprezentace znalostí*. Komunikace mezi systémem a znalostní bází se pak děje pomocí dotazů a odpovědí.

Terminologické (taxonomické) systémy reprezentace znalostí se vyznačují tím, že se v nich budují struktury termínů uspořádaných do hierarchií. Typickým

Reprezentace znalostí

využitím v praxi je zde možnost reprezentace hierarchicky strukturovaných informací na Webu.

Reprezentace znalostí založená na *konceptově orientovaném* přístupu rozlišuje mezi obecnými terminologickými znalostmi a konkrétními tvrzeními. První zacházejí s pojmy (koncepty) a jejich vztahy, druhé se týkají individuí a jejich příslušnosti konceptům a jejich vztahům.

Tyto systémy vycházejí z popisu (deskripce) pojmů - *konceptů* a jejich vlastností a vzájemných vztahů, resp. jejich rolí v modelovaném světě. Přístup tohoto typu je charakteristický též ve vývojově předcházejících systémech založených na rámcích, asociativních sítích, objektově orientované reprezentaci znalostí, sémantických datových modelech, relačních databázích.

4.3.2 Znalostní báze

Formální systém deskripční logiky je určen jazykem L_{DL} (L_{DL1}) vytvořeným speciálně pro formální zápis znalostí o nějakém konkrétním *modelovaném světě* (referenčním systému) do *znalostní báze*, která představuje soubor *speciálních axiomů* pro dedukci, a *pravidly* (mechanismem) tuto dedukci umožňujícími. Znalostní bázi přitom zpravidla tvoří její dvě součásti - *TBox* a *ABox*. Uspořádání znalostí do dvou částí znalostní báze má svůj důvod ve způsobu, jakým je možno definovat význam konceptů a rolí, reprezentovaných v deskripční logice unárními a binárními predikátovými symboly. Význam konceptu, event. role lze totiž definovat buď pomocí *extenze*, tj. výčtem prvků příslušné unární, resp. binární relace (denotátem příslušného predikátu v rámci dané interpretační struktury), nebo určením jejich *intenze*, tj. formální specifikací vlastností (pomocí formule, resp. množiny formulí) v reprezentujícím jazyce.



Definice 4.6 (znalostní báze)

Je-li dán jazyk L_{DL} (L_{DL1}) deskripční logiky pro reprezentaci znalostí o referenčním systému W , je *znalostní bázi* Σ dvojice $\Sigma = (T, A)$, kde T je *TBox intenzionálních znalostí* o W a A je *ABox extenzionálních znalostí* o W , reprezentovaných v L_{DL} (L_{DL1}).

4.3.3 TBoxy znalostních bází



Definice 4.7 (TBoxu)

TBox je *terminologický box*, který představuje množinu (významem univerzálně kvantifikovaných) formulí jazyka L_{DL} (L_{DL1}) jimiž se reprezentují *intenzionální znalosti*. Platí, že

- *Formule (termíny) TBoxu* představují ve znalostních bázích *speciální axiomy* a jsou výhradně definicemi, resp. specifikacemi některého z typů uvedených v **Chyba! Nenalezen zdroj odkazů.**

Reprezentace znalostí

- v TBoxu se objeví nejvýše jedna definice (specifikace) konceptu určitého jména,
- definice jsou (v základních typech jazyků deskripční logiky) acyklické.

Tabulka 4.1

	Typ výrazu TBoxu	syntax	sémantika
(1)	Specifikace atomického konceptu	$A \sqsubseteq T$	$A^I \subseteq \Delta^I$
(2)	Definice (rovnost) konceptů	$C \equiv D$	$C^I = D^I$
(3)	Specifikace (subsumpce) konceptů	$C \sqsubseteq D$	$C^I \subseteq D^I$

Axiomy TBoxu jsou trojího typu, tab. 4.1 obsahuje přehled syntaxe a sémantiky všech tří typů axiomů TBoxů (A je atomický koncept, C, D jsou koncepty).

Podle (1) se specifikují atomické koncepty TBoxů. *Konceptové jméno (termín)* A se vždy objeví na levé straně definice. Podobně se podle (2), (3) definuje koncept C na základě konceptu D , koncept C zde *užívá* koncept D ke své definici. V případě znalostní báze s cykly C užívá sebe sama (zde se další úvahy omezují na acyklické znalostní báze).

4.3.4 ABoxy znalostních bází

Zatímco formule TBoxu koncepty a jejich role, reprezentující intenzionální složku znalostní báze, definují v jejich vzájemných vztazích, formule ABoxu vyjadřují fakta neboli tvrzení o platnosti jednotlivých konkrétních instancí definovaných konceptů nebo rolí a reprezentují tak extenzionální složku znalostní báze

Definice 4.8 (ABoxu)

ABox (assertional box) je box formulí reprezentujících ve znalostních bázích jejich *extenzionální znalosti*. *Formule ABoxu* má některou z forem uvedených v následující tabulce (tab. 4.2), kde C je jméno konceptu, R je jméno role a a, b označují jména individuálních objektů jazyka.



Tabulka 4.2

Typ formule ABoxu	syntax	sémantika
příslušnost ke konceptu	$C(a)$	$a^I \in C^I$
příslušnost k roli	$R(a,b)$	$(a^I, b^I) \in R^I$

Konstrukce extenzionální komponenty znalostní báze, tj. ABoxu se realizuje použitím zavedených konceptů a rolí z TBoxu ve tvrzeních o individuálních objektech. Jinak řečeno, ABox lze vytvářet až tehdy, jsou-li již zavedeny příslušné koncepty a role v TBoxu (a je-li též prověřena konsistence TBoxu).

Reprezentace znalostí



Příklad 4.4

TBox :

- $\text{žena} \equiv \text{osoba} \sqcap \text{ženského_rodu}$
- $\text{muž} \sqsubseteq \text{osoba}$
- $\text{muž} \equiv \text{osoba} \sqcap \neg\text{žena}$
- $\text{rodič} \equiv \text{osoba} \sqcap \exists \text{ má_dítě}$
- $\text{matka} \equiv \text{žena} \sqcap \text{rodič}$
- $\text{otec} \sqsubseteq \text{rodič}$
- $\text{rodič} \equiv \text{matka} \sqcup \text{otec}$
- $\text{babička} \equiv \text{matka} \sqcap \exists \text{ má_dítě} . \text{rodič}$
- $\text{matka_více_dětí} \equiv \text{osoba} \sqcap \exists^{\geq 1} \text{ má_dítě}$

ABox :

- $\text{matka}(\text{marie})$
- $\text{matka_více_dětí}(\text{marie})$
- $\text{otec}(\text{petr})$
- $\text{má_dítě}(\text{petr}, \text{jan})$
- $\text{má_dítě}(\text{marie}, \text{jan})$



Příklad 4.5

Reprezentace následujících tvrzení v predikátové logice a deskripční logice DL (DL1):

"Student Jan Nový, adresa Zahradní 5, Ostrava 1, pracuje jako pomocný pedagog. Studentem je přitom každý, kdo má zapsán nějaký kurs, pedagogem je každý, kdo vyučuje nějaký kurs. Vyučujícím asistentem může být student postgraduálního studia nebo pedagog."

a) Predikátová reprezentace

$\forall x (\text{student}(x) \rightarrow (\text{osoba}(x) \ \& \ \exists y \text{ jméno}(x, y) \ \& \ \text{string}(y) \ \& \ \exists z \text{ adresa}(z) \ \& \ \text{string}(z) \ \& \ \exists w \text{ zapsán}(x, w) \ \& \ \text{kurs}(w)))$

$\forall x (\text{student}(x) \rightarrow \exists y \text{ zapsán}(x, y) \ \& \ \text{kurs}(y))$

$\forall x (\text{pedagog}(x) \rightarrow \exists y \text{ vyučuje}(x, y) \ \& \ \text{kurs}(y))$

$\forall x (\text{pomocný_pedagog}(x) \rightarrow \text{student}(x) \ \& \ \text{pedagog}(x))$

$\text{student}(\text{Jan_Nový}), \text{zapsán}(\text{Jan_Nový}, \text{DL02}), \text{kurs}(\text{DL02})$

Reprezentace znalostí

b) Reprezentace v deskripční logice

student \equiv osoba $\sqcap \exists$ jméno . string $\sqcap \exists$ adresa . string $\sqcap \exists$ zapsán . kurs

pomocný_pedagog \equiv student \sqcap pedagog

jsou formule typu (2) definující koncepty student a pomocný pedagog.

student $\sqsubseteq \exists$ zapsán . kurs

pedagog $\sqsubseteq \exists$ vyučuje . kurs

pomocný_pedagog \sqsubseteq student

pomocný_pedagog \sqsubseteq pedagog

vyučující_asistent \sqsubseteq postgrad_student \sqcup pedagog

jsou formule typu (3) – inkluze konceptů

student(Jan_Nový), zapsán(Jan_Nový, DL02), kurs(DL02) formulemi ABoxu jsou instancemi konceptů student a kurs a role zapsán.

Formule ABoxu připomínají jednotlivé instance (řádky) relační databáze s pouze unárními a binárními relacemi. Je zde však podstatný rozdíl, pokud jde o sémantiku. Zatímco instance relační databáze je konsistentní s jejím modelem, formule ABoxu, které jsou bázovými instancemi určitého konceptu, mohou být konsistentní s různými modely tohoto konceptu (téhož schématu databáze). Z chybějících prvků modelu konceptu nelze odvodit negativní tvrzení, neboť zde neplatí tzv. *předpoklad uzavřeného světa*, naopak se zde hovoří o sémantice *otevřeného světa*, kdy absence bázové instance konceptu pouze znamená absenci příslušné informace.

Pokud jde o sémantiku formulí ABoxu, je vždy odkázána, na sémantiku příslušného konceptu nebo role, jejichž je instancí.

Příklad 4.6

Stanovení extenzí komponovaných konceptů na základě extenzí atomických konceptů dospělý, muž, bohatý a atomické role dítě, jsou-li dány jejich extenze jako denotáty příslušných predikátů:

$D(\text{dospělý}) = \{\text{jana, petr, evžen, karel, david, anna, marta, klaudie}\}$, $D(\text{muž}) = \{\text{petr, evžen, karel, ivan, david, martin}\}$, $D(\text{bohatý}) = \{\text{petr, david}\}$, $D(\text{má dítě}) = \{(\text{jana, evžen}), (\text{ivan, david}), (\text{petr, david}), (\text{karel, anna}), (\text{klaudie, marta})\}$.

1. **dospělý** \sqcap **muž** má extenzi {petr, evžen, karel, david},
2. **dospělý** \sqcap **bohatý** \sqcap **muž** má extenzi {petr, david}
3. \forall **má dítě**. (**dospělý** \sqcap **muž**) má extenzi {jana, petr },
4. \exists **má dítě** má extenzi {jana, ivan, petr, karel, klaudie}.



4.4 Rozhodování v deskripční logice

Realizace terminologických systémů reprezentace znalostí, založených na deskripční logice, vyžaduje přinejmenším možnost řešení těchto problémů:

1. Rozhodování splnitelnosti znalostní báze, tj. splnitelnosti TBoxu spolu s ABoxem znalostní báze. Pozitivní odpověď vede k modelu znalostní báze, negativní znamená nebezpečí odvození sporu při formálních důkazech logických důsledků znalostní báze.
2. Rozhodování splnitelnosti konceptu C vzhledem ke znalostní bázi, tj. existence jejího modelu, v němž je extenze konceptu C neprázdná. To je podstatné nejen při zařazování dalších znalostí do TBoxu znalostní báze, ale též při řešení dotazů s vyloučením těch částí, které nepřispívají k obsahu odpovědi. Speciálně jde o rozhodování, zdali je koncept logickým důsledkem znalostní báze.
3. Rozhodování subsumpce konceptů: Je koncept D obecnější než koncept C , tj. platí $C \sqsubseteq D$? Jestliže ano, pak lze uvažovat logickou implikaci $C \rightarrow D$ ekvivalentní konceptovému výrazu $\neg C \sqcup D$.
4. Kontrola instancí: Je objekt a instancí konceptu C v nějakém modelu (ve všech modelech) znalostní báze?

Uvedené 4 problémy lze převést na problém splnitelnosti konceptu, jeho doplňku, znalostní báze rozšířené o nový koncept nebo jeho instanci.

4.4.1 Splnitelnost konceptu a znalostní báze

Pojem splnitelnosti konceptu (role) podobně jako v logice prvního řádu nelze v deskripční logice pojímat jinak než v souvislosti s danou strukturou interpretace I jejího jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) vhodně zvoleného pro popis (deskripci) nějakého referenčního systému.



Definice 4.9 (modelu, splnitelnosti konceptu, resp. role)

Struktura interpretace I jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) je *modelem* konceptu C , resp. role R , je-li C^I , resp. R^I neprázdná. Koncept (role) je *splnitelný(-á)*, má-li model, jinak je *nesplnitelný(-á)*.

K tomu, aby bylo možno definovat pojem splnitelnosti formulí znalostní báze, je třeba si ujasnit pojem splnitelnosti výrazů z jejího TBoxu, která má tvar subsumpce nebo ekvivalence konceptů (rolí), které nejsou formulemi deskripční logiky.

Věta 4.1 (o splnitelnosti formule TBoxu znalostní báze)

Nechť $T = C \sqsubseteq D$, resp. $T = C \equiv D$ je výraz TBoxu znalostní báze jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) s konstruktorem negace konceptu, C, D jsou koncepty. Potom je T v

Reprezentace znalostí

dané interpretaci I splnitelná, právě když je v I splnitelný koncept $\neg C \sqcup D$, resp. $((\neg C \sqcup D) \sqcap (\neg D \sqcup C))$.

Důkaz

Podle definice (4.5) významu subsumpce je C subsumován D , právě když $C^I \subseteq D^I$ pro každou interpretaci I , C je ekvivalentní D , právě když $C^I \equiv D^I$, tj. $C^I \subseteq D^I$ a současně $D^I \subseteq C^I$ pro každou interpretaci I . Přitom pro interpretující množiny C^I, D^I platí známý vztah inkluze $C^I \subseteq D^I$, právě když platí sjednocení $(\Delta^I \setminus C^I) \cup D^I$, jemuž odpovídá v DL (DL1) disjunkce konceptů $\neg C \sqcup D$. Ekvivalence konceptů je pak interpretována jako průnik příslušných sjednocení $((\Delta^I \setminus C^I) \cup D^I) \cap ((\Delta^I \setminus D^I) \cup C^I)$, jemuž odpovídá v DL (DL1) konjunkce příslušných disjunkcí konceptů $(\neg C \sqcup D) \sqcap (\neg D \sqcup C)$.

Obdobnou úvahu lze provést i pro formuli T TBoxu představující subsumpci $R \sqsubseteq S$, resp. ekvivalenci $R \equiv S$ rolí, s tím, že pro konstruktor negace role je doplňkem interpretace role R^I množina dvojic $(\Delta^I \times \Delta^I) \setminus R^I$.

Přímo z definice významu formule ABoxu znalostní báze vyplývá platnost následující věty.

Věta 4.2 o splnitelnosti formule ABoxu znalostní báze)

Nechť $A = C(a)$, resp. $A = R(a, b)$ je formule ABoxu znalostní báze Σ v DL (DL1). Potom je A v dané interpretaci I splněna (tzn. též splnitelná), právě když platí $a^I \in C^I$, resp. $(a^I, b^I) \in R^I$.

Definice 4.10 (modelu a splnitelnosti znalostní báze)

Struktura interpretace I je modelem TBoxu \mathbf{T} , (ABoxu \mathbf{A}) znalostní báze Σ , jestliže splňuje všechny formule z \mathbf{T} (\mathbf{A}). \mathbf{T} (\mathbf{A}) je splnitelný (konsistentní), má-li model.

Struktura interpretace I jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) je modelem znalostní báze $\Sigma = (\mathbf{T}, \mathbf{A})$, jestliže je v ní splněna každá formule z \mathbf{T} i \mathbf{A} . Znalostní báze Σ je splnitelná (konsistentní), má-li model.



4.4.2 Rozhodování splnitelnosti a subsumpce konceptů

V následujících úvahách budou používána tato označení: znalostní báze $\Sigma = (\mathbf{T}\text{Box}, \mathbf{A}\text{Box})$, koncepty C a D , individuální objekt \underline{a} a individuální objekt \underline{b} nevyskytující se v Σ . Vztahy diskutované v tomto i následujících odstavcích budou pojímány z hlediska sémantiky i z hlediska jejich formálních důkazů.

Při rozšiřování znalostní báze o další definice nebo specifikace konceptů (rolí) jde o to, aby znalostní báze zůstala po rozšíření o příslušný koncept (rolí) konsistentní.

Reprezentace znalostí

Rozhodování *splnitelnosti konceptu* C vzhledem ke znalostní bázi Σ je z hlediska sémantiky problémem zjišťování existence modelu I znalostní báze Σ takového, v němž $C^I \neq \emptyset$.



Definice 4.11

Struktura interpretace I jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) znalostní báze Σ je *modelem* konceptu C , resp. role R , je-li C^I , resp. R^I neprázdná. Koncept (role) je *splnitelný(-á)* vzhledem ke znalostní bázi Σ , má-li s ní společný model, jinak je *nesplnitelný(-á)*.

Je zřejmé, že platí:

- Koncept C je splnitelný vzhledem ke znalostní bázi Σ , právě když je splnitelná množina $\{\{TBox\}, \{ABox\}, \{C\}\}$.
- Role R je splnitelná vzhledem ke znalostní bázi Σ , právě když je splnitelná množina $\{\{TBox\}, \{ABox\}, \{R\}\}$.

Při formálním důkazu nesplnitelnosti konceptu C (role R) vzhledem ke znalostní bázi Σ jde o důkaz, že ze znalostní báze Σ spolu s konceptem C (rolí R) lze dokázat spor - absurditu \perp , tj. $\Sigma, C \vdash \perp$ ($\Sigma, R \vdash \perp$). Toho se v jazycích s konstruktorem negace konceptu využívá při nepřímém důkazu konsistence C vzhledem k Σ , kdy se dokazuje $\Sigma, \neg C \vdash \perp$.

V jazycích s konstruktorem negace konceptu lze též problém *subsumpce* převést na problém konceptové splnitelnosti:

Ve formálním systému DL i DL1 potom platí, že $\Sigma \vdash C \sqsubseteq D$, právě když $\Sigma \vdash \neg(\neg C \sqcup D) \equiv \perp$, resp. $\Sigma \vdash C \sqcap \neg D \equiv \perp$.

4.5 Logický důsledek znalostní báze



Definice 4.12 (logického důsledku znalostní báze)

Formule φ jazyka \mathbf{L}_{DL} (\mathbf{L}_{DL1}) je logickým důsledkem znalostní báze Σ , tj. $\Sigma \models \varphi$, je-li splněna ve všech modelech Σ .

V jazycích vybavených konstruktorem negace konceptu důkaz logického důsledku převést na problém konsistence znalostní báze rozšířené o negaci dokazované formule, neboť platí $\Sigma \models \varphi$, právě když $\Sigma \cup \{\neg\varphi\} \models \perp$.

Postup důkazu logického důsledku, založený na sémantickém tablu, spočívá v důkazu nesplnitelnosti znalostní báze rozšířené o negaci dokazované formule.

Tablový důkaz nesplnitelnosti konceptové formule nebo množiny konceptových formulí je formálním důkazem, který spočívá v aplikaci tablových pravidel, generujících jejich modely. Nemožnost nalezení modelu svědčí o nesplnitelnosti této množiny formulí.

Reprezentace znalostí

Definice 4.13

Kontrolou instancí ABoxu vzhledem k TBoxu znalostní báze Σ v jazyce \mathbf{L}_{DL} (\mathbf{L}_{DL1}) se rozumí zjišťování, zdali tvrzení $C(a)$ je splněno v každém modelu této znalostní báze Σ ($\Sigma \models C(a)$).



Kontrola instancí je speciálním případem důkazu logického důsledku. Při kontrole instancí totiž jde nutnost prověřovat zachování konsistence ABoxu vzhledem ke znalostní bázi u každého jejího rozšíření o další instanci.

Z hlediska sémantiky je $\Sigma \models C(a)$, právě když je nespelnitelná množina $\{\{TBox\}, \{ABox\}, \{\neg C(a)\}\}$ (v jazycích s konstruktorem negace konceptu).

Ve formálním systému DL (DL1) jde při důkazu nespelnitelnosti instance $C(a)$ vzhledem ke znalostní bázi Σ o důkaz, že ze znalostní báze Σ spolu s instancí $C(a)$, lze dokázat absurditu \perp , tj. $\Sigma, C(a) \vdash \perp$.

4.5.1 Sémantické tablo v deskripční logice DL

Tablový důkaz splnitelnosti konceptové formule nebo množiny konceptových formulí je podle autorů [4] formálním důkazem, který spočívá v aplikaci tablových pravidel jazyka \mathbf{L}_{DL} , generujících jejich modely. Nemožnost nalezení modelu svědčí o nespelnitelnosti této množiny formulí.

Tablová odvozovací pravidla jsou stejně jako v případě logiky prvního řádu (viz [64]) odvozena ze sémantiky konstruktorů a jsou generického charakteru, tj. generují postupně formule charakterizující model, pokud nevygenerují spor (nesplnitelný komplementární pár formulí). Např. ze sémantiky konstruktoru $\exists R.C$ lze odvodit, že existuje nějaký prvek c universa diskursu takový, že platí $c \in (\exists R.C)^I$. Ze sémantiky tohoto konstruktoru dále plyne, že existuje prvek d takový, že $(c, d) \in R^I$ a $d \in C^I$. To vede k zobecněnému odvozovacímu pravidlu pro konstrukty kvalifikované existenční kvantifikace (**S** je zde struktura postupně generovaného modelu, jazyk \mathbf{L}_{DL} je pro účel tablového důkazu modifikován proměnnými v závorkách za příslušnými symboly konceptů a rolí). Na obdobném zobecnění jsou založena i další tři odvozovací pravidla.

Definice 4.14 (tablových odvozovacích pravidel)

Množinu **S** formulí deskripční logiky lze rozšiřovat pomocí následujících pravidel:



Reprezentace znalostí

Tabulka 4.3

$S \longrightarrow_{\sqcap} \{C(x), D(x)\} \cup S$, jestliže 1. $(C \sqcap D)(x)$ je v S , 2. $C(x)$ a $D(x)$ nejsou současně v S .

$S \longrightarrow_{\sqcup} \{E(x)\} \cup S$, jestliže 1. $(C \sqcup D)(x)$ je v S , 2. ani $C(x)$ a $D(x)$ nejsou v S . 3. $E = C$ nebo $E = D$.

$S \longrightarrow_{\exists} \{R(x,y), C(y)\} \cup S$, jestliže 1. $\exists R . C(x)$ je v S , 2. y je nová proměnná, 3. neexistuje z takové, že $R(x,z)$ a $C(z)$ jsou v S .

$S \longrightarrow_{\forall} \{C(y)\} \cup S$, jestliže 1. $\forall R . C(x)$ je v S , 2. $R(x,y)$ je v S , 3. $C(y)$ není v S .

Autoři v různých knihách dokázali sémantickou korektnost algoritmu generování modelu znalostní báze S a též skutečnost, že algoritmus vždy skončí. Tato vlastnost činí logiku DL za cenu platnosti " *předpokladu jedinečného jména*" rozhodnutelnou.

Generování rozšířené znalostní báze tablovými pravidly logiky DL za účelem získání jejího modelu je poněkud nepřehledné, provádí-li se formou lineárně uspořádaných zvětšujících se seznamů. Proto autoři této knihy zvolili k reprezentaci tablového důkazu jeho stromovou reprezentaci.



Příklad 4.7

Tablový důkaz nesplnitelnosti formule:

$\text{student} \sqcap \forall \text{zapsán} . \text{kurs} \sqcap \exists \text{zapsán} . \neg \text{kurs}$

$$\begin{array}{l}
 (\text{student} \sqcap \forall \text{zapsán} . \text{kurs} \sqcap \exists \text{zapsán} . \neg \text{kurs})(x) \\
 | \\
 \text{student}(x), \forall \text{zapsán} . \text{kurs}(x), \exists \text{zapsán} . \neg \text{kurs}(x) \quad \sqcap, \sqcap \\
 | \quad \exists \\
 \text{student}(x), \forall \text{zapsán} . \text{kurs}(x), \exists \text{zapsán} . \neg \text{kurs}(x), \text{zapsán}(x, c), \neg \text{kurs}(c) \\
 | \quad \forall \\
 \text{student}(x), \forall \text{zapsán} . \text{kurs}(x), \underline{\text{kurs}(c)}, \exists \text{zapsán} . \neg \text{kurs}(x), \text{zapsán}(x, c), \underline{\neg \text{kurs}(c)} \\
 \times
 \end{array}$$

Výskyt komplementárního páru formule v listu tabla potvrzuje nesplnitelnost seznamu (konjunkce) formulí.

Reprezentace znalostí



Příklad 4.9

Nepřímý tablový důkaz logické platnosti subsumpce

$$\forall R . C \sqsubseteq \exists R . C \quad (1)$$

přepsané na formuli $\neg \forall R . C \sqcup \exists R . C$ s použitím algoritmu v jazyce \mathbf{L}_{DL} .

$$\neg (\neg \forall R . C \sqcup \exists R . C)$$

$$\begin{array}{c} | \\ \forall R . C, \neg \exists R . C \\ | \\ \forall R . C, \forall R . \neg C \\ \bigcirc \end{array}$$

V důkazu nelze pokračovat, protože již žádné z daných odvozovacích pravidel nelze aplikovat. Nelze tedy v rámci tablových pravidel dokázat odvozením sporného komplementárního páru formulí (očekávanou) nesplnitelnost negované formule $\neg(\neg \forall R . C \sqcup \exists R . C)$, z níž by vyplynula logická platnost původní formule $\neg \forall R . C \sqcup \exists R . C$.

4.6 Částečně rozhodnutelná deskripční logika DL1

Zatímco předcházející odstavce týkající se tématu deskripční logiky vycházely z definice jazyka deskripční logiky a její sémantiky, tak jak byly zavedeny jejími autory v publikaci, v tomto odstavci bude prezentována alternativní deskripční logika DL1 s poněkud modifikovanou sémantikou jazyka \mathbf{L}_{DL1} , a to speciálně v případě konstruktoru $\forall R . C$. Této modifikaci pak odpovídá autory této knihy zavedený tablový algoritmus, který sdílí s odpovídajícím algoritmem logiky prvního řádu vlastnost částečné rozhodnutelnosti.

4.6.1 Očekávané vlastnosti sémantiky jazyka LDL1

Na příkladech uvedených v předcházejících odstavcích, pojednávajících o jazyce \mathbf{L}_{DL} a jejím tablovém rozhodovacím algoritmu bylo patrné, že ačkoliv odvozovací pravidla generují korektně modely množin výchozích předpokladů (znalostní báze), některé závěry z důkazů vět z daných předpokladů, neodpovídající intuitivnímu pojetí dedukce, nelze bez výhrad přijmout. Nelze např., jak ukazuje řešení příkladu dokázat prostřednictvím přepisu do formule deskripční logiky platnost axiomu

$$\forall R . C \sqsubseteq \exists R . C \quad (1)$$

Reprezentace znalostí

kteř by v souladu s všeobecně sdílenou intuicí platit měl. To vyvolává otázku, proč se formální systém logiky DL jeví jako pouze sémanticky korektní, nikoliv však sémanticky úplný. Při hlubším zkoumání autoři návrhu modifikovaného jazyka \mathbf{L}_{DL1} došli k poznání, že v rámci tohoto protipříkladu nebyla odhalena sémantická neúplnost systému, ale důsledek způsobu definování sémantiky konstruktoru $\forall R . C$, který právě této všeobecně sdílené intuici neodpovídá. Nové zavedení sémantiky tohoto konstruktoru v rámci jazyka \mathbf{L}_{DL1} spolu s odpovídající modifikací tablového rozhodovacího algoritmu dává možnost dokazovat důsledky daných předpokladů, tak aby závěry odpovídaly paralelně dokázaným výsledkům z logiky prvního řádu a též intuitivnímu pojetí vztahů mezi koncepty v rámci axiomu (1).

Má-li logické platnosti výrazu (1) odpovídat i závěr z tablového důkazu jeho logické platnosti, je třeba tablový algoritmus modifikovat tak, aby odpovídal modifikované sémantice konstruktoru $\forall R . C$. Bez této modifikace nelze sestrojít uzavřené sémantické tablo negované formule (1).

Je třeba provést tyto změny:

- Definovat modifikovaně sémantiku konstruktoru $\forall R . C$ tak, jak je alternativně pro jazyk \mathbf{L}_{DL1} uvedena v definici (definice 4.3).
- Definovat způsobem odpovídajícím nově definované sémantice konstruktoru $\forall R . C$ pravidla přepisu negací kvantifikovaných konstruktorů jazyka \mathbf{L}_{DL1} .
- Přepracovat tablový rozhodovací algoritmus vzhledem k nově definované sémantice $\forall R . C$.

4.6.2 Modifikovaná sémantika $\forall R . C$ v deskripční logice DL1

V tab. 4.1 byla uvedena Backus - Naurovou formou definice syntaxe jazyka \mathbf{L}_{DL1} , která se neliší od obecně používané základní verze jazyka \mathbf{L}_{DL} deskripční logiky. Akceptována je zde též definice sémantiky (v interpretaci I) konstantních symbolů a konstruktorů jazyka \mathbf{L}_{DL} s výjimkou modifikované sémantiky $\forall R . C$ (tab 4.2).

Následující příklad objasní motiv zavedení modifikované sémantiky konstruktoru $\forall R . C$ v jazyce \mathbf{L}_{DL1} .

Příklad 4.10

Nechť $\mathit{má_dítě}(x, y)$ je role, jejíž extenze je dána takto:

$\mathit{ext}(\mathit{má_dítě}(x,y)) = \{(petr, alice), (zdeněk, viktor), (zdeněk, dana), (ondřej, jan)\}$

Koncepty $\mathit{žena}(x)$ a $\mathit{muž}(x)$ jsou definovány extenzemi

$\mathit{ext}(\mathit{žena}(x)) = \{alice, dana\}$

$\mathit{ext}(\mathit{muž}(x)) = \{petr, zdeněk, viktor, ondřej, jan, michal\}$



Reprezentace znalostí

Stanovme nyní extenze kvantifikovaných konceptů na základě sémantiky odpovídajících konstruktorů jazyka \mathbf{L}_{DL} .

a) Extenzi konceptu $\exists R . T$ se sémantikou definovanou jako

$$(\exists R . T)^I = \{x \in \Delta^I \mid \exists y R(x, y)\} \quad (2)$$

tvoří všechny objekty x universa diskursu, pro něž platí některá instance $R(x,y)$, což znamená, že v interpretující relaci predikátu $R(x,y)$ existuje pár (x,a) pro nějaké a . V našem případě je tedy extenzí konceptu $\exists má_dítě . T$

$$\text{ext}(\exists má_dítě . T) = \{\text{petr, zdeněk, Ondřej}\}.$$

b) Extenzi konceptu $\exists R . C$ při sémantice definované jako

$$(\exists R . C)^I = \{x \in \Delta^I \mid \exists y (R(x, y) \ \& \ C(y))\} \quad (3)$$

by intuitivně mělo být nějaké omezení extenze konceptu $\exists R . T$. Extenze konceptu $\exists R . C$ obsahuje všechny objekty x universa diskursu, pro něž existuje objekt o konceptu C ($o \in \text{ext}(C)$) tak že v rámci extenze role R platí $R(x,o)$. V daném případě $\text{ext}(\exists má_dítě . žena) = \{\text{petr, zdeněk}\}$.

c) Extenzi konceptu $\forall R . C$ by intuitivně mělo být vždy další omezení extenze konceptu $\exists R . C$. V rámci sémantiky definované jako

$$(\forall R . C)^I = \{x \in \Delta^I \mid \forall y (R(x, y) \rightarrow C(y))\} \quad (4)$$

však náleží extenzi konceptu $\forall R . C$ též ty objekty x universa diskursu, které splňují $\neg R(x, y)$. V tom případě neplatí očekávané $\text{ext}(\forall má_dítě . žena) = \{\text{petr}\}$. Podle výrazu (4), přepsaného do formule logiky prvního řádu $\forall x(\forall y (R(x, y) \rightarrow C(y)))$ by extenze $\text{ext}(\forall má_dítě . žena)$ zahrnovala též všechny objekty n universa, které splňují $\neg R(n, y)$.

Tedy v našem příkladě by platilo $\text{ext}(\forall má_dítě . žena) = \{\text{petr, } \langle \text{michal, alice, dana, } \dots \text{ a další objekty universa diskursu} \rangle\}$.

Je zřejmé, a to nejen na základě uvedeného příkladu, že formule (4) platí nejen pro všechny objekty x , jejichž páry (x, o_i) náleží extenzi role R , ale též pro objekty x , pro něž $R(x,o)$ je pro nějaké o nepravdivá, tj. $R(x,o) \notin \text{ext}(R)$.

Protože využití jazyka \mathbf{L}_{DL} s konstruktorem $\forall R . C$, jehož sémantika je definována (4) odporuje všeobecně přijímané interpretaci, zavedli autoři modifikovanou verzi \mathbf{L}_{DL1} jazyka deskripční logiky DL1 s takovou interpretací konstruktoru $\forall R . C$, v níž je extenze konceptu $\forall R . C$ dalším omezením extenze konceptu $\exists R . C$. Extenze konceptu $\forall R . C$ je tedy podmnožinou extenze konceptu $\exists R . C$, z čehož pak vyplývá platnost axiomu (1). Logická platnost tohoto axiomu je zřejmá po přepsání do formule deskripční logiky

$$\neg \forall R . C \sqsubset \exists R . C \quad (5)$$

a náhradě výrazu $\forall R . C$ výrazem $\exists R . C \sqcap (R(x, y) \sqcap C(y))$.

Snadno se pak dokáže (nepřímo) logická platnost (5):

Negovaná DLformule $\neg(\neg(\exists R . C \sqcap (R(x, y) \sqcap C(y))) \sqsubset \exists R . C)$ je ekvivalentní formuli

Reprezentace znalostí

$$\exists R . C \sqcap (R(x, y) \sqcap C(y)) \sqcap \neg \exists R . C,$$

kteřá je sporná ve všech interpretacích.

4.6.3 Přepisovací pravidla negací konceptů $\exists R . C$ a $\forall R . C$ v DL1

Sémantika konstruktorů $\exists R . C$ a $\forall R . C$ definovaná v [4] má z hlediska jejich negací vlastnosti formulovatelné jednoduchými výrazy:

$$\neg \forall R . C \Leftrightarrow \exists R . \neg C \quad (6)$$

$$\neg \exists R . C \Leftrightarrow \forall R . \neg C \quad (7)$$

V následujících úpravách budou použity tyto ekvivalence odpovídajících formulí logiky prvního řádu:

$$R(x,y) \ \& \ (R(x,y) \ \rightarrow \ C(y)) \ \Leftrightarrow \ R(x,y) \ \& \ (\neg R(x,y) \ \vee \ C(y)) \ \Leftrightarrow \ (R(x,y) \ \& \ \neg R(x,y)) \ \vee \ (R(x,y) \ \& \ C(y)) \ \Leftrightarrow \ R(x,y) \ \& \ C(y)$$

Sémantika konstruktoru $\forall R . C$ byla v deskripční logice DL1 definována výrazem

$$(\forall R . C)^I = \{x \in \Delta^I \mid \exists t R(x, t) \ \& \ \forall y (R(x, y) \rightarrow C(y))\}, \quad (8)$$

jíž odpovídá přepisovací pravidlo negace

$$\begin{aligned} (\neg \forall R . C)^I &= \{x \in \Delta^I \mid \forall t \neg(R(x, t) \vee \exists y \neg(R(x, y) \rightarrow C(y)))\} = \\ &= \{x \in \Delta^I \mid \forall t \neg R(x, t) \vee \exists y (R(x, y) \ \& \ \neg C(y))\} = (\forall \neg R . \top \sqcup \exists R . \neg C)^I. \end{aligned}$$

Tedy platí

$$\neg \forall R . C \Leftrightarrow \exists R . \neg C \sqcup \forall \neg R . \top \quad (6')$$

Podobně konstruktoru $\exists R . C$ odpovídá přepisovací pravidlo jeho negace:

$$\begin{aligned} (\neg \exists R . C)^I &= \{x \in \Delta^I \mid \neg \exists y (R(x, y) \ \& \ C(y))\} = \{x \in \Delta^I \mid \forall y \neg(R(x, y) \ \& \ C(y))\} = \\ &= \{x \in \Delta^I \mid \forall y (R(x, y) \rightarrow \neg C(y))\} = \{x \in \Delta^I \mid \forall y (R(x, y) \rightarrow \neg C(y)) \ \& \ (\neg R(x, y) \vee \\ &\vee \\ &R(x, y))\} = \\ &= \{x \in \Delta^I \mid \forall y ((\neg R(x, y) \ \& \ (R(x, y) \rightarrow \neg C(y))) \vee ((R(x, y) \ \& \ (R(x, y) \rightarrow \neg C(y))))\} = \\ &= (\forall \neg R . \top \sqcup \forall R . \neg C)^I, \end{aligned}$$

neboť $\neg R(x,y)$ v prvním disjunktě musí být pravdivé a z $R(x,y)$ ve druhém disjunktě vyplývá $\exists t R(x,t)$.

Platí tedy

$$\neg \exists R . C \Rightarrow \forall \neg R . \top \sqcup \forall R . \neg C \quad (7')$$

Reprezentace znalostí

4.6.4 Tablový rozhodovací algoritmus v DL1

Částečně rozhodovací tablový algoritmus, který zde bude definován, odpovídá tomu, který je definován pro logiku prvního řádu a sdílí s ním též vlastnosti sémantické korektnosti a úplnosti.

Znalostní báze, z níž jsou zde pomocí sémantického tabla dokazovány věty teorie, obsahuje formule ABoxu a přepisy axiomů TBoxu do formulí deskripční logiky. Protože formule TBoxu v axiomech mají charakter univerzálně kvantifikovaných formulí logiky prvního řádu, lze v tablových důkazech použít jejich instance tvaru $\langle C \text{formule} \rangle(x)$.



Definice 4.15 (sémantického tabla v LDL1)

Sémantické tablo množiny $S = \{X_1, \dots, X_n\}$ formulí TBoxu a ABoxu znalostní báze Σ jazyka L_{DL1} je konečný binární ohodnocený strom, jehož všechny uzly jsou ohodnoceny seznamy formulí z S , tak že platí:

- 1) Kořen je ohodnocen seznamem formulí množiny S .
- 2) K uzlu, který není listem, se vytvoří ohodnocené uzly následující úrovně vždy na základě jediné vybrané formule ze seznamu v jeho návěští podle těchto pravidel :

a) α -pravidlo: Jestliže je uzel ohodnocen seznamem formulí

$$X_1, X_2, \dots, \alpha, \dots, X_{n1}$$

obsahující jako člen formuli typu $\alpha : (C \sqcap D)(x)$ a je-li pro vytvoření následující úrovně stromu vybrána právě tato formule, pak jediný uzel bezprostředně následující je ohodnocen seznamem

$$X_1, X_2, \dots, C(x), D(x), \dots, X_{n1}.$$

b) β -pravidlo: Jestliže je uzel ohodnocen seznamem formulí

$$X_1, X_2, \dots, \beta, \dots, X_{n2}$$

obsahující jako člen formuli typu $\beta : (C \sqcup D)(\Box x)$ a je-li pro vytvoření následující úrovně stromu vybrána právě tato formule, pak dvojice uzlů bezprostředně následujících je ohodnocena po řadě seznamy

$$X_1, X_2, \dots, C(x), \dots, X_{n2} \quad \text{a} \quad X_1, X_2, \dots, D(x), \dots, X_{n2}$$

c) γ -pravidlo: Jestliže je uzel ohodnocen seznamem formulí

$$X_1, X_2, \dots, \gamma, \dots, X_n$$

obsahující jako člen formuli typu $\gamma : \forall R . C(x)$ a je-li pro vytvoření následující úrovně stromu vybrána právě tato formule, pak jediný uzel bezprostředně následující je ohodnocen seznamem

$$X_1, X_2, \dots, \forall R . C(x), C(t), \dots, X_n$$

kde t je term jazyka L_{DL1} .

Reprezentace znalostí

d) δ -pravidlo: Jestliže je uzel ohodnocen seznamem formulí

$$X_1, X_2, \dots, \delta, \dots, X_{n3}$$

obsahující jako člen formulí typu $\delta: \exists R . C(x)$ a je-li pro vytvoření následující úrovně stromu vybrána právě tato formule, pak jediný uzel bezprostředně následující je ohodnocen seznamem

$$X_1, X_2, \dots, \exists R . C(x), R(x,d), C(d), \dots, X_{n3},$$

přičemž d je nová konstanta, nevyskytující se dosud v jazyce.

e) \neg pravidlo: Jestliže je uzel ohodnocen seznamem formulí obsahující negovanou formulí $\neg \forall R . C(x)$ ($\neg \exists R . C(x)$), je možno přepsat tuto formulí na formulí

$$\exists R . \neg C \sqcup \forall R . T (\forall R . \neg C \sqcup \forall \neg R . T).$$

f) instanciací pravidlo: Jestliže je uzel ohodnocen seznamem formulí obsahující axiom TBoxu, přepsaný do formule deskripční logiky, lze do seznamu přidat instanci této formule pro libovolný term jazyka \mathbf{L}_{DL1} .

g) pravidlo uzavřené /otevřené větve: Existuje-li v seznamu návěští uzlu komplementární pár formulí deskripční logiky, stává se uzel listem stromu, který činí příslušnou větev uzavřenou (označuje se \times). V opačném případě platí: neexistuje-li již žádná možnost aplikace některého z pravidel $\alpha, \beta, \gamma, \delta$ a \neg , uzel se stává listem, který činí větev konečně otevřenou (označuje se \circ), v opačném případě se stává větev otevřenou a potenciálně nekonečnou (označuje se \circ).

Definice 4.16 (uzavřenosti / otevřenosti sémantického tabla)

Sémantické tablo množiny $\mathbf{S} = \{X_1, \dots, X_n\}$ formulí TBoxu a ABoxu znalostní báze jazyka \mathbf{L}_{DL1} je *uzavřené tablo*, jsou-li všechny jeho větve uzavřené. V opačném případě jde o *otevřené tablo*.



Lze dokázat, že tablový rozhodovací algoritmus sdílí s odpovídajícím algoritmem logiky prvního řádu vlastnosti sémantické korektnosti a úplnosti, které lze ilustrovat na obr. 4.3. To znamená též, že platí následující věta:

Věta 4.3

Sémantické tablo množiny \mathbf{S} formulí deskripční logiky se uzavře, právě když je množina \mathbf{S} nesplnitelná.

Reprezentace znalostí

4.6.6 Tablové důkazy logické platnosti a logického důsledku v LDL1

Stejně jako v logice prvního řádu je nepřímý tablový důkaz jednou z mála možností dokazování logické platnosti formulí a logického důsledku daných předpokladů. To ilustrují následující příklady.

Příklad 4.12

Nepřímý tablový důkaz logického důsledku $\neg(\text{matka} \sqcap \neg\text{žena})$ z části TBoxu znalostní báze :

$\text{rodič} \equiv \text{osoba} \sqcap \exists \text{MÁ_DÍTĚ} . \text{osoba}$

$\text{matka} \equiv \text{žena} \sqcap \text{rodič}$

$\text{žena} \equiv \text{osoba} \sqcap \text{ženského_rodu}$

S využitím tří uvedených definic konceptů rodič, matka a žena sestává seznam formulí v kořeni tabla z konjunkce jejich pravých stran a z negace dokazovaného závěru. Po několikanásobné aplikaci pravidla pro \sqcap vyloučení opakujících se prvků seznamu a aplikaci pravidla pro \sqcup dochází ke sporu - uzavření obou větví tabla, což znamená, že předpokládaný logický důsledek skutečně z daných předpokladů vyplývá.

$(\text{osoba} \sqcap \text{ženského_rodu} \sqcap \text{osoba} \sqcap \exists \text{MÁ_DÍTĚ} . \text{osoba})(x), (\text{osoba} \sqcap \text{ženského_rodu} \sqcap \neg(\text{osoba} \sqcap \text{ženského_rodu}))(x)$

$(\text{osoba} \sqcap \text{ženského_rodu} \sqcap \exists \text{MÁ_DÍTĚ} . \text{osoba})(x), (\neg\text{osoba} \sqcup \neg\text{ženského_rodu})(x)$

3x \sqcap, \sqcup

$\neg\text{osoba}(x), \text{osoba}(x), \text{ženského_rodu}(x), \text{ženského_rodu}(x), \exists \text{MÁ_DÍTĚ} . \text{osoba}(x)$	$\neg\text{ženského_rodu}(x), \text{osoba}(x), \exists \text{MÁ_DÍTĚ} . \text{osoba}(x)$
X	X

V případě, kdy jsou tablový důkaz vychází z definovaných konceptů, nikoliv z obsahů jejich definic, tvoří návěští kořene tabla seznam formulí rodič, matka, žena, matka $\sqcap \neg\text{žena}$, který vede po jediném použití pravidla pro \sqcap ke sporu.

Příklad 4.13

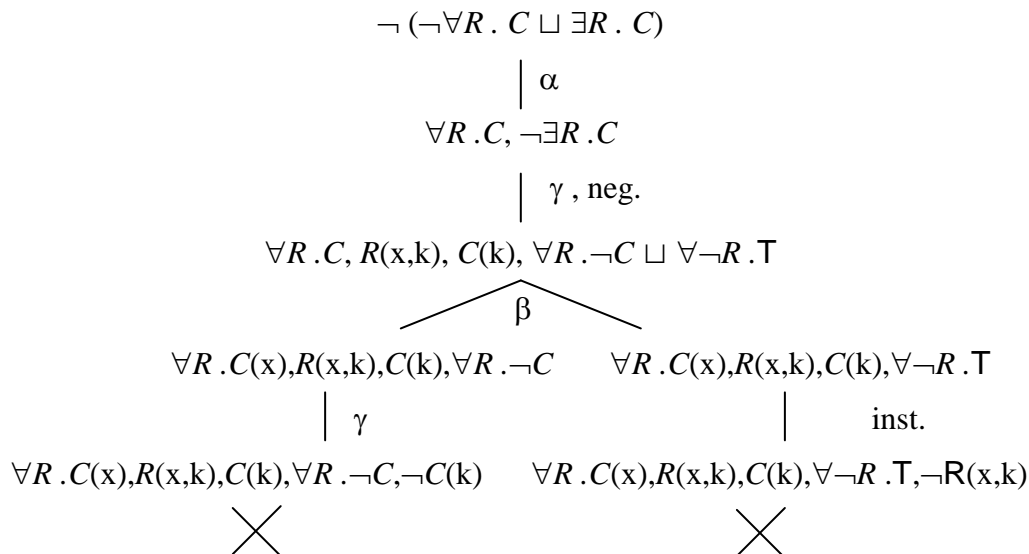
Nepřímý tablový důkaz (srovnej **Příklad**) logické platnosti subsumpce

$$\forall R . C \sqsubseteq \exists R . C \quad (1)$$



Reprezentace znalostí

přepsané na formuli $\neg \forall R . C \sqcup \exists R . C$ s použitím výše definovaného (definice 4.15) tablového algoritmu v jazyce \mathbf{L}_{DL1} .



Na rozdíl od srovnávaného příkladu řešeného v deskripční logice DL odpovídá závěr důkazu očekávanému tvrzení, že subsumpce (1) platí, neboť existuje sémantické tablo negace jejího přepisu do formule DL1, které se uzavře.



Příklad 4.14

Nepřímý tablový důkaz (fig. 3) subsumpce Z: **kráva** \sqsubseteq **vegetarian** (dále zkráceno jako "veget") z množiny předpokladů

$$\{\text{veget} \sqsubseteq \forall j\dot{i} . \neg \text{maso}, \text{kráva} \sqsubseteq \forall j\dot{i} . \text{tráva}, \text{tráva} \sqsubseteq \neg \text{maso}\}$$

přepsaných do následujících DLformulí:

$$P1 \neg \text{veget} \sqcup \forall j\dot{i} . \neg \text{maso}$$

$$P2 \neg \forall j\dot{i} . \neg \text{maso} \sqcup \text{veget}$$

$$P3 \neg \text{kráva} \sqcup \forall j\dot{i} . \text{tráva}$$

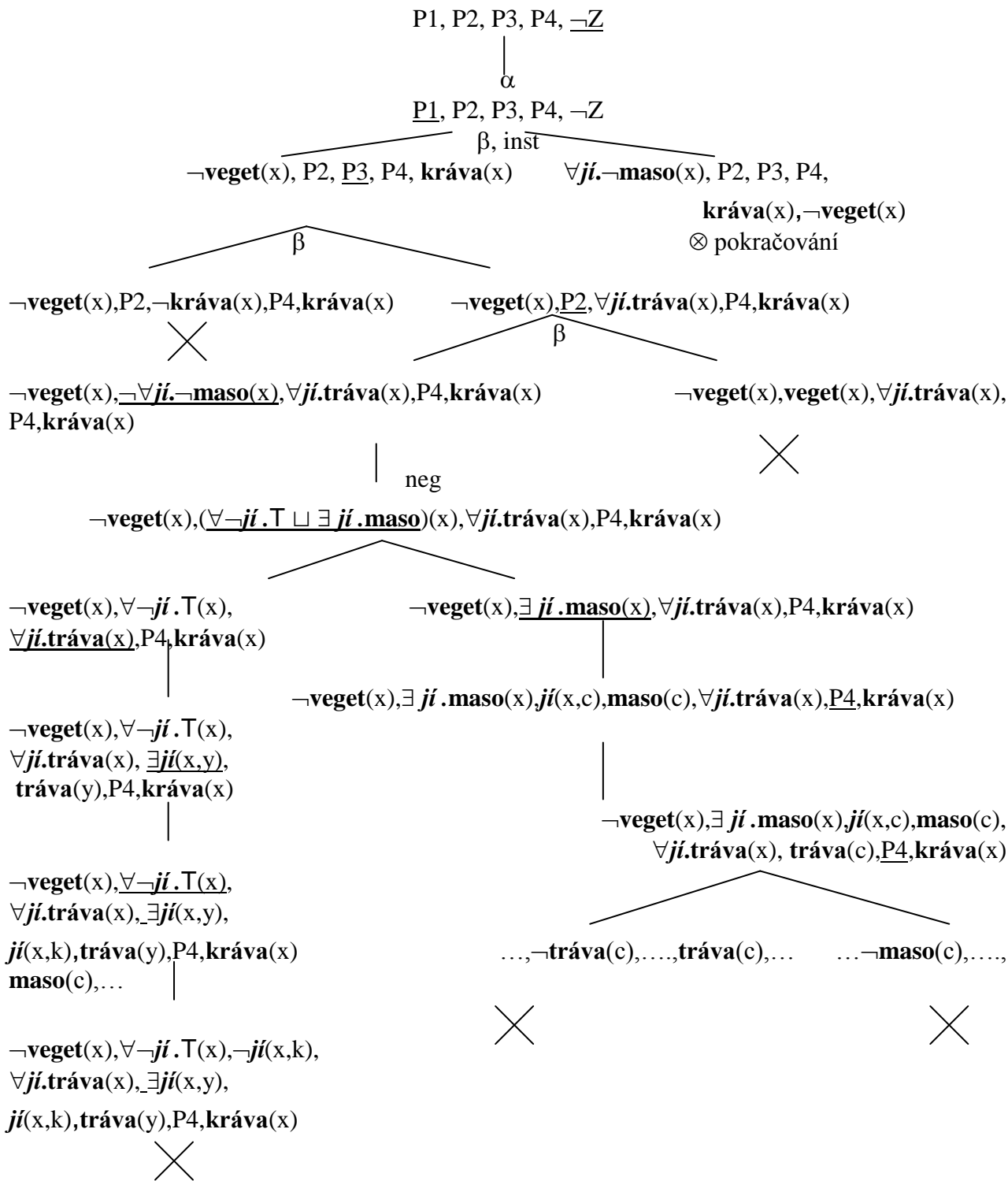
$$P4 \neg \text{tráva} \sqcup \neg \text{maso}$$

$$Z: \neg \text{kráva} \sqcup \text{veget}$$

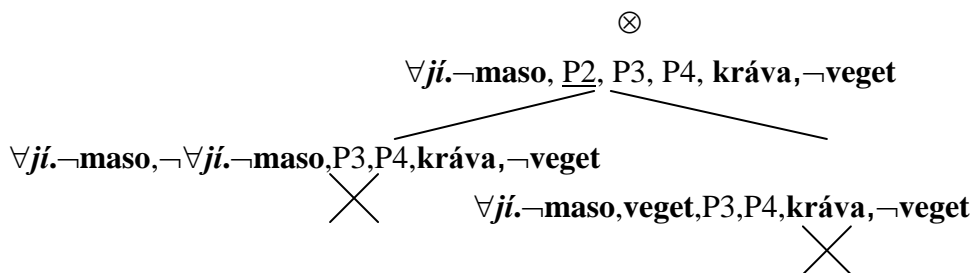
Nepřímý tablový důkaz je založen obdobně jako tomu bývá v logice prvního řádu na důkazu nesplnitelnosti množiny DLformulí sestávající z předpokladů

$$P1 - P4 \text{ a negace závěru } Z, \text{ tj z množiny } \{\neg \text{veget} \sqcup \forall j\dot{i} . \neg \text{maso}, \neg \forall j\dot{i} . \neg \text{maso} \sqcup \text{veget}, \neg \text{kráva} \sqcup \forall j\dot{i} . \text{tráva}, \neg \text{tráva} \sqcup \neg \text{maso}, (\neg \text{kráva} \sqcup \text{veget}) \}$$

Reprezentace znalostí



Reprezentace znalostí



4.6.7 Strukturální subsumpce

Deskripční logika je axiomatickým systémem. To znamená, že rozhodování, resp. dokazování znalostí v DL systému je deduktivní proces vycházející z obou částí znalostní báze $\Sigma = (\mathbf{T}, \mathbf{A})$, která představuje pro tyto postupy výchozí množinu speciálních axiomů, a že lze pomocí formálních pravidel odvozovat ze znalostní báze další implicitně skryté znalosti.

V jazycích deskripční logiky, které nedisponují konstruktorem negace, nelze problém subsumpce redukovat na problém splnitelnosti. Strukturální odvozování je založeno na myšlence, že pokud dvě srovnávané konceptové formule sestávají z podformulí, lze vzájemně srovnávat tyto podformule. Algoritmus má dvě fáze:

1. Normalizace :

- zploštění uzávorkovaných konjunkcí, tj. přepis $A \sqcap (B \sqcap C)$ na $A \sqcap B \sqcap C$,
- faktorizace tj. přepis $\forall P.C \sqcap \forall P.D$ na $\forall P.(C \sqcap D)$.

2. Nechť $C = C_1 \sqcap C_2 \sqcap \dots \sqcap C_m$, $D = D_1 \sqcap D_2 \sqcap \dots \sqcap D_n$ v normální formě. Potom D subsumuje C , právě když platí pro všechna D_i

- a) je-li D_i atomický koncept nebo koncept tvaru $\exists P$, potom existuje C_j takové, že

$$D_i = C_j$$
- b) je-li D_i koncept tvaru $\forall P.D'$, potom existuje C_j tvaru $\forall P.C'$, (atomická role P) takové, že D' subsumuje C' .

Shody:

Jde o vyhledávání shody konceptů se vzory založené na konceptově orientované normální formě spíše než na standardní subsumpci konceptů, v jazyce *ALN*.

Rozhodování subsumpce strukturálním srovnáváním je zřejmě konečným procesem (uvažují se konečné formule) rozhodnutelným s nejvýše polynomičnou složitostí. Sémantická korektnost a úplnost formálního procesu vyplývá z jednoznačné korespondence subsumpce a odpovídající množinové inkluze.

Reprezentace znalostí

4.6.8 Odvozování z ABoxu

Aby bylo možno rozhodovat konsistenci množiny formulí ABoxu \mathbf{A} , je třeba vycházet z extenzí konceptů a rolí, definovaných v TBoxu \mathbf{T} , jejichž instance tvoří \mathbf{A} . K tomu se využívá tzv. *expandovaný ABox* \mathbf{A}' , vytvořený ze všech možných instancí dotyčných konceptů a rolí přes všechny prvky jejich extenzí. Kontrolu konsistence \mathbf{A} vzhledem k \mathbf{T} , tak jak byla definována (def. 4.9), lze pak převést na kontrolu konsistence samotné \mathbf{A}' , resp. kontrolu konsistence \mathbf{A}' vzhledem k prázdnému TBoxu.

Např. nekonsistence způsobená současným výskytem instancí otec(jana) a matka(jana) v ABoxu by nebyla odhalena, kdyby se neuvažovaly extenze konceptů otec a matka.

Problém kontroly instancí ABoxu, tak jak byl definován v odstavci 4.2 se řeší pomocí \mathbf{A}' .

4.6.9 Rozšíření znalostní báze podle spouštěcích pravidel

Některé systémy DL zahrnují též tzv. *spouštěcí pravidla*, obsahující pravidla obecného tvaru $C \Rightarrow D$, pomocí nichž lze znalostní bázi rozšiřovat o další části ABoxu podle tohoto procedurálního pravidla :

Platí-li $\Sigma \models C(a)$, přidej do ABoxu i $D(a)$ pro všechny denotáty konstanty a extenze konceptu C .

Procedurální rozšíření znalostní báze je možno definovat též deklarativním způsobem. K tomu je třeba zavést epistemický operátor \mathbf{K} obohacující TBox znalostní báze o definici epistemického konceptu $\mathbf{K}C$, označujícího ty objekty, o nichž znalostní báze ví, že tvoří instance C . Potom je možno spouštěcí pravidlo $C \Rightarrow D$ převést na pravidlo inkluze konceptů $\mathbf{K}C \sqsubseteq D$.

Pravidlové znalostní báze sestávají z TBoxu, ABoxu a množiny pravidel typu $\mathbf{K}C \sqsubseteq D$.

Kontrolní otázky a úkoly

1. Jaké jsou hlavní rozdíly v syntaxi jazyků predikátové a deskripční logiky?
2. Definiujte rozdíl mezi L_{DL} a L_{DL1} ?
3. K čemu slouží sémantické tabla?



Korespondenční úkol

1. Vytvořte svůj vlastní rodokmen pomocí Deskripční logiky.
2. Dokažte pomocí deskripční logiky, že kráva žere krávu.



Reprezentace znalostí



Shrnutí

Formální systém deskripční logiky je určen jazykem, znalostní bází, která představuje soubor speciálních axiomů pro dedukci, a pravidly (mechanismem) tuto dedukci umožňujícími.

Jazyk deskripční logiky je speciální konceptový jazyk vygenerovaný množinou konstruktorů pro reprezentaci pojmů, tj. konceptů (tříd, kategorií), rolí (vztahů mezi nimi) a instancí - jejich individuí.

Abecedu jazyka deskripční logiky tvoří

- symboly pro jména konceptů - unární predikátové symboly, které reprezentují třídy individuí
- symboly pro jména rolí - binární predikátové symboly, které reprezentují relace mezi páry individuí
- symboly pro jména individuí.

Pro jména individuí platí v DL „předpoklad jedinečného jména“, tj. předpoklad, že neexistuje dvojice navzájem různých jmen pojmenovávajících též objekt.

Gramatická pravidla jsou dána množinou konstruktorů pro vytváření formulí jazyka, tj. definic konceptů a rolí pro vytváření tzv. TBoxů a ABoxů, z nichž sestávají znalostní báze.

Je-li dán konceptový jazyk L , je znalostní bází (T, A) , kde T je TBox a A je ABox v L .

TBox je terminologický box, který představuje množinu (významem univerzálně kvantifikovaných) formulí jazyka L , jimiž se reprezentují intenzionální znalosti.

ABox (assertional box) je box formulí reprezentujících extenzionální znalosti. Formule ABoxu má některou z forem uvedených v tab.4.2, kde C je koncept, R je role a a, b označují individua jazyka L .

5 MODÁLNÍ K-LOGIKA

Cílem kapitoly je:

- definice syntaxe LK,
- definice sémantiky LK,
- pravdivost a splnitelnost formulí modální logiky,
- definovat formální systém výrokové modální logiky.

Po jejím prostudování byste měli být schopni:

- formulovat syntax a sémantiku modální logiky,
- tvořit splnitelné formule modální logiky,
- odvozovat pomocí odvozovacích pravidel modální logiky.

Klíčová slova:

Modální K-logika, syntax, sémantika, formule, interpretace, axiom, odvozovací pravidla.

Čas potřebný ke studiu: 8 hodin

Průvodcem studia

V klasických formálně logických systémech je formalizované tvrzení týkající se modelovaného referenčního systému jednou provždy pravdivé nebo nepravdivé, i když se modelovaný svět může v rozsahu trvání formálního popisu měnit. Modální logika modifikuje tvrdý kontrast mezi pravdivostí a nepravdivostí formálně reprezentovaných tvrzení zaváděním tzv. modálních operátorů. Zavádí dvě modalities pravdivosti, které umožňují rozlišení případů „nutné“ pravdivosti, tedy pravdivosti vždy a všude, a „možné“ pravdivosti, která nastane alespoň v jednom případě, tedy někdy a někde, od pravdivosti tvrzení o konkrétním modelovaném světě vymezeném v daném prostoru a čase, tak jak je známa z logiky prvního řádu. Možnost a nutnost, jimiž se modifikuje pravdivost/nepravdivost tvrzení, se v modální logice nazývají aletické modalities nebo módy pravdivosti. Systémy obsahující operátory pro reprezentaci možnosti a nutnosti se nazývají aletické logiky.



Zde bude diskutována pouze poměrně jednoduchá normální modální logika vycházející z výrokové logiky, označovaná zpravidla jako *K-logika* podle svého autora, amerického filosofa a logika *Saula Kripkeho*.

5.1 Syntax jazyka LK

Všechna syntaktická pravidla jazyka **L** výrokové logiky jsou zároveň pravidly jazyka **L_K** výrokové modální logiky **K**.

Reprezentace znalostí



Rozšíření jazyka pro vyjádření módů pravdivosti spočívá v těchto krocích:

- K vyjádření možnosti a nutnosti se zde zavádějí dva modální operátory: \diamond a \square . V této symbolice je $\diamond(P) \Leftrightarrow \neg\square(\neg P)$, resp. $\square(P) \Leftrightarrow \neg\diamond(\neg P)$, proto modální logika vystačí s jedním z nich (zde operátor \square).
- Je-li F formule jazyka L a M modální operátor, pak $M(F)$ je formule modální logiky.

Jazyk L_K modální logiky je nadmnožinou jazyka L . Je třeba, aby byl schopen vyjádřit všechno, co umí logika prvního řádu.



Definice 5.1 (gramatiky jazyka LK Backus-Naurovou formou)

$\langle \text{atom} \rangle ::= \langle \text{výroková proměnná} \rangle$

$::= \langle \text{logická konstanta} \rangle$

$\langle \text{formule} \rangle ::= \langle \text{atom} \rangle$

$::= \neg(\langle \text{formule} \rangle)$

$::= \langle \text{modální operátor} \rangle (\langle \text{formule} \rangle)$

$::= (\langle \text{formule} \rangle) \langle \text{binární logická spojka} \rangle (\langle \text{formule} \rangle)$

$\langle \text{výroková proměnná} \rangle ::= a / b / c / \dots$

$\langle \text{logická konstanta} \rangle ::= \text{true} / \text{false}$

$\langle \text{binární logická spojka} \rangle ::= \& / \vee / \rightarrow / \leftrightarrow$

$\langle \text{modální operátor} \rangle ::= \square / \diamond$

5.2 Sémantika jazyka LK

5.2.1 Sémantika možných světů

Má-li být tvrzení „Tráva je zelená.“ přiřazena pravdivostní hodnota, je třeba mít na mysli, že se tato pravdivostní hodnota může změnit podle aktuálního stavu modelovaného světa.

Modální logika dokáže vystihnout rozdíl mezi tím, co je pravda, resp. nemůže být nepravda (bezpodmínečně nutná pravda), nebo co by mohlo být pravda (podmíněná pravda). Zjištění, o kterou z pravd se jedná, pak spočívá v možné či nemožné představě světa, tj. modelu, v němž je to tak nebo jinak. Pojem světa, který zůstává stálým po dobu, k níž se vztahuje znalostní bázi reprezentovaný aktuální stav světa, modální logika nahrazuje pojmem „možných světů“, který proměny modelovaného světa dokáže postihnout. Sémantika jazyka L_M je definována v pojmech *možných světů* $\{W_0, W_1, \dots\}$ a *relace přístupnosti* R mezi nimi. To určuje její *modální rámeček*.

Reprezentace znalostí

Protože logika prvního řádu pojednává o formulích, které jsou pravdivé v jednotlivých svých modelech - světech, jeví se přirozené použít modely odpovídající logiky prvního řádu, tj zde výrokové logiky, jako možné světy $\{W_0, W_1, \dots\}$ pro modální logiku.

Ideu možných světů je třeba zahrnout do samotných základů formalizace v rámci modální logiky. Má-li být nějaké tvrzení nutně pravdivé, musí být pravdivé ve všech možných světech, do nichž se aktuální svět může dostat. Podobně, má-li být nějaké tvrzení možná pravdivé, musí existovat alespoň jeden možný svět, v němž je toto tvrzení pravdivé.

Příklad 5.1

Pohádkový Honza, který se vydal do světa, nese ve svém ranečku buchtu. Tvrzení "Honza má buchtu." je tedy ve světě pohádky o Honzovi v počátečním stádiu (světě W_0) pravdivé. Jak ale známe Honzu, je nám jasné, že buchtu postupně mizí v jeho žaludku, takže tvrzení "Honza má buchtu." v nějakém světě W_1 přestává být pravdivé, neboť buchtu už je jenom část, až je nakonec ve světě W_2 , v němž už z buchtu nic nezbylo, zcela jistě nepravdivé. Ze světa W_0 lze přejít do světa W_1 nebo přímo do světa W_2 , ale obrácený přechod není možný. Podobně ze světa W_1 lze přejít do světa W_2 , ale ne naopak.



Kromě množiny možných světů je pro formalizaci sémantiky modální logiky tedy též třeba uvažovat binární relaci jejich vzájemné *přístupnosti*.

Definice 5.2 (možného světa)

Možným světem dané množiny formulí modální logiky je každý její model, tj. každá struktura pravdivostních hodnot jejich atomů, pro niž je množina formulí interpretována pravdivostní hodnotou true.



Definice 5.3 (modálního rámce)

Modálním rámcem je dvojice $\mathbf{M} = (\{W_0, W_1, \dots\}, \mathbf{R})$, kde $\{W_0, W_1, \dots\}$ je množina možných světů, \mathbf{R} je relace vzájemné přístupnosti mezi nimi.



5.2.2 Pravdivost a splnitelnost formulí modální logiky K

Podobně jako v klasické výrokové logice je definována interpretace modální výrokové formule ve světě W_k na základě pravdivosti svých atomů v rámci modelu W_k a podle interpretačních pravidel vztahujících se k jednotlivým logickým spojkám. Pravidla vztahující se k jednotlivým logickým spojení jsou pravdivostně funkční pouze v případech, kdy logickými spojkami spojované podformule jsou formulemi klasické výrokové logiky. V případě modální formule s modálním operátorem musí být ve světě W_k , který je součástí modálního rámce \mathbf{M} pravdivost modálních formulí definována s ohledem na tento modální rámec, jak je dáno následující definicí 3.4.

Reprezentace znalostí



Definice 5.4 (Interpretace formule A jazyka L_K ve světě W_i)

Interpretace formule A modální výrokové logiky v daném světě W_i modálního rámce M při valuaci ν jejích proměnných, odpovídající modelu W_i , je funkce

$$I(A[W_i]) = \text{true/false}$$

definovaná induktivně takto:

1. *Báze*

Je-li formule A symbolem pro výrokovou proměnnou p v jazyce L_K , pak

$$I(A[W_i]) = \nu(p[W_i]).$$

Je-li formule A symbolem pro logickou konstantu true/false , pak

$$I(A[W_i]) = D(A) = \text{true/false}.$$

2. *Indukce*

Jsou-li B, C formule jazyka L_K , pak

- a) $I(\neg B[W_i]) = \text{true}$, právě když $I(B[W_i]) = \text{false}$
- b) $I((B \vee C)[W_i]) = \text{true}$, právě když $I(B[W_i]) = \text{true}$ nebo $I(C[W_i]) = \text{true}$
- c) $I((B \& C)[W_i]) = \text{true}$, právě když současně platí $I(B[W_i]) = \text{true}$ a $I(C[W_i]) = \text{true}$
- d) $I((B \rightarrow C)[W_i]) = \text{true}$, právě když $I(B[W_i]) = \text{false}$ nebo $I(C[W_i]) = \text{true}$
- e) $I((B \leftrightarrow C)[W_i]) = \text{true}$, právě když současně $I((B \rightarrow C)[W_i]) = \text{true}$ a $I((C \rightarrow B)[W_i]) = \text{true}$.
- f) $I(\Box(P)[W_i]) = \text{true}$, právě když P je pravdivá ve všech světech W_j tohoto rámce takových, že W_j je přístupný z W_i , tj. platí $R(W_i, W_j)$.
- g) $I(\Diamond(P)[W_i]) = \text{true}$, právě když existuje svět W_j , tohoto rámce takový, že P je v něm pravdivá a W_j je přístupný z W_i , tj. platí $R(W_i, W_j)$.

3. *Generalizace*

Pravdivostní hodnota interpretace libovolné formule jazyka L_K výrokové modální logiky K v daném světě W_i se stanoví vždy konečným počtem aplikací kroků 1. a 2.



Příklad 5.2

Nechť jazyk L_K obsahuje výrokové proměnné p (= "prší"), q (= "je chladno") a nechť je dáno: modální rámec $M = \{\text{dnes, zítra}\}$, relace přístupnosti $R = \{(\text{dnes, dnes}), (\text{dnes, zítra}), (\text{zítra, zítra})\}$ a tyto valuace:

$$\nu(p[\text{dnes}]) = \text{true}$$

$$\nu(p[\text{zítra}]) = \text{false}$$

$$\nu(q[\text{dnes}]) = \text{false}$$

Reprezentace znalostí

$$v(\mathbf{q}[\text{zítra}]) = \text{true}.$$

Potom platí $I((\Diamond \mathbf{p})[\text{dnes}]) = \text{true}$, stejně jako $I((\Diamond \mathbf{q})[\text{zítra}]) = \text{true}$, zatímco např. $I(\Box(\mathbf{p})[\text{dnes}]) = \text{false}$, neboť ve světě "zítra", který je přístupný ze světa "dnes", má \mathbf{p} pravdivostní hodnotu false.

Definice 5.5 (pravdivosti formule v daném rámci)

Formule je pravdivá v daném rámci, je-li pravdivá v každém ze světů tohoto rámce.



Definice 5.6 (modelu a splnitelnosti formule v modálním rámci)

Formule P je splnitelná v daném modálním rámci $\mathbf{M} = (\{W_0, W_1, \dots\}, \mathbf{R})$, existuje-li valuace jejích proměnných (model formule), při níž je formule P pravdivá v daném rámci.



Formule P je platná v daném modálním rámci $\mathbf{M} = (\{W_0, W_1, \dots\}, \mathbf{R})$, je-li pravdivá v daném modálním rámci pro všechny valuace svých proměnných.

Rozšíření pojmu platnosti na logickou platnost v obdobném smyslu jako u logiky prvního řádu zde v podstatě nemá smysl, neboť pravdivost se vždy vztahuje k nějakému rámci s určitou relací přístupnosti. Platnost proto musí být uvažována vzhledem k vlastnostem relace přístupnosti.

5.3 Formální systém výrokové modální logiky

Axiomatické systémy modální logiky jsou vždy modifikací některého axiomatického systému logiky prvního řádu. Všechny axiomy logiky prvního řádu proto jsou též modálně logicky platné bez ohledu na relaci přístupnosti \mathbf{R} . Světy, které tvoří rámec, jsou totiž, stejně jako jiné světy, modely těchto axiomů. Zde bude diskutován formální systém \mathbf{K} modální logiky založený na axiomatickém systému \mathbf{H} hilbertovského typu (výrokové logiky), disponujícím pravidlem Modus Ponens.

5.3.1 Axiomy modálních logik

Základní axiom \mathbf{K} -logiky je axiom \mathbf{K} (distributivity):

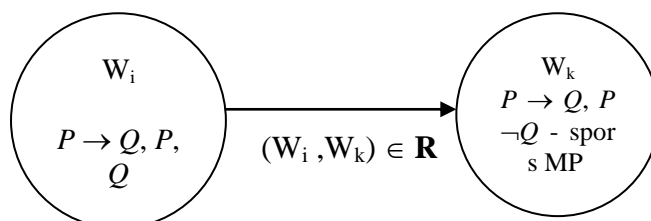
$$\Box(P \rightarrow Q) \rightarrow (\Box P \rightarrow \Box Q)$$

Důkaz (nepřímý) logické platnosti axiomu \mathbf{K} :

Aby bylo možno uvažovat podmínky kladené na relaci přístupnosti, pro něž je tento axiom platný, je třeba uvažovat případy, pro něž neplatí. Sémantika logiky prvního řádu zná pouze jediný takový případ pro spojku \rightarrow :

Reprezentace znalostí

Formule $\Box(P \rightarrow Q)$ i $\Box P$ musí být interpretovány jako true ve světě W_i a současně musí být v tomto světě $\Box Q$ interpretována jako false. To ale platí, právě když $P \rightarrow Q$ a zároveň P jsou true ve všech světech přístupných z W_i , ale formule Q je v některém z těchto světů false. Protože všechny tyto světy jsou modely logiky prvního řádu (pravidlo MP „Modus Ponens“ platí), nemůže tento případ nastat.



Obrázek 5.1

K tedy platí bez ohledu na to, jaká je relace **R** přístupnosti. Modální logika, která má axiom **K**, se nazývá *normální modální logika*.

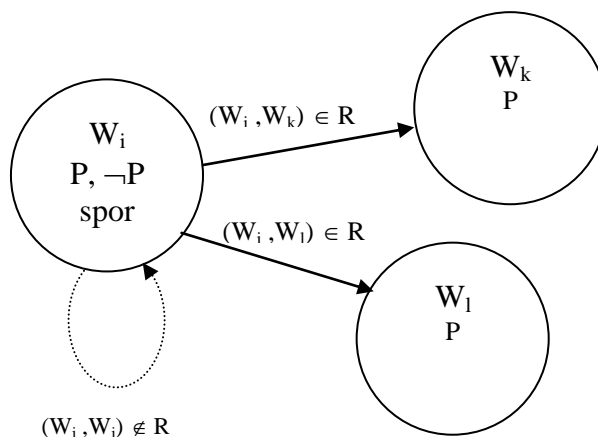
Následující axiomy jsou platné v rámci omezení položených na relaci **R**. Přidáním axiomu **T** vzniká **T**-logika apod.

Axiom **T** říká, že každá formule, která je nutně pravdivá, je pravdivá.

Axiom **T**: $\Box P \rightarrow P$

Je třeba uvažovat jako předtím jediný případ, kdy může nastat jeho nepravdivost false: P by musela být true ve všech světech přístupných z některého W_i , nikoliv však z tohoto W_i . W_i by ale musel nebýt přístupný ze sebe sama. Uvažujme, kdy to může nastat: necht' je relací přístupnosti relace „později než“. Tím se $\Box P$ interpretuje jako „stane se pravdivým a zůstane pravdivým“. Pro tento případ platí axiom **K**, nikoliv však axiom **T**. Protipříklad: „Bude pravda a zůstane pravda, že jsem mrtvý“ je pravda, ale „Jsem mrtvý“ není pravda.

T platí ve všech případech, kdy je relace přístupnosti reflexivní.



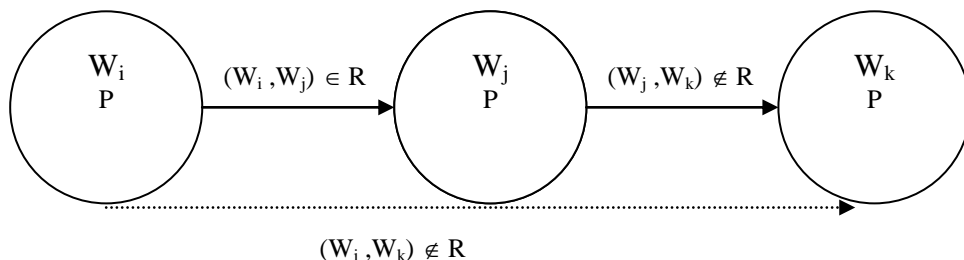
Obrázek 5.2

Další axiom říká, že když je formule nutně true, pak je nutně true.

Axiom **S4**: $\Box P \rightarrow \Box(\Box P)$

Reprezentace znalostí

Opět sledujeme možnost, kdy to není pravda. To může nastat pouze tehdy, jestliže existuje svět W_j přístupný z W_i , ale takový, že v něm $\Box P$ je false. To je možné, je-li zde další svět W_k , který je přístupný z W_j , pro nějž je P false. To ale je možné pouze když W_k není přístupný z W_i . To znamená, že aby neplatil **S4**, nesmí být relace přístupnosti tranzitivní.



Obrázek 5.3

Příklad 5.3

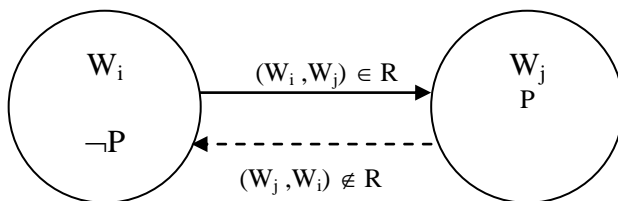
Uvažujme relaci „Je do 24 hodin.“. Tato relace je jasně reflexivní, ale nikoliv tranzitivní. Relace přístupnosti zde podporuje interpretaci modálního operátoru \Box ve smyslu „bude pravda od tohoto okamžiku do 24 hodin.“

\Box (datum je 20. nebo 21. června) je pravda, ale není pravda $\Box(\Box$ (datum je 20. nebo 21. června))

Axiom **B**: $\neg P \rightarrow \Box(\neg(\Box P))$

Tento axiom říká, že je-li P false, pak je to ten případ, že není nutně true, jinak řečeno že je zde možnost, aby byl false.

Aby axiom **B** neplatil, je třeba, aby existoval v daném rámci svět W_i , v němž by $\neg P$ byla true a zároveň $\neg(\Box P)$ false. To ale znamená, že P musí být false ve W_i a $\Box P$ musí být true v nějakém W_j . To však lze pouze tehdy, není-li W_i přístupný z W_j , neboť kdyby byl, musela by tam být true i formule P . Proto, aby axiom **B** selhal, je třeba, aby existoval pár světů (W_i, W_j) takový, že platí $R(W_i, W_j)$, ale neplatí $R(W_j, W_i)$, tj. relace přístupnosti nesmí být symetrická.



Obrázek 5.4

5.3.2 Odvozovací pravidla modální logiky

Formální systém výrokové modální logiky, založený na axiomatickém systému **H** hilbertovského typu, disponuje známým odvozovacím pravidlem **MP** (Modus Ponens), a nově zavedeným modálním pravidlem nutnosti **NEC**:

Reprezentace znalostí



Definice 5.7 (odvozovacího pravidla MP)

Pravidlo tvaru

$$\frac{U \vdash A \quad U \vdash A \rightarrow B}{U \vdash B},$$

„Z dokazatelnosti vět (formulí) A a $A \rightarrow B$ odvod' větu (formuli) B .“
se nazývá odvozovací pravidlo Modus Ponens (**MP**).



Definice 5.8 (odvozovacího pravidla NEC)

Pravidlo tvaru

$$\frac{U \vdash A}{U \vdash \Box(A)}$$

„Je-li z množiny předpokladů dokázána formule A , pak z ní odvod' formuli $\Box(A)$.“
se nazývá odvozovací pravidlo nutnosti (necessity – **NEC**).

5.3.3 Sémantická korektnost a úplnost modální logiky K

Věta 5.1

Modální logika **K** je sémanticky korektní a úplný formální systém.

Důkaz

Předpokládejme, že F je modální rámec, tj. množina $\{W_0, W_1 \dots\}$ modelů v logice prvního řádu, spolu s relací přístupnosti R , pro nějaký soubor modálních axiomů, který splňuje určité podmínky.

Jestliže existuje důkaz formule A z axiomů logiky prvního řádu a tohoto souboru modálních axiomů s použitím **MP** a **NEC**, potom A platí v rámci F .

Důkaz indukci podle délky důkazu (využívající sporu s předpokladem, že existuje kratší důkaz, pro nějž věta může selhat) :

Předpokládejme, že A je věta, jejíž nejkratší důkaz je tak krátký jako nějaký důkaz, pro nějž věta neplatí. Je třeba uvažovat tři případy :

- 1) A je axiom. Už byla dokázána korespondence mezi axiomy modální logiky a relací přístupnosti – každý axiom je platný ve vhodném rámci
- 2) A vyplývá na základě MP z dokázané formule, tj. z dokázaných B a $B \rightarrow A$, pro něž platí indukční předpoklad, se odvodí A . Protože každý svět z F je modelem logiky prvního řádu, A musí platit v každém z nich, což odporuje předpokladu, že pro něj formule selhává.

Reprezentace znalostí

- 3) A vyplývá na základě NEC z dokázané formule, pro niž platí indukční předpoklad. Předpokládejme A je ve skutečnosti $\Box(B)$, pro niž je dokázáno B . Má-li věta A selhat, musí v F existovat svět W_i , v němž $\Box(B)$ není true, tedy musí existovat svět W_j přístupný z W_i takový, že v něm je B false. To ale odporuje indukčnímu předpokladu.

Existuje mnoho způsobů, jak stanovit axiomy modální logiky.

5.4 Sémantické tablo modální K-logiky

Formuli $\Box P$ je zde třeba uvažovat ve smyslu $\forall w P(w)$ a zavést zvláštní argument označující možné světy, v nichž formule platí. Při použití tablové důkazové metody je pak potřeba držet stopu světů reprezentovaných určitými valuacemi zkoumat pak přístupnost z jednoho světa do druhého. Způsob realizace spočívá v přiřazení indexu ke každé formuli, jehož smyslem je postavit taková omezení na množině světů, v nichž má být formule true.

5.4.1 Indexování možných světů a tablová odvozovací pravidla

Definice 5.9

Indexem může být proměnná, konstanta nebo term tvaru $\text{succ}(I_1, I_2)$. Poslední z nich určuje cestu, kde I_2 je výchozí svět, I_1 je místo určení. Konstanty se značí malým písmenem, proměnné velkým písmenem.



Definice 5.10

Indexovaná formule je dvojice $F:I$ sestávající z formule F výrokové varianty modální logiky a indexu I .



Dvě indexované formule $F_1:I_1$ a $F_2:I_2$ jsou komplementární, jestliže platí

- 1) F_1 je negací F_2 a naopak,
- 2) I_1 a I_2 jsou unifikovatelné.

Definice 5.11

Modální tablo je strom indexovaných formulí sestrojený na základě následujících pravidel.



- 1) Non-modální pravidla odpovídají tablovým pravidlům logiky prvního řádu:
 - a) Je-li $\Phi \cup \{P \& Q : I\}$ návěští uzlu modálního tabla, potom následující uzel má návěští $\Phi \cup \{P \& Q : I, P : I, Q : I\}$ (podobně pro ostatní ekvivalentní logická spojení).
 - b) Je-li $\Phi \cup \{P \vee Q : I\}$ návěští uzlu modálního tabla, potom uzly dvou následujících větví mají návěští $\Phi \cup \{P \vee Q : I, P : I\}$, $\Phi \cup \{P \vee Q : I, Q : I\}$ (podobně pro ostatní ekvivalentní logická spojení).

Reprezentace znalostí

2) Modální pravidla:

- a) Je-li $\Phi \cup \{\neg(\Box(P)):I\}$ návěští uzlu modálního tabla, potom následující uzel má návěští $\Phi \cup \{\neg(\Box(P)):I, \neg P:\text{succ}(I',I)\}$, kde I' je nová konstanta.
- b) Je-li $\Phi \cup \{\Box(P):I\}$ návěští uzlu modálního tabla, potom následující uzel má návěští $\Phi \cup \{\Box(P):I, P:\text{succ}(I',I)\}$, kde I' je nová proměnná.

Tablová částečně rozhodovací procedura výrokové modální logiky **K** je založena na platnosti následující analogické věty tablového formálního systému výrokové logiky.

Věta 5.2

Formule modální výrokové logiky **K** je nespílitelná, právě když existuje sémantické tablo, které se uzavře.

Důkazy logické platnosti formulí logiky **K**, podobně jako důkazy logického důsledku daných předpokladů se provádějí nepřímým tablovým postupem.

5.4.2 Příklady modálních tablových důkazů



Příklad 5.4

Nepřímý tablový důkaz $\Box(P \& Q) \rightarrow \Box(P)$

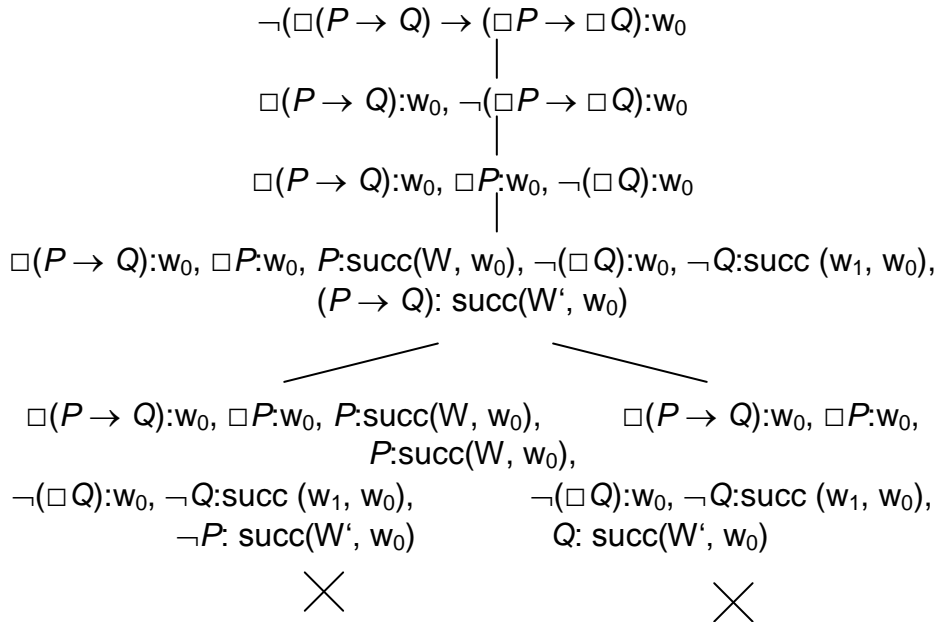
$$\begin{array}{c}
 \neg(\Box(P \& Q) \rightarrow \Box(P)):w_0 \\
 | \\
 \Box(P \& Q):w_0, \neg\Box(P):w_0 \\
 | \\
 \Box(P \& Q):w_0, (P \& Q):\text{succ}(W_0, w_0), \neg\Box(P):w_0 \\
 | \\
 \Box(P \& Q):w_0, (P \& Q):\text{succ}(W_0, w_0), P:\text{succ}(W_1, w_0), \\
 Q:\text{succ}(W_2, w_0), \neg\Box(P):w_0, \neg P:\text{succ}(w_1, w_0) \\
 \times
 \end{array}$$

Existuje sémantické tablo negované formule v jeho kořeni, které se uzavře, neboť je zde možná unifikace substitucí $\{w_1/W_1\}$. Původní formule je tedy logicky platná.

Reprezentace znalostí

Příklad 5.5

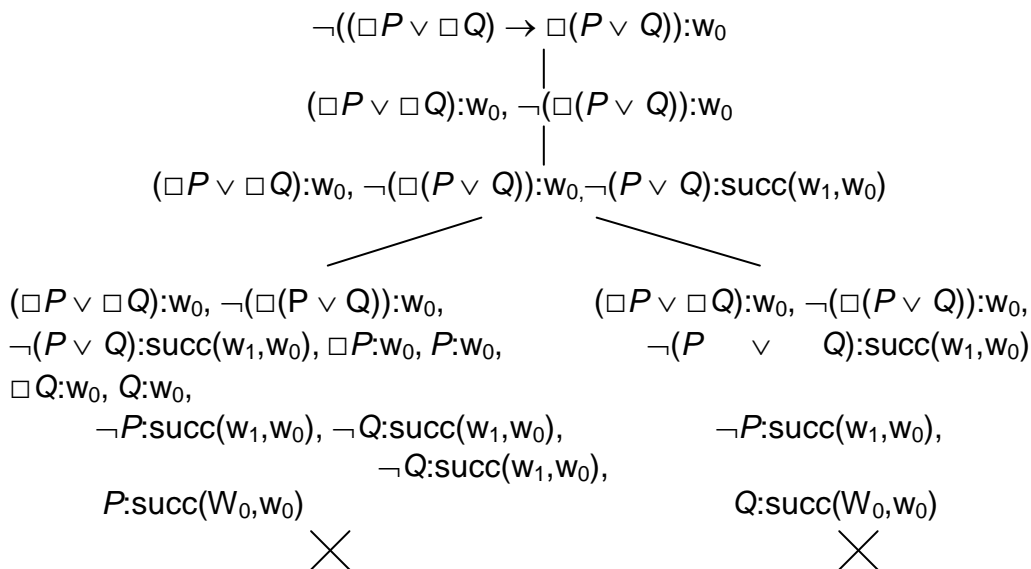
Nepřímý tablový důkaz logické platnosti axiomu **K**:



Tablo se uzavře, neboť je zde možná unifikace substitucí $\{ w_1/W, w_1/W' \}$.

Příklad 5.6

Tablový důkaz $(\Box P \vee \Box Q) \rightarrow \Box(P \vee Q)$



Reprezentace znalostí



Příklad 5.7

Důkaz, že neplatí $\Box P \rightarrow P$:

$$\begin{array}{c} \neg(\Box P \rightarrow P):w_0 \\ | \\ \Box P:w_0, P:\text{succ}(W_0, w_0), \neg P:w_0, \\ \bigcirc \end{array}$$

V daném případě je zřejmé, že neexistuje možnost konstrukce uzavřeného sémantického tabla, proto není formule v kořeni table nesplnitelná a původní formule $\Box P \rightarrow P$ neplatí.



Příklad 5.8

Důkaz, že neplatí $\Box(P \vee Q) \rightarrow (\Box P \vee \Box Q)$

$$\begin{array}{c} \neg(\Box(P \vee Q) \rightarrow (\Box P \vee \Box Q)):w_0 \\ | \\ \Box(P \vee Q):w_0, \neg(\Box P \vee \Box Q):w_0 \\ | \\ \Box(P \vee Q):w_0, \neg(\Box P):w_0, \neg(\Box Q):w_0 \\ | \\ \Box(P \vee Q):w_0, (P \vee Q):\text{succ}(W_0, w_0), \neg(\Box P):w_0, \neg P:\text{succ}(w_1, w_0), \\ \neg(\Box Q):w_0, \neg Q:\text{succ}(w_1, w_0) \\ \swarrow \quad \searrow \\ \begin{array}{c} \Box(P \vee Q):w_0, P:\text{succ}(W_0, w_0), \dots \\ \neg P:\text{succ}(w_1, w_0), \neg Q:\text{succ}(w_1, w_0) \\ \bigcirc \end{array} \quad \begin{array}{c} \Box(P \vee Q):w_0, Q:\text{succ}(W_0, w_0), \dots \\ \neg P:\text{succ}(w_1, w_0), \\ \neg Q:\text{succ}(w_2, w_0) \\ \bigcirc \end{array} \end{array}$$

Zdalo by se, že se obě větve uzavírají, jestliže se provede unifikace substitucí w_1 za proměnnou W_0 v případě levé větve a substitucí w_2 za proměnnou W_0 v případě pravé větve. To však jsou vzájemně nekompatibilní vazby, proto nepřímý důkaz logické platnosti této formule selhává.

Protipříklad : $\Box(P \vee \neg P)$ platí podle **NEC** pro libovolné P , ale nelze z toho vyvodit, že platí $\Box P \vee \Box(\neg P)$.

Tato metoda ve výrokové verzi modální logiky **K** v každém případě rozhoduje splnitelnost formule (taktéž logický důsledek). Pro její rozšíření na další typy modální logiky je třeba uvažovat alternativní pravidla, zavádějící indexy potenciální kontradikce.

Pro **T** a **S4** je třeba být připraven na vymazávání pohybů z cesty.

V případě modální logiky prvního řádu je unifikace stejně jako u predikátové logiky nutná. Existenční kvantifikace musí být spojena se skolemizací, která

Reprezentace znalostí

odkazuje na index světů, v nichž má být formule pravdivá. Držet stopu proměnných unifikací znamená jejich vymazání z indexů. Tato cesta je zdlouhavá. Efektivnější je náhrada pravidla pro formule tvaru $\Box(P):I$ takto:

Je-li $\Phi \cup \{\Box P:I\}$ návěští uzlu modálního tabla, potom návěští následujícího uzlu lze rozšířit

$\Phi \cup \{\Box P:I, P:I\}$, je-li místem určení z I proměnná, a

$\Phi \cup \{\Box P:I, P:\text{succ}(V,I)\}$, kde V je nová proměnná, je-li místem určení z I konstanta.

Příklad 5.9

Důkaz **T** ($\Box P \rightarrow P$)



$$\begin{array}{c} \neg(\Box P \rightarrow P):w_0 \\ | \\ \Box P:w_0, \neg P:w_0, P:\text{succ}(W_0, w_0) \\ \times \end{array}$$

Vzhledem k relaci \mathbf{R}_T je to akceptovatelný důkaz, neboť $\mathbf{R}_T(w_0, \text{succ}(W_0, w_0))$ v této relaci, která je reflexivní, platí .

Příklad 5.10

Důkaz **S4** ($\Box P \rightarrow \Box(\Box P)$) :



$$\begin{array}{c} \neg(\Box P \rightarrow \Box(\Box P)):w_0 \\ | \\ \Box P:w_0, \neg(\Box(\Box P)):w_0, \neg(\Box P):\text{succ}(w_1, w_0), \neg P:\text{succ}(w_2, \text{succ}(w_1, w_0)), \\ P:\text{succ}(W_0, w_0) \\ \times \end{array}$$

Vzhledem k relaci \mathbf{R}_{S4} , která je tranzitivní, je to akceptovatelný důkaz, neboť v $\mathbf{R}_{S4}(\text{succ}(w_2, \text{succ}(w_1, w_0)), \text{succ}(W_0, w_0))$ platí.

Reprezentace znalostí



Kontrolní otázky a úkoly

1. V čem spočívá rozšíření jazyka modální logiky vzhledem k logice prvního řádu?
2. Co je modální rámec?
3. Kdy je pravdivá formule $\Box(P)$ ve světě W_i v modálním rámci $(\{W_0, W_1, \dots\}, R)$?
4. Kdy je pravdivá formule $\Box(P)$ v modálním rámci $(\{W_0, W_1, \dots\}, R)$?
5. Jak je vybudován formální systém výrokové modální logiky?
6. V čem je rozdíl mezi sémantickým tablem tradiční výrokové logiky a modální výrokové logiky?
7. Jak se v modálních důkazech indexují možné světy?



Shrnutí

Modální logika se pokouší zachytit taková tvrzení přirozeného jazyka, jejichž pravdivostní hodnota není jednoznačně true nebo false, ale závisí na podmínkách vnějšího světa a jejich změnách v čase. Tato tvrzení zpravidla obsahují slova jako „možná“, „snad“, „asi“, „zpravidla“ apod., časový rozměr pak bývá vyjadřován pomocí „bylo“, „bude“, „je“.

Možnost a nutnost se nazývají aletické modality nebo módy pravdivosti. Logické systémy obsahující operátory pro „je možné, že“ (označuje se \Box), „je nutné, aby“ (označuje se \Box) se též nazývají aletické logiky.

Jazyk modální logiky je nadmnožinou jazyka L1. Všechna syntaktická pravidla jazyka L výrokové logiky jsou zároveň pravidly jazyka LM výrokové modální logiky.

Rozšíření jazyka pro vyjádření módů pravdivosti :

- K vyjádření možnosti a nutnosti se zde zavádějí dva modální operátory : \Box a \Box .
- V této symbolice je $\Box(P) \equiv \Box\Box(\Box P)$, resp. $\Box(P) \equiv \Box\Box(\Box P)$,
- proto modální logika vystačí s jedním z nich (zde operátor \Box).
- Je-li F formule jazyka L a M modální operátor, pak M(F) je formule modální logiky.

Sémantika jazyka LM je definována v pojmech možných světů $\Box W_0, W_1, \dots \Box$ a relace přístupnosti R mezi nimi. Tomu se říká modální rámec.

Modálním rámcem je dvojice $(\Box W_0, W_1, \dots \Box, R)$, kde $\Box W_0, W_1, \dots \Box$ je množina možných světů, R je relace přístupnosti mezi nimi.

Formule tvaru $\Box(P)$ je pravdivá ve světě W_i v modálním rámci $(\Box W_0, W_1, \dots \Box, R)$, právě když P je pravdivá ve všech světech W_j tohoto rámce takových, že W_j je přístupný z W_i , tj. $R(W_i, W_j)$.

Každá jiná formule je pravdivá ve světě W_i v rámci $(\Box W_0, W_1, \dots \Box, R)$, právě když W_i je její model ve smyslu logiky prvního řádu.

Reprezentace znalostí

Formule je pravdivá v daném rámci, je-li pravdivá v každém ze světů tohoto rámce.

Axiomatické systémy modální logiky jsou vždy modifikací některého axiomatického systému logiky prvního řádu. Všechny axiomy příslušného systému logiky prvního řádu proto jsou též modálně platné bez ohledu na relaci přístupnosti R . Světy, které tvoří rámec jsou, stejně jako jiné světy, modely těchto axiomů.

Formální systém výrokové modální logiky, založený na axiomatickém systému H hilbertovského typu, disponuje známým odvozovacím pravidlem MP (Modus Ponens)

a nově zavedeným modálním pravidlem nutnosti NEC: Je-li UA , pak je $U\Box(A)$. Modální logika je sémanticky korektní formální systém.

Modální tablo je binární strom ohodnocený indexovanými formulemi.

Indexem může být proměnná, konstanta nebo term tvaru $\text{succ}(I_1, I_2)$. Poslední z nich určuje cestu, kde I_2 je výchozí svět, I_1 je místo určení. Konstanty se značí malým, proměnné velkým písmenem.

Indexovaná formule je dvojice $F:I$ sestávající z formule jazyka LM výrokové varianty modální logiky a indexu.

Dvě indexované formule $F_1:I_1$ a $F_2:I_2$ jsou komplementární, jestliže

1. F_1 je negací F_2 a naopak
2. I_1 a I_2 jsou unifikovatelné.

6 NEMONOTÓNŇÍ LOGIKY

Cílem kapitoly je:

- nemonotónní budování teorie nad znalostní bází,
- charakterizovat vlastnosti monotónních formálních systémů,
- seznámení s nemonotónní teorií,
- definování Reiterova modifikace default odvozovacího pravidla,
- formalizace teorie v nemonotónních systémech.

Po jejím prostudování byste měli být schopni:

- rozšířit množinu předpokladů,
- monotónního formálního dokazování,
- používat default logiku,
- využívat Reiterovou default negaci.

Klíčová slova:

Znalostní báze, nemonotónní budování teorie, výroková logika, speciální axiomy, default logika.

Čas potřebný ke studiu: 12 hodin



Průvodcem studia

Nemonotónní logika vznikla za účelem systematicky zachytit tzv. zrušitelné uvažování, tj. typ každodenního uvažování, které vede jen k provizorně platným argumentům, jenž mohou být následně staženy s příchodem nových informací. Tím se ovšem nemonotónní logika dostává do ostrého kontrastu s klasickou logikou, která je monotónní, tj. žádné doda-tečné premisy nemohou zrušit jednou již platné argumenty. To bylo pro mnohé dostatečným důvodem k tomu, aby nemonotónní logice upřeli status logiky. V této kapitole si ukážeme, že takový závěr je příliš unáhlený a že nemonotónní logika má právo se nazývat logikou.

6.1 Nemonotónní budování teorie nad znalostní bází

Koncem sedmdesátých let minulého století se v přístupech k problematice reprezentace znalostí objevila snaha vytvořit odvozovací pravidla, která by při budování teorií nad znalostní bází umožňovala počítat s výjimkami, revidovat to, co již bylo odvozeno, a která by akceptovala jen takové odvozené důsledky, které jsou „normální“ (všeobecně akceptovatelné).

Reprezentace znalostí

6.1.1 Monotónnost budování teorie v systémech logiky prvního řádu

Pro budování teorií ve znalostních systémech založených na klasické logice prvního řádu je charakteristické to, že výchozí znalosti znalostní báze jsou zapsány ve formulích příslušného formálního jazyka jako *speciální axiomy teorie* a systém pak generuje formálními prostředky další *věty teorie* nebo rozhoduje s použitím svých formálních rozhodovacích pravidel, zdali daná formule je nebo není větou teorie. Ve formálních systémech, které jsou *sémanticky korektní a úplné*, jsou věty teorie právě *logickými důsledky* daných speciálních axiomů teorie (znalostní báze). Přidáváním formálně dokázaných vět, stejně jako přidáváním dalších znalostí do znalostní báze, se *teorie monotónně rozšiřuje*, přičemž její rozšíření nijak neovlivní platnost vět až dosud vybudované teorie. Tento model je v mnoha případech, užitečný, ale má některá závažná omezení:

- Odvozovací pravidla nikdy neodvodí novou znalost o modelovaném světě, která by ve znalostní bázi nebyla implicitně obsažena. Odvozená znalost ji pouze vyjeví v její explicitně vyjádřené podobě. Odvození důsledku daných předpokladů tedy pouze způsobí to, že se znalostem skrytým implicitně v předpokladech dostane explicitního vyjádření.
- Jestliže je věta dokázána (viz dále v příkladu) pouze na základě části znalostní báze, nezjistí se její neplatnost (spornost), která by se eventuálně vyjevila při použití celé znalostní báze.
- Znalostní bázi lze rozšiřovat o další formule, včetně těch, které ji učiní nekonsistentní, aniž by přestala v rámci teorie platit její některá již dokázaná věta. Pokud je znalostní báze ve své výchozí podobě konsistentní, je teorie nad ní vybudovaná bezesporná, tedy nemůže se stát, že bude obsahovat formuli i její negaci. Jestliže však nově dodaná formule způsobí, že se rozšířená teorie stane nekonsistentní (spornou), je možno, případně automatizovaným generováním dalších vět bez kontroly konsistence, vygenerovat spousty odporujících si nesmyslů.
- Jazykem logiky prvního řádu nelze zachytit všechno, co je potřeba říci o modelovaném světě, zejména s ohledem na průběh dalších poznání o něm, Existují souvislosti, kdy je zřejmé, že dokazování v tradičních logických systémech odvozování nevystihují usuzování tak, jak probíhá ve skutečnosti v lidských myslích. Často se stává, že inteligentní systémy pracují s předpoklady, které jsou všeobecně přijímány, ale které mohou být později vyvráceny nebo modifikovány, neboť ne všechna tvrzení, která je třeba v rámci umělé inteligence formalizovat, jsou trvale platná. Důsledky se dodáním dalších informací o stavu modelovaného světa mohou stát neplatnými a je třeba je revidovat.
- Logika prvního řádu nemožňuje formulovat tvrzení alespoň s takovou mírou neurčitosti, která by dávala možnost revize již odvozených tvrzení na základě nových znalostí dodaných do znalostní báze.

Vlastnost formulovaná ve druhém bodě je typickým důsledkem vlastnosti *monotónnosti* tradičních formálních systémů logiky prvního řádu. Dedukce v logice prvního řádu vychází z určité znalostní báze $\mathbf{S} = \{P_1, P_2, \dots, P_n\}$, z níž

Reprezentace znalostí

je pak pomocí odvozovacích pravidel odvozena formule A . Stejně dobře však lze A odvodit z nadmnožiny \mathbf{R} . Pro všechny zde diskutované formální teorie totiž platí v logice prvního řádu známá *věta o rozšíření množiny předpokladů*.



Věta 6.1 (o rozšíření množiny předpokladů)

Jsou-li P_1, P_2, \dots, P_n formule tvořící množinu předpokladů a P je další formule tuto množinu rozšiřující, potom platí – li $P_1, P_2, \dots, P_n \vdash A$, platí též $P, P_1, P_2, \dots, P_n \vdash A$.

Tato věta je důsledkem kompaktnosti logiky prvního řádu, která tvrdí, že formule A je dokazatelná z množiny předpokladů \mathbf{S} , tj. $\mathbf{S} \vdash A$, jestliže pro nějakou konečnou podmnožinu $\mathbf{S}_A \subseteq \mathbf{S}$ platí $\mathbf{S}_A \vdash A$. Ostatní formule, které nevstupují do důkazu, nejsou předmětem zájmu. Tato vlastnost se nazývá vlastností *lokality* tradiční formální logiky.



Příklad 6.1

Nechť $\mathbf{S} = \{\text{má_řád}(\text{bb}, \text{zvířata}), \text{žena}(\text{bb}), \text{herec}(\text{bb}, \text{slavný}), \forall x \text{ má_řád}(x, \text{zvířata}) \rightarrow \neg \text{nosí}(x, \text{kožešiny}), \forall x (\text{herec}(x, \text{slavný}) \rightarrow \text{bohatý}(x))\}$ je množina formulí (speciálních axiomů) tvořících znalostní bázi v některém z uvažovaných axiomatických systémů.

Formálně odvozenou větou teorie ze speciálních axiomů \mathbf{S} (logickým důsledkem \mathbf{S}) je zde mimo jiné formule

$$\neg \text{nosí}(\text{bb}, \text{kožešiny}).$$

K formálnímu odvození této věty (důsledku) ze znalostní báze \mathbf{S} ale stačí její podmnožina $\mathbf{S}' \subset \mathbf{S}$, kde

$$\mathbf{S}' = \{\text{má_řád}(\text{bb}, \text{zvířata}), \forall x \text{ má_řád}(x, \text{zvířata}) \rightarrow \neg \text{nosí}(x, \text{kožešiny})\}.$$

Omezení formulované v posledním bodě je závažné zejména z hlediska možnosti přiblížení modelovaného usuzování logikou prvního řádu skutečnému usuzování, jaké probíhá v lidských myslích. Prakticky všechny znalosti, které může člověk získat ať už během svého vývoje nebo vývoje poznatků o světě, jsou více či méně dynamického charakteru. Vše se mění, něco častěji, něco zase méně často, ale ke změně, ať už je nepatrná či zásadní, dochází. Tomu je třeba přizpůsobit jazykové prostředky formálního usuzování.



Příklad 6.2

Představme si, že dítě žije v bytě s rodiči a přijde zima. Maminka ví, že dítě již samo chodí, a že je ho tedy třeba upozornit na, že se nemá dotýkat radiátoru ústředního topení, protože je horký. Dítě tomu ale neuvěří, dokud to nezkusí. Pro dítě je totiž tento objekt určitého (například obdélníkového tvaru a bílé barvy) chladný, protože si jej zřejmě již například v létě nebo na podzim, kdy už chodilo, ohmatalo. Teď je ale zima a po doteku dítě s pláčem zjistí, že

Reprezentace znalostí

radiátor skutečně pálí. Dojde tedy k revidování jeho dosavadních znalostí a k upravení znalostní báze v jeho myšlení.

Pravidlo, podle něhož je možno usuzování dítěte zapsat by mohlo vypadat takto:

"Je-li x radiátor topení a je-li konzistentní podle všeho, co je dosud dítěti známo, předpokládat, že radiátor topení je chladný, pak radiátor topení je chladný."

Jakmile ale nastane zima, toto tvrzení neplatí, protože v domě se topí a radiátor topení tedy určitě není chladný, ale je horký. Není tedy konzistentní se situací, předpokládat, že radiátor topení je chladný. Zmíněné pravidlo tedy již není aplikovatelné. Proto je důležité definovat metodu budování teorie způsobem, který umožňuje toto nemonotónní chování.

Nově vytvořené pravidlo by pak např. vypadalo takto:

"Je-li x radiátor topení, je-li zimní období a je-li konzistentní podle všeho, co je dítěti nově známo, předpokládat, že radiátor topení je horký, pak radiátor topení je horký."

Zmíněný příklad demonstruje model skutečného myšlení nejen u dítěte, ale i v běžném životě každého z nás. Nějaký okamžitý dojem (prerekvizita) vede bezprostředně k závěru, který může být založen na předcházející zkušenosti. Tento závěr však byl učiněn na neúplné informaci. Tak se to děje i ve skutečnosti. Lze totiž usuzovat, a v praxi se to tak děje, i bez předcházejícího shromáždění všech relevantních informací.

6.1.2 Charakteristické vlastnosti monotónních formálních systémů

Souhrnně lze konstatovat, že charakteristickou vlastností tradičních formálních systémů je skutečnost, že jejich teorie, tj. množiny speciálních axiomů (znalostní báze) a jejich logických důsledků, rostou *monotónně* s množinou formulí odvozených nebo zařazených dodatečně do výchozí znalostní báze. Při monotónním odvozování přidání formule do znalostní báze nikdy neznehodnotí pravdivost jejich předcházejících důsledků. Tato vlastnost však je evidentně v rozporu s představou formální reprezentace vývoje poznání. Vlastnost monotónnosti je typická pro tradiční formální systémy logiky prvního řádu.

Znamé vlastnosti formálních systémů logiky prvního řádu lze shrnout takto ($P_1, P_2, \dots, P_n, A, Q, R$ jsou formule):

1. Reflexivnost :

$\{P_1, P_2, \dots, P_n, A\} \vdash A$,

kteřá zajišťuje možnost odvodit důsledek totožný s některým z předpokladů.

2. Monotónnost :

je-li $\{P_1, P_2, \dots, P_n\} \vdash A$, pak je i $\{P_1, P_2, \dots, P_n, R\} \vdash A$, která zajišťuje, že odvozený důsledek nebude znehodnocen další dedukcí ani při rozšíření znalostní báze.



Reprezentace znalostí

3. Tranzitivnost :

je-li $\{P_1, P_2, \dots, P_n\} \vdash R$ a $\{P_1, P_2, \dots, P_n, R\} \vdash Q$ pak je i $\{P_1, P_2, \dots, P_n\} \vdash Q$, která zajišťuje, že mezivýsledky lze využít pro další odvozování.

Všechny tradiční typy logických formálních systémů generují pouze teorie těchto uvedených vlastností.



Definice 6.1

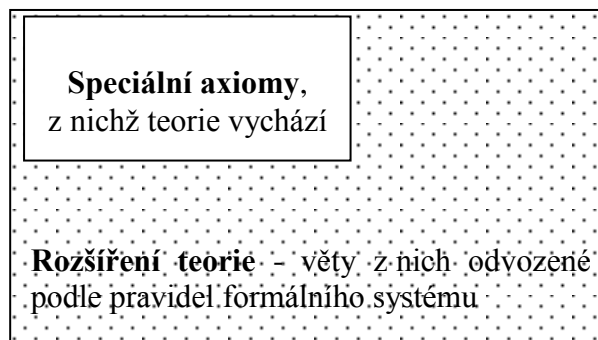
Formální systém je *monotónní*, jestliže v něm přidání nové znalosti o modelovaném světě do znalostní báze zvětší nebo alespoň nezmenší množinu odvozených důsledků.

Věta 6.2 (o monotónnosti formálního dokazování)

Označíme-li teorii $Tn(\mathbf{X})$ vybudovanou jako množinu všech vět, které jsou dokázanými důsledky dané množiny formulí (znalostní báze) \mathbf{X} (tzv. logický uzávěr množiny \mathbf{X}), pak pro monotónní dokazování platí následující vlastnosti:

1. \mathbf{X} je podmnožinou $Tn(\mathbf{X})$.
2. Je-li \mathbf{X} podmnožinou \mathbf{Y} pak $Tn(\mathbf{X})$ je podmnožinou $Tn(\mathbf{Y})$.
3. $Tn(Tn(\mathbf{X}))$ je rovno $Tn(\mathbf{X})$.
4. $Tn(\mathbf{X})$ je sjednocením množin $Tn(\mathbf{Y})$, kde \mathbf{Y} jsou konečné podmnožiny množiny \mathbf{X} .

Obrázek 6.1 znázorňuje situaci při budování teorie je v monotónních systémech na základě (konsistentní) množiny speciálních axiomů, takové že do jejího rozšíření náleží každé odvozené tvrzení (věta) ze speciálních axiomů a též další věty odvozené z již odvozených vět. Přidáváním dalších později získaných tvrzení, která se dodatečně ukázala jako pravdivá, neovlivní nijak obsah až dosud vybudované teorie. Jestliže nově dodaná pravdivá tvrzení způsobí, že se rozšířená teorie stane spornou, je možno bez kontroly konsistence vygenerovat korektním způsobem řady nesmyslů.



Obrázek 6.1

Reprezentace znalostí

6.1.3 Nemonotónnost teorií

Aby se přiblížila skutečnému myšlení, musí formální logika vzít v úvahu i myšlení s neúplnými nebo nespolehlivými informacemi a především revidovatelné usuzování. Cílem tradičního (monotónního) odvozování je bezpochyby též docílení co nejvyšší podobnosti s reálným usuzováním. Ale toto odvozování sebou nese nedostatky, které tuto shodu se skutečným *všeobecně sdíleným usuzováním* „selským rozumem“ (commonsense reasoning), zahrnujícím schopnost docílit smysluplných závěrů i s neúplnými nebo nejistými znalostmi, neumožňuje. Pak je ale třeba použít i nemonotónních prostředků dokazování. Právě o tomto typu logického usuzování a dokazování bude převážná část dalšího textu této kapitoly.

V monotónních teoriích, jak již bylo konstatováno, jsou výchozí i odvozené znalosti pevně dány a nevyvrátitelné. Při definování formálního systému je to určitě výhoda, nevýhody se ale projeví při odvozování v případě, kdy nastane nějaká změna. Je zřejmé, že pro zajištění *nemonotónnosti* (revidovatelnosti) teorie, vybudované nad znalostní bází je třeba definovat vstupní podmínky poněkud měkčeji. Nelze zadávat nevyvrátitelné vstupní znalosti a přísná odvozovací pravidla, jak tomu bývá v klasických monotónních systémech logiky prvního řádu. Nemonotónní logické systémy mají základ z velké části v monotónních logických systémech.

Definice 6.2

Teorie je *nemonotónní*, jestliže připouští, aby přidání nové znalosti o modelovaném světě do znalostní báze znehodnotilo některé její již odvozené věty.



V následující větě jsou odpovídajícím způsobem (Věta 6.2) formulovány vlastnosti nemonotónního dokazování.

Věta 6.3 (o nemonotónnosti dokazování)

Označíme-li teorii $T_n(X)$ vybudovanou jako množinu všech vět, které jsou dokázanými důsledky množiny formulí X (tzv. logický uzávěr množiny X), pak pro nemonotónní dokazování platí:

1. X je podmnožinou $T_n(X)$.
2. Když X je podmnožinou Y pak $T_n(X)$ nemusí být podmnožinou $T_n(Y)$.
3. $T_n(T_n(X))$ je rovno $T_n(X)$.
4. $T_n(X)$ je sjednocením množin $T_n(Y)$, kde Y jsou konečné podmnožiny množiny X .

Právě u druhého bodu této věty je vidět rozdíl proti monotónnímu dokazování.

6.2 Vývoj nemonotónních přístupů

První pokusy o eliminaci nevýhod tradiční monotónní logiky jsou datovány do 70. a 80. let minulého století.

Objevily se např. snahy postavit systémy nemonotónního odvozování na principu hierarchického uspořádání předpokladů z hlediska jejich relevantnosti Touretzky [viz literatura].

Tři první opravdu významná díla o nemonotónní logice, jejichž autory byli McDermott a Doyle, Reiter a McCarthy, se objevila v roce 1980. V každém z těchto případů byly prezentovány formalismy jako výsledky snah, které trvaly řadu let. Všechna tato důležitá díla byla vydána téměř ve stejném roce, proto není pochyb o jedinečnosti každého z těchto tří přístupů.

6.2.1 Dva způsoby nemonotónní modifikace logiky prvního řádu

Většina přístupů k nemonotónnímu odvozování vychází z principů tradiční logiky prvního řádu. Monotónní logika prvního řádu tvoří tedy základní rámec pro nemonotónní logiky. Existují dvě hlavní možnosti, jak doplnit tradiční predikátovou logiku, tak aby zahrnovala neurčitost umožňující nemonotónní odvozování. Těmito dvěma možnostmi jsou:

- zavedení dalších odvozovacích pravidel, kterým se vyznačuje Reiterova default logika,
- rozšíření jazyka, které je v rámci autoepistemické logiky realizováno modální operátorem L "věří se, že...".

Vedle těchto dvou hlavních směrů patří mezi základní typy nemonotónních logických systémů, které představují jisté přiblížení všeobecně sdílenému uvažování, podle řady autorů

- techniky založené na dědičnosti a specifičnosti předpokladů,
- techniky založené na předpokladu uzavřeného světa a vymezení (circumscription) - McCarthy [viz literatura],

6.2.2 Reiterova modifikace default odvozovacími pravidly



Přístup *default logiky*, která přidává k logice prvního řádu *default odvozovací pravidla*, představuje jednu ze dvou hlavních výše diskutovaných možností rozšíření jazyka logiky prvního řádu.

Myšlenku *default logiky* poprvé uvedl Ray Reiter [viz literatura] takovouto úvahou:

Představme si formalizaci logikou prvního řádu všeho toho relevantního, co víme o přiměřeně složitém světě. Protože tradiční logika prvního řádu je monotónním systémem, platí pro ni (věta 6.2): Je-li tvrzení A důsledkem množiny předpokladů U , pak je též důsledkem množiny U' takové, že platí $U \subseteq U'$. Postupně je tedy možno v monotónním systému nové znalosti o

Reprezentace znalostí

referenčním systému přidávat k předpokladům, a to i tehdy, stane-li se tím rozšířená množina nekonsistentní a odvozená teorie spornou.

Default pravidla fungují jako meta-pravidla určující, jak teorii rozšiřovat (vytvářet její extenzi) pouze pomocí těch tvrzení, kterým věříme. Obecně způsob takového rozšíření není jediný, mohou tedy existovat různá rozšíření téže teorie. Reiterova pravidla říkají něco v tom smyslu, že pokud nemáme informaci o jeho opaku, přijmeme uvažované tvrzení.

Obecně lze říci, že ve většině případů nelze o reprezentovaném světě (referenčním systému) vědět všechno, ve znalostech o něm jsou vždy mezery. Proto na nich vybudovaná teorie v logice prvního řádu bude neúplná. Úkolem defaultů, podobně jako dalších nemonotónních technik, je pomocí tyto mezery znalostních bází a teorií nad nimi vybudovaných vyplňovat. Jedním ze způsobů realizace nemonotónního odvozování je proto definování logiky, která vedle formulí standardní logiky prvního řádu používá nestandardní nemonotónní odvozovací default pravidla.

Řada dalších nemonotónních logik se vymezuje právě vůči Reiterově *default* logice.

Definice 6.3 (default pravidla)

Nechť $\alpha = \alpha_1, \dots, \alpha_m$ $\beta = \beta_1, \dots, \beta_n$ a γ jsou dobře utvořené uzavřené formule logiky 1. řádu.

Potom pravidlo

$$\frac{\alpha : \beta}{\gamma}$$

nebo též

$$(\alpha : \beta) \rightarrow \gamma,$$

„Platí-li α a je-li β konsistentní s tím, co je všeobecně považováno za platné, potom je možno usuzovat, že platí γ .“

je default pravidlo (stručně default).

$\alpha = \alpha_1, \dots, \alpha_m$ představuje *prerekvizity* pravidla, β_1, \dots, β_n jsou jeho *oprávnění*, γ je *závěr*.

Znalosti zapsané default pravidlem nad čarou se skládají ze dvou částí prerekvizit a oprávnění, oddělených dvojtečkou. První je množina formulí v logice prvního řádu a představuje dané předpoklady, které musí být stanoveny (resp. dokázány ve standardním deduktivním smyslu logiky prvního řádu) jako pravdivé pro použité pravidlo. Další část je označena jako test konzistentnosti s tím, co je všeobecně známo. Tyto výrazy musí být konsistentní se současnou znalostní bází (v rámci daného problému). To znamená, že musí být dokázáno, že nelze dokázat jejich negaci. Jestliže konsekvent je takto dokázán, je přidán do *rozšíření* dané znalostní báze.



Reprezentace znalostí



Definice 6.4 (default rozšíření)

Množina odvozených důsledků default teorie se nazývá jejím *rozšířením* (*extenzí*)

Speciální případy default pravidla:

1. $(\alpha :) \rightarrow \gamma$
znamená, že „když platí α , pak je možno usuzovat, že platí i γ “ (což není totéž jako $\alpha \rightarrow \gamma$ (platí-li α , pak nutně platí i γ)).
2. $(: \beta) \rightarrow \gamma$
znamená, že když je β konsistentní s tím, čemu se všeobecně věří, pak můžeme usuzovat, že platí γ .
3. **normální** default pravidlo $(\alpha : \gamma) \rightarrow \gamma$
znamená, že „platí-li předpoklady a závěr je konsistentní s tím, čemu se všeobecně věří, pak lze odvodit, že závěr platí.“



Příklad 6.3

a) Předpokládá se, že každý pták za normálních okolností může létat, což lze zapsat normálním default pravidlem

$$\frac{\text{pták}(x) : \text{létá}(x)}{\text{létá}(x)},$$

které vyjadřuje skutečnost, že

„Je-li x pták a je-li tvrzení „ x může létat“ konsistentní s tím, co je všeobecně známo, pak x může létat.“

b) Předpokládá se, že každý univerzitní profesor, pokud nenastanou zvláštní okolnosti, je též vyučujícím, což lze zapsat normálním default pravidlem

$$\frac{\text{profesor}(x) : \text{vyučuje}(x)}{\text{vyučuje}(x)}$$

Zvláštní okolností může být např. skutečnost, že se profesor stane rektorem a pracovní vytíženost v čele univerzity mu nedovolí paralelní pedagogické působení.



Příklad 6.4

V každém z těchto příkladů default pravidlo $[\alpha:\beta/\gamma]$ čteme jako „Je-li α true, přičemž je možné věřit β , pak závěr γ je true.“

a) Typické děti mají dva rodiče. To lze vyjádřit pomocí defaultů takto:

$$\frac{\text{dítě}(x) : \text{má_dva_rodiče}(x)}{\text{má_dva_rodiče}(x)}$$

To lze číst takto: „ Pokud je rozumné věřit, že dítě má dva rodiče, pak tomu věřme.“

b) odvozování bez rizika

Reprezentace znalostí

Předpokládáme, že obžalovaný je nevinný, pokud se nedokáže opak.

$$\frac{\text{obžalovaný}(x) : \text{nevinný}(x)}{\text{nevinný}(x)}$$

c) nejlepší odpověď

Předpokládáme, že nejlepší nalezené řešení je tím nejlepším řešením.

$$\frac{\text{předpokládané_nejlepší_řešení}(x) : \text{nejlepší_řešení}(x)}{\text{nejlepší_řešení}(x)}$$

d) pravidlo Sherlocka Holmesa

$$\frac{\text{jediné_zbývající_řešení}(x) : \text{možné_řešení}(x)}{\text{řešení}(x)}$$

e) komunikační konvence

Jestliže vlak z Londýna do Glasgow v 10.30 není v jízdním řádu, pak je bezpečné předpokládat, že žádný vlak z Londýna do Glasgow v 10.30 nejede.

$$\frac{\neg \text{je v jízdním řádu}(x) : \neg \text{odjíždí v}(x)}{\neg \text{odjíždí v}(x)}$$

Tento typ negace je znám z Prologu, který nepoužívá klasickou negaci, ale negaci selháním důkazu. tj., jestliže interpreter Prologu není schopen dokázat, že klauzule je důsledkem programu, pak ji považujeme za nepravdivou.

6.2.3 Modifikace rozšířením jazyka v autoepistemické logice

Autoepistemická logika rozšiřuje logiku prvního řádu o modální operátor L. Poprvé byla autoepistemická logika uvedena McDermotem a Doylem [viz literatura] jako modální logika používající modální operátor L, přičemž $L\alpha$ vyjadřuje „věří se, že platí α “, resp. „je známo, že platí α “. Jazyk autoepistemické logiky byl navržen k účelu formalizace způsobu, jak rozumem vybavený agent může konstruovat množinu tvrzení, kterým věří. Autoři rozlišují ne-modální formule k reprezentaci skutečností týkajících se modelovaného světa a vlastní modální formule k reprezentaci způsobů nemonotónního odvozování.

Formuli autoepistemické logiky

$$\neg L\neg\alpha \rightarrow \beta$$

je třeba číst

„Jestliže nelze věřit, že platí $\neg\alpha$, potom platí β “,

resp.

„V případě absence informace vyvracející α odvod β “.

V daném jazyce autoepistemické logiky definuje Moore [viz literatura] extenzi E teorie T tak, že platí:

Reprezentace znalostí

$$E = \text{Cn}(\mathbf{T} \cup \{L\alpha \mid \alpha \in E\} \cup \{\neg L\alpha \mid \alpha \notin E\})$$

Cn je *operátor důsledku* logiky prvního řádu, který pracuje s formulemi $L\alpha$ jako s proměnnými. Moore odůvodňuje tuto rovnici tím, že extenze by měla sestávat právě z těch formulí, které lze odvodit z \mathbf{T} a z formulí odvozených na základě pozitivních a negativních zkušenosti toho, čemu agent věří. Mooreova extenze teorie \mathbf{T} má vlastnosti, které ji činí vhodnou pro modelování množiny tvrzení, kterým rozumný agent věří.

Pro množiny tvrzení, kterým se věří, platí axiomy:

- **B1:** $\text{Cn}(E) \subseteq E$ (axióm racionality)
- **B2:** Platí-li $\alpha \in E$, potom platí $L\alpha \in E$ (uzavřenost vzhledem k pozitivní introspekci)
- **B3:** Platí-li $\alpha \notin E$, potom platí $\neg L\alpha \in E$ (uzavřenost vzhledem k negativní introspekci).

I když motivace jsou rozdílné, sleduje autoepistemická logika podobné odvozovací vzory jako default logika.

Např. univerzitní příklad lze popsat v modálním jazyce jedinou teorií.

$$\mathbf{T} = \{ \text{profesor}(x), \text{rektor}(x) \rightarrow \neg \text{vyučuje}(x), L \text{profesor}(x) \& \neg L \neg \text{vyučuje}(x) \rightarrow \text{vyučuje}(x) \}$$

Tato teorie má jediné rozšíření, které pro případ instance **profesor(J)** obsahuje **vyučuje(J)**. Přidáním **rektor(J)** bude teorie mít rovněž jediné rozšíření, ale bude obsahovat tvrzení $\neg \text{vyučuje}(J)$

Autoepistemická logika vychází ze *stabilní množiny*, tj množiny obecných tvrzení, kterým racionální agent věří, a z ní vytvořeného rozšíření o odvozené věty.

Základními vlastnostmi stabilních množin jsou

- konsistence a
- *negativní introspekce*: nepatří-li sentence P do stabilní množiny, potom tam patří sentence $\neg L P$ ("nevěří se, že platí P").

Pravidlo autoepistemické logiky odpovídající default pravidlu z předcházejícího odstavce má tvar

$$(L(\text{pták}(x)) \& \neg L(\neg \text{létá}(x))) \rightarrow \text{létá}(x).$$

6.2.4 Další možnosti modifikace rozšířením jazyka

Přístup označovaný v literatuře jako *circumscription* (vymezení) formalizuje pojmy, jakými jsou např. normálnost a typičnost pomocí

- *standardního predikátu normální*
 $\forall x (\text{pták}(x) \& \text{normální}(x) \rightarrow \text{létá}(x)),$
- nového *modálně logického operátoru typicky*
 $\forall x (\text{typicky}(\text{pták}(x)) \rightarrow \text{létá}(x)).$

Reprezentace znalostí

V literatuře se vyskytují ještě další možnosti, které jsou možné, ale méně používané, a ne tak účinné jako dvě výše zmíněné. Jsou to např. tato řešení:

- využití specifikujících predikátů k vyjmenování případů, kdy určité tvrzení neplatí,
- doplnění logiky prvního řádu o standardní predikát **unless**.

Příklady využití specifikujících predikátů k vyjmenování případů, kdy tvrzení neplatí, jsou

$$\forall x (\text{pták}(x) \ \& \ \neg(\text{tučňák}(x) \vee \text{poraněný}(x) \vee \dots) \rightarrow \text{létá}(x)),$$

$$\forall x (\text{člověk}(x) \ \& \ \neg(\text{zraněný}(x) \vee \text{kojenec}(x) \vee \dots) \rightarrow \text{chodí}(x))$$

První zmíněné řešení, tedy využití *specifikujících predikátů* k vyjmenování případů, kdy určité tvrzení neplatí, nese už ve svém popisu jeden problém, který by mohl být za určitých okolností mohl problematicky řešitelný. Vyžaduje totiž nutnost přepracování podmínky po zjištění každé nové výjimky.

Další v literatuře uváděnou možností je doplnění logiky prvního řádu o standardní predikát **unless**. Predikát **unless**(P) je **true** pro formulí P , právě tehdy, nelze-li P odvodit ve z dané množiny předpokladů **A**. Tento predikát lze, jak bude blíže vysvětleno dále, použít v definicích odvozovacích pravidel, která umožňují odvodit nové tvrzení za podmínky, že jiné tvrzení odvodit nelze.

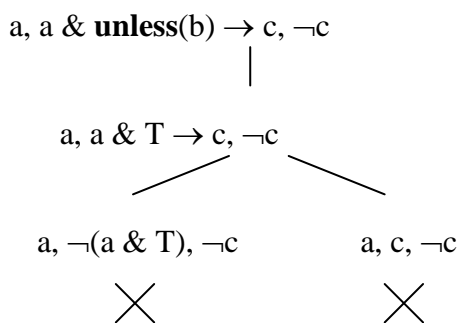
Deduktivní systém výrokové logiky doplněný pravidly obsahujícími predikát **unless** je sice nemonotónní, ale nemá požadované vlastnosti. Může se totiž stát, jak bude ukázáno na příkladech, že P a **unless**(P) bude odvozeno zároveň.

Příklad 6.5

Nechť $\mathbf{A} = \{a, a \ \& \ \text{unless}(b) \rightarrow c, a \ \& \ \text{unless}(c) \rightarrow b\}$, kde a, b, c , jsou výrokové proměnné, je množina předpokladů. Následující dva tablové důkazy ukazují, že z \mathbf{A} lze odvodit buď b nebo c , nikoliv však zároveň.

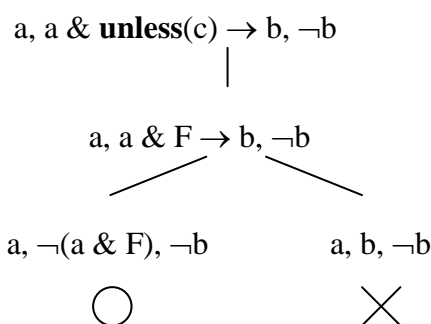


Tablový důkaz c vycházející z předpokladu, že nelze dokázat b , tj. pro případ, kdy **unless**(b) je **true**:



Protože c dokázat lze, je hodnota predikátu **unless**(c) **false**. Potom ale, jak ukazuje následující tablo, nelze dokázat b .

Reprezentace znalostí



Směry, které používají např. predikát **normální**, resp. **typický** v syntaxi

$$\forall x \ \mathbf{normální}(\mathbf{pták}(x) \rightarrow \mathbf{létá}(x))$$

$$\forall x \ \mathbf{typický}(\mathbf{pták}(x) \rightarrow \mathbf{létá}(x)),$$

posouvají formální reprezentaci do oblasti logiky druhého řádu.

Následující odstavce jsou věnovány základním směrům praktické realizace nemonotónního odvozování.

6.2.5 Nemonotónnost založená na dědičnosti a specifičnosti předpokladů

Na nemonotónnost logického systému lze nahlížet jako na způsob rozšíření systému tradiční monotónní logiky o daný kontext. V rámci určitého kontextu lze pak např. rozlišovat, jakou závažnost mají výchozí předpoklady dedukce. V tomto smyslu odvozuje možnost nemonotónní dedukce z principu dědičnosti. Následující příklad, často uváděný v literatuře, ilustruje rozdíl mezi monotónním a nemonotónním přístupem k formálním odvozováním, z nichž druhý bere v úvahu kontext, v němž dané předpoklady lze z hlediska relevantnosti hierarchicky uspořádat.



Příklad 6.6

Z předpokladů P_1 "Ptáci létají.", P_2 "Tweety je pták.", jak ukazuje následující důkaz, v hilbertovském axiomatickém systému H_1 (viz např. [64]) lze pomocí pravidla Modus ponens (MP) odvodit závěr "Tweety létá.". Po přidání dalších tří předpokladů P_3 "Tučňáci jsou ptáci.", P_4 "Tučňáci nelétají." a P_5 "Tweety je tučňák." lze ale odvodit závěr "Tweety nelétá.".

Důkaz v H_1 :

- | | |
|---|----------------|
| 1. $P_1 \vdash \forall x (\mathbf{pták}(x) \rightarrow \mathbf{létá}(x))$ | předpoklad |
| 2. $\vdash \forall x (\mathbf{pták}(x) \rightarrow \mathbf{létá}(x))$ | axiom A4 H_1 |
| 3. $P_1 \vdash (\mathbf{pták}(\mathbf{Tweety}) \rightarrow \mathbf{létá}(\mathbf{Tweety}))$ | MP na 1,2 |
| 4. $P_2 \vdash \mathbf{pták}(\mathbf{Tweety})$ | předpoklad |
| 5. $P_1, P_2 \vdash \mathbf{létá}(\mathbf{Tweety})$ | MP na 3,4 |

Reprezentace znalostí

6. $P_3 \vdash \forall x (\text{tučňák}(x) \rightarrow \text{pták}(x))$ předpoklad
7. $\vdash \forall x (\text{tučňák}(x) \rightarrow \text{pták}(x)) \rightarrow (\text{tučňák}(\text{Tweety}) \rightarrow \text{pták}(\text{Tweety}))$
axiom A4 H_1
8. $P_3 \vdash (\text{tučňák}(\text{Tweety}) \rightarrow \text{pták}(\text{Tweety}))$ MP na 6,7
9. $P_4 \vdash \forall x (\text{tučňák}(x) \rightarrow \neg \text{létá}(x))$ předpoklad
10. $\vdash \forall x (\text{tučňák}(x) \rightarrow \neg \text{létá}(x)) \rightarrow (\text{tučňák}(\text{Tweety}) \rightarrow \neg \text{létá}(\text{Tweety}))$
axiom A4 H_1
11. $P_4 \vdash (\text{tučňák}(\text{Tweety}) \rightarrow \neg \text{létá}(\text{Tweety}))$ MP na 9,10
12. $P_5 \vdash \text{tučňák}(\text{Tweety})$ předpoklad
13. $P_4, P_5 \vdash \neg \text{létá}(\text{Tweety})$ MP na 11,12

V monotónních systémech, mezi něž H_1 patří, mají všechny předpoklady vstupující do procesu odvozování, stejnou závažnost, tedy nelze rozhodnout, který z odvozených sporných závěrů $\text{létá}(\text{Tweety})$, $\neg \text{létá}(\text{Tweety})$, je přijatelnější. Z hlediska kontextu se ale předpoklad P_5 , tvrdící že Tweety je tučňák, jeví jako relevantnější než ten, který tvrdí, že Tweety je pták, neboť všichni tučňáci jsou ptáci, nikoliv naopak. Předpoklad P_4 o tučňácích je více specifický než předpoklad P_1 o ptácích, neboť P_1 se týká ptáků, P_4 jejich podmnožiny – tučňáků. Tady lze spatřovat měřítko relevantnosti argumentů. Specifičtější tvrzení je více relevantní, proto „Tweety nelétá“. Je-li cílem využívat především ty nejrelevantnější předpoklady, je namístě odvodit, že Tweety nelétá.

Diference ve specifičnosti zde určuje relevantnost předpokladů. Předpoklad P_4 , který vyjadřuje všeobecně sdílenou znalost, je zde klíčový. Předpoklad P_1 naproti tomu je zde pravidlem s výjimkami, které na počátku nebyly známy. Tak ale probíhá skutečné myšlení – často činíme závěry, aniž bychom čekali na příchod dalších informací. Přijímáme v daném okamžiku předpoklad uzavřeného světa.

Další známý příklad z literatury ukazuje, že někdy nelze takto uvažovat, neboť specifičnosti předpokladů nelze vždy uspořádat.

Příklad 6.7

V tomto případě jsou k dispozici předpoklady: P_1 "Kvakeři mají holubí povahu.", P_2 "Republikáni mají jestřábí povahu.", P_3 "Dick je republikán.", P_4 "Dick je kvaker." P_5 "Jestřáb není holub."



Důkaz, v hilbertovském axiomatickém systému H_1 :

1. $P_1 \vdash \forall x (\text{kvaker}(x) \rightarrow \text{holub}(x))$
2. $P_2 \vdash \forall x (\text{republikán}(x) \rightarrow \text{jestřáb}(x))$
3. $P_3 \vdash \text{republikán}(\text{Dick})$
4. $P_4 \vdash \text{kvaker}(\text{Dick})$
5. $P_5 \vdash \forall x (\text{jestřáb}(x) \rightarrow \neg \text{holub}(x))$
6. $\vdash \forall x (\text{kvaker}(x) \rightarrow \text{holub}(x)) \rightarrow (\text{kvaker}(\text{Dick}) \rightarrow \text{holub}(\text{Dick}))$
axiom A4 H_1
7. $P_1 \vdash (\text{kvaker}(\text{Dick}) \rightarrow \text{holub}(\text{Dick}))$ MP na 1,6
8. $P_1, P_4 \vdash \text{holub}(\text{Dick})$ MP na 4,7

Reprezentace znalostí

9. $\vdash \forall x (\text{republikán}(x) \rightarrow \text{jestřáb}(x)) \rightarrow (\text{republikán}(\text{Dick}) \rightarrow \text{jestřáb}(\text{Dick}))$ axiom A4
10. $P_2 \vdash (\text{republikán}(\text{Dick}) \rightarrow \text{jestřáb}(\text{Dick}))$ MP na 2,9
11. $P_2, P_3 \vdash \text{jestřáb}(\text{Dick})$ MP na 3,10
12. $\vdash \forall x (\text{jestřáb}(x) \rightarrow \neg \text{holub}(x)) \rightarrow (\text{jestřáb}(\text{Dick}) \rightarrow \neg \text{holub}(\text{Dick}))$ axiom A4
13. $P_5 \vdash (\text{jestřáb}(\text{Dick}) \rightarrow \neg \text{holub}(\text{Dick}))$ MP na 5,12
14. $P_2, P_3, P_5 \vdash \neg \text{holub}(\text{Dick})$ MP na 11,13

Důkaz, v hilbertovském axiomatickém systému H_1 zde vede ke sporným závěrům $\text{holub}(\text{Dick})$, $\neg \text{holub}(\text{Dick})$. Jde o příklad, kdy specifičnosti nelze uspořádat. Problémem je skutečnost, že chybí informace o hierarchii předpokladů, jak tomu bylo například v předchozím příkladu. Nelze tedy přijmout odvozený závěr, že Dick má holubí povahu, ani že Dick má jestřábí povahu.

To znamená, že posuzování relevantnosti předpokladů problém nemonotónního usuzování obecně neřeší. Formální logické systémy nemonotónního usuzování se proto, jak bude ukázáno dále, zaměřují na možnosti přidání dalších pravidel nebo operátorů schopných vzít v úvahu konsistenci odvozené znalosti s tím, co je v dané oblasti všeobecně známo.

6.3 Formalizace teorie v nemonotónních systémech

6.3.1 Význam konsistence v nemonotónních systémech

Deduktivní systém formalizující nemonotónní uvažování musí být schopen odvodit (dokázat) přijatelné logické důsledky z konsistentní množiny předpokladů. Nekonsistentní množiny předpokladů nemá smysl brát v úvahu, neboť z těch je možno odvodit cokoli. Dokázané důsledky ale musí být též s množinou předpokladů konsistentní, tj. z hlediska sémantiky platné alespoň v jednom z jejích modelů.

Formulace role konzistence v nemonotónních systémech mívá z pravidla vágní podobu, např.: Je-li x člověk a je-li tvrzení „ x může chodit“ konsistentní s tím, co je známo, pak x může chodit. Pro formalizaci odvozování na bázi konzistence je však potřeba jejího přesnějšího formálního vyjádření.

6.3.2 Konsistence z formálního hlediska

Dokazatelnost v nemonotónním formálním systému se zpravidla označuje symbolem \vdash na rozdíl od dokazatelnosti v monotónních systémech používající symbolu \vdash .

Problém formálního odvozování v nemonotónních logických systémech v podstatě sestává ze dvou sub-problémů, a to

Reprezentace znalostí

- vytvoření formálních prostředků, tj. nových pravidel, resp. nových operátorů pro jejich realizaci a
- formální analýzy opodstatnění (oprávnění) nemonotónních závěrů vzhledem k daným předpokladům.

Druhý z nich bývá zpravidla formalizován jako problém *konsistence* odvozeného závěru s danými předpoklady, resp. s tím, co je všeobecně známo.

Pro všechny typy nemonotónních logických systémů je charakteristické, že jsou postaveny na pojmu *konsistence* a jeho formalizaci.

Pojem *konsistence* množiny formulí bývá v logice prvního řádu zpravidla definován z *hlediska sémantického*, a to na základě existence modelu této množiny formulí. Z *hlediska formálního* platí následující věta:

Věta 6.4

Formule β je konsistentní s množinou předpokladů S , jestliže z množiny S nelze dokázat $\neg\beta$, tj. $S \not\vdash \neg\beta$.

Problém konsistence je v logice prvního řádu obecně nerozhodnutelný, ve výrokové logice sice rozhodnutelný, ale při budování nemonotónních teorií v řadě případů nerealizovatelný. Pro omezené použití však byla vyvinuta řada efektivních systémů

Zásadní význam formalizace konsistence znalostních bází v nemonotónních systémech lze ilustrovat na řadě příkladů. Jedním z nich je následující příklad ve výrokové default logice.

Příklad 6.8

V poslední době jsou nejen obyvatelé České republiky sužováni povodněmi. Takže se určitě může stát, že nějaká rodina Nováků přijede z delší dovolené a zjistí, že mají ve sklepech vodu. Už kvůli pojištění musí tento „problém“ řešit.



Znalostní bázi uvedeného příkladu tvoří tato tvrzení:

- Právě tehdy když byly povodně, byl sklep rodiny Nováků zatopen povodněmi.
- V uvedené době, kdy byli Novákovi na dovolené, nepršelo.

Z této znalostní báze je tedy možno usoudit, že Novákovi nebyli zatopeni povodněmi. Je proto potřeba zjišťovat, jak došlo k zatopení jejich sklepa.

Označení výroků:

Sklep Nováků byl zatopen povodněmi.	p
V uvedené době byly povodně.	a
V dané době pršelo.	r

Reprezentace znalostí

Formule znalostní báze:

Sklep Nováků byl zatopen povodněmi, právě tehdy když byly povodně.

$$p \leftrightarrow a$$

V dané době nepršelo.

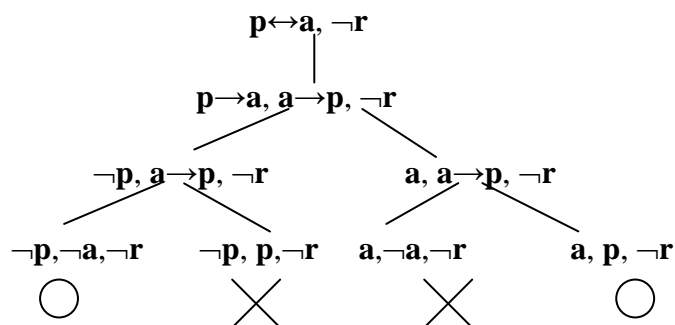
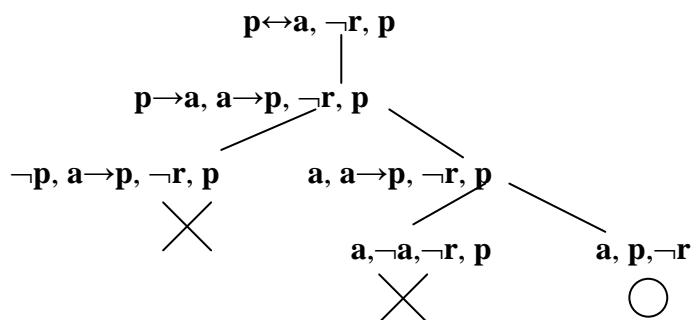
$$\neg r$$

Jde tedy o dedukci

$$p \leftrightarrow a, \neg r \models \neg p,$$

která říká, že za daných předpokladů Novákovi zatopeni povodněmi nebyli.

Příklad ukazuje neadekvátnost klasického monotónního odvozování a naopak vysoce účinný mechanismus default logiky. Monotónně, zde nepřímým tablovým důkazem, nelze totiž odvodit, že předpokládaný závěr „Novákovi povodněmi zatopeni nebyli“, je logickým důsledkem množiny uvedených předpokladů, neboť neexistuje možnost konstrukce uzavřeného sémantického tabla z dané množiny formulí v jeho kořeni. Sémanticky je to zřejmé ze skutečnosti, že množina předpokladů má, jak ukazuje přímý tablový důkaz její splnitelnosti, viz dole, dva modely $\neg a, \neg p, \neg r$ a $a, p, \neg r$, ale závěr $\neg p$ je splněn pouze v prvním z nich.



Při aplikaci *normálního default pravidla* jde o důkaz podle default pravidla

$$\underline{p \leftrightarrow a, \neg r} : \neg p$$

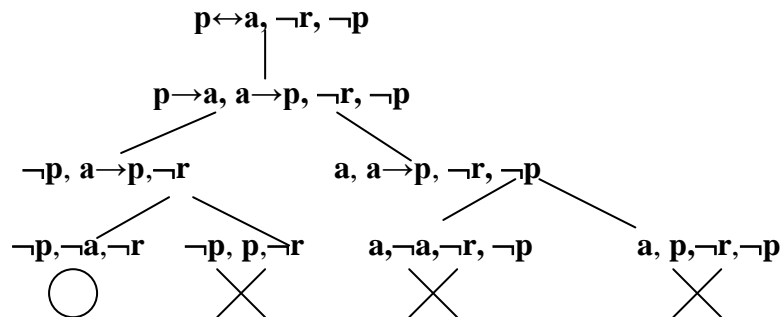
$$\neg p$$

které říká, že při platnosti předpokladů $p \leftrightarrow a, \neg r$ a za podmínky konsistence $\neg p$ s těmito předpoklady lze odvodit závěr $\neg p$.

Reprezentace znalostí

Stačí tedy dokázat konsistenci $\neg p$ s množinou formulí $\{p \leftrightarrow a, \neg r\}$, což lze uskutečnit např. nepřímým tablovým důkazem, že z daných předpokladů nelze dokázat p (nelze sestrojít uzavřené sémantické tablo).

Formule $\neg p$ potom patří do normálního default rozšíření teorie.



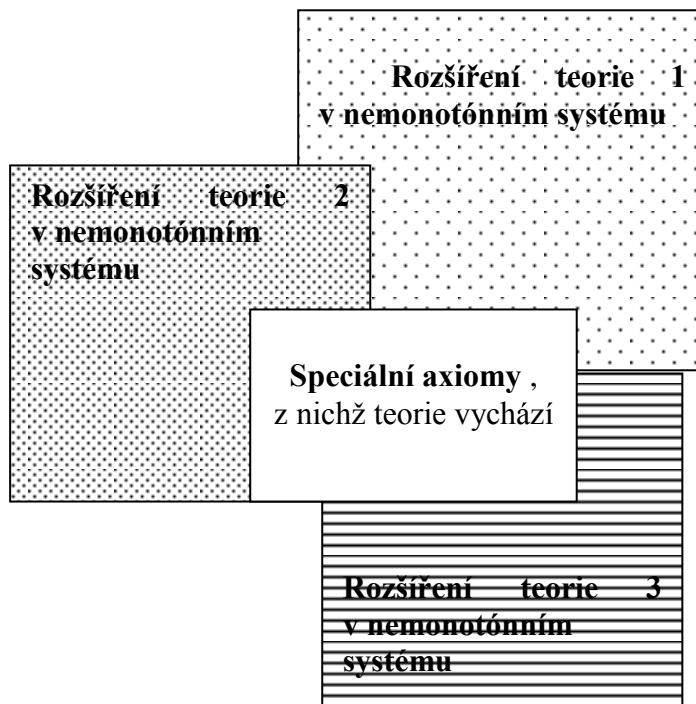
6.3.3 Budování teorií v nemonotónních systémech

Budování teorií v nemonotónních systémech je sice rizikovější než budování teorií v systémech monotónních, ale pokud je systém dobře navržen, tak nemůžeme dojít k odvození sporných důsledků, jak tomu bývá v klasické logice v důsledku rozšíření znalostní báze o další nová tvrzení.

Nemonotónní systémy berou v úvahu skutečnost, že do základní množiny speciálních axiomů teorie mohou náležet i tvrzení, která jsou důsledky empirických generalizací. V případě default logiky jde o důsledky odvozené pomocí default pravidel. Tato provizorní tvrzení a v důsledku i věty z nich odvozené mohou mít rizikovou existenci. Jednou vybudované rozšíření teorie tak nemusí mít věčnou platnost. Dodání dalších opravných tvrzení, získaných o zmíněných předmětech empirického zkoumání, může způsobit znehodnocení celého nebo části původního rozšíření teorie a vybudování nového rozšíření na základě opravené množiny poznatků.

Rozšíření teorie vždy respektuje speciální axiomy (stejně jako u systémů monotónních), ale toto rozšíření teorie se může měnit. Toto je právě podstatný a rozhodující rozdíl v budování těchto teorií oproti teoriím monotónním. Znamená to tedy, že po doplnění množiny speciálních axiomů o nějakou empirickou generalizaci se může změnit i rozšíření teorie.

Reprezentace znalostí



Obrázek 6.2

V reprezentaci znalostí se formální logika prakticky omezuje na relaci důsledku, ať už se jedná o její monotónní nebo nemonotónní variantu. Je třeba zavést speciální formální prostředky, tj. jazyk a odvozovací pravidla, pro případy, kdy jsou odvozené důsledky bezpečné, ale též pro případy, kdy tomu tak není. Jde o dva druhy tvrzení, které způsobují či vyvolávají nemonotónnost:

- empirická generalizace (speciální axiomy s možnými výjimkami)
- provizorně odvozené věty (věty odvozené z těchto empirických generalizací)

Generalizaci je třeba pojímat tak, aby byla informačně stabilní v tom smyslu, že ji odvozená rozšíření neovlivní. Jediné, co může generalizaci postihnout, je omezení její aplikovatelnosti, tedy užší specifikace, doprovázená eventuálními novými specifickými pravidly.

6.3.4 Formalizace default teorie

Default teorie je v podstatě pokusem formulovat zachycení myšlenky, že určitá sada domněnek může být spolu s jistými závěry tolerována, aniž by byly tyto závěry logickými důsledky daných předpokladů. Právě toto nedovoluje klasická logika, kde každý závěr musí logicky vyplývat z předpokladů v teorii uvedených. Jde o velice silnou zbraň nemonotónních logik vůbec, především pak tedy default logiky. Default pravidlo se využívá jen tehdy, když závěr nemůže být deduktivně odvozený pouze z existujících předpokladů. A závěr může být odvozený v tomto tvaru jen v tom případě, jestliže existuje default pravidlo, které podporuje toto odvození.

Reprezentace znalostí

Definice 6.5 (default teorie)

Default teorie $\Delta = (\mathbf{T}, \mathbf{D})$ je dvojice množin, kde \mathbf{T} je množinou uzavřených formulí logiky prvního řádu a \mathbf{D} je množina default pravidel.

Default teorie $\Delta = (\mathbf{T}, \mathbf{D})$ je normální, jsou-li všechna její default pravidla z \mathbf{D} normální.

Zde \mathbf{T} má reprezentovat znalosti (které jsou obecně neúplné), \mathbf{D} má poskytovat meta-pravidla, která vyplňují mezery ve znalostech.

Nechť $\Delta = (\mathbf{T}, \mathbf{D})$, \mathbf{S} je teorie logiky prvního řádu, uzavřená vzhledem k logickému důsledku. Δ pak může sloužit k revizi \mathbf{S} . Revidovaná teorie by měla obsahovat \mathbf{T} , navíc by měla být uzavřena vzhledem k odvozeným logickým důsledkům logiky prvního řádu a vzhledem k odvozeným default důsledkům, které s prvními nejsou v rozporu.

Tento proces revize teorie lze formalizovat pomocí operátoru Γ_{Δ} takového, že pro každou množinu formulí S (ne nutně uzavřenou vůči logickému důsledku) je $\Gamma_{\Delta}(S)$ definována jako nejmenší množina U formulí logiky prvního řádu splňujících tyto podmínky:

1. U je uzavřena vzhledem k FOL dokazatelnosti
2. $\mathbf{T} \subseteq U$
3. Pro každý default $d \in \mathbf{D}$ platí, jestliže $p(d) \subseteq U$ a pro každé $\beta \in j(d)$, $S \not\vdash \neg\beta$, potom $c(d) \in U$.

Pevné body operátoru Γ_{Δ} reprezentují množiny tvrzení, kterým se věří, které jsou stabilní vzhledem k Δ - ty už nemohou být odstraněny v rámci revize. Reiter je nazývá extenzí.

Definice 6.6

Nechť Δ je default teorie. FOL teorie S je extenzí Δ , jestliže $S = \Gamma_{\Delta}(S)$.

6.3.5 Požadované vlastnosti nemonotónních teorií

Revidovatelnost

V nemonotónních formálních systémech musí být možné v některých případech stáhnout zpět již odvozený důsledek. Jde o hlavní výhodu nemonotónních systémů oproti systémům monotónním. K tomu je potřeba revizních odvozovacích pravidel, tj. pravidel dynamického charakteru, která umožní ještě před odvozením verifikovat, že tvrzení, které má být odvozeno, je konsistentní se vším, co již předtím bylo v systému z premis odvozeno. Tato verifikace se může dynamicky měnit s narůstající množinou formulí již v systému odvozených. Všechny systémy založené na empirické generalizaci nebo víře jsou náchylné k nekonzistenci.

Reprezentace znalostí

Každý formální způsob nemonotónního odvozování musí obsahovat postup, jak naložit s nekonzistencí – jak ji rozpoznat, jak ji vyřešit a jak rozlišit správné odvození v rámci nekonzistentního systému od nesprávného. Většinou jde o nelehký úkol, ale výsledný systém je o to použitelnější a mnohem kvalitnější než systém definovaný v klasické monotónní logice bez použití empirických generalizací.



Regulárnost

Rozšíření teorie, které je vybudováno tak, aby zůstalo konsistentní i po přidání všech odvozených závěrů, se nazývá regulární. Požadavek regulárnosti je třeba ošetřit zejména při automatizovaném dokazování. Při aplikaci každého pravidla se zajištěním regularity je třeba kontrolovat konsistenci s celým rozšířením znalostní báze, protože nekonzistentní systém není, stejně jako v logice monotónní, v ničem prospěšný. Naopak, dochází ke zbytečným nesrovnalostem a zmatkům, které je třeba samozřejmě předcházet.



Založenost

Požadavek, aby každá formule z rozšíření byla odvoditelná na základě skutečností tvrzených ve výchozí množině speciálních axiomů, se nazývá požadavkem založenosti (anglicky groundedness).



Saturovanost

Před formální specifikací rozšíření je třeba si ještě položit požadavek, aby všechna aplikovatelná pravidla byla aplikována, tj. aby rozšíření bylo saturováno. Tento požadavek či vlastnost není povinná, ale má jistě svůj smysl a měla by být součástí kvalitního systému. Nač by nám také byla v systému pravidla, která nejsou používána, nemají tedy vůbec pro daný problém smysl.

6.3.6 Reiterova default-negace

Jak již bylo konstatováno, konzistenci formule A s množinou formulí S lze v logice prvního řádu zjišťovat na základě dvou možností:

- sémantické hledisko (prověření existence modelu množiny formulí $A \cup S$),
- formální hledisko (Věta 6.4)

Právě tento druhý způsob sledování a zajišťování konzistence je základem myšlenky default-negace podle Reitera.



Definice 6.7

Default-negace teorie je dána dvojicí, která sestává z množiny tvrzení T a množiny pravidel D . Každé pravidlo v D je specifikováno jako trojice (<pozitivní antecedenty>, <negativní antecedenty>, <konsekvent>).

Reprezentace znalostí

Default - negace pravidlo má obecně tvar:

$$\frac{\alpha_1 \& \dots \& \alpha_m : \neg\beta_1 \& \dots \& \neg\beta_n}{\gamma}$$

Podmínky pravidla $\alpha_1 \& \dots \& \alpha_m$, $\neg\beta_1 \& \dots \& \neg\beta_n$ jsou splněny vzhledem k množině předpokladů \mathbf{E} (současnému stavu rozšíření) a lze odvodit závěr γ , jestliže všechny pozitivní antecedenty jsou přítomny v \mathbf{E} a žádný negativní antecedent není přítomen v \mathbf{E} . V dané teorii nelze tedy dokázat opak těchto tvrzení.

Příklad 6.9

Pravidlo pro lékařskou praxi neboli pravidlo běžného rozhodování praktického lékaře:

„Trpí-li pacient bolestí a je-li skutečností, že aspirin uvolňuje bolest, a pacient není alergický na aspirin, pak odvod' závěr, že je možno předepsat pacientovi aspirin.“

Toto pravidlo lze pomocí default - negace vyjádřit takto:

$$\frac{\text{pacient_trpí_bolestí, aspirin_uvolňuje_bolest} : \neg\text{pacient_alergický_na_aspirin}}{\text{předepsat_pacientovi_aspirin}}$$



6.3.7 Extenze default negace podle Reitera

Reiter předložil následující definici rozšíření default-negace teorie. $\text{Th}_L(\mathbf{E})$ v ní označuje množinu všech důsledků dobře utvořených formulí logiky \mathbf{L} . V každém kroku vytváření posloupnosti \mathbf{E}_i se k teorii přiřazují všechny důsledky v klasickém smyslu a též všechny důsledky aplikovatelných default pravidel. Definice proto neobsahuje saturační podmínku, která říká, aby všechna aplikovatelná pravidla byla aplikována, protože tuto podmínku zajistí samotný algoritmus vytváření této posloupnosti.

Definice 6.8

Nechť $\Delta = (\mathbf{T}, \mathbf{D})$ je uzavřená default teorie. Potom množina uzavřených dobře utvořených formulí \mathbf{E} je Reiterovou extenzí teorie Δ , právě když

$$1. \mathbf{E} = \bigcup_{i=0}^{\infty} \mathbf{E}_i$$

kde

$$2. \mathbf{E}_0 = \mathbf{T} \text{ a}$$

Pro každé i , $i \geq 1$

$$\mathbf{E}_{i+1} = \text{Th}_L(\mathbf{E}_i) \cup \{\gamma \mid [\alpha:\beta_1, \dots, \beta_n/\gamma] \in \mathbf{D} \& \alpha \subseteq \mathbf{E}_i \& \neg\beta_1, \dots, \neg\beta_n \notin \mathbf{E}_i\},$$



Reprezentace znalostí

kde $\text{Th}_L(\mathbf{E}_i)$ je vygenerovanou teorií do i -tého kroku, γ představuje důsledek odvozený default pravidlem v rámci kroku $i+1$.

V této definici je extenze definována tak, že je saturována a založena v \mathbf{T} . Extenze je též regulární. To znamená, že do extenze jsou zařazeny pouze ty důsledky, jsou-li jejich oprávnění konsistentní s výslednou extenzí ($\neg\beta_1, \dots, \neg\beta_n \notin \mathbf{E}$). Tento způsob kontroly konsistence však činí tuto definici nekonstruktivní. Kdyby byl totiž index n příliš vysoký, tak by toto ošetření konzistence bylo velice pracné.

6.4 Default důkazy jako argumenty

Důležitou vlastností default závěrů je to, že nejsou kategorické. Default pravidel se používá ke konstrukci *podmíněných důkazů*, tj. takových, které mohou být na základě dodatečné informace zpochybněny. Takové důkazy mají spíše charakter *argumentů*.

Default důkazy lze, jak ukazují následující příklady, považovat za vyvratitelné argumenty.



Příklad 6.10

Doktor F očekává příjezd Profesora L do Londýna na plánované setkání, a to jako obvykle, určitým vlakem V. Jeho sekretářka ale zjistí, že londýnská nádraží budou v předpokládanou dobu příjezdu vlaku V mimo provoz, proto argumentuje, že L přijede na setkání se zpožděním. F ale uvažuje takto : Jestliže L stihne dřívější vlak, nemusí přijet pozdě. Tím zpochybní sekretářčin argument. Alternativně L může přijet autobusem, čímž se zpochybní předpoklad jeho příjezdu vlakem V a tím i celý sekretářčin argument.

Znalostní bázi $T = \{t_0, t_1, t_2\}$ tohoto příkladu tvoří na počátku věty:

t_0 : **uzavřené_stanice**

t_1 : $\forall x$ (**stihne_dřívější_vlak**(x) \rightarrow \neg **zpožděný**(x))

t_2 : $\forall x$ (**cestuje_autobusem**(x) \rightarrow \neg **cestuje_vlakem_V**(x))

Sekretářčin argument lze formalizovat default pravidlem:

$$\frac{\mathbf{T} : \text{cestuje_vlakem_V}(x)}{\text{zpožděný}(x)} \quad (\delta)$$

Zapsané default pravidlo δ , využívající jako pre-rekvizity znalostní bázi \mathbf{T} , má stejně jako jeho dvě další tvrzení t_1 a t_2 znalostní báze, univerzální charakter. Pravidlo říká, že když budou stanice skutečně uzavřené, a je-li fakt, že osoba x cestuje vlakem V, konzistentní se světem reprezentovaným znalostní bázi, pak se osoba x zpozdí.

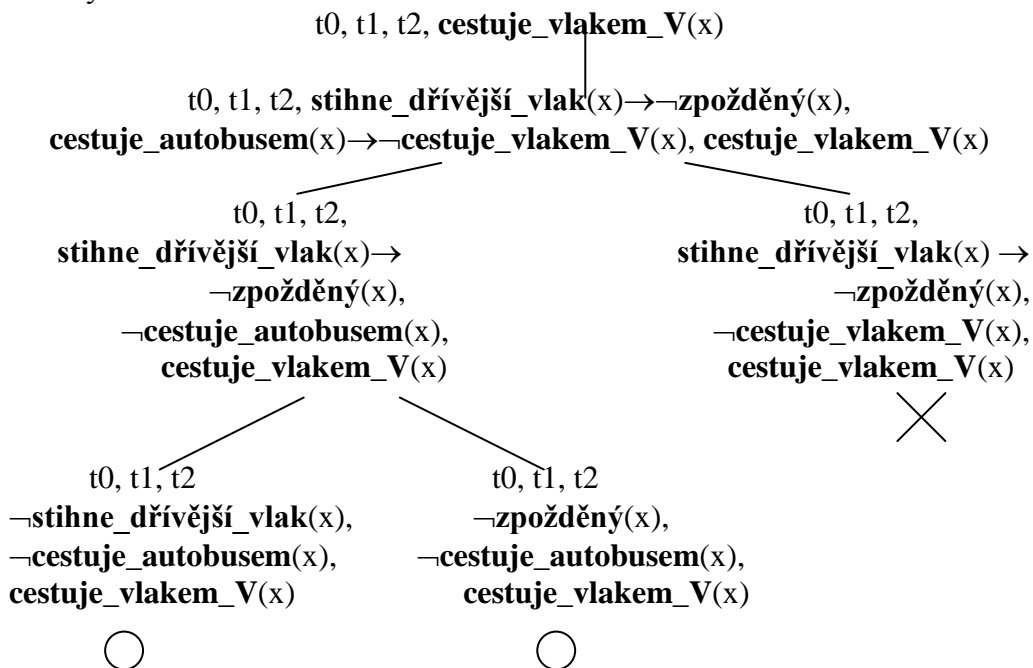
Reprezentace znalostí

Budování default teorie $\Delta = (\mathbf{T}, \mathbf{D})$, kde $\mathbf{T} = \{t0, t1, t2\}$ má na počátku následující formální vyjádření:

$$\Delta = \{ \{t0, t1, t2\}, \{\delta\} \}$$

Formální důkaz konsistence tvrzení **cestuje_vlakem_V(x)** se znalostní bází $\{t0, t1, t2\}$ lze provést pomocí sémantického tabla tak, že se dokáže, že \neg **cestuje_vlakem_V(x)** není z dané znalostní báze dokazatelná.

Obrázek představuje formální důkaz pomocí sémantického tabla provedený na bázevých instancích univerzálních formulí dané znalostní báze.



Reprezentace znalostí

- | | |
|---|---------------|
| 3. $(\text{stihne_dřívější_vlak}(x) \rightarrow \neg \text{zpožděný}(x))$ | MP na 1,2 |
| 4. $\text{stihne_dřívější_vlak}(x)$ | nový poznatek |
| 5. $\neg \text{zpožděný}(x)$ | MP na 3,4 |

Teorie, vybudovaná postupně aplikací default pravidla (δ), doplněním znalostní báze \mathbf{T} o nový poznatek a aplikací pravidla Modus ponens při reprezentaci problému v rámci axiomatického systému hilbertovského typu, obsahující formule

$t_0, t_1, t_2, \text{cestuje_vlakem_V}(x), \text{zpožděný}(x), \text{stihne_dřívější_vlak}(x), \neg \text{zpožděný}(x)$, je evidentně nesplnitelná - sporná.

Formule $\text{zpožděný}(x)$ nyní již není s rozšířenou znalostní bází konsistentní, proto ji nelze odvodit jako závěr default pravidla (δ).

Podobně je možno předpokládat, že L cestuje autobusem. Nyní je

$$\Delta^{\prime\prime} = (\mathbf{T}^{\prime\prime}, \mathbf{D}), \text{ kde } \mathbf{T}^{\prime\prime} = \mathbf{T}^{\prime} \cup \{ \text{cestuje_autobusem}(x) \}.$$

V hilbertovském axiomatickém systému \mathbf{H}_1 je nyní možno odvodit:

- | | |
|--|---------------|
| 1. $\forall x (\text{cestuje_autobusem}(x) \rightarrow \neg \text{cestuje_vlakem_V}(x))$ | tvrzení t_2 |
| 2. $\forall x (\text{cestuje_autobusem}(x) \rightarrow \neg \text{cestuje_vlakem_V}(x)) \rightarrow (\text{cestuje_autobusem}(x) \rightarrow \neg \text{cestuje_vlakem_V}(x))$ | axiomA4 |
| 3. $\text{cestuje_autobusem}(x) \rightarrow \neg \text{cestuje_vlakem_V}(x)$ | MP na 1,2 |
| 4. $\text{cestuje_autobusem}(x)$ | nový poznatek |
| 5. $\neg \text{cestuje_vlakem_V}(x)$ | MP na 3,4 |

Teorie rozšířená o novou odvozenou větu je opět nesplnitelná, i když z ní byl po předcházejícím kroku odstraněn závěr $\text{zpožděný}(x)$.

Tento příklad je typickým a vzorovým příkladem toho, že default pravidla mají skutečně charakter argumentů, které jsou vyvratitelné.



Kontrolní otázky a úkoly

1. V čem se liší monotónní a nemonotónní formální systémy?
2. Proč nemonotónní odvozování lépe odpovídá skutečnému usuzování?
3. Jakým způsobem lze modifikovat logiku prvního řádu, aby umožňovala nemonotónní odvozování?
4. Co jsou default pravidla?
5. Jaký význam má konsistence v nemonotónním odvozování?



Korespondenční úkol

Vypracujte pomocí nemonotónní logiky, že třínohý pes je stále pes.

Reprezentace znalostí

Shrnutí

Formální systém je monotónní, jestliže v něm přidání nové znalosti o modelovaném světě do znalostní báze zvětší nebo alespoň nezmenší množinu odvozených důsledků. Formální systém je nemonotónní, jestliže připouští, aby přidání nové znalosti o modelovaném světě do znalostní báze znehodnotilo některé jeho již odvozené věty.

Většina přístupů k nemonotónnímu odvozování vychází z tradiční logiky 1. řádu rozšířené tak, aby zahrnovala neurčitost umožňující nemonotónní odvozování:

- rozšířením jazyka
- zavedením dalších odvozovacích pravidel

Speciálním případem default pravidla je normální default pravidlo $(a : g) \textcircled{R} g$

znamená, že „platí-li předpoklady a závěr je konsistentní s tím, čemu se všeobecně věří, pak lze odvodit, že závěr platí.“

Deduktivní systém formalizující nemonotónní uvažování musí být schopen odvodit přijatelné logické důsledky (označujeme $\square\sim$). Ty ale musí být s množinou předpokladů konsistentní, tj. platné alespoň v jednom z jejích modelů.



7 Sémantický web

V této kapitole se dozvíte:

- co je to univerzální informační prostor webu a jaké požadavky se na něj kladou,
- o přechodu od syntaktického webu k sémantickému,
- co nebo kdo je to konsorcium W3C a jaké jsou jeho cíle,
- ideu sémantického webu,
- co je to ontologie a jak souvisí se sémantickým webem.

Po jejím prostudování byste měli být schopni:

- správně určit nezbytné vlastnosti pro další řízený rozvoj informačního prostoru,
- rozeznat syntaktický web od sémantického webu,
- popsat vývoj významu dat,
- definovat správný význam sémantického webu a jak to souvisí s ontologiemi.

Klíčová slova této kapitoly:

Informační prostor, sémantický web, konsorcium, WWW, syntaktický web, sémantický web, XML, taxonomie, ontologie.

Doba potřebná ke studiu: 2 hodiny

Průvodce studiem

Pojem sémantický web představuje v současné době konkrétně specifikovanou vizi, jak by měla být významová stránka informací a znalostí na webu reprezentována prostřednictvím takových formálních struktur, aby k nim bylo možno přistupovat, a to bez ohledu na rozdílné typy zdrojů (data, služby, webové stránky aj.) prostřednictvím agenta softwarového typu.

Na sémantický web se klade spousta požadavků a my se právě těmito požadavkům budeme v této kapitole věnovat. Proto prosím, abyste studiem této kapitoly věnovali minimálně 2 hodiny.



7.1 Univerzální informační prostor webu

V 90. letech minulého století vývoj webových technologií zaznamenal prudký rozvoj, tentokrát orientovaný především na kvalitu a kultivaci této informační báze. V této době již byl vytvořen (na principu hypertextu) univerzální navigovatelný informační prostor, otevřený dalšímu vývoji.

Reprezentace znalostí

Web jako univerzální informační prostor již tedy měl nezbytné vlastnosti pro další řízený rozvoj, jako

- decentralizaci, založenou na peer-to-peer architektuře,
- existenci řídicího a garantujícího orgánu W3C,
- interoperabilitu,
- univerzální přístupnost široké škále uživatelů, která bere v úvahu rozdílnosti jazyků, kultur, vzdělanosti a zvyklostí zúčastněných.

Vyhledávací služby webu, podporující hlavně možnost vyhledávání uživatelem podle klíčových slov, se staly nedostatečně adekvátními vzrůstajícímu objemu informací na webu. Potřeba modelu, který by splňoval představu webu jako znalostní báze o všem co existuje, přístupné komukoliv, vyústila v koncepci standardního mechanismu *výměny dat*. Tento mechanismus měl podporovat integrovaný a uniformní přístup k informačním zdrojům, inteligentní aplikace a služby, včetně zacházení s jejich rozdílnými formáty.

Speciálně zdůrazněnou *významovou stránkou dat* byla ve skutečnosti deklarována potřeba *webu nové generace*, v němž budou informace a znalosti automatizovaně vyhledatelné a zpracovatelné na základě jejich významu. Konkrétně tato nová koncepce znamenala přechod od dosavadního tzv. *syntaktického webu* k nové vizi *sémantického webu*.

7.2 Od syntaktického k sémantickému webu

7.2.1 Konsorcium W3C a jeho cíle

Organizace World Wide Web (WWW, resp. W3) byla vytvořena jako projekt Evropské Organizace pro nukleární výzkum (CERN), jejíž vědeckovýzkumný tým měl vytvořit vizi dalšího vývoje WWW. *Konsorcium W3C* bylo pak založeno v roce 1994 jako výsledek spolupráce mezi MIT (Massachusetts Institute of Technology) a CERN (European Organization for Nuclear Research) za podpory americké organizace DARPA (Defense Advanced Research Project Agency) a Evropské komise.

V současné době konsorcium W3C zodpovídá za vývoj technologií, které vedou web k plnému využití jeho potenciálu. W3C je fórum pro informace, komerci, komunikaci a kolektivní porozumění. Jako dlouhodobé cíle si konsorcium stanovilo:

1. *Univerzální přístup* - vytvořit web přístupný všem pokročilým technologiím, který by zohledňoval širokou variabilitu kultur, vzdělání, jazyků, materiálních základů a fyzických omezení pro uživatele ze všech kontinentů.
2. *Sémantický web* - vyvinout softwarové prostředí, které umožní uživateli nejlepší možné využití zdrojů přístupných na webu, založené na významové stránce webových informačních zdrojů.

Reprezentace znalostí

3. *Spolehlivý web* - řídit vývoj webu s ohledem na nejnovější právní, komerční a sociální aspekty, které tato technologie sebou přináší.

7.2.2 Web jako aplikace nad internetem

Web samotný je aplikace vytvořená nad internetem a jako taková musí sdílet jeho následující základní návrhové principy:



- *Interoperabilita* - specifikace týkající se webovských jazyků a protokolů musí být vzájemně kompatibilní a umožňovat interoperabilitu hardwaru a softwaru pro přístup k webu.
- *Vývojeschopnost* - web musí být schopen přizpůsobit se novým technologiím. Musí respektovat návrhové principy jako jednoduchost, modularita a rozšiřitelnost.
- *Decentralizace* - jde bezpochyby o nejobtížnější princip, neboť jeho architektura musí limitovat závislosti na centrálních prvcích. Internet má od počátku vlastnost peer-to-peer systému, která jde ruku v ruce s decentralizací. Naproti tomu je zde nutnost centrální definice konceptů a slovníků, která představuje dlouhodobé úkoly výzkumných týmů, s potřebnými aktualizacemi.

Web jako aplikace vychází z určitého kulturního dědictví, ovlivněného geografickým, kulturním a časovým kontextem. Respektuje rozdílné potřeby uživatelů vyhledávajících informace. Široká škála uživatelů od profesionálů, přes vědce, praktiky až k laikům pocítuje potřebu inteligentního adaptivního systému, založeného na strukturovaných, vzájemně propojených datech. Tomu však je zapotřebí přizpůsobit i základní stavební prvky webu – dokumenty.

7.2.3 Idea sémantického webu

Úsilí specialistů v oblasti formální reprezentace znalostí, inspirované praktickými problémy s informacemi uloženými na webu, se v posledních desetiletích soustřeďovalo na vytvoření moderních inteligentních systémů schopných na jedné straně obhospodařovat rozsáhlá data a na druhé straně poskytovat mechanismy „inteligentního“ vyhledávání na základě významu dat a odvozování ze znalostníchází. Vyhledávání na webu se totiž v současné etapě syntaktického webu děje pomocí softwarových prostředků, které zpravidla vyžadují pro uživatele, postupně se „proklikávajícího“ k požadovaným informacím, mnohdy zdlouhavou a nepřijemnou práci.

V roce 2001 zveřejnil Tima Berners-Lee, nynější prezident konsorcia W3C, vizionářskou představu, o přechodu od současného *syntaktického webu* k webu, který si zaslouží přívlastek sémantický. Tento přechod se podle autora vize měl uskutečnit doplněním současného syntaktického webu o možnost automatizovaného vyhledávání na základě významu (sémantiky), a to prostřednictvím

- strojově srozumitelných *sémantických informací* o obsahu dokumentů,

Reprezentace znalostí

- všeobecně sdílených *terminologických slovníků – ontologií*,
- *automatizovaných agentů*,
- *webových služeb*, které zprostředkovávají zpracování agenty.

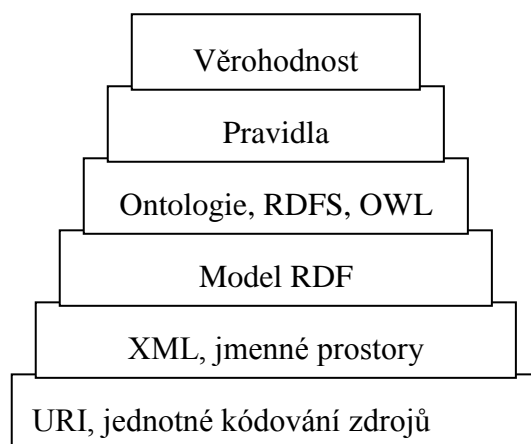
Pojem *sémantický web* představuje v současné době konkrétně specifikovanou vizi, jak by měla být významová stránka informací a znalostí na webu reprezentována prostřednictvím takových formálních struktur, aby k nim bylo možno přistupovat, a to bez ohledu na rozdílné typy zdrojů (data, služby, webové stránky aj.) prostřednictvím agenta softwarového typu.

Hlavní ideou *sémantického webu* je docílit možnosti dotazování různého typu na základě speciálního, dostatečně *expresivního konceptuálního modelu*, schopného popisu vzájemných vztahů mezi základními reprezentačními jednotkami – formalizovanými koncepty, jejich rolemi a vlastnostmi. Protože symbol reprezentujícího jazyka může mít různý význam v rozdílných aplikacích, resp. v různých přirozených jazycích, je nutno v rámci konceptuálního modelu vždy též prostřednictvím definice explicitně vyjádřit jeho *význam* formálními prostředky. Definice *významů webových pojmů* bývají organizovány do speciálních struktur, které jsou nazývány *ontologiemi*.

Sémantický web je navržen jako souhrn několika na sebe navazujících úrovní. Jedno z potenciálních schémat je uvedeno na obrázku dole:

- Adresování webových dokumentů je založeno na jednotném kódování dokumentů prostřednictvím URI (Unified Resource Identifier).
- Univerzální syntaktický standard představuje jazyk XML.
- úroveň výše stojí RDF (Resource Description Framework) jako hlavní datový model struktury reprezentace fakt, pravidel a vzájemných závislostí, obsažených ve webových dokumentech, na konceptuální úrovni.
- Na další úroveň jsou zařazeny ontologie (viz následující kapitolu), které prvkům modelu dodávají sémantiku.
- Z ontologií samotných ale nelze formulovat ani dedukovat nová tvrzení kromě těch, která pro individua přímo vyplývají z jejich členství ve třídě. Bylo proto třeba rozšířením modelu RDF na RDFS poskytnout jazykové prostředky, zavádějící do RDF prvky formálního vyjádření sémantiky reprezentovaných tvrzení, a vytvořit na bázi RDF spolu s formalismem *deskripční logiky* dostatečně vybavený ontologický jazyk OWL pro realizaci dedukce a tvorbu aplikací.
- Pracuje se proto na vývoji vrstvy pravidel, která bude zřejmě vycházet ze zkušeností jak s deduktivními databázemi, tak s věcnými pravidly v informačních systémech, aktivovanými prostřednictvím událostí.
- Posledním úkolem tvůrců *sémantického webu* je zajištění věrohodnosti (spolehlivosti) vyhledávaných i odvozených informací.

Reprezentace znalostí



Obrázek 7.1: Schéma vrstev sémantického webu

7.2.4 Vývoj popisu významu dat

Obecně se z hlediska popisu významu dat rozlišují 4 vývojová stádia:

- Text a databáze (před XML): data jsou vlastněna aplikací, ta je primárním cílem.
- XML dokumenty v jednoduchých doménách. Data zde již jsou v rámci domény nezávislá na aplikacích. XML může popsat standardní sémantiku např. v oblasti zdravotnictví, obchodu apod.
- Taxonomie a dokumenty se smíšenými slovníky: V tomto stádiu jsou uvažována a klasifikována data z různých domén. Jsou odhalovány jednoduché vztahy mezi kategoriemi v taxonomiích různých domén.
- Ontologie a pravidla: V tomto stádiu se nová data odvozují ze stávajících podle pravidel. Logická odvození se dějí prostřednictvím "sémantické algebry". V tomto stádiu již data neexistují jako blob, ale jsou součástí sofistikovaného mikrokosmu.

7.2.5 Strojově srozumitelné informace, všeobecně sdílená terminologie

Možnost doplnění strojově čitelných informací o obsahu dokumentů vyžaduje, aby tyto informace byly formulovány ve všeobecně sdílených základních pojmech (konceptech), postavených na jejich vlastnostech (atributech) a vzájemných vztazích, nikoliv, jak bylo zvykem předtím, na klíčových slovech. "Svět", který má být formálními prostředky modelován, musí být viděn prostřednictvím abstrakcí vyčleněných relevantních *konceptů*, jejich vlastností a vzájemných vztahů, tedy prostřednictvím konceptově orientovaného paradigmatu.

Ani v této oblasti nezačínali tvůrci koncepce sémantického webu „na zelené louce“. Pro zavedení všeobecně sdílených termínů pro popis významu konceptů, jejich atributů a vztahů bylo možno navázat na

Reprezentace znalostí

- v oblasti reprezentace znalostí již dlouhodobou snahu vyčlenit a formálně definovat základní pojmy, na nichž by bylo možno vybudovat „všeobecně sdílené“ („common-sense) definice pojmů, jejich vztahů a též principů usuzování - jazyk KIF (Knowledge Interface Format), Sowovy konceptuální grafy a další,
- již zavedený způsob definování základních údajů o webovém dokumentu prostřednictvím značkovacích jazyků (XML) - všeobecně sdílené metadatové jádro o dokumentu se nazývá podle místa vzniku Dublin Core.

Dokument DCMI (Dublin Core Metadata Initiative) specifikuje metadatové termíny popisující obsah webových dokumentů tak, že každý termín je specifikován minimální množinou atributů (tab. 7.1).

Tabulka 7.1

Jméno	jedinečné označení termínu
URI	zdroj, který jednoznačně identifikuje termín
Návěští	čitelné (člověku) návěští přiřazené termínu
Definice	tvrzení reprezentující pojem a základní povahu termínu
Typ termínu	typ termínu, jako je prvek, kódové schéma apod., tak jak je popsán v DCMI gramatických principech
Status	přiřazený termínu v DCMI procesu užití
Datum	kdy byl termín poprvé deklarován

Příklad definice termínu "creator" v DCMI je uveden na obr. 7.2. Realizace myšlenky, opatřit webové dokumenty strojově srozumitelnými sémantickými informacemi o jejich obsahu, spočívá ve vytvoření metadat, která vycházejí z opatření každého dokumentu určitými jeho obsah charakterizujícími *značkami* (tagy). Jejichž užití v rámci realizace sémantického webu navazuje na vývojově předcházející značkování dokumentů v rámci jazyka HTML (Hyper Text Markup Language).

Term Name: creator	
URI:	http://purl.org/dc/elements/1.1/creator
Label:	Creator
Definition:	An entity primarily responsible for making the content of the resource.
Comment:	Examples of a Creator include a person, an organisation, or a service. Typically, the name of a Creator should be used to indicate the entity.
Type of Term:	element
Status:	recommended
Date Issued:	1999-07-02

Obrázek 7.2

Web vděčí za rozšíření svého využití do značné míry jednoduchosti a průhlednosti jazyka HTML, který umožňuje i běžnému uživateli webu vytvořit během několika minut (formou znovupoužití existujícího zdrojového kódu)

Reprezentace znalostí

funkční webové stránky. Dokumenty na Internetu jsou vytvořené v HTML, jsou sice opatřeny značkami, ale nedostatek těchto HTML značek spočívá v tom, že dokument je jimi formátován do podoby, které rozumí pouze člověk, ale softwarový agent si s pochopením informací v něm obsažených nemůže vědět rady. Značky se totiž zaměřují na formátování vzhledu výstupu tak, aby sdělovaná informace byla co nejčitelnější pro lidského čtenáře. Dnešní internetový prohlížeč podle značek, ve kterých je informace uzavřena, pozná, že daná část textu má být napsána kurzívou nebo že na určitém místě je obrázek a dokáže podle toho zformátovat výstup odpovídajícím způsobem. Ale o významu dané informace nedokáže zjistit nic, neví, co text popisuje či co je vyobrazeno na obrázku.

Vyřešit otázku lepší spolupráce člověka a počítače a tím i efektivního vyhledávání na Internetu je cílem sémantického webu. Značky sémantického webu musí popisovat informace v dokumentu takovým způsobem, aby jim byl schopen porozumět softwarový agent. A aby též umožňovaly zachytit vztahy mezi rozdílnými množinami dat. Počítač pak např. musí rozpoznat, že položka IČO v databázi označuje totéž jako pole formuláře webové stránky s označením IDENTIFIKAČNÍ_ČÍSLO_OBČANA. To navíc napomáhá systému efektivněji shromažďovat informace z různých zdrojů.

7.2.6 Ontologie

Pro realizaci vize sémantického webu je k popisu obsahu dokumentů na základě konceptů, jejich atributů a vztahů potřeba vytvářet, resp. dále rozšiřovat základní soubory *všeobecně sdílených termínů* týkajících se *modelovaného světa - domény*. Jinak řečeno, je třeba jednoznačně definovat základní pojmový aparát schopný vytvářet tvrzení o všem, co existuje v zájmové doméně, ať v reálném nebo abstraktním světě. Takovými tvrzeními, týkajícími se bytí, se ale v klasické filosofii zabývalo jedno z dvou jejích odvětví - *ontologie* (*teorie bytí*, druhým pak je *gnoseologie* - *teorie poznání*).

V informatice se v současné době pojmem *formální ontologie* (podrobněji viz následující kapitulu Formální ontologie) rozumí takový popis významu termínu reprezentujícího uvažovanou entitu, který je založen na jeho pozici vůči termínům náležejícím jiným entitám, nadřazeným či podřazeným. V rámci tohoto přístupu vznikají produkty - hierarchicky uspořádané *slovníky termínů* určitých vymezených oblastí bytí, které jsou v informatice rovněž nazývány *ontologiemi*.

Rozlišuje se *ontologie nejvyšší úrovně* (top level) od *ontologií doménových*, resp. vztahujících se ke konkrétním aplikačním oblastem.

Nové termíny lze z již existujících termínů, zařaditelných do ontologií, vytvářet kombinacemi, které jsou z hlediska logiky konjunkcemi, disjunkcemi, resp. kvantifikovanými omezeními (viz prostředky deskripční logiky), na základě nichž je možno formálně reprezentovat další koncepty.

Potřeba budování terminologie, na jejímž základě by se metadata o webových dokumentech stala srozumitelná softwarovým agentům, plnícím specializované požadavky uživatelů, vedla ke vzniku nového odvětví v oblasti formální

Reprezentace znalostí

reprezentace znalostí, který se nazývá *formální ontologií*. Té je věnována následující kapitola.

Cílem sémantického webu je zpřístupnění webových zdrojů automatickým procesům, a to pomocí

- rozšíření stávajících značkovacích jazyků o možnost zachycení sémantiky v metadatech popisujících obsah/funkce webových zdrojů,
- použití ontologií jako formálně specifikovaných slovníků, tak aby se formální specifikace staly automaticky přístupnými.

Definice 7.1

Sémantický web je infrastruktura pro vyhledávání a odvozování na webu, založená na formálních ontologiích a metadatech o webových zdrojích.



Kontrolní otázky a úkoly

1. Jaké jsou cíle konsorcia W3C?
2. Jaké jsou základní návrhové principy webu?
3. O co by se měl podle T. Berners-Lee doplnit syntaktický web?
4. Co je to ontologie a k čemu slouží?
5. Jaký je rozdíl mezi ontologií a formální ontologií?



Shrnutí

Organizace World Wide Web (WWW, resp. W3) byla vytvořena jako projekt Evropské Organizace pro nukleární výzkum (CERN), jejíž vědeckovýzkumný tým měl vytvořit vizi dalšího vývoje WWW. *Konsorciium W3C* bylo pak založeno v roce 1994 jako výsledek spolupráce mezi MIT (Massachusetts Institute of Technology) a CERN (European Organization for Nuclear Research) za podpory americké organizace DARPA (Defense Advanced Research Project Agency) a Evropské komise.

Web jako aplikace vychází z určitého kulturního dědictví, ovlivněného geografickým, kulturním a časovým kontextem. Respektuje rozdílné potřeby uživatelů vyhledávajících informace. Široká škála uživatelů od profesionálů, přes vědce, praktiky až k laikům pocítuje potřebu inteligentního adaptivního systému, založeného na strukturovaných, vzájemně propojených datech. Tomu však je zapotřebí přizpůsobit i základní stavební prvky webu – dokumenty.

Úsilí specialistů v oblasti formální reprezentace znalostí, inspirované praktickými problémy s informacemi uloženými na webu, se v posledních desetiletích soustřeďovalo na vytvoření moderních inteligentních systémů schopných na jedné straně obhospodařovat rozsáhlá data a na druhé straně poskytovat mechanismy „inteligentního“ vyhledávání na základě významu dat a odvozování ze znalostních bází. Vyhledávání na webu se totiž v současné etapě syntaktického webu děje pomocí softwarových prostředků, které



Reprezentace znalostí

zpravidla vyžadují pro uživatele, postupně se „proklikávajícího“ k požadovaným informacím, mnohdy zdlouhavou a nepříjemnou práci.

Pro realizaci vize sémantického webu je k popisu obsahu dokumentů na základě konceptů, jejich atributů a vztahů potřeba vytvářet, resp. dále rozšiřovat základní soubory *všeobecně sdílených termínů* týkajících se *modelovaného světa - domény*. Jinak řečeno, je třeba jednoznačně definovat základní pojmový aparát schopný vytvářet tvrzení o všem, co existuje v zájmové doméně, ať v reálném nebo abstraktním světě. Takovými tvrzeními, týkajícími se bytí, se ale v klasické filosofii zabývalo jedno z dvou jejích odvětví - *ontologie (teorie bytí, druhým pak je gnoseologie -teorie poznání)*.

V informatice se v současné době pojmem *formální ontologie* (podrobněji viz následující kapitolu Formální ontologie) rozumí takový popis významu termínu reprezentujícího uvažovanou entitu, který je založen na jeho pozici vůči termínům náležejícím jiným entitám, nadřazeným či podřazeným. V rámci tohoto přístupu vznikají produkty - hierarchicky uspořádané *slovníky termínů* určitých vymezených oblastí bytí, které jsou v informatice rovněž nazývány *ontologiemi*.

8 FORMÁLNÍ ONTOLOGIE

V této kapitole se dozvíte:

- o ontologii a vize sémantického webu,
- o problému nezávislosti znalostí,
- ontologickou konceptualizaci,
- jak od datového modelu ke konceptuálnímu modelu,
- o historii vývoje ontologií,
- o ontologiích vyšších úrovní,
- některé systémy na podporu tvorby ontologií.

Po jejím prostudování byste měli být schopni:

- definovat formální ontologii,
- vytvářet koncepty,
- určit typ a hierarchii ontologie,
- využívat možné přístupy k doménovým formálním ontologiím,
- definovat koncepty pomocí formální konceptové analýze.

Klíčová slova této kapitoly:

Sémantický web, metadata, formální ontologie, znalost, paradigma, konceptualizace, koncept, formální konceptová analýza.

Doba potřebná ke studiu: 11 hodin

Průvodce studiem

Ontologie popisuje určitou oblast zájmu (doménu) formálním způsobem – definuje třídy objektů, které se v dané oblasti nacházejí, a vztahy, které mezi nimi mohou existovat. Význam ontologie spočívá především v usnadnění komunikace mezi lidmi, zlepšení spolupráce softwarových systémů a ve zdokonalení systémového inženýrství. Ontologie do všech těchto oblastí přináší možnost sjednocení pohledu, udržení konzistence a jednoznačnosti.



8.1 Ontologie a vize sémantického webu

Idea *sémantického webu*, inspirovaná skutečností, že inteligentní vyhledávání informací na webu prostřednictvím softwarových agentů nelze zajistit pouze v rámci syntaktických pravidel, značnou měrou přispěla závažnosti tématu ontologie. Potřeba formulovat přesné otázky po významové stránce, které by „inteligentní“ internetový vyhledávač vedly podle jejich obsahu přímo k cíli, se

Reprezentace znalostí

neobejde bez *formalizované sémantiky* webovských dokumentů, založených na všeobecně sdílených ontologiích.

8.1.1 Strojová čitelnost metadat webových dokumentů

Když v roce 2001 Tim Berners-Lee zveřejnil svoji vizionářskou představu o přechodu od současného syntaktického webu k "sémantickému webu", měl na mysli, že by se tento přechod měl uskutečnit doplněním současného *syntaktického webu* o možnost automatizovaného vyhledávání na základě formalizované sémantiky webových dokumentů, a to prostřednictvím termínů všeobecně sdílených terminologických slovníků – ontologií. K realizaci této vize je ale především třeba vytvořit prostředky a technologie, které pracují s *metadaty* o datech, tj. se strojově čitelnými informacemi o obsahu webových dokumentů, podporují technologickou i sémantickou interoperabilitu a zaručují spolehlivost.



Definice 8.1

Metadaty o informačních zdrojích se obecně rozumí data k identifikaci, popisu významu a lokaci informačních zdrojů, ať již fyzických nebo elektronických.

Metadaty jsou např. knihovnické katalogové karty. Realizace strojové čitelnosti metadat pak vyžaduje:

1. obecný datový model pro reprezentaci metadat založený na konceptově orientovaném paradigmatu - vidění modelovaného světa v konceptech, jejich atributech a vzájemných vztazích, který v současné době představuje *konceptuální model RDF*,
2. *jednotnou identifikaci zdrojů* (v současnosti řešenou pomocí Unified Resource Identifier - URI),
3. přístup *softwarových agentů* k metadatům na webu, který se řeší odpovídajícími technickými a programovými prostředky,
4. obecné slovníky neboli *ontologie* uvažovaných zájmových domén, které jsou již dnes pro řadu těchto domén vytvořeny.



Průvodce studiem

V rámci vývoje informačních technologií „ontologický“ přístup způsobil posun zaměření od reprezentace znalostí (procedurálních i deklarativních) v dané doméně k reprezentaci znalostí o doméně samotné.

8.2 Znalosti a ontologie

8.2.1 Problémová nezávislost znalostí

Znalostí ve smyslu umělé inteligence se rozumí v podstatě cokoli, co lze agentovi popsat tak, že následné naprogramování způsobí jeho rozumné (inteligentní) chování vzhledem k cíli, který má agent řešit. Tvůrci znalostních bází se proto často zaměřují nikoliv pouze na pravdivé zobrazení modelovaného světa, ale též na jeho funkční užitečnost pro dosažení tohoto cíle. Obsah znalostní báze, která by měla být využitelná i pro jiné agenty, zaměřené na jiné cíle, by se však měl vztahovat k objektivní realitě, nikoliv k zaměření jedné konkrétní agentovy činnosti. Je proto zřejmé, že by se znalosti měly vztahovat více k tomu, co je pravdivé vzhledem k reprezentované doméně (modelovanému světu) a méně k cíli, který agent řeší. Modelování pak pouze zajišťuje korespondenci mezi znalostní bází a dvěma separátními subsystémy: činností agenta, který řeší problém, a jeho prostředím, tj. formálně reprezentovanou problémovou doménou.

Zdá se, že až do současnosti stále převládá přístup, v rámci něhož doménové znalosti jsou těsně závislé na konkrétní řešené úloze. Berou se tak v úvahu jen doménové znalosti relevantní vzhledem k tomuto cíli. Jestliže však takové relevantní znalosti nejsou uvažovány jako část znalostí objektivní reality v doméně jako celku, nezachycuje modelovací hledisko dostatečně *vnitřní strukturu domény*.

8.2.2 Konceptuální úroveň modelování zájmové domény

Vzrůstající požadavky na *sdílení znalostních bází* musí dnes být založeny na solidním konceptuálním základě. Výše zmíněný aktuální požadavek integrace by ale neměl být zaměřen jen na metodiku *konceptualizace*, ale na jasnou komunikaci celků, založených na detailně a jednoznačně specifikovaných konceptech a jejich vzájemných vztazích (s možností vyjádření v různých jazycích), ponechávající každému uživateli volnost při vytváření své specializované konceptualizace. Nejlepším kandidátem řešení této problematiky se právě jeví nové odvětví v oblasti reprezentace znalostí, které s klasickou filosofií sdílí název *ontologie*.

Formalizace znalostí o dané doméně (referenčním systému) začíná na *konceptuální úrovni*, a to jak v případě modelování deklarativních, tak i v případě modelování procedurálních znalostí. Tu však předchází ještě neformalizované vyjádření konceptualizace přirozeným jazykem (lingvistická úroveň). Tato základní konceptualizace na dvou nejvyšších úrovních abstrakce zahrnuje výběr *relevantních objektů* modelované domény, viděných prostřednictvím zvoleného paradigmatu, jejich seskupení do tříd (konceptů), jejich vzájemné závislosti a vztahy. Konceptuální model pak na jejich základě vytváří, zpravidla již formalizovaná, základní pravdivá tvrzení o modelované doméně. Ta tvoří výchozí základ znalostní báze, bez ohledu na způsob, jakým budou znalosti na závěrečné úrovni implementovány.

Reprezentace znalostí

Při tradičním pojetí etap modelování by za abstraktní úrovní konceptuální následovala již jen konkrétní úroveň implementační, tj. *úroveň logické formalizace*.

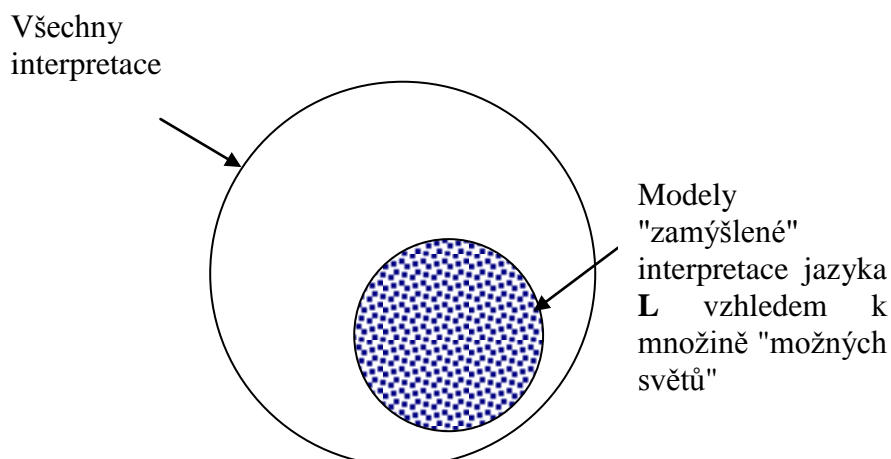
8.2.3 Zamýšlená interpretace a možné světy

Obvyklým logickým přístupem je, že znalosti ve znalostních bázích představují formálně reprezentované předpoklady, z nichž lze odvodit další znalosti. Při tradičně pojatém logickém přístupu se znalostní báze nějaké konkrétní domény (světa, referenčního systému) vytváří jako formálně logický popis aktuálního stavu této domény. Soubor logických formulí zvoleného formálního jazyka, reprezentujících vlastnosti modelovaného světa, tvoří pak množinu *speciálních axiomů* teorie, z níž lze podle odvozovacích pravidel v rámci uvažovaného systému (zpravidla logiky prvního řádu) generovat další *věty teorie*.

S formálním popisem domény souvisí stanovení slovníku potřebných primitivních prostředků reprezentace, tj. predikátů, funkcí a konstant pro následující úroveň abstrakce v rámci modelování formálními prostředky. Predikáty, funkce a konstanty zavedené v logickém jazyce pro reprezentaci modelovaných objektů, jejich vlastností a vztahů se zpravidla volí tak, aby byly dostatečně expresivní, ale zároveň co nejobecněji použitelné. O povaze těchto vztahů se však v rámci tradiční formální logiky jako modelovacího jazyka nic nepředpokládá, jsou obecné a obsahově nezávislé. *Logická úroveň* modelování proto umožňuje *formální interpretaci* primitivních prostředků jazyka, která je v podstatě libovolná. Jedním ze základních rysů sémantiky tradiční formální logiky prvního řádu, je tedy skutečnost, že se zabývá obecnými vlastnostmi různých aplikovatelných struktur interpretace (tradičně především matematických), které mohou být modely zvoleného jazyka (obr. 8.1). Přístup na logické úrovni modelování, a to i tehdy, má-li se jednat o reprezentaci znalostí o konkrétním modelovaném světě, je v zpravidla typicky „ontologicky neutrální“, tj. vyznačuje se větším důrazem na formu a formální principy dedukce než na sémantiku ve smyslu ontologicky co nejvěrnější reprezentace konkrétních znalostí o reprezentované doméně. Z hlediska tradiční logiky *zamýšlená interpretace* není jedinou možnou interpretací vytvořené znalostní báze. Mohou existovat i další modely, tj. struktury interpretací vztahující se k jiným doménám, v nichž platí všechny formule znalostní báze. Mohou tedy existovat i jiné modifikacemi vzniklé *možné světy*, pro něž se vytvořená reprezentující znalostní báze hodí.

K tomu, aby bylo možné znovupoužití znalostí, je ale třeba stanovit pravidla omezující možné volby interpretace. Řešením je přechod na ontologickou úroveň, kde koncepty a role mají *standardní formální interpretaci*. V rámci tradičního přístupu mohou mít koncepty, reprezentované predikáty ve znalostní bázi, rozdílné interpretace, zatímco při ontologickém pojetí modelování by denotáty konkrétních jazykových prostředků (predikátů, funkcí, konstant) použitých v této zamýšlené interpretaci měly odpovídat modelovanému světu, event. zahrnující též jeho možné proměny, tj. měly by určovat interpretaci v rámci dané množiny stanovených *možných světů*.

Reprezentace znalostí



Obrázek 8.1

8.2.4 Od konceptualizace k implementaci báze znalostí

V rámci ontologického pojetí konceptualizace a následné logické implementace vlastností a vztahů modelovaného světa s jeho *zamýšlenou interpretací* (obrázek 8.1 **Chyba! Nenalezen zdroj odkazů.**), odvozenou z modelované skutečnosti, se vkládají mezi úrovně konceptuální a implementační ještě další dvě úrovně (tab. 8.1), a to *úroveň ontologická*, jejímž účelem je co nejpřesnější explicitní vymezení *zamýšleného modelu* vzhledem k modelované skutečnosti, a *úroveň epistemologická*.

Tabulka 8.1

Úroveň	Prostředky	Interpretace	Hlavní rysy
Lingvistická	lingvistické termíny	subjektivní	jazyková závislost
Konceptuální	konceptuální vztahy	subjektivní	lingvistická nezávislost
Epistemologická	strukturální vztahy	libovolná	struktura
Ontologická	ontologické vztahy	vymezená realitou	význam
Logická, implementační	formální logika	libovolná	formalizace

Predikáty zavedené v jazyce pro vyjadřování vlastností a vztahů modelovaných objektů je třeba volit tak, aby byly dostatečně expresivní, ale zároveň co nejobecněji použitelné. To se řeší na epistemologické úrovni modelování, která vyplní chybějící *úroveň struktury* mezi konceptuální úrovní a ontologickou úrovní, kde se již vyžaduje určitý zamýšlený význam.

V rámci formálně ontologického přístupu je právě epistemologická úroveň touto *úrovní struktury*. Jde na ní (na úrovni konceptuální) o zachycení struktury znalostí prostřednictvím konceptuálních jednotek, sestávajících ze vzájemně

Reprezentace znalostí

provázaných konceptů, jejich atributů a rolí, které koncepty v rámci jednotky sehrávají. Tyto koncepty se zpravidla opírají se o jazykové primitivy generické povahy (kategorie). Ontologickou povahu primitivů lze pak definovat na základě konceptuálních kategorií, jako jsou koncepty, role, atributy, vlastnosti, části, stavy, procesy, akce,..., jimž jsou podrobněji věnovány následující odstavce.

Na konceptuální úrovni by v případě ontologického pojetí modelování primitivní prostředky formální reprezentace (koncepty, role, vlastnosti, atributy,...) neměly postrádat definitní *kognitivní interpretaci*, odpovídající jazykově i implementačně nezávislým pojmům dané aplikační domény. Ty pak vytvářejí *základní strukturu domény*, tvořící určitou kostru, a uživatel je směřován k vyjádření znalostí ve tvaru užší specifikace, vycházející z této kostry.

Rozdíl mezi epistemologickou úrovní a logickou úrovní lze ilustrovat na následujících příkladech.



Příklad 8.1

a) Reprezentace tvrzení, že "každý tvor s velmi malým mozkem je hloupý" lze realizovat řadou způsobů, nichž první je evidentně nejméně vhodný.

$$\forall x (\text{tvor_s_velmi_malým_mozkem}(x) \rightarrow \text{hloupý}(x))$$

$$\forall x (\text{tvor}(x) \ \& \ \text{mozek}(x, \text{malý}, \text{velmi}) \rightarrow \text{hloupý}(x))$$

$$\forall b (\text{tvor}(x) \ \& \ \text{velmi_malý}(\text{mozek}(x)) \rightarrow \text{hloupý}(x))$$

$$\forall b (\text{tvor}(x) \ \& \ \text{malý}(\text{mozek}(x), \text{velmi}) \rightarrow \text{hloupý}(x))$$

b) Reprezentace konceptu "červený míč":

Na logické úrovni výraz v logice prvního řádu může mít tvar

$$\exists x (\text{míč}(x) \ \& \ \text{červený}(x)).$$

Při způsobu reprezentace na logické úrovni je v podstatě lhostejné, zdali použité predikáty reprezentují vlastnost věci, věc samotnou nebo úlohu, kterou sehrává.

Na epistemologické úrovni je však třeba se rozhodnout vzhledem k účelu této formalizace a vzhledem k využitelnosti v rámci modelovaného celku, co má být konceptem a co rolí. Např. v reprezentaci deskripční logikou je míč koncept a rolí je barva s výplní "červený":

$$\text{míč} \sqcap \exists \text{BARVA} . \text{červený}$$

nebo jednodušeji, jak míč, tak i červený jsou koncepty:

$$\text{míč} \sqcap \text{červený}.$$

Na ontologické úrovni jsou explicitně specifikovány pojmy v souladu s jazykovým hlediskem za účelem zúžení možností interpretace, a to na bázi základních ontologických kategorií. Termínu míč, stejně jako termínu červený je jednoznačně přiřazen význam pomocí pozice každého z pojmů v příslušných *ontologických strukturách*, určujících jejich *ontologickou sémantiku*.

Reprezentace znalostí

8.2.5 Ontologická formalizace

Centrem zájmu ontologicky formalizované sémantiky vzhledem k dané doméně je snaha formálním jazykem co nejlépe a na základě co nejobecnějšího konsensu strukturovaně reprezentovat prostřednictvím množiny formálních definic (hierarchicky uspořádaných slovníků) ty entity a jejich vzájemné vztahy, které představují relevantní *esence* zamýšlených modelů dané domény.

Na začátku tohoto snažení však evidentně stojí problém stanovení, které entity se svými vzájemnými vztahy jsou tou *relevantní esencí* konkrétního modelovaného světa (referenčního systému) - zde se jedná o přístup ve vazbě na *speciální, aplikační resp. doménové ontologie* - a zdali jsou zde entity spolu s jejich vzájemnými vztahy, které jsou svou obecností přítomny v celých třídách modelovaných referenčních systémů - v tomto případě se modelovací přístup opírá o *ontologii (nej)vyšší úrovně (top-level ontology)*.

Ontologie bývá definována podle Grubera následující, poněkud vágní definicí, v níž pro odlišení od své klasické předchůdkyně získává ontologie jako oblast informatiky přívlastek formální.

Definice 8.2

Formální ontologie je reprezentace všeobecně sdílené konceptualizace v určité oblasti, založená na terminologickém slovníku (ontologii) pro danou oblast.



Z uvedené definice je vidět, že pro ujasnění podstaty formální ontologie, která má sehrát zásadní úlohu v tvorbě znalostních bází, je třeba především věnovat pozornost samotnému pojmu konceptualizace.

8.3 Ontologická konceptualizace

Z potřeby *ontologicky adekvátních znalostních bází* a z nich vycházejících informačních systémů, která je typická pro vývoj posledních let, vyvstala též potřeba nového pojetí konceptuálního modelování – *ontologická konceptualizace*.

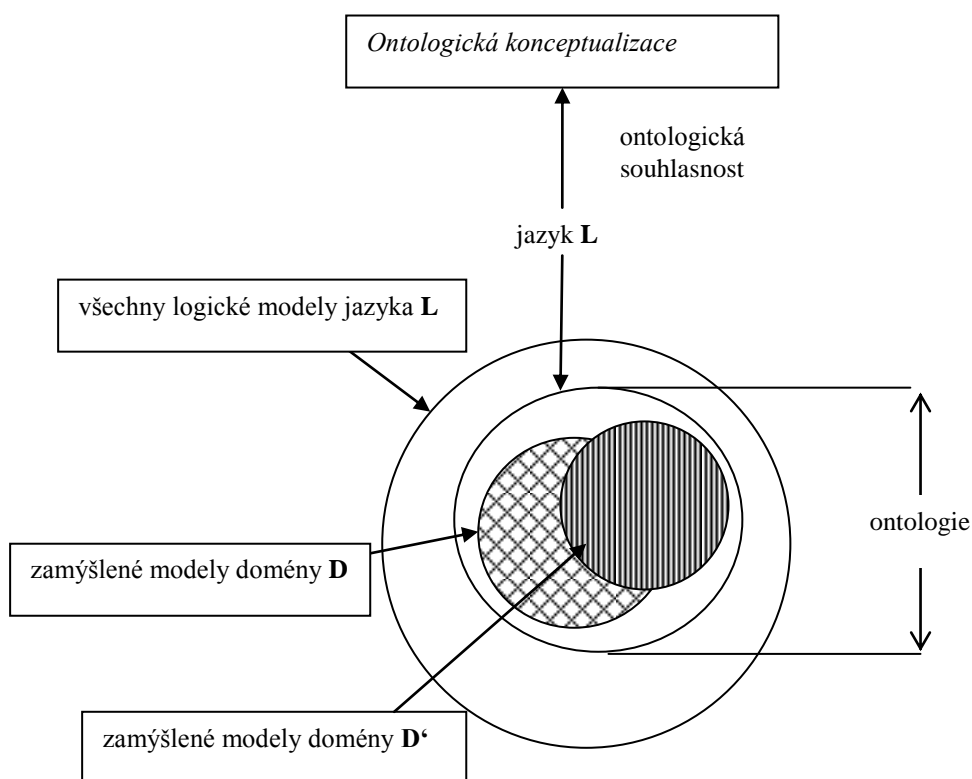
8.3.1 Paradigma, metamodel a jazyk

Na počátku procesu konceptualizace, jehož cílem je vytvoření konceptuálního modelu, tvůrce vidí modelovaný svět prostřednictvím určitého paradigmatu. Paradigmatem se přitom rozumí určitý způsob vidění modelovaného světa (referenčního systému). Adekvátně tomuto způsobu vidění tvůrce zvolí abstraktní metamodel, tj. soubor vhodných pojmů a jejich vzájemných vztahů s danou sémantikou. Nástrojem umožňujícím vlastní modelování je pak modelu odpovídající zvolený jazyk.

Reprezentace znalostí

Pojmy a prostředky téhož konceptuálního modelu je možno reprezentovat prostřednictvím celé řady různě zvolených jazykových prostředků. Např. model vycházející z konceptově orientovaného paradigmatu může pro reprezentaci znalostí o modelovaném světě použít stejně dobře symbolických jako grafických jazykových prostředků.

Konceptualizace v ontologickém pojetí je pak formální reprezentací struktury, tvořené entitami modelované domény (referenčního systému) v jejich vzájemných vztazích, tak jak má být vnímána a organizována agentem, prostřednictvím vhodně zvoleného formálního jazyka.



Obrázek 8.2

Agenti mohou komunikovat pouze tehdy, jestliže se jejich zamýšlené modely překrývají. Toto překrytí lze zajistit příslušnou společnou ontologií (obr. 8.2). Dvě různé ontologie, relevantní vzhledem k těmto modelům, se totiž mohou překrývat (být totožné), i když se jejich zamýšlené modely nepřekrývají. Naopak integrace doménově specifických ontologií zdola nahoru nezaručí konsistenci zamýšlených modelů. Tu ale může zaručit existence ontologie vyšší úrovně.

Obecná ontologie vyšší úrovně zjednodušuje návrh, zvyšuje kvalitu a srozumitelnost. Ontologické konceptuální modelování se zaměřuje na zachycení relevantních entit příslušné domény s využitím specifikačního jazyka, založeného na malém počtu základních doménově nezávislých ontologických kategorií tvořících ontologii nejvyšší úrovně.

Reprezentace znalostí

8.3.2 Od datového modelu ke konceptuálnímu modelu

Pojem datového modelování se objevil asi před třemi desítkami let v souvislosti s rozvojem relačních databází. Postupným vývojem se datový model stával nástrojem analýzy *sémantiky organizace informací* a její struktury.

V současné době se sémantika dat stala středem zájmu sama o sobě. Znalostní inženýři zabývající se reprezentací znalostí dospěli k poznání, že sémantické hledisko je nepostradatelné, mají-li následně vyvíjené systémy být schopny navzájem komunikovat a má-li tato komunikace probíhat prostřednictvím programových agentů.

Datovým modelem se obecně rozumí popis významové (sémantické) struktury modelované domény, jejích objektů a vztahů mezi nimi, a to zpravidla formou vhodného i laikům srozumitelného grafického jazyka. Zatímco původně, a to v souvislosti s vývojem databází, byl datový model pojímán jako prostý *návrh databáze*, postupně se datové modelování stalo *konceptuálním modelováním* na nejvyšší úrovni abstrakce, bez kterého se v podstatě neobejde žádný návrh programového nebo databázového systému bez ohledu na jeho povahu a účel.

8.3.3 Syntax jazyka konceptuálního modelování, (meta)model

V případě jazyků lexikálního typu je abstraktní syntax definována bezkontextovou gramatikou, tj. abecedou výchozích symbolů a souborem gramatických pravidel, umožňujících ze symbolů abecedy vytvářet formule jazyka.

V grafickém jazyce nelze abstraktní syntax definovat jako formuli v termínech řetězců symbolů, ale jako *metamodel* v termínech svých konstruktorů - obrazových znaků.

Množina grafických modelovacích prvků tak odpovídá symbolům abecedy jazyka a abstraktní syntax je nad ní definována jako soubor přípustných sestav obrazových prvků jako *metamodel* jazyka, který umožňuje tvorbu konkrétních modelů v daném jazyce.

Definice 8.3

Metamodel grafického jazyka je popisem abstraktní syntaxe jazyka, který definuje:

1. abecedu jako množinu grafických symbolů - konstruktorů jazyka,
2. atomický prvek grafického jazyka,
3. množinu pravidel tvorby dobře utvořených, gramaticky správných modelů.



Reprezentace znalostí

V souladu s touto definicí je v kapitole 2 definován grafický jazyk asociativních sítí prostřednictvím základních symbolů jazyka, atomu sítě a pravidel vytváření sítí (Definice 2.2, Definice 2.3).

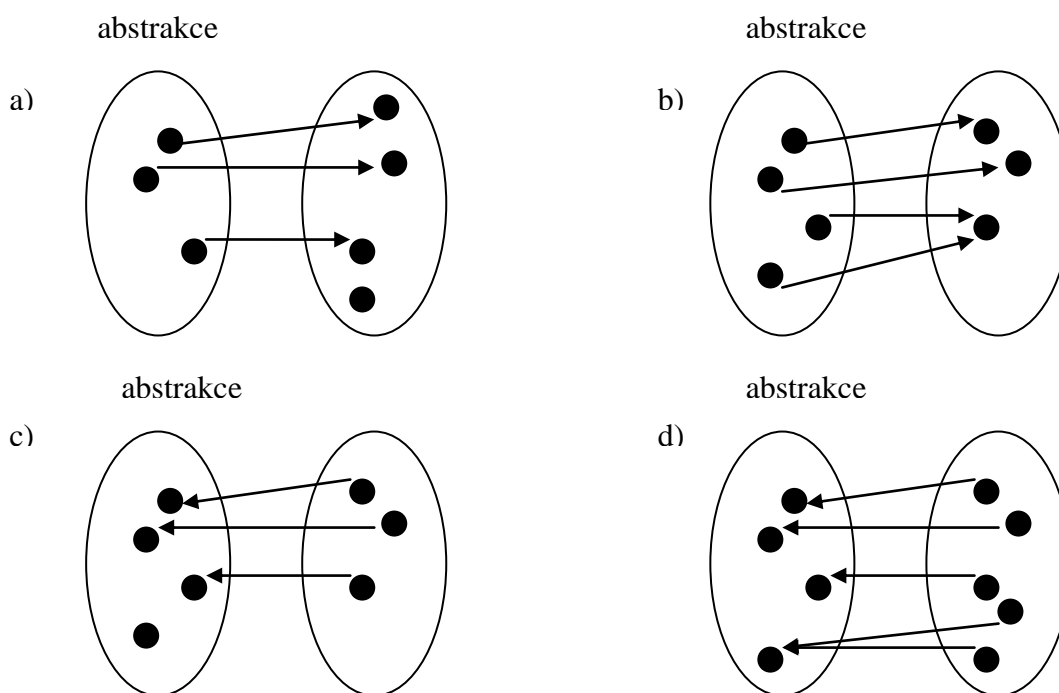
8.3.4 Vlastnosti jazyka konceptuálního modelování domény

Mezi ontologií reprezentované domény D a metamodelem příslušného grafického doménového jazyka by mělo existovat isomorfické zobrazení, aby interpretace jazyka na základě příslušné ontologie odpovídala přesně a jedinečně reprezentované abstrakci.

Vlastnosti, které by tedy měl mít jazyk vhodný pro konceptuální modelování, speciálně pro reprezentaci doménových ontologií, jsou dány následujícími zobrazeními mezi abstrakcí vymezenými modelovými prvky jazyka a koncepty ontologie O dané domény:

- Jasnost (přehlednost) - jazyk L je přehledný vzhledem k doméně D , právě když každý modelový prvek jazyka reprezentuje nejvýše jeden doménový koncept ontologie O této domény D .
- Korektnost - jazyk L je korektní vzhledem k doméně D , právě když je každý modelový prvek jazyka má svoji reprezentaci v termínech doménových konceptů ontologie O .
- Úspornost - jazyk L je úsporný vzhledem k doméně D , právě když každý koncept ontologie O této domény je reprezentován nejvýše jednou v metamodelu tohoto jazyka.
- Úplnost - jazyk L je úplný vzhledem k doméně D , právě když každý koncept ontologie O této domény je reprezentován modelovým prvkem tohoto jazyka.

Reprezentace znalostí



Obrázek 8.3

K popisu struktury jazyka L lze přistupovat prostřednictvím specifikace konceptuálního modelu stojícího za jazykem, tj. pomocí vidění světa (paradigmatu) zakódovaného do grafických jazykových modelovacích prvků.

To bývá nazýváno *ontologickým metamodelem jazyka* nebo též *ontologií jazyka*. Např. Chenův E-R model odpovídá vidění světa zahrnujícímu tři typy věcí: entity, vztahy a atributy.

8.3.5 Ontologie, metamodel a konceptualizace

Doménová konceptualizace by měla být schopna specifikovat všechny možné abstrakce, vztahující se k této doméně.

Modelovací jazyk by měl mít schopnost vymezení všech možných specifikací, které lze s použitím tohoto jazyka sestavit, tj. všechny gramaticky správné specifikace jazyka.

V následující formalizaci je konceptualizace pojímána současně se zamýšlenou strukturou interpretace ve všech možných světech a metamodel jako reprezentovatelný logickými teoriemi.

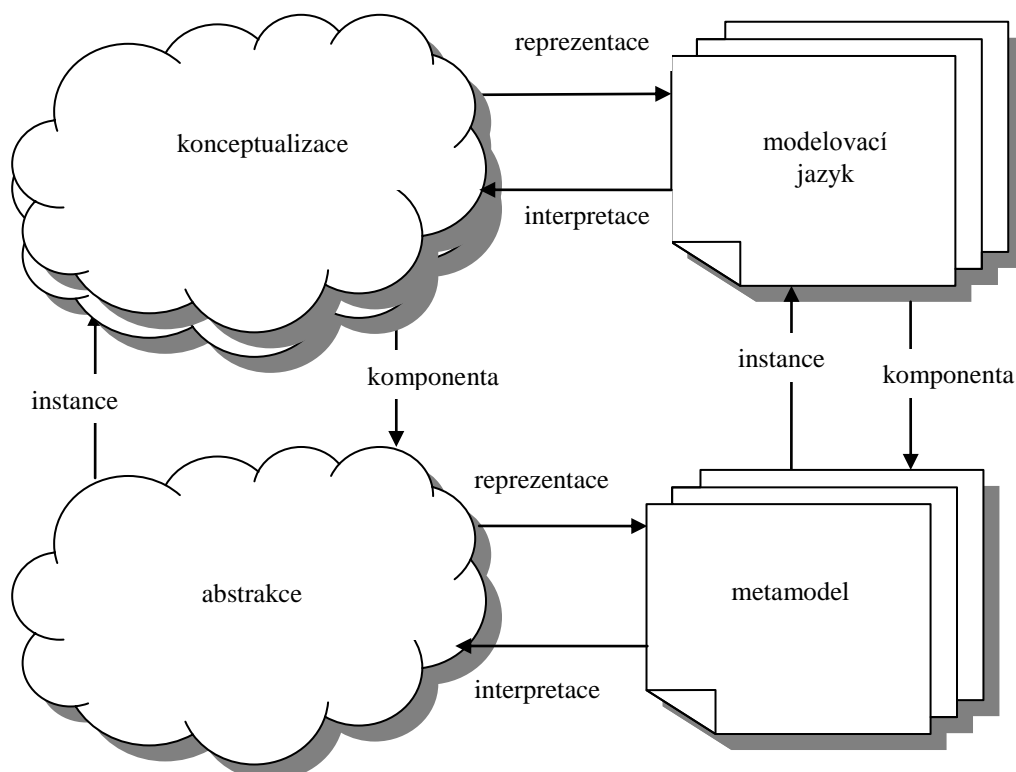
Definice 8.4 (konceptualizace)

Konceptualizace C je intenzionální struktura $\langle W, D, R \rangle$ taková, že W je (neprázdna) množina možných světů, D je doména individuí a R je množina n-árních relací (konceptů) uvažovaných v C . Prvky R jsou intenzionální



Reprezentace znalostí

(konceptuální) relace $p^n: W \longrightarrow p(D^n)$, tak, že každá n -ární relace je funkcí z možných světů do n -tic individuů v doméně.



Obrázek 8.4



Definice 8.5 (zamýšlené struktury interpretace)

Každému světu $w \in W$ konceptualizace C domény D odpovídá struktura $\langle D, R_w C \rangle$, $R_w C = \{ \rho(w) \mid \rho \in R \}$, která je jeho zamýšlenou strukturou interpretace.

V dalších definicích bude uvažován jazyk L , obsahující termíny pro reprezentaci všech konceptů z C .



Definice 8.6 (logického modelu)

Logický model jazyka L je určen dvojicí $\langle S, I \rangle$, přičemž S je struktura $\langle D, R \rangle$, kde D je doména individuů a R je množina extenzionálních relací, I je interpretační funkce $V \longrightarrow D \cup R$ přiřazující konstantám z V prvky domény D a predikátovým symbolům z V prvky (relace) z R . Logický model určuje extenzionální interpretaci jazyka L .



Definice 8.7 (intenzionální interpretace)

Intenzionální interpretace je definována dvojicí $\langle C, S \rangle$, kde $C = \langle W, D, R \rangle$ je konceptualizace a S je intenzionální interpretační funkce $V \longrightarrow D \cup R$ přiřazující konstantám z V prvky domény D a predikátovým symbolům z V prvky (relace) z R .

Reprezentace znalostí

Guarino nazývá tuto intenzionální strukturu ontologickou souhlasností jazyka L vzhledem ke konceptualizaci C . proto uvažujeme intenzionální relaci jako odpovídající reprezentační relaci.

Definice 8.8 (ontologické souhlasnosti)

Je-li dán logický jazyk L se slovníkem V , potom ontologická souhlasnost modelu $K = \langle C, S \rangle$ je určena kompatibilitou:



Model $\langle S, I \rangle$ jazyka L je se nazývá kompatibilním s K , jestliže platí:

- i) $S \in S_c$,
- ii) pro každou konstantu c $I(c) = S(c)$,
- iii) existuje svět w takový, že I zobrazuje každý predikátový symbol p do možné extenze z $S(p)$, tj. existuje konceptuální relace ρ a $\rho(w) = I(p)$.

Množina $Ik(L)$ modelů jazyka L , které jsou kompatibilní s K , se nazývá množinou zamýšlených modelů L vzhledem ke K .

Definice 8.9 (logického ztvárnění - renderingu)

Je-li dána specifikace X ve specifikačním jazyce L , pak logickým ztvárněním X je logická teorie T prvního řádu této specifikace.



Příklad 8.2

$W = \{w, w'\}$, $D = \{a, b, c\}$, $R = \{\text{osoba}, \text{otec}\}$, $\text{osoba}(w) = \{a, b, c\}$, $\text{otec}(w) = \{a\}$, $\text{osoba}(w') = \{a, b, c\}$, $\text{otec}(w') = \{a, b\}$. Tato konceptualizace akceptuje dvě možné skutečnosti reprezentovatelné strukturami $S_w C = \{\{a, b, c\}, \{a, b, c\}, \{a\}\}$, $S_{w'} C = \{\{a, b, c\}, \{a, b, c\}, \{a, b\}\}$.



Mějme jazyk L obsahující slova (termíny) osoba, otec. Metamodel jazyka L tedy obsahuje prvky ztvárnění 1. 2 a specifikaci 3:

1. $\exists x \text{ osoba}(x)$
2. $\exists x \text{ otec}(x)$
3. $\forall x (\text{otec}(x) \rightarrow \text{osoba}(x))$

Logické modely pak jsou zamýšlenými modely.

Doménovou konceptualizaci C lze vidět jako popis všech možných stavů světa v daném universu diskursu U . Necht' V je slovník, jehož termíny odpovídají intenzionálním relacím v C . Necht' X je konceptuální specifikace (konkrétní reprezentace) universa diskursu U v termínech V a necht' T_x je logický rendering X , tak že jeho axiomatizace omezuje možné interpretace prvků V . Potom X (a T_x) je ideální ontologií U vzhledem k C , právě když logické modely T_x popisují všechny stavy světa, které jsou akceptovány v C .

Obr. popisuje vztahy mezi jazykem, slovníkem, konceptualizací, ontologickou souhlasností a ontologií.

Reprezentace znalostí

Ontologie nemusí vždy být ideální a proto tato definice:

Ontologie je konceptuální specifikace popisující znalosti o doméně způsobem, který je nezávislý na epistemickém stavu a stavu světa. Navíc se snaží omezit možné interpretace slovníku jazyka tak, aby její logický model co nejlépe aproximoval množinu zamýšlených struktur konceptualizace příslušné domény.

8.4 Vývoj přístupů formální ontologie

Podoby ontologie mají v průběhu vývoje širokou variabilitu, a to od jednoduchých slovníků, přes kategorizovaně organizované thesaury, taxonomie, jejich termíny jsou již uspořádány stromově hierarchicky, až k ontologiím se schopností definovat nové koncepty a jejich vztahy.

Z hlediska šířky záběru jsou zde na jedné straně doménové ontologie, na druhé straně ontologie nejvyšší úrovně (top level).

Všechny ontologie obsahují část, kterou lze nazvat *terminologickou* složkou, která v podstatě definuje strukturu a termíny zájmového prostoru (domény), který má být ontologií reprezentován. Druhou složkou je znalostní báze obsahující definice (axiomy) obecně platné v rámci tohoto prostoru.

Pro formální specifikaci ontologií byla v rámci vývoje této oblasti vytvořena řada jazyků, která vyústila v současný standard, jímž je jazyk OWL.

8.4.1 Předchozí vývoj

Informační zdroje se srovnatelnou strukturou se již dříve vyskytovaly v různých odvětvích oborů informačních technik, proto je možno problematiku ontologií rozdělit na tři hlavní vývojová stádia, které lze chápat jako součásti vývoje tradičních vědních oborů:

1. *Terminologickou* či *lexikální* ontologii lze ztotožnit s pokročilými tezaury, používanými v knihovnictví a dalších oborech orientovaných na textové zdroje. Jejich charakteristickým rysem je ústřední role *termínů*, které již nejsou dále (formálně) definovány. Používané *relace* mají z velké části taxonomický charakter (vymezení vztahu obecnějšího a speciálnějšího termínu), vedle toho bývá vyjádřena synonymie, meronymie (vztah termínů označujících celek a jeho část) a další relace obecného charakteru. Ontologie se v tomto lingvistickém pojetí spíše podobají specializovaným slovníkům pro tyto oblasti vytvořených pojmů. Termíny bývají ztotožňovány, jak s entitami reality, tak i jejich lingvistickými protějšky. Prostředky reprezentace, vybudované na čistě lingvistické úrovni, pak nejsou jazykově nezávislé, neboť se vztahují ke slovesům a jménům výchozího přirozeného jazyka. Nejznámější terminologickou ontologií byl pravděpodobně *WordNet* ve své původní podobě.
2. *Informační ontologie* jsou rozvinutím *databázových* konceptuálních schémat. Jejich úkolem je být nadstavbou nad primárními

Reprezentace znalostí

(strukturovanými, např. relačně-databázovými) zdroji, pro které zabezpečují jak konceptuální abstrakci potřebnou pro pojmové dotazování, tak vyšší úroveň kontroly integrity, než je tomu u běžných nástrojů.

3. *Formální (znalostní) ontologie* navazuje na výzkum v oblasti reprezentace znalostí v rámci *umělé inteligence*. Ontologie je zde chápána jako *logická teorie*, a její vazba na reálné objekty (koncepty, jejich referenty) je oproti informačním ontologiím relativně volnější a obecnější. Třídy (koncepty), vlastnosti a vztahy jsou systematicky definovány prostřednictvím formálního jazyka. Na rozdíl od čistě lingvistického přístupu formulovaného v bodě 1 vychází z *filosofické lingvistiky*, která se zabývá významy základních *pojmu* a tím se dokáže odpoutat od závislosti na jejich reprezentantech v přirozených jazycích.

8.4.2 Reprezentace struktury domény třídami, stromy a svazy

Vývoj ontologií až k ontologii nesoucí přívlastek formální, lze sledovat též z hlediska matematických prostředků jejich reprezentace. Jedná se na počátku o prostou *klasifikaci* konceptů do tříd, později reprezentaci taxonomií stromy, až po reprezentaci ontologií konceptovými svazy.

Klasifikací se rozumí systematické shromažďování (zařazování) objektů do skupin nebo kategorií podle daného kritéria. Z matematického hlediska klasifikaci reprezentuje rozklad množiny objektů do tříd, nepřipouštějící jejich překrytí.

Pokud klasifikace konceptů do skupin a jejich podskupin má hierarchický charakter, reprezentovatelný *stromem*, hovoří se zpravidla o *taxonomii*. V případě taxonomie, která se rozvinula zejména v oblasti biologických věd, podmínkami, vytvářejícími taxony a jejich hierarchické uspořádání, měly původně být přírodními zákonitostmi určené vztahy mezi koncepty reprezentující biologické taxony (rody, druhy). Stromový charakter reprezentačního prostředku taxonomií zaručuje, že každý taxon reprezentovaný uzlem stromu (s výjimkou kořene) má jediného rodiče, který je jeho supertaxonem.

Rozvíjející se oblast reprezentace znalostí v pracích některých autorů přebírá pojem taxonomie i pro účel specifikace současné formální ontologie. Např. podle mínění příslušné pracovní skupiny W3C, která se ontologiemi zabývá a klade důraz na strojovou čitelnost definic významů konceptů v ontologiích, má ontologie být „strojově čitelnou množinou definic, které tvoří taxonomii tříd a podtříd a vztahů mezi nimi“. Formální ontologie však ve své terminologické části nevystačí se stromovou reprezentací svých multidimenzionálních hierarchických struktur konceptů. Formálním reprezentantem ontologie je zde *Galoisův svaz*.

8.5 Současné pojetí a přístupy formální ontologie

8.5.1 Trojí význam pojmu ontologie



Především je potřeba si uvědomit některé rozdíly v současném používání pojmu ontologie. Přinejmenším jsou zde tři následující významy:

1. Teorie bytí, která je spolu s teorií poznání tvoří náplň klasické filozofie.
2. Oblast v rámci reprezentace znalostí v umělé inteligenci, v níž se jedná o formalizovaný popis sémantiky objektů reprezentované zájmové domény pomocí slovníků všeobecně sdílených pojmů. Pro tuto oblast umělé inteligence se má ontologie přivlastek „formální“.
3. Ontologie jako produkt bodu 2. – terminologický slovník hierarchicky uspořádaných pojmů z dané problémové domény, doplněný souborem axiomů (pravidel) platných pro tuto doménu.

Vztah současné formální ontologie ke klasické teorii bytí z 1. bodu je dán skutečností, že ontologie v podstatě navazuje na *tradiční (filosofickou) ontologii* jako *teorii bytí* tím, že se zabývá

- formálními definicemi vlastností, rozdílů a vzájemných vztahů entit (konceptů) v rámci určitého referenčního systému - světa modelovaného formálními prostředky,
- stanovením univerzálních kategorií (konceptů), které jsou potřebné k tomu, aby se dalo obecně hovořit o tomto modelovaném světě.

Pokud jde o význam uvedený ve 2. bodě, objevovaly se rozdílné názory na to, co je ontologie a v souvislosti s tím řada pokusů definovat formální ontologii jako oblast informatiky. Někteří autoři připouštěli neformální označení ontologie jako hierarchie, zaměřené pouze na uspořádání výrazů přirozeného jazyka; jiní vyžadovali, aby ontologie byla *teorií* založenou na axiomech, zabývající se pojmy samotnými, nikoliv jejich jazykovými reprezentanty.

Různé názory na vymezení role ontologie ve výzkumném světě sice obohatily vědeckou debatu, ale závěry této debaty rozhodně nezastínily strategickou roli ontologií především v problematice návrhu informačních systémů.

8.5.2 Typy a hierarchie ontologií



Jedná se o obvyklé členění s množstvím variant navržených různými autory, z nichž nejdůležitějšími typy jsou:

- *Generické* (top-level) ontologie nejvyšší úrovně usilují o zachycení obecných zákonitostí, které platí napříč věcnými oblastmi, např. problematiky času, vzájemné pozice objektů (topologie), skladby objektů z částí (mereologie) apod.
- Někdy se ještě výslovně vyčleňují tzv. ontologie *vyšší úrovně* („upper-level“), které usilují o zachycení obecnějších pojmů a vztahů, jako základu struktury např. doménové ontologie. Ontologie typu *common-*

Reprezentace znalostí

sense („přirozeného rozumu“) mohou naopak obsahovat řadu velmi specifických, avšak relativně doménově-nezávislých znalostí, které lidé používají v každodenním životě. Nejznámějším příkladem je Cyc.

- *Doménové ontologie* jsou nejrozšířenějším typem. Jejich předmětem je vždy určitá specifická věcná oblast, vymezená širšími okolnostmi (např. celá problematika medicíny nebo fungování firmy), nebo užšími okolnostmi (problematika určité choroby, poskytování úvěru apod.). Příkladem doménové ontologie se širokým záběrem je lékařská ON9.
- Jako *problémově orientované ontologie* bývají označovány generické modely znalostních úloh a metod jejich řešení. Na rozdíl od ostatních ontologií, které zachycují znalosti o světě, se zaměřují na procesy odvozování. Mezi úlohy tradičně zachycené pomocí takových znalostních modelů patří např. diagnostika, zhodnocení, konfigurace, nebo plánování.
- *Aplikační ontologie* jsou nejvíce specifické. Jedná se o směs modelů převzatých a adaptovaných pro konkrétní aplikaci, zahrnující zpravidla doménovou i úlohovou část (a tím automaticky i generickou část)

Souhrnně je třeba rozlišit:

- *Specializované ontologie*: Týkají se problému stanovení relevantních esenciálních entit a jejich vzájemných vztahů v rámci konkrétního modelovaného světa. Jde o speciální ontologie *problémově orientované*, *aplikační* resp. *doménové*. Souhrnně se zpravidla hovoří o *doménových ontologiích*.
- *Obecné ontologie* (top-level, upper level, common sense): Řeší otázku, zdali jsou zde entity spolu s jejich vzájemnými vztahy, které jsou svou obecností přítomny v celých třídách modelovaných referenčních systémů a které by mohly tvořit ontologie (nej)vyšší úrovně. Obecná ontologie je víceméně aplikovatelná na každou speciální ontologii (s přidáním axiomů speciálních pro danou oblast).



8.5.3 Přístupy k doménové formální ontologii

Na základě specifikací požadavků na formální ontologii dané domény, které byly formulovány v předcházejících odstavcích, jsou zde diskutovány čtyři z přístupů k tvorbě ontologií, uváděné v literatuře. Z jejich vyhodnocení pak je odvozena i specifikace (definice) pojmu formální ontologie.

V poslední době lze sledovat skutečnost, že přístupy k problematice formální ontologie vycházejí z kombinací čtyř rozdílných koncepcí a s nimi spojených prostředků formální reprezentace.

- První koncepcí je koncepce vycházející z analytického prostředku - *formální konceptové analýzy* založené na matematické formální reprezentaci konceptovými (Galoisovými) svazy. Formální konceptová analýza je speciální analýzou mnohorozměrných dvouhodnotových dat popisujících problémovou doménu, definovanou v rámci určitého kontextu, jejímž výstupem je zpravidla graficky reprezentovaný



Reprezentace znalostí

konceptový svaz, poskytující základní hrubou představu o možné struktuře konceptů (ontologii) pro danou doménu.

- Druhá koncepce navazuje doslova na Gruberovu myšlenku, sdílenou řadou dalších specialistů, definující formální ontologii jako „formální specifikaci konceptualizace“. Pro matematickou reprezentaci koncepce formální konceptualizace rozpracoval A. Savinov myšlenku *konceptově orientovaného datového modelu*, reprezentovatelného stejně jako v prvním případě Galoisovým svazem. V případě dostatečně obecně pojaté modelované domény se vytvořený konceptově orientovaný datový model může stát součástí příslušné doménové ontologie.
- Třetí je formalizovaná filosoficko - lingvistická koncepce spolu s prostředky *formální ontologické analýzy* podle Guarina a Weltyho. Autoři v ní vycházejí ze skutečnosti, že tvorba ontologií v informatice poslední doby probíhá v těsné spolupráci informatiků, filosofů a specialistů z oblasti empirických věd. Ontologie se tak pojetím opět přibližuje ke klasické filosofické ontologii, neboť se stává disciplínou založenou na reprezentaci univerzálního a partikulárního, skutečně existujícího ve zkoumaném světě.
- Z hlediska tvorby ontologií v praxi je relativně rozšířen čtvrtý přístup, který lze nazvat "lexikografickým", neboť je založen na analýze "klíčivosti" slov vyskytujících se v relevantních textech dané problémové domény. Tento přístup navazuje na již zpracované metodologie analýzy přirozeného jazyka. Výstupem analýzy je soubor klíčových slov, více či méně korespondujících s pojmy zařaditelnými do terminologické části ontologie vybrané domény. V kombinaci s výše uvedenými přístupy, zejména s třetím z nich, tj. filosoficko-lingvistickým přístupem, je lexikografický přístup, navíc proveditelný automatizovaně, vhodným přístupem ke tvorbě doménových ontologií.

Tím, co mají všechny čtyři uvedené koncepce společného, je v podstatě pojetí formální ontologie v rámci *konceptově orientovaného* vidění modelované domény. Konceptová orientace dává možnost reprezentace vlastností a vztahů mezi koncepty v ontologiích poměrně jednoduchými logickými prostředky založenými na teorii svazů a uspořádaných množinách. Základní ideou je, že model musí být hierarchický a současně multidimenzionální.

Pro současnou formální ontologii je dále typické to, že kromě formálně specifikovaných významů termínů, explicitně vymezujících příslušnost ke konceptům, se zpravidla do ontologií zařazují další logické formule, které vyjadřují např. ekvivalenci/subsumpci konceptů, jejich disjunktnost, rozklad na subkoncepty apod. Nejčastěji bývají tyto formule označovány jako *axiomy*; které mají v ontologii samostatné postavení. V ontologických jazycích těsněji spjatých s určitým odvozovacím mechanismem se spíše označují jako *pravidla* a mají i tomu odpovídající sémantiku.

8.6 Formální konceptová analýza a formální ontologie

8.6.1 Formální konceptová analýza dat

Formální konceptová analýza, jejíž základ položili Bernhard Ganter a Rudolf Wille, je analýzou mnohorozměrných dat vybudovanou na základě vlastností Galoisových svazů, zde označovaných jako *konceptové svazy*. Metodika vychází z datové matice, jejímiž řádky jsou vektory reprezentující zkoumané objekty a jejímiž sloupce jsou hodnoty vektoru atributů pro jednotlivé objekty. Datová matice s řádky představujícími jednotlivé objekty analýzy a sloupce představujícími jejich atributy, která představuje *formální kontext*, reprezentuje binární relaci.

Smyslem analýzy je nalezení *přirozených shluků objektů* nebo *přirozených shluků atributů* na základě dat z formálního kontextu. Přirozeným shlukem objektů se rozumí množina všech objektů, které sdílejí nějakou množinu atributů. Podobně přirozeným shlukem atributů se rozumí množina všech atributů, kterými se vyznačuje nějaká množina objektů. *Koncept* je potom takovou množinou objektů s jejich atributy, která má současně charakter přirozeného shluku objektů i přirozeného shluku atributů. Atributy v rámci formální konceptové analýzy jsou analogií primitivních konceptů konceptově orientovaného modelu dat, charakterizovaného v předcházejících odstavcích.

8.6.2 Konceptový svaz

Algebraickou strukturu tvoří množina s jednou nebo více na ní definovanými operacemi. Svaz je algebraická struktura sestávající z množiny \mathbf{L} s částečným uspořádáním \leq a dvěma binárními operátory \cup a \cap . Jsou-li x, y prvky \mathbf{L} , $x \cap y$ se nazývá infimem x a y , $x \cup y$ supremem x a y .



Pro libovolné prvky x, y, z z \mathbf{L} splňují operátory \cup a \cap tyto axiomy :

- $x \cap y \leq x, x \cap y \leq y$
- je-li z prvek patřící \mathbf{L} , pro nějž platí $z \leq x$ i $z \leq y$, potom $z \leq x \cap y$
- $x \leq x \cup y, y \leq x \cup y$
- je-li z prvek patřící \mathbf{L} , pro nějž platí $x \leq z$ i $y \leq z$, potom $x \cup y \leq z$

Operátory \cup a \cap označením odpovídají množinovému sjednocení a průniku, neboť všechny podmnožiny univerzální množiny \mathbf{U} tvoří svaz s částečným uspořádáním \subset .

Ohraničený svaz je svaz s prvky \top (top) a \perp (bottom), kde pro všechny prvky a svazu platí $\perp \leq a \leq \top$. Všechny konečné svazy jsou ohraničené a též mnohé nekonečné svazy. Ve svazu podmnožin univerzální množiny \mathbf{U} je \top univerzální množina \mathbf{U} a \perp prázdná množina.

Pro svazové operátory platí podobně jako pro množiny tyto zákony:

- Idempotence: $x \cap x$ stejně jako $x \cup x$ je identické x .
- Komutativnost: $x \cap y$ je identické $y \cap x$, $x \cup y$ je identické $y \cup x$

Reprezentace znalostí

- Asociativnost: $x \cap (y \cap z)$ je identické $(x \cap y) \cap z$, $x \cup (y \cup z)$ je identické $(x \cup y) \cup z$
- Absorbce: $x \cap (x \cup y)$ je identické x , $x \cup (x \cap y)$ je identické x .

Uvedené zákony platí pro všechny svazy. V distributivních svazech platí navíc distributivní zákon. V komplementárních svazech je zaveden operátor komplementu a platí DeMorganovy zákony. Svaz podmnožin nějaké množiny je příkladem ohraničeného distributivního komplementárního svazu.

Všechny hierarchické struktury konceptů pro účely reprezentace znalostí jsou částečná uspořádání, řada z nich tvoří svazy.

8.6.3 Formální koncept v daném kontextu

Autory teorie, která zde bude uvedena, jsou Ganter a Wille. Podle této teorie je koncept definován svou intenzí a extenzí v rámci daného kontextu. V rámci problematiky *formální konceptové analýzy* se hovoří o typu formálních objektů a typu formálních atributů. Množina formálních objektů a formálních atributů spolu tvoří formální kontext reprezentovatelný binární relací.

Základním datovým formátem formální konceptové analýzy je relace nazývaná *formální kontext*, určující jakými atributy se jednotlivé dané objekty nebo podmnožiny uvažované množiny objektů vyznačují. Ústředním pojmem *formální teorie konceptů* je dualita extenze a intenze konceptu, zvaná *Galoisovo spojení*. Toto spojení znamená, že zvětšování jedné ze dvou množin způsobuje zmenšování druhé z nich. Např. dokumenty a termíny při získávání informací z webu spolu souvisí tak, že čím je vyšší počet dokumentů, tím nižší počet termínů sdílí a naopak, čím menší počet termínů je zadán, tím větší počet je obsahujících dokumentů systém nabízí.



Definice 8.10

Formální kontext je struktura určená trojicí $(\mathbf{O}, \mathbf{A}, \mathbf{I})$, kde \mathbf{O} je množina objektů, \mathbf{A} je množina atributů a $\mathbf{I} \subseteq \mathbf{O} \times \mathbf{A}$ je binární relace.

Relace \mathbf{I} má vyjadřovat, že nějaký objekt o má vlastnost danou atributem a . Dvojice $(o, a) \in \mathbf{I}$ tedy znamená, že objekt o se vyznačuje atributem a .

Každá množina objektů $\mathbf{C} \subseteq \mathbf{O}$ je spojena s podmnožinou množiny \mathbf{A} atributů a tvoří tak její *intent*

$$\text{int}(\mathbf{C}) = \{a \in \mathbf{A} \mid \forall o \in \mathbf{C}. (o, a) \in \mathbf{I}\}.$$

Analogicky každá množina atributů $\mathbf{B} \subseteq \mathbf{A}$ je spojena s množinou objektů a tvoří tak její *extent*

$$\text{ext}(\mathbf{B}) = \{o \in \mathbf{O} \mid \forall a \in \mathbf{B}. (o, a) \in \mathbf{I}\}.$$

Reprezentace znalostí

Stručně řečeno, intenze množiny C je vyjádřena jako její intent množinou společných atributů, extenze množiny B je jako její extant dána množinou objektů, které mají společné vlastnosti. To vede k pojmu formálního konceptu.

Definice 8.11

Formální koncept v kontextu $(\mathbf{O}, \mathbf{A}, \mathbf{I})$ je pár (\mathbf{X}, \mathbf{Y}) , kde $\mathbf{X} \subseteq \mathbf{O}$, $\mathbf{Y} \subseteq \mathbf{A}$ tak, že platí $\mathbf{X} = \text{ext}(\mathbf{Y})$ a $\mathbf{Y} = \text{int}(\mathbf{X})$.



Podle autorů tedy koncept C má svoji intenzionální část - formalizovanou množinu atributů \mathbf{Y} a extenzionální část určenou množinou objektů \mathbf{X} , které se danými atributy vyznačují. Obě části určení konceptu závisí na daném kontextu.

Poznámka:

Z hlediska formální logiky Σ lze koncept je pojímat jako trojici $(C, \text{int}(C), \text{ext}(C))$, kde C je jméno konceptu, $\text{int}(C)$ je teorie spojená s konceptem C nad logikou Σ , $\text{ext}(C)$ je model této teorie. Potom lze uvažovat třídy ekvivalence teorií jako intenze konceptů.

Na množině konceptů lze definovat částečné uspořádání jako

$$(\text{obj1}, \text{attr1}) \leq (\text{obj2}, \text{attr2}) \Leftrightarrow \text{obj1} \subseteq \text{obj2} \Leftrightarrow \text{attr1} \supseteq \text{attr2}$$

Z toho je vidět, že koncepty spolu s částečným uspořádáním tvoří svaz, který má nejmenší prvek $(\text{ext}(\mathbf{A}), \mathbf{A})$ a největší prvek $(\mathbf{O}, \text{int}(\mathbf{O}))$.

Pro $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}$ platí tato pravidla:

$$1) \mathbf{X}_1 \subseteq \mathbf{X}_2 \Rightarrow \text{int}(\mathbf{X}_2) \subseteq \text{int}(\mathbf{X}_1), \quad 1') \mathbf{Y}_1 \subseteq \mathbf{Y}_2 \Rightarrow \text{ext}(\mathbf{Y}_2) \subseteq \text{ext}(\mathbf{Y}_1),$$

(Větší množina objektů se vyznačuje menším počtem společných atributů)

$$2) \mathbf{X} \subseteq \text{ext}(\text{int}(\mathbf{X})) \text{ a } \text{int}(\mathbf{X}) \subseteq \text{int}(\text{ext}(\text{int}(\mathbf{X}))), \quad 2') \mathbf{Y} \subseteq \text{int}(\text{ext}(\mathbf{Y})) \text{ a } \text{ext}(\mathbf{Y}) \subseteq \text{ext}(\text{int}(\text{ext}(\mathbf{Y}))).$$

(Zpětně se množině všech atributů, kterými se vyznačují všechny prvky z \mathbf{X} , přiřadí množina všech objektů, a to nejen \mathbf{X} , které tyto atributy mají)

Struktura $\mathbf{S} = (\mathbf{O}, \mathbf{A}, \mathbf{I})$ je popsána pomocí tzv. hlavní věty konceptových svazů.

Věta 8.1

(1) Struktura $\mathbf{S} = (\mathbf{O}, \mathbf{A}, \mathbf{I})$ množiny konceptů \mathbf{J} tvoří úplný svaz vzhledem k relaci \leq , kde supremum a infimum jsou dána

$$\Lambda(\mathbf{X}_j, \mathbf{Y}_j) = (\cap \mathbf{X}_j, \text{ext}(\text{int}(\cup \mathbf{Y}_j))), \quad \mathbf{V}(\mathbf{X}_j, \mathbf{Y}_j) = (\text{int}(\text{ext}(\cup \mathbf{X}_j)), \cap \mathbf{Y}_j)$$

(2) Libovolný konceptový svaz $\mathbf{V} = (\mathbf{V}, \leq)$ je isomorfní s $\mathbf{S} = (\mathbf{O}, \mathbf{A}, \mathbf{I})$, právě když existuje zobrazení $\gamma = \mathbf{O} \rightarrow \mathbf{V}$, $\mu = \mathbf{A} \rightarrow \mathbf{V}$ takové, že

- (i) $\gamma(\mathbf{O})$ je \mathbf{V} -hustá ve \mathbf{V} , $\mu(\mathbf{A})$ je Λ -hustá ve \mathbf{V} ,
- (ii) $\gamma(\mathbf{X}) \leq \mu(\mathbf{Y})$, právě když $(\mathbf{X}, \mathbf{Y}) \in \mathbf{I}$.

Reprezentace znalostí

8.6.4 Uzávěrové vlastnosti konceptů

Vlastnost uzávěru je další zajímavou vlastností Galoisova spojení. Je-li výchozí množinou množina formálních objektů, lze stanovit množinu formálních atributů, které tyto objekty sdílejí. K této množině atributů lze pak podle pravidla 2 předcházejícího odstavce přidat i další formální objekty, které se jimi vyznačují. V tom smyslu lze dospět k „uzavřenosti“, neboť další rozšíření množiny atributů sdílených rozšířenou množinou objektů už není možné. Podobně lze dosáhnout uzavřenosti, je-li výchozí množinou množina atributů, k ní nalezena množina objektů je sdílejících a doplnění podle pravidla 2' z předcházejícího odstavce výchozí množiny atributů o ty atributy, které jsou rovněž množinou objektů sdíleny. Výsledná množina objektů, sdílejících uzavřeně nějakou množinu atributů, se nazývá formální koncept, přičemž její objekty tvoří extenzi konceptu, její atributy tvoří intenzi konceptu. V rámci daného kontextu jsou koncepty jednoznačně určeny.



Definice 8.12

Operátor φ je vzhledem k množině \mathbf{X} (\mathbf{Y}) *uzávěrový operátor*, jestliže splňuje tři základní axiomy:

1. $\mathbf{X} \subseteq \mathbf{X} \cdot \varphi$,
2. $\mathbf{X} \subseteq \mathbf{Y} \Rightarrow \mathbf{X} \cdot \varphi \subseteq \mathbf{Y} \cdot \varphi$
3. $\mathbf{X} \cdot \varphi \cdot \varphi \subseteq \mathbf{X} \cdot \varphi$.



Definice 8.13

Množina $\mathbf{X} \subseteq \mathbf{U}$ (podmnožina universa diskursu \mathbf{U}) je *uzavřená*, je-li $\mathbf{X} \cdot \varphi = \mathbf{X}$.

Dvojice (\mathbf{U}, φ) se nazývá *uzávěrový prostor*.

Následující vlastnosti uzávěrů vyplývají přímo z axiomů 1-3 (**Definice**).

- a) je-li \mathbf{C} uzavřená a $\mathbf{X} \subseteq \mathbf{C} \subseteq \mathbf{X} \cdot \varphi$, potom $\mathbf{C} = \mathbf{X} \cdot \varphi$ (tj. $\mathbf{X} \cdot \varphi$ je nejmenší uzavřená množina obsahující \mathbf{X}).
- b) $\mathbf{X} \cdot \varphi \cup \mathbf{Y} \cdot \varphi \subseteq (\mathbf{X} \cup \mathbf{Y}) \cdot \varphi$
- c) $(\mathbf{X} \cap \mathbf{Y}) \cdot \varphi \subseteq \mathbf{X} \cdot \varphi \cap \mathbf{Y} \cdot \varphi$
- d) $\mathbf{X} \cdot \varphi \cap \mathbf{Y} \cdot \varphi$ je uzavřená.
- e) $\emptyset \cdot \varphi = \emptyset$
- f) $\mathbf{U} \cdot \varphi = \mathbf{U}$

Reprezentace znalostí

Definice 8.14

Nechť \mathbf{R} je daný kontext. *Uzávěr* $\varphi_{\mathbf{R}}$ objektu \underline{o} množiny \mathbf{O} vzhledem k \mathbf{R} je maximální množina objektů, která sdílí stejné atributy jako \underline{o} . *Uzávěr* $\psi_{\mathbf{R}}$ atributu \underline{a} množiny \mathbf{A} vzhledem k \mathbf{R} je maximální množina atributů, kterou sdílejí stejné objekty jako \underline{a} .

Uzávěr podmnožiny $O \subseteq \mathbf{O}$ objektů o v daném kontextu \mathbf{R} je

$$\varphi_{\mathbf{R}} = \{o \mid \forall a, a \in \mathbf{A} \Rightarrow (o, a) \in \mathbf{R}\},$$

podobně je *uzávěr podmnožiny* $A \subseteq \mathbf{A}$ atributů a v daném kontextu \mathbf{R}

$$\psi_{\mathbf{R}} = \{a \mid \forall o, o \in \mathbf{O} \Rightarrow (o, a) \in \mathbf{R}\}.$$

Ganter a Wille ukázali, že uzávěry $\varphi_{\mathbf{R}}$ a $\psi_{\mathbf{R}}$ tvoří Galoisovo spojení, reprezentující koncepty, jsou isomorfní a lze je reprezentovat svazem $\mathcal{L}_{\mathbf{R}}$ uzavřených množin, které jsou částečně uspořádány inkluzí. Návěštím každého uzlu je pár uzavřených množin, spojených Galoisovým spojením, např. v kontextu \mathbf{R} z příkladu 8.3 následujícího odstavce jde o dvojice jako $(abg, 123)$, reprezentující koncepty v daném kontextu \mathbf{R} . Množina abg je uzavřená v \mathbf{A} , množina 123 je uzavřená v \mathbf{O} .

8.6.5 Konceptové grafy FCA

Koncept je možno v daném kontextu pojímat (viz předcházející odstavec) jako základní prvek myšlení sestávající ze dvou částí – extenze a intenze. Extenze pokrývá všechny objekty náležející konceptu a intenze shromažďuje všechny atributy, kterými se koncept vyznačuje.

Výhodou této koncepce je možnost vizualizace konceptů a jejich vzájemných vztahů subkonceptů a superkonceptů využívající vlastností konceptového svazu.

V konceptovém svazu hrají zvláštní roli koncepty \top - univerzální a \perp - absurdní koncept. Extenzí \top konceptu jsou všechny objekty daného kontextu, zatímco jeho intenze je zpravidla prázdná. Relace subkoncept – superkoncept je tranzitivní. Jestliže formální koncept má určitý formální atribut, pak tento atribut je děděn všemi jeho subkoncepty.

Pro snadné pochopení způsobu grafické reprezentace konceptů a jejich atributů, tak jak je zvykem ve formální konceptové analýze, poslouží následující dva příklady z literatury.

Při tvorbě minimálního svazu byla v následujících příkladech aplikována metoda formální konceptové analýzy FCA (Ganter – Wille), vycházející z definice 8.4 konceptu v daném kontextu.

Reprezentace znalostí

Příklad 8.3

Příklad podle J. L. Pfalze a Ch. M. Taylora:

Je dán kontext R (tabulka 8.2), tj. relace mezi množinou atributů A a množinou objektů O (tabulka 8.3).

Tabulka 8.2

R	a	b	c	d	e	f	g	h	i
1	×	×					×		
2	×	×					×	×	
3	×	×	×				×	×	
4	×		×				×	×	×
5	×	×		×		×			
6	×	×	×	×		×			
7	×		×	×	×				
8	×		×	×		×			

Tabulka 8.3 a)

	objekty
1	pijavice
2	cejn
3	žába
4	pes
5	plevel
6	třtina
7	bob
8	kukuřice

b)

	atributy
a	potřebuje k životu vodu
b	žije ve vodě
c	žije na souši
d	potřebuje k tvorbě potravy chlorofyl
e	klíčí dvěma malými lístky
f	klíčí jedním malým lístkem
g	může se pohybovat
h	má páteř
i	kojí potomky

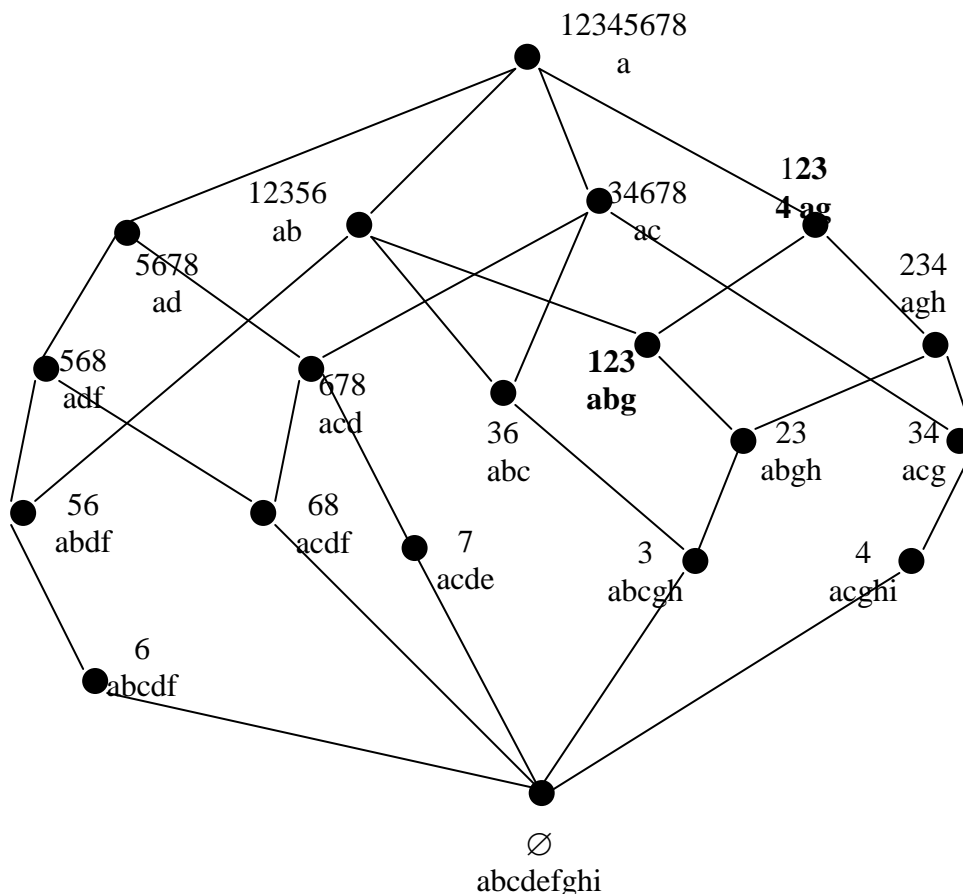
Svaz L_R na obrázku 8.11 je vizuálním modelem obsahu kontextu R . V obrázku je svaz orientovaný vzhledem k O , kde univerzální koncept (top-koncept) T , jehož extent zahrnuje všechny objekty, které zde sdílejí atribut a, je supremem svazu, absurdní koncept (bottom-koncept) \perp je prázdný koncept vyznačující se všemi atributy abcdefghi, je infimem svazu. Konceptu s extemem 123, který se vyznačuje atributy abg (potřebující vodu k životu, žijící ve vodě a mající možnost pohybu) určujícími jeho intent lze např. přiřadit jméno „vodní živočich“.

Reprezentace znalostí

Každý koncept začleněný do grafu představuje uzel se dvěma částmi návěští: V horní části návěští je uveden extent konceptu, tj. objekty, které koncept reprezentuje, a v dolní části návěští jeho intent, tj. množina atributů, kterými se koncept vyznačuje.

Pro orientaci ve výsledném konceptovém svazu orientovaném vzhledem k množině objektů O zde platí pravidlo:

Koncept C_1 je subkonceptem konceptu C_2 , právě když existuje cesta grafem směrem dolů od uzlu konceptu C_2 k uzlu konceptu C_1 .



Obrázek 8.5: Konceptový graf

8.6.6 Dolování znalostí z konceptových svazů

Pojmem *dolování znalostí z dat* se zpravidla rozumí vyhledávání podmnožin množiny sledovaných objektů, reprezentovaných zpravidla prvky relace, které jsou si navzájem podobnější než je tomu v případě objektů z navzájem různých podmnožin. Nalezený rozklad množiny objektů na podmnožiny - *shluky*, resp. hierarchické struktury shluků, pak představuje nějakou hypotézu o *klasifikační struktuře* objektů reprezentovaných daty, která je-li akceptovatelná, může představovat novou znalost o zkoumané množině objektů. Znalost klasifikační struktury dat pak umožňuje, aby nový objekt byl k některému z výsledných shluků přiřazen s tím předpokladem, že sdílí i další interpretující charakteristické vlastnosti objektů tohoto shluku.

Reprezentace znalostí

V jiných případech je dolování znalostí z dat zaměřeno na zkoumání vzájemných závislostí atributů, charakterizujících zkoumané objekty, za účelem zjištění, zdali neexistují nějaké vzájemně závislé skupiny atributů, představující faktory, výrazně se podílející na variabilitě v rámci množiny objektů reprezentovaných danými daty. Nalezené faktory zde mohou představovat novou znalost, která je dobře využitelná pro zjednodušení sledování objektů uvažovaného typu na základě menšího počtu vysoce reprezentativních komponovaných atributů.

Dolování znalostí z dat, postavené zde na algebraických a logických principech analýzy pojmů (konceptů) a jejich vzájemných vztahů, speciálně na konceptových svazech a deskripční logice, se pokouší sáhnout k samotným principům kategorizace objektů lidského myšlení, tj. pojmů (konceptů).

Paradigma těchto *vědeckých znalostí* lze charakterizovat následujícím symbolickým zápisem:

$$(\forall o \in \mathbf{O})(P(o) \rightarrow Q(o)),$$

kde \mathbf{O} je zájmové universum diskursu, sestávající z objektů (o), P a Q jsou unární predikáty, které jsou konjunkcemi /disjunkcemi atributů objektů.

8.7 Ontologie vyšších úrovní

Důležitým tématem formální ontologie je vývoj ontologií vyšší, resp. nejvyšší úrovně, tj. specifikací, (teorií) obecných kategorií, jakými jsou čas, prostor, inherence, instanciace, identita, proces, událost, atribut, relace apod.

Společným rysem níže uvedených návrhů ontologií vyšší úrovně, zejména návrhu obecné ontologie Russela a Norviga ontologie GOL, top-level ontologie J.F. Sowy, Geneserethova a Fikesova systému výměny znalostí KIF, Lenatovy encyklopedie CYC, tvůrců znalostní báze SUMO i návrhů dalších autorů je cíl, poskytnout tvůrcům doménových ontologií široký společný klasifikační základ, vybavený souborem axiomů, do něhož lze umístit specializovaný soubor konceptů, charakterizujících danou problémovou doménu, a pomocí dalších klasifikačních specifikací nevelkého rozsahu vytvořit pak ontologii této domény.

8.7.1 Obecná ontologie vyšší úrovně Russela a Norviga

Podle autorů odlišují obecnou ontologii od ontologií pro speciální účely, resp. doménových ontologií dvě hlavní vlastnosti:

- *Obecná ontologie by měla být víceméně aplikovatelná na každou speciální ontologii (s přidáním axiomů speciálních pro danou oblast).*
- Pro různé oblasti by různé typy znalostí o nich měly být unifikovány, aby bylo možno odvozovat z navzájem různých oblastí současně.

Reprezentace znalostí

8.7.2 Obecný ontologický jazyk GOL (General Ontological Language)

Podle autorů GOL musí ontologie vyšší úrovně splňovat tato kritéria:

- Musí obsahovat nejméně tři ontologické kategorie: individuály, univerzály a množiny.
- Musí obsahovat rozšíření o další základní relace, především ty, které se týkají času, prostoru a obrysu, stejně jako topologické relace jako ohraničení, souvislost apod.

Principy, na nichž je jazyk GOL vybudován, se neliší od obecných principů, k nimž dospěla tradiční filosofická ontologie.

8.7.3 Formát výměny znalostí KIF (Knowledge Interchange Format)

KIF (Knowledge Interchange Format) je formální jazyk výměny informací mezi různými programy, vytvářený a doplňovaný řadou programátorů. Lze jej využívat i k modelování na nižších doménových úrovních: po načtení znalostní báze KIF aplikačním programem proběhne konverze do jeho vnitřního formátu, vhodného pro další využití, a naopak, je-li potřeba komunikace s jiným programem, děje se tak prostřednictvím konverze jeho vnitřního formátu do KIF.

Podle Geneseretha se KIF vyznačuje těmito hlavními rysy:

- Nejobecnější kategorií entit je objekt.
- Objekty mohou být konkrétní nebo abstraktní (soud, číslo 2...).
- Objekty mohou být jednoduché nebo komponované.
- V ontologii je základním rozdílem rozdíl mezi individuálem a množinou.
- Relace a funkce jsou zavedeny jako množiny konečných seznamů. Obvykle tomu v GOL odpovídá extenzionální relace.
- KIF nepředkládá ontologicky základní relace jako inherence, part-whole...
- GOL lze pokládat za rozšíření KIF, KIF je množinově-teoretickou částí GOL.

8.7.4 Sowův návrh ontologie nejvyšší úrovně

Sowův návrh hierarchie kategorií reprezentace znalostí na nejvyšší úrovni vychází z primitivních kategorií

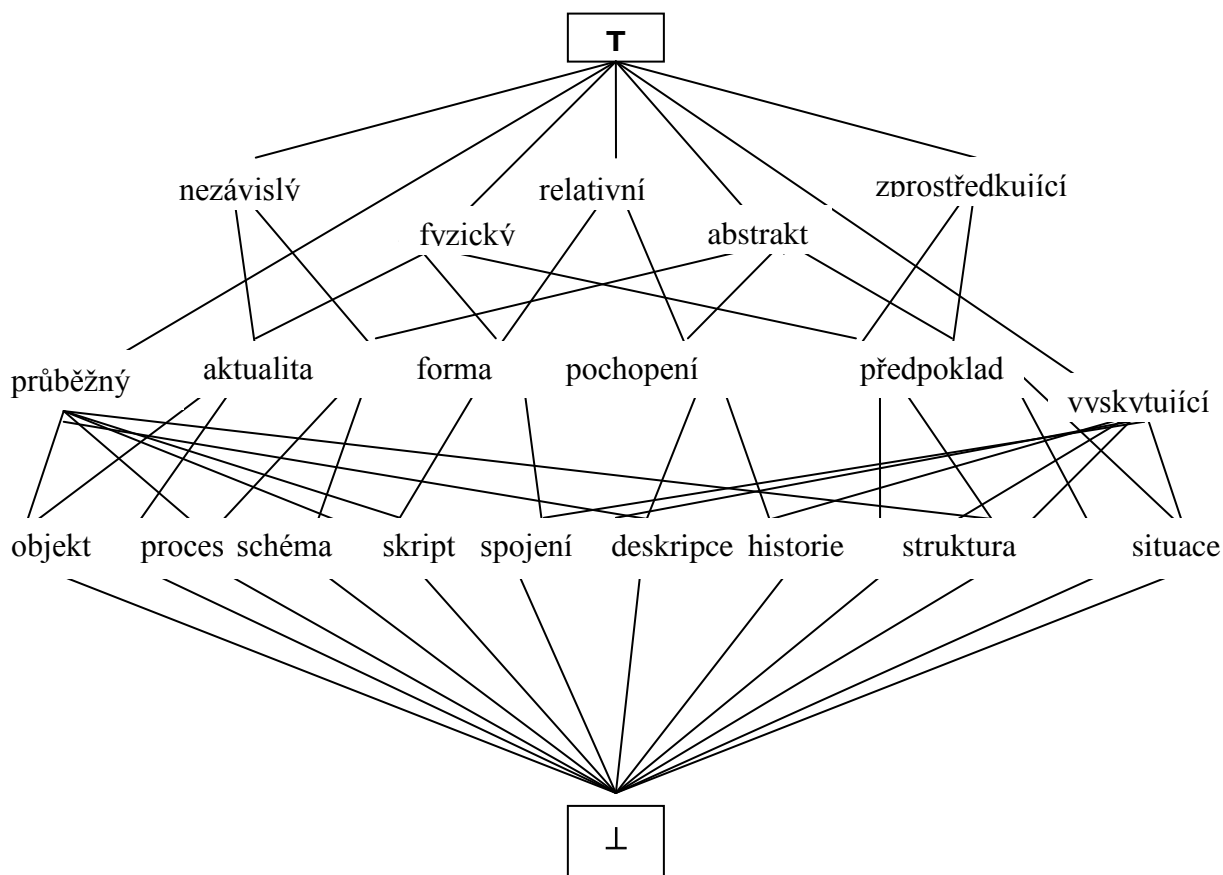
- nezávislý, relativní, zprostředkovaný,
- fyzický, abstraktní
- průběžný, vyskytující se

Reprezentace znalostí

Tabulka (8.4) zobrazuje matici kombinatorických struktur kategorií podle J.W.Sowy.

Tabulka 8.4

	fyzický		abstraktní	
	průběžný	vyskytující se	průběžný	vyskytující se
nezávislý	objekt	proces	schéma	skript
relativní	spojení	participace	deskripce	historie
zprostředkovaný	struktura	situace	důvod	účel



Obrázek 8.6

V obr. 8.12 je znázorněno 9 základních kategorií – typů konceptů: **T**, **⊥**, Independent (nezávislý), Relative (relativní), Mediating (zprostředkovatelný), Physical (fyzický), Abstract (abstraktní), Continuant (průběžný), Occurent (objevující se). Každý subtype je definován pomocí infima (nejvyššího společného subtypu reprezentovaného \cap) dvou supertypů, jejichž vlastnosti (axiómy) dědí. Např. koncept forma je nezávislý a abstraktní.

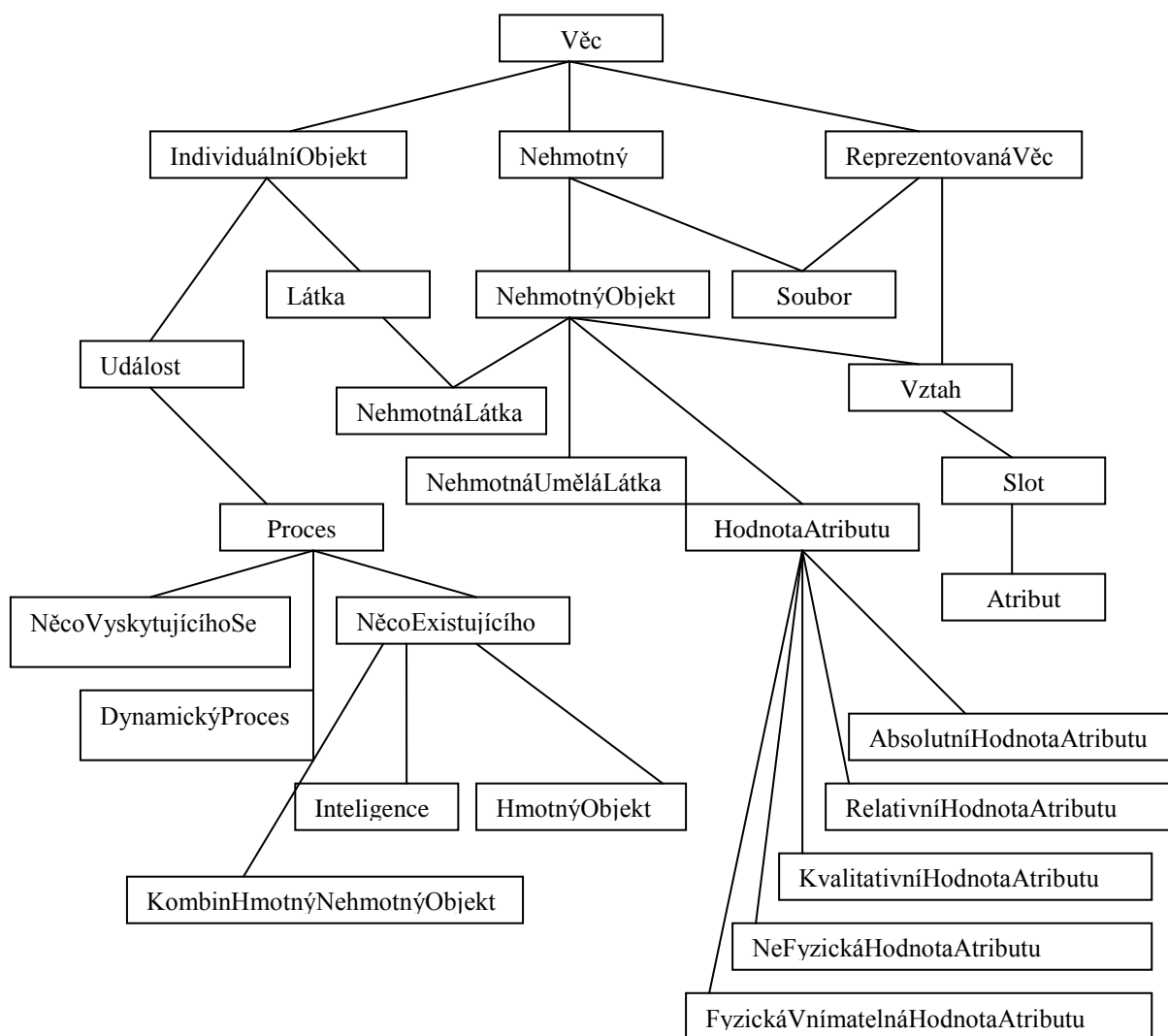
Reprezentace znalostí

8.7.5 CYC ontologie

Jedním z prvních pokusů zachytit ve velkém rozsahu znalosti o světě je (i v současnosti stále “živý”) projekt (*Cyc*), jehož název je odvozen ze slova “enCYClopedia”. Projekt společnosti *CycCorp* usiluje o shromáždění *všeobecně sdílených* („common sense“) znalostí, které by ve znalostních systémech fungovaly komplementárně ke znalostem expertním, a zabraňovaly absurdnímu chování. Svou podstatou i zaměřením aplikací jde o projekt, který jednoznačně patří do oblasti ontologií, ačkoliv jeho tvůrci tento termín příliš často nepoužívají. Hovoří namísto toho o „mikroteoriích“ složených z dílčích tvrzení. Jako formální reprezentaci *Cyc* využívá svůj vlastní jazyk *CycL* (se základní notací převzatou z funkcionálního jazyka LISP), který má plnou vyjadřovací sílu predikátového kalkulu, a kombinuje jej s prvky rámcových jazyků.

Cyc systém obsahuje 100 000 typů konceptů použitých v pravidlech a faktech obsažených ve znalostní bázi. Systém obsahuje axiomy a definice konceptů. Volně přístupná verze má název *OpenCyc*. *Cyc* ontologie na obr. 8.13 je acyklický graf, v němž některé uzly mají více rodičů, které podporují vícenásobnou dědičnost.

Reprezentace znalostí



Obrázek 8.7

8.7.6 WordNet

George Miller se spoluautory z University v Princetonu vytvořil WordNet jako hierarchii 166 000 termínů.

WordNet je rozsáhlá lexikální databáze anglického jazyka. Podstatná jména, slovesa, přídavná jména, příslovce jsou seskupena do skupin (synsety) kognitivních synonym, příslušejících rozdílným konceptům. Synsety jsou propojeny konceptuálně-sémantickými a lexikálními vztahy.

Dotazovací systém WordNet Search sděluje (symbolem "S"), zdali odpověď obsahuje sémantické (synset-ové) vztahy k dotazovanému termínu nebo (v případě symbolu "W"), jde-li o přidružené lexikální vztahy.

Např. odpovědí na dotaz "informatics" je:

S: (n) information science, informatics, information processing, IP (the sciences concerned with gathering, manipulating, storing, retrieving, and classifying recorded information).

Reprezentace znalostí

8.7.7 SUMO ontologie

Cílem tvůrců této ontologie nejvyšší úrovně bylo vytvoření základu doménové ontologie střední úrovně, která zvýší interoperabilitu, zlepší možnosti vyhledávání, automatizované dedukce a zpracování přirozeného jazyka. Sestává z přibližně 4000 tvrzení zahrnujících 800 pravidel a 1000 konceptů. Ontologie SUMO je napsána v SUO-KIF jazyce logiky prvního řádu, ale byla převedena do různých formátů reprezentace.

Ontologie je podobná slovníkům, ale s větším zaměřením na detaily a strukturu domény. Sestává z množiny konceptů, jejich vztahů a axiomů, resp. pravidel, vztahujících se k zájmové doméně.

Ontologie nejvyšší úrovně se specializuje na koncepty charakteru generického, abstraktního, filosofického, proto je dostatečně obecná k zahrnutí širokých doménových oblastí.

Koncepty specifické pro jednotlivé zájmové domény nejsou součástí ontologie nejvyšší úrovně, ta ale skýtá vhodnou strukturu, na základě níž mohou být doménové ontologie sestrojeny. MILO (mid-level) ontologie slouží jako most mezi abstrakcí na nejvyšší úrovni a detailní pozicí doménové ontologie.

SUMO je jedinou formální ontologií, která má zmapovaný celý slovník WordNet.

8.8 Některé systémy na podporu tvorby ontologií

Tvorba doménových ontologií probíhá zpravidla jako iterativní proces tímto postupem:

- Shromáždění znalostí o předmětné doméně
- Definování konceptuální struktury domény s ohledem na zamýšlenou hloubku detailní reprezentace.
- Vyladění výsledku - strukturování podle míry obecnosti, odstranění nekonsistencí, soulad s ontologií nejvyšší úrovně.

8.8.1 OntoClean

OntoClean byl vytvořen na základě principů pro vytváření a používání ontologií vyšších úrovní, pro analýzu jádra domény ontologie, jejich změn a vývoje. Metodologie OntoClean je založena na obecných ontologických pojmech koncipovaných z filozofické ontologie, zvláště z toho, co se nyní nazývá „analytická metafyzika“, a je dostatečně obecná pro použití v kterémkoli typu ontologií, nezávisle na konkrétní doméně.

Reprezentace znalostí

Jak prohlašují autoři: „Metodologie OntoClean poskytuje formální, adekvátní a přímou cestu k vysvětlení některých nejčastějších nedorozumění v konceptuálním modelování s ohledem na taxonomické nebo jiné relace.“

Systém pro realizaci metodologie OntoClean pomocí dotazů a odpovědí pomáhá modelujícímu ve výběru vhodných meta-vlastností jednotlivých vlastností na základě znalostní báze, která je vstupem do systému.. Dotaz jako „Je tato vlastnost nositelkou identity?“ Odpověď je ano/ne/nevím. Po odpovědi nevím následuje užší specifikace otázky. Např. „Jsou instance vlastnosti počítatelné?“

Po vstupu těchto vlastností se kontroluje konsistence informací a odvodí se některé důsledky, např. $\sim R \rightarrow -R$. Systém je implementován v CLASSIC systému dedukce v deskripční logice.

Používané pojmy a jejich formální reprezentace

Tabulka 8.5

<i>Reprezentovaný pojem</i>	<i>Formální reprezentant</i>
koncept	třída, unární relace
objekt	individuum, instance třídy
vztah/vlastnost	binární relace
funkční vztah/vlastnost	funkce jedné proměnné
hierarchie konceptů	taxonomie, konceptový svaz

8.8.2 Ontolingua

V určitém kontrastu k „uzavřené“ koncepci Cyc byla iniciativa vedená na začátku 90. let. T. Gruberem a jeho spolupracovníky ze stanfordské Knowledge System Laboratory (KSL). Jejich cílem bylo zkonstruovat dostatečně silný a současně přehledný jazyk, který by umožňoval sdílet ontologie v rámci specializovaných komunit používajících vzájemně nekompatibilní znalostní systémy.

Jazyk označený jako *Ontolingua* je koncipován jako nadstavba jazyka *KIF* (Knowledge Interchange Format), což je varianta predikátového kalkulu využívající (stejně jako CycL a mnohé další jazyky) syntaxe LISP. Základními konstruktory jazyka Ontolingua jsou definice tříd, relací a funkcí, přičemž vymežující podmínky pro příslušnost instancí jsou vyjádřeny právě v KIF.

Ontolingua byla od začátku zamýšlena jako „mezi-jazyk“, primárně určený k výměně ontologických informací mezi systémy, které interně používají svou vlastní (zpravidla omezenější, avšak výpočtově efektivnější) reprezentaci. Z toho vyplývají i omezené možnosti odvozování přímo v tomto jazyce. Na druhé straně ovšem Ontolingua, vzhledem ke své rozšířenosti, plnila po řadu

Reprezentace znalostí

let i roli jazyka „první volby“ pro vytváření ontologií nezávisle na konkrétním znalostním systému. Základním nástrojem pro zpřístupňování ontologií v jazyce Ontolingua se stal tzv. *Ontolingua server*, umístěný právě na KSL.

8.8.3 ONIONS, OnTopic

ONIONS (Ontological Integration Of Naive Sources) je soubor metod pro rozšíření neformálních dat z terminologických zdrojů na status datových typů formálních ontologií. OnTopic se týká vytváření závislostí mezi hierarchiemi témat a ontologiemi. Obsahuje metody pro odvozování základů ontologie, které popisují dané téma, a metody pro vytváření „aktivních“ témat, definovaných vzhledem k závislosti na jedinci, konceptu nebo vztahu v ontologii.

Ontologie vytvořené prostředky metodologie ONIONS podporují:

- *Formální rozšíření* terminologických systémů, která spočívá v tom, že klasifikace a definice termů je dostupná v obecném, expresivním formálním jazyce.
- *Konceptuální jasnost* terminologických systémů: (místní) definice termů jsou dostupné, přestože je zdroje explicitně neobsahují.
- *Možnost konceptuálního rozšíření* terminologických systémů: klasifikace a definice termů jsou vysvětleny tak, že mohou být přidány do každé ontologické knihovny, která má podmnožinu tvořenou odpovídajícími všeobecnými ontologiemi.
- *Ontologická srovnatelnost* s předtím již existujícími ontologickými knihovnami týkajícími se různých oblastí, které byly z velké části použity.

8.8.4 OCML

Omezené schopnost odvozování v jazyce Ontolingua motivovaly E. Mottu z Open University ve Velké Británii k návrhu jazyka, který by výrazněji usnadňoval přímý vývoj programových aplikací, aniž by bylo nutno model překládat do jiného jazyka. Vývoj *Operational Conceptual Modelling Language* (OCML) proto těsně propojil s tvorbou jeho interpretu, implementovaného v prostředí CommonLISP. Základem interpretu jsou algoritmy pro Prologovské dokazování a dědění v hierarchii tříd. Třídy a jejich atributy jsou ovšem důsledně chápány jako unární resp. binární relace, takže primární (vnitřní) reprezentací jsou Hornovy klauzule.

“Deklarativní” část OCML je prakticky shodná s jazykem Ontolingua. Vedle toho je však podporována řada konstruktorů, převzatých z procedurálních jazyků a expertních systémů (podmínky, cykly, produkční pravidla), a navíc je zde i snadné volání LISP-ovských funkcí. V OCML je proto možné napsat libovolnou aplikaci, která s ontologiemi nemusí ani mít nic společného.

OCML si v komunitě znalostního inženýrství získal značný respekt. Je ovšem skutečností, že se (zčásti asi i vinou vázanosti na LISP jako implementačním prostředím) příliš nerozšířil.

Reprezentace znalostí

8.8.5 OKBC a XOL

V roce 1993 začal vznikat návrh aplikačního rozhraní, které by umožňovalo otevřenou komunikaci mezi rámcově–orientovanými znalostními systémy – *Open Knowledge Base Connectivity* (OKBC). Protokol specifikuje metodu předávání jak konstruktorů znalostních bází (jako jsou třídy, individua nebo sloty), tak i volání operací nad jejich konstrukty. Soubor konstruktorů podporovaných OKBC (tzv. znalostní model) vychází z analýzy řady existujících znalostních systémů a sémanticky do značné míry odpovídá souboru konstruktorů ontologických jazyků, jako je Ontolingua.

Právě proto se jednoduchá verze OKBC, nazvaná OKBC-Lite, stala v roce 1999 výchozím bodem pro návrh dalšího ontologického jazyka: *eXtensible Ontology Language* (XOL). Jeho motivací byla potřeba *bioinformatické* komunity sdílet struktury znalostí o genovém výzkumu, které žádný ze stávajících jazyků plně nevyhovoval. Hlavní inovací na rozdíl od předcházejících jazyků bylo zakotvení v syntaxi XML, která umožnila efektivní využití řady obecných nástrojů, které již existovaly pro tento značkovací jazyk. Na rozdíl od *XML Schematu* XOL zavádí pouze jedinou, generickou definici typu dokumentu (DTD), která umožňuje definovat strukturu tříd, jejich sloty (pouze binární), obory hodnot slotů apod.



Kontrolní otázky a úkoly

1. Co jsou to metadata?
2. V čem spočívá smysl přechodu datového modelu ke konceptualizaci?
3. Jaký je trojí význam pojmu ontologie?
4. Jaké typy ontologií znáte?
5. Setkali jste se již s nějakou takovou ontologií?



Korespondenční úkol

Pomocí formální konceptové analýzy zachyťte vlastnosti (např. koníčky) Vaší nejbližší rodiny. Výsledkem analýzy pak bude konceptový graf.



Shrnutí

Ontologický přístup je v současné *formální ontologii* přístupem formalizované sémantiky, který nezahrnuje relativní interpretační "svévolnost". Při ontologickém přístupu je zpravidla jazyk logiky prvního řádu nebo jeho vhodná modifikace zvolen za prostředek reprezentace znalostí tak, aby účelově posloužil určitému konkrétnímu modelování nějaké zájmové *domény*, tj. určitým způsobem vymezeného modelovaného světa. Je tedy třeba, aby jeho modely byly výhradně těmi strukturami, které jsou odvozeny speciálně z vlastností a vztahů entit vymezené domény modelované reality, odpovídajících právě této zamýšlené interpretaci. Starost o obecné vlastnosti

Reprezentace znalostí

ostatních modelů - interpretačních struktur, které nemají nic společného s interpretacemi zamýšlenými, zde tedy zcela odpadá.

Lze říci, že výzkum v problematice reprezentace znalostí (ve své epistemologické části) se donedávna více soustřeďoval k principům usuzování než k povaze modelovaného světa. Toto inferenční hledisko bylo pro epistemologii typické. Studium povahy znalosti bylo omezeno na formální přístupy. Ontologický výzkum naproti tomu v podstatě spočívá ve studiu organizace a povahy světa nezávislého na formě vyjádření znalostí o něm. Znalostní báze v případě ontologického přístupu odráží pevně vymezený modelovaný svět, neboť jen v případě takové přísné vazby znalostní báze na *zamýšlenou interpretaci* lze každý koncept popsat stanovenými vlastnostmi, umožňujícími následné stanovení jeho významu. Ontologický přístup nevyklučuje úvahy o dalších možných světech, ale na rozdíl od přístupu tradičně logického jsou tyto světy svázány s původním světem zamýšlené interpretace relací vzájemných možných přístupů - *relací přístupnosti*.

9 MODELY A JAZYKY PRO SÉMANTICKÝ WEB

V této kapitole se dozvíte:

- o obecném datovém modelu RDF,
- ideu a formu reprezentace v RDF,
- co je to zdroj a jak ho identifikujeme,
- o formální sémantice RDF a to ve vztahu k jazyku prvního řádu,
- ontologické jazyky webu OWL.

Po jejím prostudování byste měli být schopni:

- definovat a používat abstraktní syntax modelu RDF,
- vytvářet tvrzení RDF grafu pomocí trojice,
- specifikovat třídy a vlastností v RDFS,
- zhodnotit přínos RDF/OWL modelování a jejich výhody a nevýhody.

Klíčová slova této kapitoly:

RDF model, syntax, zdroj, URI, RDF trojice, koncepty, třídy, vlastnosti, RDFS, slovník, OWL.

Doba potřebná ke studiu: 7 hodin



Průvodce studiem

Kapitola, jejíž název nese Modely a jazyky pro sémantický web je orientována na několik témat. Prvním tématem bude RDF jazyk, který je doporučený konsorciem W3C jako základ pro reprezentaci sémantického webu. RDF je vlastně drup datového modelu, tak jak jej známe z databází. Této části, prosím, věnujte největší pozornost.

Dále se bude zabývat i dalšími ontologickými jazyky, například když budeme navazovat na RDF model ve spojení s deskripční logikou, nám pak vzniká OWL jazyk.

9.1 Obecný datový model RDF

Z hlediska transformace syntaktického webu na sémantický web jde o přechod od strukturálně propojených datových zdrojů na bázi klíčových slov k jejich propojení na bázi významových asociací.

Potřeba strukturovat data na webu podle jejich významu v rámci zaměření na sémantický web obrací pozornost k datovému modelování na konceptuální

Reprezentace znalostí

úrovni návrhů znalostních bází nad webovými zdroji. Jako prostředek datového modelování vlastností a vztahů mezi datovými zdroji byl vyvinut model RDF (Resource Description Framework) jako základní rámec komplexního pojetí reprezentace znalostí na webu.

Propojení webových zdrojů na bázi významů dokumentů, které jsou jejich obsahem, realizuje datový model RDF prostřednictvím speciálních informací (metadat) o datových zdrojích. Základem této reprezentace jsou popisy vlastností webových zdrojů prostřednictvím metadat vycházejících ze stanovených konkrétních vlastností (atributů) zdrojů.

9.1.1 Vlastnosti (atributy) zdrojů

Datový model RDF popisuje zdroje a vztahy mezi nimi v termínech *pojmenovaných vlastností* a jejich *hodnot*, které mohou reprezentovat, jak samotné vlastnosti (atributy) zdrojů, tak též vztahy mezi zdroji. Z tohoto hlediska je třeba model RDF vidět jako konceptuální datový model, podobně jako je tomu např. u E-R modelu.

Model vychází z myšlenky známé již z asociativních (sémantických) sítí (viz kapitolu 2), spočívající v jednoduchých tvrzeních, týkajících se vztahů mezi objekty (zde zdroji), a to ve formě vektorů (trojic)

„subjekt - má vlastnost - objekt“.

Původním záměrem bylo vytvořit model využívající *metadat* (název, autor, datum vytvoření apod.) všeobecně považovaných za základní informace o webových zdrojích s využitím jazyka XML (eXtensible Markup Language), ale postupně se model stal obecnou metodou konceptuálního modelování znalostí obsažených v dokumentech webu, bez ohledu na formát jejich syntaxe. Reprezentace metadat z webu se totiž ve své podstatě neliší od reprezentace dat obecně. RDF lze proto používat jako obecný konceptuální rámec pro výměnu dat, a to nejen těch, která jsou dostupná na webu. Metadata neboli strukturovaná data o datech, na nichž je RDF model postaven, zlepšují významně možnost přístupu k informacím.

RDF jako datový model je pro sémantický web základním reprezentačním prostředkem, podobně jako tomu bylo a dosud je v případě jazyka HTML a syntaktického webu.



9.1.2 Idea a formy reprezentace v RDF

Základem koncepce RDF jako konceptuálního modelu jsou dva principy, které představují v konceptuálním modelování nové ideje. Jde o

- způsob popisu konceptů, jejich vlastností a vzájemných vztahů v termínech *vlastností* (atributů) a jejich *hodnot* (tj. způsobem typickým pro konceptově orientované datové modelování) a

Reprezentace znalostí

- identifikaci prvků modelu, kterými jsou zde webové zdroje, pomocí *uniformních identifikátorů zdrojů URI* (Uniform Resource Identifier).



RDF má *abstraktní syntax* vycházející z reprezentace jednoduchých tvrzení prostřednictvím *RDF trojic* (viz následující odstavec) *textovou formou* i *grafickou formou*, a tomu odpovídající formální, na teorii modelů založenou, sémantiku. Grafický způsob reprezentace trojic činí model srozumitelným i těm, kteří nemají odpovídající základní teoretické znalosti problematiky.

RDF model je popsán v dokumentech konsorcia W3C, z nichž první (RDF) se týká abstraktní syntaxe, druhý (RDFS) se týká schémat (slovníků, ontologií) další pak sémantické stránky modelu.

Vedle abstraktní syntaxe RDF je definována též *RDF XML syntax* modelu RDF s vlastní sémantikou pro účely strojového porozumění a přenosu informací obsažených v modelu. V následujících odstavcích budou pouze pro získání představy o syntaxi jazyka RDF XML k některým RDF grafům vytvořeny též jejich RDF XML popisy.

9.2 Abstraktní syntax modelu RDF

9.2.1 Uniformní identifikace zdrojů



Definice 9.1

Zdroj je obecně entita, jejíž vlastnosti a vztahy k jiným entitám lze popsat RDF modelem.

Zdrojem může být

- webová stránka nebo její část,
- soubor webových stránek,
- literál (prostý nebo typovaný),
- element XML ve zdrojovém dokumentu,
- jiný objekt, dostupný přímo pomocí webu – kniha, obrázek,...

Model RDF pracuje uniformně pouze s *URI-odkazy (URI-ref)* identifikujícími, resp. odkazujícími na zdroje. *URL* je speciálním případem URI-odkazu.



URI-odkaz je UNICODE řetězcem, který

- neobsahuje vymezené řídicí znaky,
- může obsahovat volitelný fragmentový identifikátor (odděleným #).

Reprezentace znalostí

9.2.2 RDF - trojice a graf

Základním jednotkou modelu RDF je *tvrzení* (statement) o zdrojích tvořící vektor (trojici), tj. tvrzení, že

"subjekt" "má vlastnost" určenou "objektem",

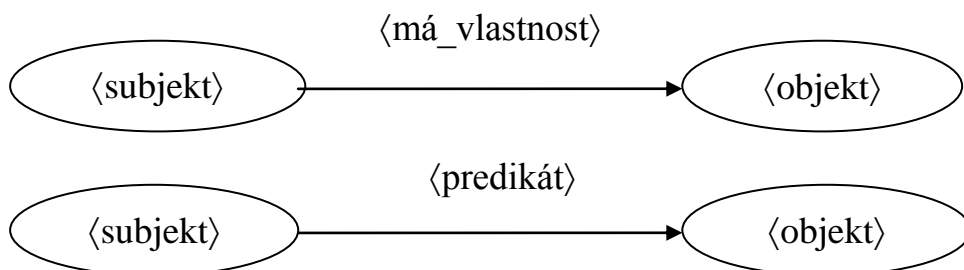
reprezentovatelné graficky (**Chyba! Nenalezen zdroj odkazů.**) nebo nebo v podobě textu jako

⟨subjekt⟩ ⟨má_vlastnost⟩ ⟨objekt⟩,

resp.

⟨subjekt⟩ ⟨predikát⟩ ⟨objekt⟩.

Grafická reprezentace:



Obrázek 9.1

Definice 9.2 (RDF-trojice a RDF-grafu)

RDF-trojice je uspořádaný ohodnocený graf reprezentující *tvrzení*

<subjekt> má vlastnost *<predikát>* s hodnotou *<objekt>*.

RDF-graf je množina RDF-trojic.

RDF-trojice má tři komponenty:

- zdroj *<subjekt>* je graficky zpravidla* reprezentovaný elipsou jako ohodnocený uzel
- spojený se zdrojem *<objekt>* graficky zpravidla* reprezentovaným rovněž elipsou jako ohodnocený uzel
- pomocí ohodnocené hrany *<predikátu>*.

Návěštmi ohodnocení uzlů (subjektů a objektů) mohou být literály (textové řetězce), *URI-odkazy* nebo návěští mohou být prázdná. Návěští hran (predikáty) tvoří vždy *URI-odkaz*.

RDF adaptér může přiřadit jednoznačný identifikátor zdroje *URI-odkaz* těm zdrojům, které jej nemají. Potom cokoliv, co má *URI-odkaz*, je považováno za zdroj náležející webu.

Reprezentace znalostí

9.2.3 Způsoby reprezentace RDF grafu

Následující příklady ilustrují grafickou a textovou reprezentaci elementárního grafu pomocí trojice v rámci abstraktní syntaxe ve srovnání s reprezentací v RDF XML.

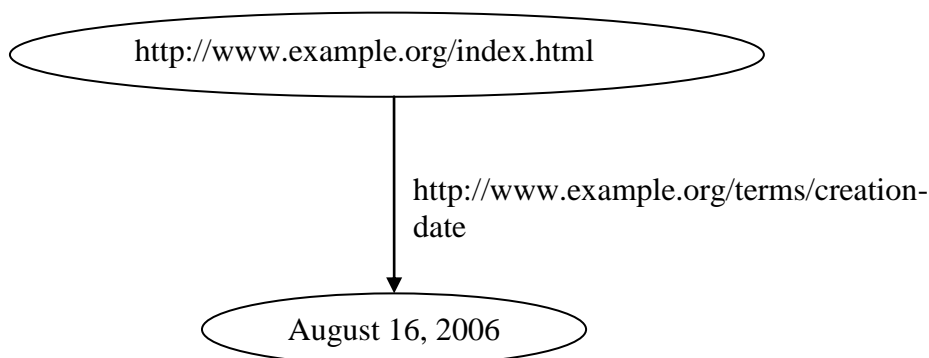


Příklad 9.1

Příklad jednoduchého tvrzení (inspirovaný příkladem publikovaným v dokumentu RDF Primer) o datu vytvoření webové stránky reprezentované:

- grafickými prostředky - vektorem trojice RDF,
- trojici v textové podobě abstraktní syntaxe RDF a
- RDF XML kódem, používajícím *jmenné prostory* (xmlns).

a)



b)

`<http://www.example.org/index.html> <http://www.example.org/terms/creation-date>
<"August 16, 2006">`

c)

V RDF XML jazyce:

```
1. <?xml version="1.0"?>
2. <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3.     xmlns:exterms="http://www.example.org/terms/">
4.     <rdf:Description rdf:about="http://www.example.org/index.html/">
```

Obrázek 9.2

a) Elementární graf (trojice) na obr. 9.2 má jako návěští subjektu URI-odkaz <http://www.example.org/index.html>, který sděluje v lidsky i strojově čitelné formě, že se tvrzení v grafu týká jako subjektu příslušné webové stránky, literál "August 16, 2006" (adaptovaný jako zdroj) sděluje, co je objektem

Reprezentace znalostí

reprezentovaného tvrzení, a návěští predikátu, kterým je URI-odkaz <http://www.example.org/terms/creation-date>, odkazuje na specifikaci vlastnosti "datum vytvoření".

b) Textová reprezentace abstraktního modelu daného tvrzení obsahuje URI-odkazy subjektu a predikátu, uvedené v lomených závorkách, literál specifikující objekt daného tvrzení je uveden v uvozovkách.

c) Řádek 2. RDF XML kódu začíná `<rdf:RDF`, což znamená, že následující XML kód (končící `</rdf:RDF>`) reprezentuje RDF-graf. Následující prefix `xmlns:` definuje jmenný prostor prvku `rdf:RDF`. Podobně je definován jmenný prostor pro prefix `exterms:` v řádku 3.

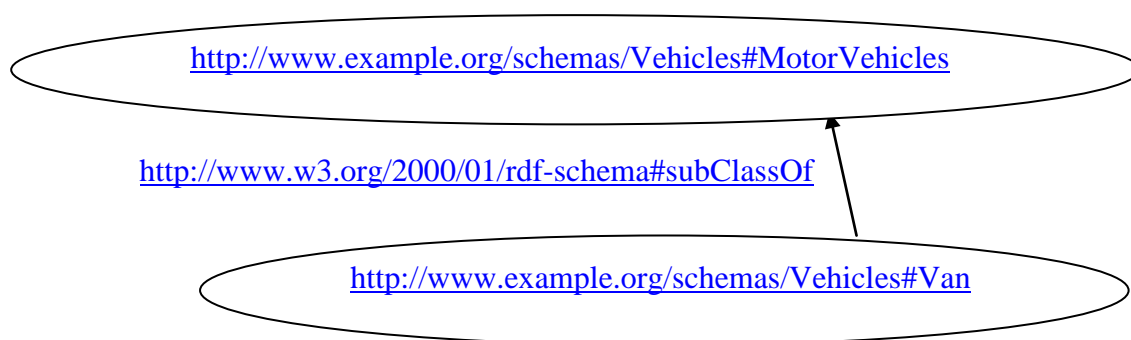
K vyjádření o reprezentovaném RDF tvrzení se používá popisu `description-about` týkající se subjektu tohoto tvrzení, predikátu `exterms:creation-date` a objektu, kterým je literál August 16, 2006.

Příklad 9.2



Tvrzení, že nákladní auta tvoří podtřídu třídy motorových vozidel (převzato z W3C dokumentu RDF Primer), na obr. 9.3 má jako návěští subjektu URI-odkaz <http://www.example.org/schemas/Vehicles#Van>, který sděluje v lidsky i strojově čitelné formě, jakého subjektu se reprezentované tvrzení týká (zde třídy nákladních aut),

URI-odkaz <http://www.example.org/schemas/Vehicles#MotorVehicles>, sděluje, co je objektem reprezentovaného tvrzení (zde třída motorových vozidel), a URI-odkaz <http://www.w3.org/2000/01/rdf-schema#subClassOf>, jako návěští predikátu odkazuje na specifikaci vlastnosti "být podtřídou" (`subClassOf`) třídy motorových vozidel.



Obrázek 9.3

Protože každý zdroj je identifikován pomocí uniformního identifikátoru zdroje, který je pro zdroj jedinečný, je možné spojování RDF grafů na základě shody zdrojů, které se v nich vyskytují (možno i distribuovaně). Některá vývojová prostředí slučují grafy automaticky.

9.3 Systém konceptů v RDF modelu

Požadování strojově čitelných sémantických informací o obsahu dokumentů vyžaduje, aby tyto informace byly uvažovány ve všeobecně sdílených základních pojmech (konceptech), postavených na jejich vlastnostech a vzájemných vztazích (které jsou obecně též koncepty). "Svět" (doména, referenční systém), který má být formálními prostředky modelován, musí být viděn prostřednictvím abstrakcí vyčleněných relevantních *konceptů*, jejich vlastností (atributů) a vzájemných vztahů. Způsob vidění modelovaného světa je zde *konceptově orientovaný*.



RDF model rozlišuje tři typy konceptu, identifikovatelné pomocí URI-odkazu:

1. Základní koncepty modelu RDF na nejvyšší úrovni obecnosti, kterými jsou
 - *zdroj* s identifikátorem `rdf:Resource`,
 - *vlastnost* s identifikátorem `rdf:Property` a
 - *tvrzení* s identifikátorem `rdf:Statement`.
2. Koncepty definující *schéma* (pro definování terminologických slovníků).
3. Koncepty *utilit* (užitečné pro některé aplikace).

Zdroj je základním prvkem a zároveň konceptem modelu. K definování konceptů uživatelem využívá RDF(S) konstruktoru třídy s názvem `Class` pro ty koncepty, které v objektovém pojetí odpovídají pojmu třída. Pro definování konceptů vyjadřujících vlastnosti má uživatel k dispozici konstruktor s názvem `Property`. To odpovídá konceptově orientovanému pojmání modelovaného světa, neboť jak třída, tak i vlastnost jsou v pojetí RDF modelu je též základními koncepty. Tvrzení, schémata a utility jsou komponovanými prvky modelu, ale jejich URI-identifikace umožňuje, aby byly používány jako prvky elementární. RDF formální systém se tímto „trikem“ stává systémem logiky prvního řádu.

9.4 Specifikace tříd a vlastností v RDFS

Jak bylo zřejmé z předcházejících odstavců, model RDF samotný poskytuje pouze prostředky vytváření jednoduchých tvrzení o vlastnostech a vztazích mezi zdroji. Pro popis významů zdrojů, jejich vlastností a vzájemných vztahů neposkytuje žádný mechanismus. To je úkolem RDF Schema (RDFS). RDF model tedy má dvě komponenty, z nichž první - RDF se týká syntaxe, druhý - RDFS se týká adres slovníků (schémat), a dalších prostředků určujících významovou stránku modelu.



Příklad 9.3

Popis webové stránky v RDF XML pomocí vlastností definovaných v Dublin Core dokumentu.

Reprezentace znalostí

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1">
  <rdf:Description> rdf:about="http://www.dlib.org">
    <dc:title>D-Lib program = Research in Digital Libraries</dc:title>
    <dc:description> D-Lib program supports the community of people with research
    interest...</dc:description>
    <dc:publisher>Corporation for National Research Initiatives</dc:publisher>
    <dc:date>1995-01-07</dc:date>
    <dc:subject>
      <rdf:Bag>
        <rdf:li>Research; statistical methods</rdf:li>
        <rdf:li>Education, research, related topics</rdf:li>
        <rdf:li>Library use studies</rdf:li>
      </rdf:Bag>
    </dc:subject>
    <dc:type>World Wide Web Home Page</dc:type>
    <dc:format>text/html</dc:format>
    <dc:language>en</dc:language>
  </rdf:Description>
</rdf:RDF>
```

RDFS nespecifikuje přímo slovník deskriptivních prvků zdrojů, jako např. "autor". Místo toho specifikuje mechanismus potřebný k definování těchto prvků, definuje použitelné třídy zdrojů, jejich omezení, resp. kombinace tříd nebo vlastností. Tímto mechanismem je v RDFS *typový systém tříd*.

RDF(S) datový model na rozdíl od databázových systémů nemá žádný „vestavěný“ soubor datových typů, pouze poskytuje způsob explicitního určení, jakého typu má prvek být.

Prostředky, kterými se definují zdroje, resp. koncepty (třídy) v RDFS jsou

- Typování - individuum náleží určité třídě
- Podtřídy - instance jedné třídy je zároveň instancí jiné třídy.

9.5 Formální sémantika RDF

9.5.1 RDF jazyk prvního řádu

Deklarativní jazyky, a RDF mezi ně patří, jsou určeny k tomu, aby vytvářely formalizovaná platná tvrzení o modelovaném světě (doméně). RDF je deklarativní jazyk k vytváření tvrzení o dané doméně pro účely přístupu a

Reprezentace znalostí

dalšího využití v rámci webu. Při tvorbě všech těchto tvrzení se používají výhradně přesně specifikované významy slov, určené svou pozicí v příslušných RDFS-slovnících.

Sémantika formálního jazyka RDF musí mít obdobně jako v případech deklarativních jazyků logiky prvního řádu definována jasná pravidla postupu stanovení pravdivostních hodnot základních a komponovaných tvrzení jazyka. V případě RDF(S) jazyka jde stejně jako v případě logiky prvního řádu o denotační sémantiku, založenou na přístupu teorie modelů. Zvláštností je zde skutečnost, že všechny jazykové symboly jsou URI identifikátory webových zdrojů. RDF model dané tematické nebo problémové domény, charakterizované relevantními webovými zdroji, má tvořit základní rámec dalšího odvozování, dotazování a tvorby aplikací.

Modelovaný svět (doména) určuje po významové stránce *zamýšlenou interpretaci* formálního jazyka RDF, poskytuje o tomto světě základní fakta a platná pravidla. Jako u každého jiného formálního jazyka mohou existovat vedle zamýšlené interpretace odvozené z modelovaného světa i další možné světy interpretující zvolený formální jazyk. Tvrzení o vlastnostech a vzájemných vztazích participujících zdrojů tvoří tedy *znalostní bázi* domény, nad níž se zmíněné úkony provádějí. Je proto zapotřebí, aby tato znalostní báze měla charakter logiky prvního řádu. RDF dociluje splnění tohoto požadavku tím, že jeho sémantika ignoruje významy samotných URI-odkazů, tj. uvažuje je pouze jako *jména*, která v logice prvního řádu hrají úlohu termů, nikoliv predikátů, v attributech daných RDF-predikátů. Tato abstrakce umožňuje řadit RDF k jazykům prvního řádu. URI-odkaz má vždy stejný význam, bez ohledu na změny vnitřního obsahu jeho odpovídajícího zdroje.

9.5.2 Interpretace slovníku RDF(S)

Interpretace formálního jazyka je vždy vázána na jeho syntax, pravidla interpretace proto korespondují se syntaxí RDF(S). Interpretační pravidla, podobně jako v případě logiky prvního řádu, poskytují předpisy, jak stanovit pravdivostní hodnoty základních a dalších v rámci modelu komponovaných a odvozených tvrzení.

Je třeba především definovat, jakých jazykových prvků se interpretace jazyka týká. Následující definice vymezuje pojem *jména* jako základního prvku slovníku.



Definice 9.3 (jména a slovníku)

Jméno je URI-odkaz nebo literál (kterému lze též přiřadit URI-odkaz). Množina jmen tvoří *slovník* jazyka RDF.

RDF(S) je jazyk deklarativního charakteru, v němž jména pojmenovávají entity (objekty) universa diskursu modelovaného světa. Těmi jsou v rámci modelu webové zdroje.

Reprezentace znalostí

Každá trojice reprezentuje samostatné tvrzení, jemuž je možno přiřadit pravdivostní hodnotu. Cílem interpretace je zde, stejně jako v případě logického jazyka prvního řádu, stanovení pravdivostních hodnot základních tvrzení RDF grafu, tj. RDF trojic a na jejich základě pak stanovení pravdivostní hodnoty celého grafu.

Pravidla interpretace jazyka RDF proto musí zahrnovat:

- způsob stanovení *denotátů jmen* ze slovníku modelu RDF,
- postup přiřazení *pravdivostní hodnoty* *bázové trojici*,
- způsob stanovení *pravdivostní hodnoty* *bázového RDF grafu* na základě trojic, kterými je graf tvořen.

Pravidla interpretace jazyka modelu RDF je možno stejně jako u jazyků formální logiky aplikovat až tehdy, je-li pro tuto interpretaci dána její *struktura*. Ta obdobně jako v logice prvního řádu s její denotační sémantikou určuje ke každému jménu v pozici subjektu, predikátu nebo objektu tvrzení, vyskytujícímu se ve slovníku \mathbf{V} , jako jeho denotát interpretující prvek universa diskursu. Ke každému tvrzení grafu pak přiřazuje interpretující relaci, umožňující stanovení jeho pravdivostní hodnoty.

Interpretace zaměřená na některý speciální podslovník (RDF slovník, RDFS slovník) bývá nazývána interpretací RDF slovníku, resp. interpretací RDFS slovníku. Není-li interpretace zaměřena na některý z podslovníků, hovoří se obecněji o jednoduché interpretaci slovníku.

Interpretace \mathbf{I} slovníku \mathbf{V} jazyka RDF (RDFS) je definována podle následující definicí.

Definice 9.4

Jednoduchá interpretace \mathbf{I} slovníku $\text{rdf}\mathbf{V}/\text{rdfs}\mathbf{V}$ jazyka RDF/RDFS je dána následující *strukturou*:



1. Neprázdnou množinou \mathbf{IR} *zdrojů*, zvanou doménou, resp. *universem diskursu* interpretace \mathbf{I} .
2. Množinou \mathbf{IP} *vlastností* v interpretaci \mathbf{I} .
3. Zobrazením \mathbf{IEXT} z \mathbf{IP} do podmnožiny kartézského součinu $\mathbf{IR} \times \mathbf{IR}$, tj. množinou dvojic $\langle x, y \rangle$, $x, y \in \mathbf{IR}$.
4. Zobrazení \mathbf{IS} z množiny URI-odkazů slovníku \mathbf{V} do sjednocení $\mathbf{IR} \cup \mathbf{IP}$.
5. Zobrazení \mathbf{IL} z množiny typovaných literálů z \mathbf{V} do \mathbf{IR} .
6. Zvláštní podmnožinou \mathbf{LV} množiny \mathbf{IR} , nazývanou množinou literálních hodnot, která obsahuje všechny prosté literály z \mathbf{V} .

Poznámky k definici:

Množina zdrojů \mathbf{IR} bodu 1 podle definice též zahrnuje zdroje odpovídající literálům typovaným (bod 5) i prostým (bod 6), zatímco množina zdrojů,

Reprezentace znalostí

náležících vlastnostem (bod 2), musí být uvažována odděleně, zde jako **IP**, neboť se jedná z hlediska logiky o množinu predikátů, jejímž každému prvku x je třeba přiřadit jako denotát pomocí zobrazení $IEXT(x)$ binární relaci nad $IR \times IR$, určující pro každou vlastnost x její extenzi, umožňující následně každému tvrzení modelu přiřadit jeho pravdivostní hodnotu. Je třeba si uvědomit, že typovaný literál (bod 5) je určen dvěma jmény: sebou samotným a odpovídajícím URI-odkazem na jeho typ. V bodě 4 je stanoveno, že každému zdroji, včetně těch, které určují vlastnosti, je třeba pomocí zobrazení **IS** přiřadit jeho URI-odkaz.

9.5.3 Prázdné uzly jako existenční proměnné

Zobrazení popsaná v definici (def. 9.5) tvoří *strukturu interpretace I*, která umožňuje stanovení pravdivostní hodnoty každého tvrzení grafu a následně pak stanovení pravdivostní hodnoty grafu. Přiřazení pravdivostní hodnoty danému tvrzení však je možné pouze tehdy, tvoří-li trojici reprezentující tvrzení jména, tj. URI-odkazy na příslušné konkrétní zdroje. Existenční tvrzení o tom, že existuje jméno zdroje, které dosazeno na místo prázdného uzlu činí tvrzení pravdivým, je třeba vhodnou substitucí převést na bázové tvrzení týkající se konkrétního substituovaného zdroje.

Substitucí splňujícího jména za prázdné návěští se tvrzení obsahující prázdný uzel, které mělo charakter existenčně kvantifikovaného tvrzení, stane pravdivým tvrzením.

Např. (viz obr. 9.4) z původního tvrzení, že

„zdroj s identifikátorem urn:ISBN:... má nějakou kapitolu“,

se substitucí jména, tj. identifikátoru zdroje, obsahujícího např. konkrétní kapitolu o deskripční logice dotyčné knihy, stane pravdivé tvrzení

„zdroj s identifikátorem urn:ISBN: má kapitolu Deskripční logika (s daným jménem).“

V této souvislosti je třeba definovat pojmy instancí tvrzení, bázových tvrzení a bázových grafů.



Definice 9.5 (instance tvrzení, bázového tvrzení a bázového grafu)

Substitucí jména za prázdný uzel v grafu (trojici) vzniká jeho *instance*.

Bázový graf (bázová trojice) RDF je takový graf (trojice), v němž se nevyskytuje prázdný uzel.

Pro množinu prázdných uzlů $blank(E)$ je třeba rozšířit interpretační pravidla následující definicí.

Reprezentace znalostí

Definice 9.6

Nechť I je interpretace a A je zobrazení z množiny $\text{blank}(E)$ prázdných uzlů z E do universa diskursu \mathbf{IR} , které přiřazuje každému prázdnému uzlu prvek z \mathbf{IR} . Potom platí:

Je-li E prázdný uzel a $A(E)$ je definováno, potom v rozšířené interpretaci $[I+A](E) = A(E)$.

Je-li E RDF graf, potom je $I(E) = \text{true}$, jestliže platí pro nějaké zobrazení A' z $\text{blank}(E)$ do \mathbf{IR} $[I+A'](E) = \text{true}$. V opačném případě je $I(E) = \text{false}$.



9.5.4 Interpretace báзовých grafů

V rámci interpretace RDF jazykových prvků je třeba rozlišovat:

- jménu je přiřazen jeho denotát, kterým je vždy objekt universa diskursu, tedy příslušný zdroj, zatímco
- báзовé trojici a báзовému grafu je přiřazena pravdivostní hodnota jí/jemu odpovídajících tvrzení.

Stanovení denotátu jména, ať už jde o URI-odkaz nebo literál (v případech a) - d)), a stanovení pravdivostní hodnoty báзовého grafu (v případech e),f)) v interpretaci I jsou v následující definici dána *pravidly interpretace*.

Definice 9.7 (pravidel interpretace báзовého grafu E)

- Je-li E prostý literál "aaa" z \mathbf{V} , potom je $I(E) = \text{aaa}$.
- Je-li E prostý literál "aaa"@ttt z \mathbf{V} (kde @ttt je jazykový ukazatel), potom je $I(E) = \langle \text{aaa}, \text{ttt} \rangle$.
- Je-li E typovaný literál z \mathbf{V} , potom je $I(E) = \mathbf{IL}(E)$.
- Je-li E URI-odkaz z \mathbf{V} , potom $I(E) = \mathbf{IS}(E)$.
- Je-li E báзовá trojice $s p o$ (tj. $\langle \text{subjekt} \rangle \langle \text{predikát} \rangle \langle \text{objekt} \rangle$), potom $I(E) = \text{true}$, jestliže s a p je z \mathbf{V} , $I(p)$ je z \mathbf{IP} a dvojice $\langle I(s), I(o) \rangle$ náleží extenzi $\mathbf{IEXT}(I(p))$. Jinak je $I(E) = \text{false}$.
- Je-li E báзовý graf RDF, potom $I(E) = \text{false}$, jestliže pro některou trojici E' platí $I(E') = \text{false}$. Jinak je $I(E) = \text{true}$.



9.5.5 Modely a logické důsledky RDF grafů

Podobně jako v logice prvního řádu je i zde na základě sémantiky definován model a logický důsledek grafu nebo množiny grafů.

Definice 9.8

Struktura interpretace I je modelem množiny \mathbf{S} grafů, právě když všechny grafy množiny \mathbf{S} jsou v interpretaci I pravdivé (splněny).



Reprezentace znalostí



Definice 9.9

Graf E je logickým důsledkem množiny S grafů, právě když pro všechny modely množiny S platí, že graf E je v nich splněn.

Podobně jako v logice prvního řádu též platí

- každý jednotlivý graf množiny S je jejím logickým důsledkem,
- logickým důsledkem grafu s prázdnými uzly je jeho instance a
- následující věta.

Věta 9.1 (o kompaktnosti)

Je-li konečný graf E logickým důsledkem grafu S , potom je E logickým důsledkem nějakého subgrafu S' grafu S .

9.6 Ontologický jazyk webu OWL

9.6.1 RDF a OWL

Jak je vidět z předcházejících odstavců, účelem modelu RDF je definovat mechanismus pro popis zdrojů, který sice předpokládá návazné budování aplikací nad nimi, avšak primárně nic nepředpokládá o aplikačních doménách, např. nedefinuje v nich žádný způsob reprezentace omezení, ani nemá prostředky vyjádření příslušnosti objektu ke třídě. Definice tohoto mechanismu je doménově neutrální, neboť tento mechanismus má být vhodný k popisům informací o vazbách mezi zdroji pro jakékoliv domény. Nespecifikuje se žádný netriviální odvozovací mechanismus, ten je třeba vybudovat nad RDF.

Ontologický jazyk webu (OWL) je jazykem pro definování a používání webových ontologií prostřednictvím vhodných instancí. Jazyk OWL, stejně jako RDFS je určen k tomu, aby poskytoval prostředky definování tříd, jejich vlastností a vzájemných vztahů. Rozdíl je především v tom, že RDFS poskytuje pouze nejzákladnější z těchto prostředků, neobsahuje možnosti definovat omezení ani prostředky odvozování, zatímco OWL, vybavený logickými prostředky deskripční logiky, umožňuje vyjádření omezení na třídách (konceptech) a poskytuje prostředky odvozování založené na logickém základě.

OWL využívá všech schopností definování struktur tříd (konceptů) a vlastností a jejich organizací do hierarchií poskytovaných RDFS. OWL je jazykem vybudovaným nejen na modelu RDF(S) a na bázi formální ontologie, ale též na bázi dedukčních vlastností deskripční logiky (DL).

Vybavení jazyka prostředky deskripční logiky dává možnost vyjádření konceptu nebo vlastnosti jako logické kompozice jiných konceptů nebo vlastností nebo jako výčet objektů specifikovaných vlastností. Tyto možnosti v rámci RDFS nejsou. Navíc, je-li vytvořena pro nějakou doménu ontologie, umožňuje formální sémantika OWL odvození jejích logických důsledků, tj. dalších tvrzení, která nejsou v ontologii přímo obsažena, ale vyplývají z její

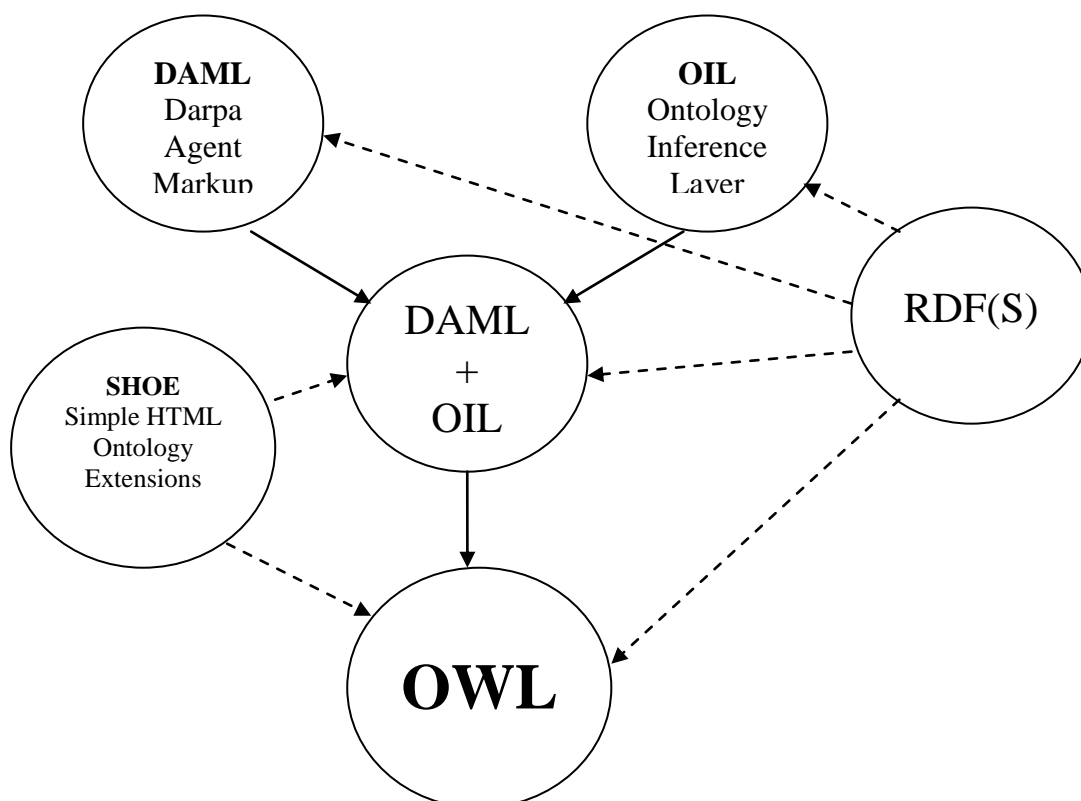
Reprezentace znalostí

významové struktury. To umožňuje vtažení formalizované sémantiky webových zdrojů, jejich vlastností a vzájemných vztahů, vázaných na terminologické slovníky příslušných domén, do logických struktur a odvozovacích metod deskripční logiky.

9.6.2 Historie vývoje OWL

Jazyk OWL byl vytvořen na základě výsledků úsilí dvou navzájem nezávislých pracovních skupin. První z nich, vytvořená jako "Joint Working Group on Agent Markup Languages" ze dvou pracovních týmů z USA a Velké Británie a částečně podporovaná agenturou "US Defense Advanced Research Projects Agency - DARPA", podala návrh webového ontologického jazyka s názvem DAML+OIL (Darpa Agent Markup Language + Ontology Inference Layer). Druhou z nich byla skupina evropských specialistů, jejichž práce na návrhu ontologického jazyka byla podporována EU projektem IST.

Jazyk OWL, vytvořený skupinou "Web Ontology Working Group" konsorcia W3C, byl předložen poprvé v roce 2001, po doladění a vytvoření potřebných formálních náležitostí pak uveden pod garanci W3C do provozu v roce 2004 .



Obrázek 9.4

OWL má stejně jako RDF abstraktní syntax založenou na RDF trojicích a pro účely výměny informací používá stejně jako RDF syntax RDF XML. V rámci realizace sémantického webu náleží OWL, akceptující *předpoklad otevřeného světa*, ke klíčovým prvkům systému.

Reprezentace znalostí

9.6.3 Podjazyky OWL

OWL tvoří rodinu jazyků pro reprezentaci znalostí ve formě ontologií, která je ve svých verzích OWL-DL a OWL-LITE založena na deskripční logice (její SHOE verzi), její sémantice a inferenčních mechanismech, zatímco verze OWL-Full je založena na sémantice kompatibilní s RDFS.

OWL lite verze, která je jednodušší než zbývající dvě verze, je určena těm uživatelům, kteří vystačí s využitím principů klasifikační hierarchie a jednoduchými omezeními.

OWL DL, jak je patrné z jeho názvu, využívá prostředky reprezentace a odvozování, vyvinuté v rámci deskripční logiky. Je prostředkem formální reprezentace, určeným těm tvůrcům ontologií, kteří jsou schopni maximálně využít expresivitu OWL a odvozovací schopnosti jazyka DL deskripční logiky.

OWL full je určen těm vyspělým tvůrcům ontologií, kteří jsou schopni využít jeho relativní syntaktické svobody při definování konceptů (tříd), využívání konstruktorů datových typů. Umožňuje využití již zkonstruovaných OWL ontologií k vytváření jejich potřebných modifikací doplňováním dalších vlastností. Přitom jde o rozšiřování znalostní báze monotónním způsobem, který neznehodnotí to, co již bylo předtím vytvořeno.

9.6.4 Reprezentace ontologie v jazyce OWL

Dokument v ontologickém jazyce OWL reprezentuje ontologii příslušné domény a je organizován tak, aby byl strojově "čitelný" a použitelný pro odvozování důsledků.

Základní koncepty *světa (domény)*, kterého se vytvářená ontologie týká, tvoří třídy, uspořádané do hierarchických struktur. Přitom každý prvek reprezentující objekt světa náleží univerzální třídě (Top konceptu) `owl:Thing`. Pojmenované třídy, specifické vzhledem k popisovanému světu, jsou pak definovány prostřednictvím svých vlastností, které představují rovněž relevantní koncepty reprezentovaného světa. Systém konceptově orientované reprezentace, který má charakter multidimenzionální hierarchie, doplňuje prázdná třída `owl:Nothing`, zastupující absurdní koncept (Bottom koncept).

Poznámka

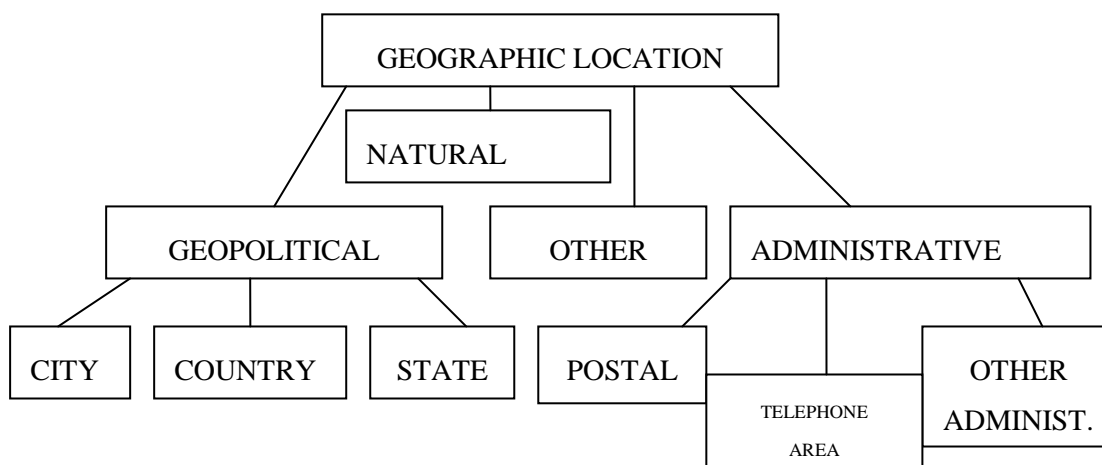
W3C dokument s názvem OWL Web Ontology Language Guide, postavený na konkrétním příkladu konstrukce ontologie domény vinných produktů, může sloužit jako návod postupu vytváření ontologie v podstatě libovolné domény, neboť použití jednotlivých konstrukčních prvků doprovází obecnějším výkladem jejich smyslu a možností využití.

Reprezentace znalostí

9.6.5 Ontologie v OWL jako XML dokument systému Protégé

Jak již bylo uvedeno, existuje celá řada programových produktů určených k vytvoření OWL dokumentů definujících ontologie daných domén.

Následující příklad ukazuje postup převodu hierarchicky uspořádané množiny (taxonomie) konceptů z domény geografických lokací (obr. 9.9) do ontologie definované v jazyce OWL formou XML dokumentu.



Obrázek 9.5

Příklad 9.4

Převod taxonomie konceptů z domény geografických lokací zobrazené na obr. 9.9 do OWL dokumentu definujícího ontologii pomocí systému Protégé. Příklad byl vytvořen na základě publikovaného webového dokumentu D.C. Haye.



Datovým modelem, představujícím startovací bod tvorby ontologie v jazyce OWL je zde taxonomie, v níž subkoncepty mají vždy jediného společného rodiče s týmiž atributy. Taxonomie geografických ploch se proto vyznačuje tím, že každá třída v ní obsažená náleží jediné supertřídě. V systému Protégé roli univerzální supertřídě všech ontologií sehraává třída `owl:Thing`.

Definice ontologie začíná definováním jmenných prostorů RDF, OWL a jiných prvků ontologického jazyka. Značka `owl:Ontology` deklaruje, že se jedná o OWL definici ontologie. Ontologie „Geography“ je zde použita jako jmenný prostor a následně definované termíny jsou popisovány jako prvky tohoto bazového jmenného prostoru.

Protégé vytváří vedle záznamu o průběhu práce na bázi ontologie „Geography“ soubor `.owl`, který obsahuje XML popis ontologie.

Reprezentace znalostí

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
.....
  xmlns="http://www.owl-ontologies.com/Geography.owl#"
xml:base="http://www.owl-ontologies.com/Geography.owl"
```

Třída GEOGRAPHIC LOCATION je definována jako podtřída univerzální třídy Thing. Podtřídy ADMINISTRATIVE AREA a OTHER AREA nyní lze obdobným způsobem definovat jako podtřídy již definované třídy GEOGRAPHIC LOCATION. Podobně je tomu i na další úrovni podtříd.

Třídy definované jako podtřídy již předtím definovaných tříd sice sdílejí jejich atributy, ale nemají zaručenu jejich vzájemnou disjunktnost. Je-li např. potřeba definovat třídu OTHER AREA jako disjunktní vzhledem ke třídě ADMINISTRATIVE AREA Protégé poskytuje k tomu, stejně jako pro účel dalších omezení na třídách své prostředky.

```
<owl:Class rdf:about="#GEOGRAPHIC LOCATION">
  <rdfs:subClassOf>

<rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  .....
  </rdfs:subClassOf>

  <owl:disjointWith>
    <owl:Class rdf:ID="ADMINISTRATIVE AREA"/>
  </owl:disjointWith>

  <owl:disjointWith>
    <owl:Class rdf:ID="OTHER AREA"/>
  </owl:disjointWith>

</owl:Class>
```

Definice *vztahu* specifikujícího umístění města ve státě je v OWL realizována pomocí vlastnosti owl:objectProperty.

```
<owl:Class rdf:ID="City"/>
<owl:Class rdf:ID="State"/>
<owl:objectProperty rdf:ID="located_in_state">
  <rdfs:domain rdf:resource="#City"/>
  <rdfs:range rdf:resource="#State"/>
</owl:objectProperty>
```

Obdobně lze specifikovat v OWL *atribut* pomocí vlastnosti datatypeProperty.

9.7 Ontologické jazyky a datové modelování

9.7.1 Datový model a ontologie

D.C. Hay vymezuje pojmy ontologie a datový model takto: Ontologie je katalog věcí, které existují v zájmové doméně. Datový model je druhem ontologie, který začíná definováním kategorií dat, doplněným stanovenými pravidly seskupování dat do kategorií a organizovaným pro prezentaci člověku. Ontologie v tom smyslu, jak bývá konstruována, tedy může a nemusí být podle Haje datovým modelem, neboť, jak ukazuje praxe, nemusí nutně začínat definováním kategorií dat, ale v řadě případů začíná identifikací instancí aktuálních dat a teprve následně tyto instance klasifikuje a organizuje do vhodných struktur, doplněných pravidly pro možné odvozování.

Pokud ale, jak je všeobecně přijímáno, má ontologie být všeobecně sdílenou konceptualizací dané domény, která je nemyslitelná oddělena od datového modelu, vymezujícího její prostředky, je možno konstatovat, že kvalitní ontologie dané domény a konceptuální datový model této domény jsou jedno a totéž.

9.7.2 Prostředky RDF(S), OWL a E-R datového modelování

Na rozdíl od prostředků datového modelování s širokou adaptabilitou jsou ontologické jazyky postaveny na XML, což znamená, že jsou limitovány principem předem daných značek.

Pro porovnání prostředků reprezentace E-R datového modelu a jazyků RDF, RDFS a OWL slouží následující trojice tabulek.

9.8 Zhodnocení přínosu RDF/OWL modelování

9.8.1 RDF datový model a jazyk OWL

Datová úroveň RDF poskytuje obecný model reprezentace výměny a integrace dat. Na ontologické úrovni RDFS podporuje specifikaci slovníků použitých v RDF a OWL/DL obohacuje reprezentační prostředky o formální prostředky dedukce a bližší specifikace, poskytované deskripční logikou. OWL/Lite je zjednodušeným podjazykem OWL/DL. OWL/full představuje základní rozšíření RDFS, obohacující množinu axiomů pro definování tříd a vlastností v rámci modelované domény o další atributy vlastností, možnosti omezování tříd a vyjádření vztahů mezi třídami.

9.8.2 Typické vlastnosti modelování v RDF/OWL

V RDF jsou zdroje, vlastnosti a třídy určeny svými identifikátory URI. Pokud by určování URI nepodléhalo žádnému centrálnímu řízení, mohlo by se stát, že

Reprezentace znalostí

různým URI odpovídá tentýž element (koncept) modelu. Jinými slovy, neplatí "předpoklad jedinečného jména".

Sémantický web poskytuje způsob pojmenovávání ontologií (schémat) ve dvou úrovních, tj. jméno má dvě části:

- jméno schématu
- specifické jméno ve schématu.

URL je URI, které přiděluje umístění zdroje na webu. Jméno schématu a prvního prvku specifického jména jsou regulovány v rámci organizace WWW. Pro použití svého vlastního URL je třeba získat svolení ICANN (Internet Corporation for Assigned Names and Numbers).

Dalším zásadním předpokladem je "předpoklad otevřeného světa" RDF/OWL model nepředpokládá, že v době modelování je o modelované doméně známo všechno, co je v ní relevantní. To znamená otevřenost nově přichozím informacím a znalostem o této doméně. Z tohoto předpokladu vyplývá, jak již bylo popsáno v kap. problém stanovení pravdivostních hodnot některých tvrzení.

9.8.3 Výhody a nevýhody modelování RDF/OWL

Podle řady autorů je základní silnou stránkou RDF/OWL

- podpora integrace informací a jejich znovupoužití na základě jednotné identifikace a sdíleného slovníku,
- možnost manipulace s částečně strukturovanými daty,
- oddělení datového modelování od syntaxe reprezentačního jazyka,
- začlenění webových zdrojů prostřednictvím metadat o jejich obsahu (formální sémantiky) do problematiky znalostních bází,
- podpora klasifikace a inference na bázi formální sémantiky,
- schopnost reprezentovat třídu a její instanci stejným způsobem a možnost jejich kombinování.

Slabými stránkami pak jsou

- nízká schopnost hodnocení dokumentů,
- limitovaná expresivita zdroji a definovanými vlastnostmi,
- nedostatečně vyjasněná role XML reprezentace,
- nevybavenost prostředky reprezentace procesu a změny.

RDF/OWL je speciálně vhodný pro modelování aplikací, které se potýkají s problémem distribuované informace, neboť umožňuje integraci dat z více různých zdrojů, sdílený slovník umožňuje interoperabilitu. Pro případy uzavřených problémů se stabilními datovými modely není nejvhodnější.

Reprezentace znalostí

9.8.4 Ontologie, XML Schema a objektový princip UML

Na první pohled je zřejmé, že mnoho tvrzení v OWL ontologiích je možno formulovat též např. pomocí XML Schéma nebo objektového jazyka UML. Tyto reprezentační prostředky sdílejí společné rysy jako hierarchii typů, schopnost specifikace kardinalit relací a stanovení typů atributů. Porovnání těchto prostředků je možné pouze na základě úvahy, čeho by se mělo modelováním docílit.

V případě ontologie je cílem zachytit a modelovat entity a vztahy, které tvoří vnitřní komponenty domény, s cílem umožnit interoperabilitu mezi různými systémy, vázanými na tuto doménu.

Při tvorbě schématu je cílem specifikovat, jak mají být informace a znalosti strukturovány pro ukládání do paměti, vzhledem k možnému přístupu potřebnému pro aplikace. Primárním cílem je tedy podpora efektivního a bezpečného procesního využití. Totéž je typickým rysem objektového modelování v UML.

Tyto přístupy se tedy zjevně překrývají.

Nejvýraznější předností RDF/OWL je na rozdíl od obou zbývajících schopnost integrovat informace a znalosti z násobných heterogenních zdrojů. To umožňuje sémantika postavená na ontologiích a zavedení globálních URI pro identifikaci zdrojů, konceptů, jejich vlastností a vzájemných vztahů. Ontologie (slovníky) jsou publikovány na webu tak, že jejich URL odpovídají URI jmenných prostorů těchto slovníků.

Konceptuální přístup RDF/OWL umožňuje modelování nezávislé na syntaxi reprezentačního jazyka. Vedle standardizovaného RDF XML je možno zvolit jiné vhodné jazyky, jako je např. N3 formát, jehož autory jsou Tim Berners-Lee a Dan Connolly.

Další předností RDF/OWL je schopnost odvozování znalostí založená na formální sémantice modelu a využívající inferenčních mechanismů poskytovaných deskripční logikou.

Kontrolní otázky a úkoly

1. K čemu slouží RDF a RDFS? Jaká je mezi nimi rozdíl?
2. Je možné pro RDF model využít XML?
3. K čemu slouží URI a jaký má vztah k URL?
4. Jaké jsou slabé stránky modelování RDF/OWL?



Shrnutí

Jazyk RDF je doporučený konsorciem W3C jako základ pro reprezentaci sémantického webu. RDF je vlastně drup datového modelu, tak jak jej známe z databází.

Základními objekty jsou zdroje, které jsou identifikovány pomocí URI. URI může být přiřazeno jakémukoliv objektu, pojmu nebo např. osobě. V praxi



Reprezentace znalostí

může obsahovat URL anebo také jednoznačný identifikátor (například rodné číslo, unikátní kód,...). Z toho vyplývá, že jediný požadavek na URI je, aby byl jednoznačný.

Výroky v RDF přiřazuje zdroji vlastnost: hodnota. Tedy výroky jsou složeny z trojic: zdroj (předmět), vlastnost (predikát) a hodnoty vlastnosti (objekt):

- Zdroj - je předmětem, na který se můžeme odkazovat pomocí URI. V praxi je zdroj anonymní a odkazuje se na něj pomocí "_:n" (n je přirozené číslo).
- Vlastnost - (predikát) vytváří binární relaci mezi zdroji a atomickými hodnotami poskytnutými definicemi primitivních datových typů v XML. Vlastnost může být dána také pomocí URI.
- Hodnota vlastnosti - (objekt) jak již z názvu vyplývá, specifikuje vlastnost nějakého predikátu. Objektem může být literál nebo další zdroj.

Výrok v RDF lze zapsat například takto:

```
<uživatel:Gunner, uživatel:celéJméno, uživatel: Petr Hanyáš>
```

V lidské řeči by se dala z tohoto výroku vytvořit věta: "Zdroj informací identifikovaný jako uživatel:Gunner má celé jméno Petr Hanyáš".

Terminologie RDF používá speciální termín - RDF Graf (někdy také RDF Model nebo RDF Data) definovaný jako množina zmíněných trojic. A pro výměnu dat využívá serializaci v XML nazývanou RDF/XML. Pro elementy rdf se používá předpona rdf a pro vyjádření prostorů jmen pomocí atributů XML se využívá předpona xmlns. Pak se například jmenný prostor pro Dublin Core označuje takto:

```
xmlns:dc = 'http://www.purl.org/DC/'.
```


Literatura

- Baader, F. et al.: *The Description Logic Handbook*. Cambridge University Press, 2004.
- Baader, F. et al.: *The Description Logic Handbook*. Cambridge University Press, 2004.
- Barry, S.: *Tools of Formal ontology*. In Nicola Guarino (ed.): *Formal Ontology in Information Systems*. IOS Press (Frontiers in Artificial Intelligence and Applications), 1998, s.19-28.
- Barry, S.: *Tools of Formal ontology*. In Nicola Guarino (ed.): *Formal Ontology in Information Systems*. IOS Press (Frontiers in Artificial Intelligence and Applications), 1998, s.19-28.
- Bechhofer, S., Horrocks, I., Patel-Schneider, P. F.: *Tutorial on OWL*. ISWC, Sanibel Island, Florida, USA, 2003.
- Bělohávek, R., Sklenář, V., Zacpal, J.: *Concept lattices constrained by attribute Dependencies*. Proc. Dateso, 2004, Ostrava, 2004.
- Berka, Petr a kolektiv: *Expertní systémy*. 1. vyd., Praha, Vysoká škola ekonomická v Praze 1998, s. 160.
- Borgida, A.: *On the relative expressiveness of description logics and predicate logics*. Artificial Intelligence 82, 1996, s. 353-367.
- Brachman, R. J.: *On the Epistemological Status of Semantic Network*. In N. Findler (ed.) *Associative Network: Representation and Use of Knowledge by Computers*. Academic Press, 1979.
- Brachman, R., Levesque, H. J.: *Readings in Knowledge Representation*. Los Altos: Morgan Kaufman, 1985.
- Brachman, R., Levesque, H. J.: *Readings in Knowledge Representation*. Los Altos: Morgan Kaufman, 1985.
- Broekstra, J., Kampman, A., van Harmelen, F., Sesame, A.: *Generic Architecture for Storing and Querying RDF and RDF Schema* [Online].2002. [Citace: 5. ledna 2012] <http://www.openrdf.org/doc/papers/Sesame-ISWC2002.pdf>, 2002.
- Buchheit, M., Donini, F., Schaerf, A.: *Decidable Reasoning in Terminological Knowledge Representation Systems*. Journal of Artificial Intelligence Research 1, 109 – 138, 1993.
- Buchheit, M., Donini, F., Schaerf, A.: *Decidable Reasoning in Terminological Knowledge Representation Systems*. Journal of Artificial Intelligence Research 1, 109 – 138, 1993.
- Calvanese, D., Lenzerini, M. G., Nardi, D.: *Description Logics for Conceptual Data Modeling*. Logics for Databases and Information systems, J. Chomicki and G. Saake eds., Kluwer, 1998.

Reprezentace znalostí

Calvanese, D., Lenzerini, M. G., Nardi, D.: *Description Logics for Conceptual Data Modeling*. Logics for Databases and Information systems, J. Chomicki and G. Saake eds., Kluwer, 1998.

Catone, J.: *Web 3.0: Is It About Personalization?* [Online]. 2008. [Citace: 5. ledna 2012].

http://www.readwriteweb.com/archives/web_3_is_it_about_personalization.php

Cocchiarella, N.B.: *Formal Ontology*. Data and Knowledge Engineering 39, Elsevier Science B.V., 2001.

Cocchiarella, N.B.: *Formal Ontology*. Data and Knowledge Engineering 39, Elsevier Science B.V., 2001.

Decker, S., Mitra, P., Melnik, S.: *Framework for the Semantic Web*. In RDF Tutorial Stanford University, 2000. s. 68 – 73.

Donini, F. M., Lenzerini, M., Nardi, D., Schaerf, A.: *Reasoning in Description Logic*. A Great Collection U. Gnowho and U. Gnowho-else, eds, CSLI Publications, 1997.

Donini, F. M., Lenzerini, M., Nardi, D., Schaerf, A.: *Reasoning in Description Logic*. A Great Collection U. Gnowho and U. Gnowho-else, eds, CSLI Publications, 1997.

Feigenbaum, E. A. [editor]: *Themes and Case Studies of Knowledge Engineering*. Edinburgh: Edinburgh University Press, 1979. Expert Systems in the Micro-Electronic Age.

Feigenbaum, E. A. [editor]: *Themes and Case Studies of Knowledge Engineering*. Edinburgh: Edinburgh University Press, 1979. Expert Systems in the Micro-Electronic Age.

Feigenbaum, L.: *TechnicalLee Speaking: Using RDF on the Web: A Vision* [Online]. 2007. [Citace: 10. února 2012] http://www.thefigtrees.net/lee/blog/2007/01/using_rdf_on_the_web_a_vision.html

Feigenbaum, L.: *TechnicalLee Speaking: Using RDF on the Web: A Vision* [Online]. 2007. [Citace: 5. ledna 2012] http://www.thefigtrees.net/lee/blog/2007/01/using_rdf_on_the_web_a_vision.html

Guarino, N. (ed.), *Formal Ontology in Information Systems*, Amsterdam, IOS-Press, 1998.

Guarino, N. (ed.), *Formal Ontology in Information Systems*, Amsterdam, IOS-Press, 1998.

Guarino, N., Carrara, M., Giaretta, P.: *An Ontology of Meta-Level, Categories*. In Doyle, J., Sandewall, E., Torasso, P. (eds.) principles of Knowledge representation and Reasoning. proceedings of the fourth International Conference (KR94). Morgan Kaufman, San Mateo, 1994.

Guarino, N., Carrara, M., Giaretta, P.: *An Ontology of Meta-Level, Categories*. In Doyle, J., Sandewall, E., Torasso, P. (eds.) principles of Knowledge

Reprezentace znalostí

representation and Reasoning. proceedings of the fourth International Conference (KR94). Morgan Kaufman, San Mateo, 1994.

Guarino, N., Giaretta, P.: *Ontologies and Knowledge Bases. Towards a Terminological Clarification*. In: Mars, N. (ed.). *Towards very large knowledge bases. Knowledge building and knowledge Sharing*, Ios Press, 1995.

Guarino, N., Giaretta, P.: *Ontologies and Knowledge Bases. Towards a Terminological Clarification*. In: Mars, N. (ed.). *Towards very large knowledge bases. Knowledge building and knowledge Sharing*, Ios Press, 1995.

Guarino, N., Welty, Ch.: *A Formal Ontology of Properties*. Proc. of 12th Conf. On Knowledge Engineering and Knowledge Management. Lecture Notes on Computer Science, Springer Verlag, 2000.

Guarino, N., Welty, Ch.: *A Formal Ontology of Properties*. Proc. of 12th Conf. On Knowledge Engineering and Knowledge Management. Lecture Notes on Computer Science, Springer Verlag, 2000.

Guarino, N., Welty, Ch.: *Conceptual Modeling and Ontological Analysis* [Online]. Proc. FOIS 2001 Formal Ontology in Information Systems. 2001. [Citace: 26. října 2011] <http://www.cs.vassar.edu/faculty/welty/>, 2001.

Guarino, N.: *Formal Ontology, Conceptual Analysis and Knowledge representation*. In: Guarino, N., Poli, R. (eds.). *International Journal on Formal Conceptual Analysis*, 1999.

Guarino, N.: *Formal Ontology, Conceptual Analysis and Knowledge representation*. In: Guarino, N., Poli, R. (eds.). *International Journal on Formal Conceptual Analysis*, 1999.

Hájek, P.: *Metamathematics of Fuzzy Logic*. Springer, 2001. s. 312.

Harmelen, F., Lifschitz, V., Porter, B., eds: *Conceptual Graphs*, Handbook of Knowledge Representation, Elsevier, 2008, s. 213-237.

Huan Keat, T.: *Semantic Networks* [Online]. 2004. [Citace: 11. října 2011] <http://seantoh.tnlsolutions.my/project/Semantic%20Networks.pdf>

Champin, P. A.: *RDF tutorial*. Bechhofer, S. (ed.): *Ontology Language Standardization Efforts*. 2001.

Kosek, J.: *Tutoriál Topic Maps* [Online]. 2006. [Citace: 12. prosince 2011] <http://www.kosek.cz/xml/tmtut/>

Kramer, B.; Mylopoulos, J.: *Knowledge Representation*. In: Shapiro, S. C. (ed.): *Encyclopedia of Artificial Intelligence*. 2. vyd., New York, John Wiley & Sons 1992, s. 1792.

Kremer, R.: *Visual Languages for Knowledge Representation* [Online]. Proc. of KAW'98, Eleventh Workshop on Knowledge Acquisition, Voyager Inn, Banff, Alberta, Canada, 1998. [Citace: 11. listopad 2011]. <http://www.hum.auc.dk/cg/>

Lukasová, A., Habiballa, H., Telnarová, Z., Vajgl, M.: *Formální reprezentace znalostí*. Ostrava: Universum, 2010. str. 345.

Reprezentace znalostí

Lukasová, A., Habiballa, H., Telnarová, Z., Vajgl, M.: *Formální reprezentace znalostí*. Ostrava: Universum, 2010. str. 345.

Lukasová, A.: *Formální logika v umělé inteligenci*. 1. vyd. Brno: Computer Press, 2003. 300 s.

Lukasová, A.: *Reasoning with Semantic Tableau Binary Trees in Description Logic*. Proc. DL2005, Edinburgh, 2005.

Lukasová, A.: *Reprezentace znalostí v asociativních sítích*. Proc. Znalosti 2001.

Mazzieri, M., Dragoni, A.F.: *A Fuzzy semantics for semantic web languages*. In Proceedings of the Proceedings of the ISWC Workshop on Uncertainty Reasoning for the Semantic Web (URSW-05). CEUR Workshop Proceedings, 2005.

Mazzieri, M.: *A Fuzzy RDF semantics to represent trust metadata*. In Proceedings of the 1st Italian Semantic Web Workshop: Semantic Web Applications and Perspectives (SWAP 2004), 2004.

McBride, B.: *The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS*. In Staab, S., Studer, R. eds.: *Handbook on Ontologie*, Springer, 2004.

Miller, E. *An Introduction to the Resource Description Framework D-LibMagazine* [Online], 1998. [Citace: 2. ledna 2012].
<http://www.dlib.org/dlib/may98/miller/05miller.html>

Obitko, M.: *Conceptual Graphs* [Online]. 2007. [Citace: 9. listopad 2011]
<http://www.obitko.com/tutorials/ontologies-semantic-web/conceptual-graphs.html>

Quillian, M. R.: *Semantic memory*. In Minsky, M. (ed.): *Semantic Information processing*, MA: MIT Press, pp. 27-70, 1968.

Richards, T.: *Clausal Form Logic. An Introduction to the Logic of Computer Reasoning*. Addison-Wesley, 1989.

Russel, S., Norwig, P.: *Artificial Intelligence. A Modern Approach*. (2nd edition), Prentice Hall, 2003.

Russel, S., Norwig, P.: *Artificial Intelligence. A Modern Approach*. (2nd edition), Prentice Hall, 2003.

Sowa, J.F.: *Conceptual graph* [Online]. 2005. [Citace: 20. prosinec 2011]
<http://www.jfsowa.com/cg/index.htm>

Sowa, J.F.: *Conceptual Graph Standard*, NCITS.T22 Committee on Information Interchange and Interpretation, 2000.

Sowa, J.F.: *Ontology* [Online]. 2003. [Citace: 10. prosinec 2011]
<http://www.jfsowa.com/ontology/>

Straccia, U.: *A Minimal Deductive System for General Fuzzy RDF*. In Proceedings of the 3rd International Conference on Web Reasoning and Rule Systems (RR-09), 2009.

Vajgl, M., Lukasová, A.: *RDF model as Associative Network*. Datakon 2009.

Reprezentace znalostí

- W3C: *Primer: Getting into RDF et Semantic Web using N3* [Online]. 2005. [Citace: 3. ledna 2012] <http://www.w3.org/2000/10/swap/Primer>
- W3C: *RDF Semantics* [Online]. 2004. [Citace: 3. ledna 2012] <http://www.w3.org/TR/2004/REC-rdf-mt-20040210/>
- W3C: *Resource description Framework (RDF). Concepts and abstract Syntax* [Online]. 2004. [Citace: 5. ledna 2012] <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>
- W3C: *Resource description Framework (RDF). RDF Primer* [Online]. 2004. [Citace: 7. ledna 2012] <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>
- W3C: *Web ontology Language Guide* [Online]. 2004. [Citace: 7. ledna 2004] <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>
- W3C: *Web ontology Language Overview* [Online]. 2004. [Citace: 7. ledna 2004] <http://www.w3.org/TR/owl-features/>
- W3C: *XML Schema* [Online]. 2004. [Citace: 8. ledna 2004] <http://www.w3.org/XML/Schema>
- Wikipedia: *Resource Description Framework* [Online]. [Citace: 6. ledna 2012] http://en.wikipedia.org/wiki/Resource_Description_framework
- Wikipedia: *Topic Maps* [Online]. 2011. [Citace: 12. prosince 2012] http://en.wikipedia.org/wiki/Topic_Maps
- Wikipedie: *Einsteinova hádanka* [Online]. 2012. [Citace: 10. června 2012] http://cs.wikipedia.org/wiki/Einsteinova_h%C3%A1danka

Vysvětlivky k používaným symbolům



Průvodce studiem – vstup autora do textu, specifický způsob, kterým se studentem komunikuje, povzbuzuje jej, doplňuje text o další informace



Příklad – objasnění nebo konkretizování problematiky na příkladu ze života, z praxe, ze společenské reality, apod.



Pojmy k zapamatování.



Shrnutí – shrnutí předcházející látky, shrnutí kapitoly.



Literatura – použitá ve studijním materiálu, pro doplnění a rozšíření poznatků.



Kontrolní otázky a úkoly – prověřují, do jaké míry studující text a problematiku pochopil, zapamatoval si podstatné a důležité informace a zda je dokáže aplikovat při řešení problémů.



Úkoly k textu – je potřeba je splnit neprodleně, neboť pomáhají dobrému zvládnutí následující látky.



Korespondenční úkoly – při jejich plnění postupuje studující podle pokynů s notnou dávkou vlastní iniciativy. Úkoly se průběžně evidují a hodnotí v průběhu celého kurzu.



Úkoly k zamyšlení.



Část pro zájemce – přináší látku a úkoly rozšiřující úroveň základního kurzu. Pasáže a úkoly jsou dobrovolné.



Testy a otázky – ke kterým řešení, odpovědi a výsledky studující najdou v rámci studijní opory.



Řešení a odpovědi – vážou se na konkrétní úkoly, zadání a testy.