

EXPERTNÉ SYSTÉMY

**KONCEPCIE, PRINCÍPY A ARCHITEKTÚRA,
PROSTRIEDKY REPREZENTÁCIE ZNALOSTÍ
ROZVINUTÉ A NEKLASICKÉ TECHNIKY ODVODZOVANIA,
TVORBA SYSTÉMOV**

Mikuláš Popper
MFF UK

Bratislava 1997-2002

PREDSLOV A ÚVODNÉ POZNÁMKY

Predložený text uvádza čitateľa do jednej z oblastí *umelej inteligencie*¹ - do problematiky **expertných systémov**.² Ponúkaný rozsah textu si kladie za cieľ poskytnúť čitateľovi dobré predpoklady k orientácii v problematike: v konceptuálnom pozadí týchto systémov, v princípoch, ktoré sú nimi stelesňované, v zásadach ich navrhovania, vyvíjania a vhodných kategóriách problémov, pri riešení ktorých je ich aplikácia výhodná, alebo aspoň opodstatnená. Hlavným poslaním textu je však ponuka poznatkov, ktorých osvojenie umožňuje v širokom rozsahu osvojiť nadobudnúť znalosti a aj zručnosti nevyhnutné pre tvorbu **ES**.

Hoci texty sú určené poslucháčom univerzitného štúdia informatiky, najmä pre špecializáciu **UI**, môžu byť nepochybne prínosom aj pre iné štúdijné smery, ako aj odborníkom z potenciálnych aplikačných oblastí. Čitateľova orientácia aspoň vo východiskových tákach informatiky a v uvádzajúcich predmetoch **UI** zvyšuje čitateľnosť týchto textov.

ES tvoria špecifickú podtriedu *znalostne-intenzívnych* (alebo iba *znalostných*) **systémov**. Tie môžu byť nositeľmi netriviálneho rozsah **explicitne formulovateľných a symbolovo reprezentovaných znalostí** z aplikačnej oblasti, v ktorých sa uplatňujú. Od iných *znalostne-intenzívnych* systémov sa líšia tým, že sú nositeľmi predovšetkým **odborných** znalostí uplatňovaných pri riešení problémov danej kategórie.

Predkladaný text sa orientuje predovšetkým na kľúčové zložky **ES**, ale treba uviesť, že, okrem iného aj z rozsahových dôvodov, nezahrňuje alebo nerozvádza niektoré témy, ktoré by si pozornosť zaslúžili. Napriek tomu v dostatočnom rozsahu uvádzajú čitateľa do živej a aktuálnej teoretickej aj implementačnej problematiky. Vo viacerých častiach môže čitateľa motivovať k prípadnému hlbšiemu štúdiu aj vedeckému bádaniu.

Pre tých, čo v tej či inej podobe sa budú chcieť, ba dokonca z pracovných dôvodov budú musieť, danou problematikou hlbšie zaoberať, sú tieto texty nepochybne zdrojom poznatkov a východísk ku konkrétnym praktickým aplikáciám. Nejde však o príručku tvorby **ES** – nekladie si úlohu poskytovať k tomu konkrétne návody. Pozornosť zameriva na všeobecné poznatkové zázemie. Popri základných prístupoch k symbolovej reprezentácii znalostí, menovite na rôzne metódy, princípy, prostriedky a techniky tvorby systémov, vrátane prístupov k ich uplatňovaniu. Uvádza, resp. upozorňuje aj na novšie a perspektívne vývojové trendy **ES** s potenciálom ich teoretického aj praktického uplatňovania.

Produkty vznikajúce v oblasti tvorby **ES** sa dajú členiť na

- ✓ **problémovo zamerané ES** – sú vystrojené konkrétnymi kategóriami symbolovo reprezentovaných znalostí – nazývajú sa **báza znalostí**³ - z určitej problémovej oblasti,
- ✓ **prázdne expertné systémy** – programové systémy zodpovedajúce konkrétnemu funkčnému typu s vymedzenými štruktúrami, vlastnosťami a prostriedkami reprezentácie znalostí, ale **BZ** nie sú ich súčasťou,
- ✓ **bázy znalostí** – dátová štruktúra kompatibilná s **prázdny ES**, je nositeľkou symbolovej reprezentácie znalostí - produkt náročnej súčinnosti dvoch odborností: z aplikčnej oblasti a z tvorby **ES/BZ**,

¹ Symbolovo (skrátene) **UI**. V slovenskom jazykovom prostredí sa udomácnil termín *umelá inteligencia*, je prekladom anglického originálu *artificial intelligence*. Občas sa však dá stretnúť aj so synonymickými termínmi ako sú *strojová inteligencia*, či *intelektika*.

² Skrátene **ES**.

³ Skrátene **BZ**.

- ✓ **vývojové prostredie tvorby ES** – softvérový produkt tvorby prázdnych alebo problémovo zameraných systémov.

Finančné náklady na vývoj alebo zakúpenie **ES**, prípadne aj **BZ**, obdobne ako na vývoj alebo zakúpenie netriviálneho *vývojového prostredia tvorby ES*, sú značné. V čase spisovania týchto textov sa môžu pohybovať rádovo až v rozmedzí od 4 až 6 ciferných číslíc v korunovom vyjadrení. Preto poznatky a informácie uvádzané v tomto texte sa dajú považovať za významné aj pre

- formulovanie aplikačných potrieb a požiadaviek na používanie expertného systému a teda pre jeho cenovo a funkčne adekvátnu špecifikáciu,
 - tvorbu vlastného prázdneho či aplikačného **ES**,
 - výber a zakúpenie prázdneho alebo aplikačne zameraného **ES**,
 - výber komerčne dostupných prázdnych **ES** alebo ich vývojových prostredí
- a pod.

Prednášky z predmetov **Expertné systémy I. a II.** korešpondujú s týmto textom, je ich východiskom. Avšak pri "živom" výklade sú spravidla viaceré témy podrobnejšie rozvádzané aj rozšírené, prípadne aktualizované. Rozsah, výber a príprava tém prednášok je nezriedka aj výslednicou interakcie s poslucháčmi.

Autor zatiaľ tieto texty nepovažuje za uzatvorené. Vznikajú, doplňujú a modifikujú sa priebežne. Najmä preto, že neustále pribúdajú nové poznatky zverejňované v rozmanitých literárnych prameňoch, alebo sú artikulované na významných vedeckých podujatiach. Takže základné, vlastne už klasické témy, sa priebežne aktualizujú a doplňujú o nové vybrané state. Keďže texty zatiaľ neprechádzajú ani odbornou ani jazykovou recenziou, rozsah potenciálnych nedostatkov, chýb, obsahovo či štylisticky nevydarených formulácií je určite nemalý. Autor s vďakou uvíta všetky upozornenia na akékoľvek nedostatky, žiaduce opravy aj úpravy, ako aj odporúčenia na doplnenie a vylepšenie textov.

1. EXPERTNÉ SYSTÉMY - UVEDENIE DO PROBLEMATIKY

1.1 Konceptuálne hľadiská

Expertné systémy v kontexte umelej inteligencie

Umelá inteligencia (UI) je vedný odbor s vlastným teoretickým zázemím a vlastnou oblasťou praktických aplikácií. **Expertné systémy** tvoria jednu z významných častí UI – majú svoje špecifické teoretické základy a sú významné pre svoju praktickú uplatniteľnosť.

Problematika ES sa zameriava na

- a) tvorbu počítačom realizovaných javov, ktoré dostatočne presvedčivo pripomínajú prirodzený fenomén ľudskej inteligencie pri riešení odborných problémov (a v tesnej súvislosti s tým)
- b) hľadanie hraníc a možností symbolovej reprezentácie poznatkov, vrátane procesov ich nadobúdania, uchovávanía, využívania a rozvíjania.

Témy súvisiace s oboma zložkami tejto problematiky sú nosné. Zaoberáme sa nimi v nerovnakom rozsahu a s líšiacou sa detailnosťou – pozornosť kladieme skôr na druhú z uvedeých tém. Na ich uvedenie a priblíženie uvažujme nasledujúce otázky:

Úvaha o charaktere znalostí a ich uplatnenia pri riešení úloh

- Keď máme *k dispozícii systém parciálnych diferenciálnych rovníc, ktoré reprezentujú fyzikálne znalosti týkajúce sa zmien tlaku, prietoku a teploty média pretekajúceho v potrubí jadrového reaktora, a použijeme metódu numerického riešenia takého systému rovníc*, dá sa považovať programová realizácia zodpovedajúcich algoritmov v niektorom zo všeobecných programovacích jazykov, resp. tých čo sa používajú v umelej inteligencii (Prolog, LISP, FRL [Frame Representation Language], KRL [Knowledge Representation Language], Clips a iné) za prejav inteligencie?
- Keď máme *k dispozícii rozsiahlu sústavu algebraických rovníc opisujúcich štrukturálne väzby medzi veľkým počtom ekonomických fenoménov a použijeme metódu lineárneho programovania na nájdenie optimálnych parametrov skúmaného systému*, dá sa považovať programová realizácia zodpovedajúcich algoritmov v niektorom zo všeobecných programovacích jazykov, resp. tých čo sa používajú v umelej inteligencii (pozri v predošlom) za prejav inteligencie?
- Keď máme *za úlohu podporiť diagnostický proces, v ktorom je potrebné interpretovať pozorovaniu dostupné prejavy chybného správania skúmaného systému tak, aby sa zistila zodpovedajúca príčina (chybový stav) a použijeme k tomu známe vzťahy pre výpočet podmienených pravdepodobností napr. na základe Bayesovho vzťahu*, dá sa považovať programová realizácia zodpovedajúcich algoritmov v niektorom zo všeobecných programovacích jazykov, resp. tých čo sa používajú v umelej inteligencii (pozri v predošlom) za prejav inteligencie?

Áno? Nie? A prečo?

Dajú sa očakávať záporné odpovede. Aj napriek tomu, že pri riešení spomínaných úloh sa uplatňuje symbolová reprezentácia poznatkov (stelesnené

Mechanické, reproduktívne, postupy nie sú považované za prejav inteligencie

príslušnými matematickými vzťahmi). Zdôvodniteľnosť zápornej odpovede vyplýva najmä z toho, že zodpovedajúce výpočtové procedúry, potenciálne akokoľvek zložité, *sú implikované formuláciou úlohy, sú deterministické*, sú pomerne jednoducho *mechanizovateľné (algoritmizovateľné)* a *kvantitatívne*. Jadrnejšie sa to dá vyjadriť ešte aj nasledujúcim spôsobom

ak postup riešenia problému spočíva v **reprodukovaní** vopred daného algoritmu, ktorý vyplýva z formulácie problému a stelesňuje návod na riešenie, tak takýto postup je považovaný za **mechanický**.

Charakteristické rysy inteligentného riešenia problémov

Fenomén prirodzenej ľudskej inteligencie⁴ sa spája s **kognitívnymi (poznávacimi) prejavmi myslenia**. Tie sú charakterizovateľné

- **diskrétnymi, symbolovými, kvalitatívnymi,**
- **prehľadávacimi, rozpoznávajúcimi,**
- **nemonotónnymi, t.j. predpokladajúcimi, skúšajúcimi, nahrádzajúcimi a najmä nezáväznými (t.j. pripúšťajúcim odstupenie od predchádzajúcich postupov) a aj nedeterministickými krokmi.**

Na rozdiel od mechanických prejavov, iba ak sa riešiacie postupy prejavujú (aspoň zčásti) uvedenými charakteristikami – nezávisle od toho, či ich nositeľom je človek alebo (jeho výtvor) počítač – sme im náchylní pripísať *atribút inteligentného správania*.

Počítačom realizované symbolové procesy, ktoré potenciálne považujeme za nemechanické, teda také čo pripúšťajú možnosť priradiť im atribút inteligencie, sa dajú charakterizovať takto:

Produktívne riešenie problémov

Keď riešenie problému nespočíva na vopred danom návode,

- ale uplatňuje algoritmy **prehľadávania, odskúšavania a zretazovania vhodných operácií** - riešiacich krokov,
- pričom v prípade nezdaru sa od nich **odstupuje a nahrádzujú** sa inými krokmi a ich postupnosťami,

vtedy sa riešiaci proces postupne produkuje.

Pre kognitívne (inteligentné) správanie, na rozdiel od náhodilého či exhausívneho (preskúmania všetkých do úvahy prichádzajúcich možností) – teda pre produkovanie riešiacich postupov je nevyhnutné

využívať súbory všeobecných aj špecifických stratégií spočívajúcich na všeobecných a najmä špecifických znalostiach spôsobom, ktorý minimalizuje výpočtovú zložitosť vlastnú riešeným úlohám.

Z tohto pohľadu možno vnímať expertný systém ako

ES ako programový produkt

programový produkt stelesňujúci znalosti umožňujúce stratégiami riadené exploračné (preskúmavanie) problémového priestoru s cieľom efektívneho produkovania správnych a vyhovujúcich riešiacich postupov.

Expertné systémy majú špecifické danosti a charakteristiky. Patria medzi ne

⁴ Spomeňme si na Turingov test inteligencie.

a niektoré ich
charakteris-
tické rysy

- **ich aplikácie:** riešenie problémov, pre ktoré nie sú známe klasické algoritmičné riešiacie postupy (nedobre formulova[teľ]né, štruktúrova[teľ]né, podurčené [nedostatočne vymedzené, informované] problémy);
- **riešenia:** produktívne riešiacie postupy spočívajúce na kognitívnych stratégiách, ktoré - hoci dosiahnutie výsledku nezaručujú - spravidla vedú k požadovaným výsledkom, aj keď tie môžu byť zaťažené neurčitostami a sú teda nekategorickej či nejednoznačnej povahy;
- **funkčné vlastnosti:** stelesňujú situačné, údajmi (aj nekategorickými, kvalitatívnymi, neurčitými - fuzzy, nezáväznými, predpokladateľnými) podmienené a riadené prehl'adavacie procesy založené na symbolovej reprezentácii znalostí z oblasti, v ktorej vznikol riešený problém,
- **dynamika väzieb funkčných prvkov štruktúry systému:** situačná (oportunistická) výmena a zdielanie údajov, situačný prenos riadenia medzi zložkami systému;
- **metodika tvorby, rozvíjania a udržiavania systému:** prípustnosť relatívne nezávislého - autonómneho - vytvárania a úprav jednotlivých funkčných prvkov systému, čiastkové a postupné vytváranie, rozvíjanie a modifikovanie reprezen-tovaných znalostí a stratégií ich používania;
- **charakteristiky používania:** on-line interakcia s používateľom alebo s prostredím, zriedkavejšie aj vyhodnocovanie a interpretácia údajov v pozadí; pri interakcii s používateľom je žiaduca spôsobilosť systému vysvetľovať a zdôvodňovať postup riešenia danej úlohy a odvodených výsledkov, prípadne aj adaptácia interakcií v závislosti od daností a potrieb.

Rôznosť po-
hľadov na ES

Filozofické
hľadiská

Kognitívno-
psychologické
hľadiská

Expertné systémy sú netriviálnymi softvérovými dielami. Možno ich vnímať z viacerých iných zorných uhlov a tým sa sústreďovať na líšiace netriviálne otázky. Napríklad oprávnené otázky *filozofického* typu sa zaoberajú tým či sa vôbec dajú počítačom realizovať procesy zodpovedajúce ľudskej mysli - (problém '*mysel v neživej matérii*' či '*rozsah a ohraničenia realizovateľných prejavov a funkcií myslenia v neživej matérii*'). Sú to témy, s ktorými sa zaoberá filozofia mysle.

Množstvo otázok patrí do kategórie, ktorá je predmetom pozornosti *kognitívnych vied a psychologie*. Týkajú sa vrozených i vývinom a učením nadobúdaných a rozvíjaných psycho-biologických a psycho-lingvistických spôsobilostí, toho ako prebieha poznávanie a poznanie sveta existujúcich alebo uvažovaných objektov, javov, udalostí, toho ako sa realizujú a prebiehajú myšlienkové procesy, ako sa vytvárajú '*mentálne modely sveta*', ako sa utvárajú emócie, postoje, presvedčenia, vzťahy a vlastná rola vo vzťahu k druhým. A to aj vo väzbách od nižších k vyšším formám živého. Keďže doména umelej inteligencie sa považuje aj za súčasť kognitívnych vied, je prirodzené, že predmetom pozornosti je aj to, či sú '*mentálne modely sveta*' realizovateľné nebiologickými štruktúrami (počítačovými technológiami) a v akom rozsahu, či korešpondujúca '*reprezentácia sveta*' sa dá skúmať aj nezávisle na mysli, ktorá sa viaže k mozgu. Súvisí to so psychologickým poňatím takých fenoménov akými sú '*schémy "logického" usudzovania*', '*dedukcie*', '*abdukcie*', '*indukcie*', '*asociovania*', '*komplementovania*', '*porovnávanía*, *hodnotenia a usudzovania na úrovni kvalitatívneho (nie kvantitatívneho)*', '*pravdepodobnosti, očakávateľnosti a vieryhodnosti*', '*mentálnej reprezentácie času a priestoru*', '*zameriavania pozornosti*',

Abstraktné a
konceptuálne
témy

'usudzovania a interpretácie opierajúcich sa o kontext a analógiu', usudzovania na základe 'hierarchizovateľných kategórií a príčinnno-dôsledkových súvislostí medzi nimi', usudzovania na základe "zdravého" rozumu', 'skúsenosť', 'intencií', 'postojov', 'presvedčení' a množstva iných fenoménov. Pochopenie takýchto fenoménov a im zodpovedajúcich dejov v mysli by určite obohatilo možnosti ich realizácie počítačom. Na druhej strane, dá sa oprávnenne predpokladať, že okolnosti, za ktorých sa uvádzané procesy kognície dajú reliazovať počítačom, môžu pozitívne vplývať aj na spoznávanie a porozumenie kognitívnym procesom prebiehajúcim v mysli človeka.

Formálno-
logické
a informa-
ticky-teore-
tické

Jestuje veľký počet *všeobecných, abstraktných a konceptuálnych tém*, ktoré v kontexte ES a UI pútajú pozornosť. Bez rozboru uvádzame rad takých tém: kognitívna riešiteľnosť problémov a podmienky riešiteľnosti □ symbolová reprezentácia domény, v ktorých také problémy vznikajú □ súvisiaca vyjadrovacia účinnosť reprezentácie, odvodzovacia a výpočtová účinnosť, ktorú umožňuje □ formy stavového priestoru riešenia problému a zodpovedajúce princípy a procesy jeho prehľadávania □ metódy a postupy účinného rozkladu problému na podproblémy □ (ne-)monotónne hierarchické (stupňovité, kaskádovité) riešiace postupy □ spontánna aktivácia decentralizovaných a distribuovaných riešiacich postupov □ princípy pružných (fluidných) kognitívnych procesov, ich realizácie v procesoch spracúvania symbolov □ neuniverzálne a nehomogénne reprezentačne prostriedky reprezentácie znalostí (formalizmy logiky a neurónových sietí) □ prostriedky reprezentácie a uplatňovania ohraničujúcich podmienok a vzájomných väzieb medzi parametrami problémov □ rôznorodé súčasne uplatňovateľné prostriedky a metódy spracúvania neurčitostí/vierohodností.

Inteligentné
rozhrania

Medzi *formálno-logické a informaticky-teoretické* témy a otázky možno radíť aj uplatniteľnosť a ohraničenia logických formalizmov (výrokovej, predikátovej logiky, modálnej, viachodnotových, temporálnej a iných neklasických logík), obohatenie produkčných pravidiel o mimologické konštrukty a sémantická diferenciacia ich zložiek (a s tým súvisiace odvodzovacie postupy), cielené a zdôvodnené odstupovanie od prebiehajúcej nemonotónnej inferencie, prostriedky udržiavania konzistentnosti BZ a výsledkov odvodzovania, dynamické (vývoja schopné) BZ obsahujúce aj reprezentáciu nekonzistentných a spochyniteľných poznatkov, znalostí, či postojov (napríklad kontextom). Týka sa to aj otázok zabezpečovania bezspornosti, konzistentnosti a adekvátnosti reprezentovanej množiny znalostí, súcisiacou výpočtovou a odvodzovacou účinnosťou, včleňovania metód evolučného zdokonaľovania stratégií odvodzovania aj neuropočítania, uplatňovania paralelných a distribuovaných odvodzovacích a výpočtových procesov, tvorby a uplatňovania konceptuálnych sietí, subsymbolovej reprezentácie poznatkov, rozhodovania medzi alternujúcimi možnosťami riešiacich stratégií, metód, prostriedkov a postupov, ich evidovania, revidovania a nahradzovania,

Metodické
hľadiská

Medzi ďalšie problémové okruhy, ktoré vyžadujú skúmanie, rozpracovanie a hľadanie odpovedí na otvorené otázky sú aj tieto:

Inteligentné rozhrania: efektívna obojstranná interakcia používateľ-ES aj miomorečovými prostriedkami (grafmi, schémami, obrazcami), prostredníctvom napísaných či vyslovených textov (hoci iba ohraničeného) prirodzeného jazyka, vrátane spôsobilosti systému komentovať a zdôvodňovať proces odvodzovania aj

odvodených výsled

kov, schopnosť systému rozoznať významy ovplyvňované kontextom či situáciou, rozpoznávania úmyslu, cieľov, snáh používateľa a v súvislosti s tým cielene s nim interagovať pri odstraňovaní alebo aspoň minimalizovaní nejednoznačností a nedorozumení vznikajúcich v komunikačnom procese.

Dizajnerské a metodické aspekty: metodika navrhovania a dizajnu systému, vytvárania a testovania BZ, foriem interakcie používateľ-systém ako aj dátovým a programovým prostredím systému, grafickými a multimédiovými rozhraniami, riadením a trasovaním procesov odvodzovania, výberu prázdneho ES, resp. vývojového prostredia jeho tvorby.

Súčinnosť odborníkom z aplikačnej oblasti: metodika získavania, vyjadrovania znalostí aplikačného odborníka a ich transformácia do symbolovej reprezentácie pri tvorbe bázy znalostí, návrh a implementácia stratégií a techník odvodzovania, vysvetľovania a zdôvodňovania odvodených výsledkov, priebežné overovanie funkčnosti na prototypoch systému v procese jeho evolučnej (iteračnej) tvorby, overovanie a zdokonaľovanie predstáv jeho použiteľnosti v danej problémovej oblasti za daných podmienok a okolností používania, tvorba prostriedkov umožňujúcich testovanie systému na testovacích polygónoch (hoci aj softvérovo realizovaných) keď to spoľahlivosť systému vyžaduje.

Hoci s viacerými z vyššie uvedených tém sa zaoberáme v nasledujúcich textoch, predsa len zďaleka nie všetkými. V niektorých prípadoch iba náznakom, či stručne, s inými podrobne a aj z viacerých pohľadov. Ale radu otvorených otázok a málo rozpracovaných tém nevenujeme už ďalšiu pozornosť. Je to dané cieľmi výučby, prípustným rozsahom prednášok a prijateľným rozsahom týchto textov. Ich uvedenie si kladie za cieľ informovať čitateľa o šírke problematiky a prípadne ho aj motivovať k vlastným bádateľským a implementačným činnostiam (povedzme aj v rozsahu diplomových prác).

1.1 Poslanie expertných systémov

Ako už bolo uvedené, programové produkty stelesňujúce princípy *umelej inteligencie* majú zabezpečovať také procesy v hardvéri, resp. nim ovládaného zariadenia, ktoré v určitej aplikačnej oblasti dostatočne presvedčivo pripomína prejavy prirodzeného fenoménu ľudskej inteligencie. Aplikačných oblastí tohto druhu je nepreberné množstvo. Patria medzi ne napríklad

- intelektuálne náročné hry (napr. šach či go), krížovky, rébusy, hlavolamy, kryptoaritmetické úlohy a mnoho ďalších,
- dokazovanie teorémov,
- výpočtové procesy v algebre, riešenie úloh z geometrie, derivovanie a integrovanie, riešenie sústavy rovníc a nerovností, kvalitatívne modelovanie a simulácie, riešenie slovne formulovaných výpočtových úloh,
- interpretovanie napísaných aj vyslovovaných textov prirodzeného jazyka, preklad textov z jedného do iného prirodzeného jazyka, analýza príbehov a ich generovanie,
- konextom podmienená transformácia významov, identifikácia analógií,
- počítačové videnie, rozpoznávanie a porozumenie obrazov a scén,
- kognitívna robotika (inteligentné roboty),

- automatizovaná tvorba programov,
- medicínske, psychologické, biologické, technické, ekonomické, sociálne diagnostikovanie - identifikácia priamemu pozorovaniu nedostupného stavu skúmaných systémov, návrh spôsobu a postupu odstraňovania neželaných stavov (porúch, chorôb),
- plánovanie, navrhovanie organizačných štruktúr a postupov, navrhovanie zariadení a konštrukčných postupov.

Neinformovane -
né postupy -

Je zrejmé, že keby sa uvedené typy úloh riešili náhodnými postupmi alebo takými, ktorými sa preskúmajú všetky do úvahy prichádzajúce možnosti, teda náhodilo alebo exhaustívne, boli by to **naivné – neinformované – teda slepé postupy**. Nie sú to postupy, ktoré by sa dali považovať za prejav inteligencie človeka, napokon vzhľadom na ich neefektívnosť a neprijateľnú výpočtovú zložitnosť by pri riešení netriviálnych problémov boli zriedka uplatniteľné. V oblasti umelej inteligencie sa vylučujú – s výnimkou stuácií, v ktorých nevyhnutný počet možností vyžadujúcich preverenie je už iba veľmi malý.

slabé metódy
riešenia
problémov

Rané obdobie rozvoja UI bolo poznačené vyvíjaním **univerzálnych (všeobecných) riešiacich postupov (mechanizmov)**. Spočívali na nachádzaní všeobecných operácií, t.j. *elementárnych krokov usudzovania* a ich postupným reťazením s cieľom *identifikovať/odvodiť/nájsť výsledok*. Uplatňovanie všeobecných riešiacich krokov - práve pre ich univerzálnosť – je lákavé aj z hľadiska programovej realizácie. Ale, a to je podstatné, pri riešení netriviálnych problémov sú všeobecné operácie málo účinné - sú slabé⁵. *Všeobecné operácie (usudzovania)* práve kvôli svojej univerzalite nemôžu byť informované o konkrétnej realite ovplyvňujúcej a podmieňujúcej povahu problémov a ich riešení. Práve preto riešiace metódy spočívajúce iba na takých operáciách klasifikujú ako **slabé metódy**. Prekonať sa dajú iba použitím *poznatkov z oblasti, v ktorej problém vzniká*, teda **informovanými postupmi**.

a ich
prekonávanie

Poznatky sú prostriedkom umožňujúcim uskutočňovať **cielené (špecifické, problémovo-orientované)** riešiace kroky, ktoré zodpovedajú postupom experta (vybaveného spravidla množstvom poznatkov aj skúseností).

Riešenia vyššie uvedených ilustračných kategórií úloh patria medzi také, ktoré

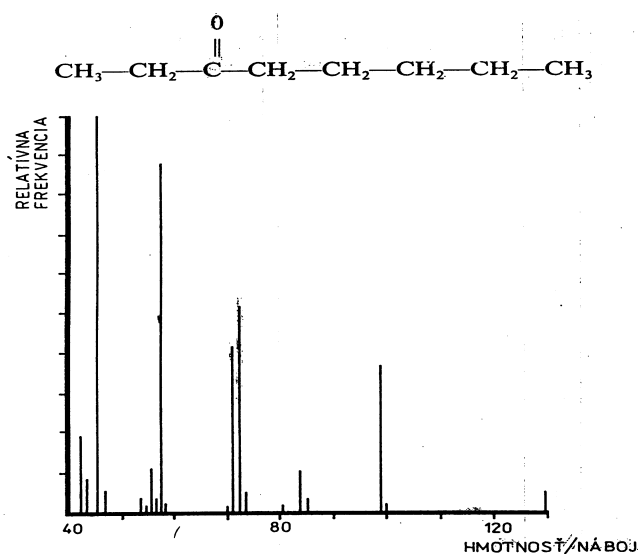
popri všeobecných znalostiach a postupoch v rozhodujúcom rozsahu uplatňujú špecifické znalosti a postupy. A pokiaľ tieto sú profesionálne - odborné - z danej oblasti a uplatňujú sa postupmi pripomínajúcimi príslušných expertov patria do domény expertných systémov.

DENDRAL -
implikácia
nevyhnutnosť

⁵ Nepochybne, všeobecný gravitačný zákon, za formuláciu ktorého vďačíme Newtonovi, nemožno ani na okamih spustiť zo zreteľa. Avšak úspešný štart a pristávanie lietadla, popri striktnnej konformite so všeobecnými zákonmi gravitácie, mechaniky a aerodynamiky, nevyhnutne vyžaduje zohľadňovanie ďalších špecifických fyzikálnych a technologických zákonitostí vyplývajúcich aj z vlastností daného lietadla, špecifickej aktuálnej poveternostnej situácie, smeru vetra, viditeľnosti, charakteru štartovacích/pristávacích dráh, aktuálnej premávky na letisku i v jeho okolí, vrátane potenciálneho výskytu vtáctva v danom priestore, ako hoci aj prípadného opotrebovania, či. poškodenia niektorej pozornosť si zasluhujúcej súčiastky. Jednoducho, len s gravitačným a ani s ostatnými všeobecnými fyzikálnymi zákonmi sa nedá vystačiť, hoci je ich nevyhnutné rešpektovať.

i špecifických
poznatkov

Takéto vymedzenie nebolo samozrejším v počiatkových fázach vývoja ES. V tejto súvislosti je vhodné stručne uviesť historicky mimoriadne významnú úlohu, ktorú zohral systém DENDRAL (Feigenbaum, Buchanan, Lederberg, 1969). Tento pozoruhodný a úspešný programový produkt sa stal pragmatickým dôkazom opodstatnenosti uvedeného vymedzenia. Jeho autori⁶ sa rozhodli vytvoriť systém schopný odvodiť štruktúru chemickej molekuly z údajov, ktoré poskytuje hmotový spektrometer. Vstupmi do programu boli jednak sumárny vzorec molekuly (napr. $C_6H_{13}NO_2$) spolu s nameraným spektrogramom informujúcim o hmotnosti rôznych zložiek molekuly (pozri obr. 1.1). Program obsahoval množstvo poznatkov o rôznych možných tvaroch a štruktúrach spektrogramov zodpovedajúcich reálne existujúcim chemickým látkam.



Obr. 1.1

Tak napríklad stelesňoval poznatok, že pík (vrchol) v pozícii zodpovedajúcej hmotnosti $m=15$ v hmotovom spektrograme molekuly svedčí o prítomnosti metylovej zložky (CH_3) v tejto molekule.

Princíp pôvodnej naivnej verzie systému odvodzovania štruktúry molekúl z ich sumárneho chemického vzorca a zodpovedajúceho hmotového spektrogramu je možné vyjadriť v nasledujúcej podobe:

- (1) systém vygeneruje všetky možné chemické štruktúry konzistentné so zadaným sumárnym vzorcom,
- (2) každej vygenerovanej chemickej štruktúre priradí (vygenerovaný alebo už pamätaný) tvar zodpovedajúceho hmotového spektrogramu,
- (3) vhodnou metódou nachádzania zhody vyhladá spomedzi priradených spektrogramov taký, ktorý je v najväčšej zhode s práve skú-

⁶ Ed Feigenbaum (pôvodne študent Herberta Simona), Bruce Buchanan (pôvodom filozóf, ktorý sa stal informatikom) a Joshua Lederberg (genetik, nositeľ Nobelovej ceny).

⁷ Pravidlá tohto typu sa nazývajú **produkcčné**. Podrobne sa nimi zaoberáme v ďalšom. Podoba pravidla zodpovedá tvaru, v ktorom je uvádzaný v (Russell, Norvig, 1995).

maným spektrogramom,
(4) vybraný spektrogram určuje zisťovanú štruktúru molekuly.

Aplikácia tejto metódy narazila však na vážny problém *bariéry výpočtovej zložitosti*. Je to dané tým, že jedinému sumárnemu vzorcu už nevelkej molekuly môže zodpovedať až prekvapujúco veľký počet prípustných spektrogramov.

Autori systému DENDRAL-u konzultovali tento problém u expertov z oblasti analytickej chémie. Zistili, že títo namiesto exhaustívnych postupov uplatňujú postupy cielené: *Vyhľadávajú pre reálne existujúce podštruktúry molekuly s daným sumárnym vzorcom dobré známe zoskupenia (útvary, vzorce) pík v hmotovom spektrograme*. Také, čo sú typické. Napríklad, keď sa dá predpokladať, že sa v skúmanej molekule vyskytuje *ketónová podskupina* (C=O), pri analýze spektrogramu, v snahe jej výskyt potvrdiť, použijú nasledujúce pravidlo⁷:

Ilustrácia
špecifického
poznatku v
podobe
produkčného
pravidla

AK v hmotovom spektrograme existujú dve pozície x_1 a x_2 , v ktorých sú píky také, že

- (a) $x_1 + x_2 = M + 28$ (M je hmotnosť celej molekuly),
- (b) $x_1 - 28$ je vysoký pík,
- (c) $x_2 - 28$ je vysoký pík,
- (d) aspoň jeden z x_1 , x_2 obsahuje vysoký pík,

TAK molekula obsahuje ketónovu podskupinu.

Rozpoznanie výskytu určitých konkrétnych prvkov štruktúry molekuly výrazne znižuje počet alternatív následnej analýzy a tým sa riešenie problému stáva priechodným, alebo aspoň priechodnejším. Zmenou pôvodného prístupu k tvorbe DENDRAL-u tým, že sa v ňom uplatnili pravidlá (sú reprezentáciou špecifických poznatkov) uvedeného typu vznikol praktický a výkonný systém. Autori to vysvetlili takto:

Poznatkovo
intenzívny
systém

"Všetky relevantné teoretické znalosti potrebné na vyriešenie predmetných problémov sa pretransformovali z ich všeobecného tvaru [v zložke generovania zodpovedajúcich hmotových spektrogramov] ("počiatočný princíp") do účinných špeciálnych tvarov ("receptov v kuchárskej knižke")." (Feigenbaum et al. 1971)

Význam úspešného DENDRAL-u spočíva(1) v tom, že odhalil

- (1) *možnosť a spôsob spôsob vytvárania poznatkovo-intenzívneho (informovaného) systému uplatňovaním explicite symbolovo reprezentovaných (špecifických) znalostí reprezentovaných adekvátnym počtom expertami používaných špeciálnych účelových pravidiel,*
- (2) *princíp umožňujúci vytvárať programy schopné riešiť problémy, ktorých riešenia pri klasických metódach programovania narážali na medze výpočtovej zložitosti.*

DENDRAL sa stal zdrojom poznania:

Kľúčová úlo-
ha znalostí

Efektívne riešiacie postupy takých úloh, ktoré svojou formuláciou riešiaci postup neimplikujú, spočívajú na účelnom používaní potrebného rozsahu všeobecných a/alebo špecifických znalostí z oblasti, v ktorej riešený problém vznikol.

Znalostné systémy

ZNALOSTI sú prostriedkom umožňujúcim stratégiami riadené exploračné problémového priestoru pri efektívnom produkovaní správnych riešiacich postupov.

Expertné systémy

V období formulovania týchto výrokov vôbec sa nezdali takým samozrejým a prirodzeným ako ich vnímame v súčasnosti, keď sa všeobecne prijímajú nasledujúce dve vymedzenia

Programové systémy, ktoré opierajú svoju činnosť o symbolovú reprezentáciu znalostí, nazývame znalostné (na znalostiach založené) systémy

a tie z nich

čo sú stelesnením odborných znalostí z určitej profesnej oblasti a slúžia na podporu (prípadne aj nahradenie) odborníkov (expertov) pri riešení odborných problémov (tvoria podmnožinu znalostných systémov) zodpovedajú expertným systémom.

1.2 Hypotézy v pozadí inteligentných symbolových systémov

Úvahy o tvorbe programových prostriedkov spracovajúcich symboly vytvárajúce efekt zodpovedajúci *fenoménu inteligencie* je vhodné aspoň v krátkosti zasadiť do určitého rámca. Nejde pritom o snahu poskytnúť ucelený a už vôbec nie vyčerpávajúci pohľad na problematiku. Odporúčanú prezentáciu kvalifikovaných stanovísk k tejto téme možno nájsť v článku *Kognícia bez mentálnych procesov od J. Šefránka*⁸. Tento článok iba naznačí "príchut" témy so zámerom motivovať čitateľa k samostatnej analýze náročnej problematiky na základe uvádzaných hypotéz.

Bežné laické, ale často aj odborné pozorovanie a sebaopozorovanie (introsepekcia) úspešných procesov myslenia obchádza otázku **povahy týchto procesov, spôsobov ich uskutočňovania**, ba ani túto otázku si nekladie⁹. Vedeckým skúmaním procesov myslenia sa zaoberajú *kognitívne vedy*. Tie sa z rôznych pohľadov a na líšiach úrovniach štruktúr vytvárajú **modely procesov myslenia**.

Myslenie

Ústredná idea kognitívne sa dá formulovať takto:

Myslenie spočíva v interných manipuláciách s mentálnymi modelmi reality či predstáv o nej, ktoré si ľudia vytvárajú neartikulovateľnými mentálnymi procesmi

Mentálny model – začlenená sústava poznatkov

A mentálny model sa dá považovať z

aspoň minimálne nevyhnutné zoskupenie znalostí o existencii, príčinách, princípoch a spôsoboch vzniku, priebehu (trvania) a zániku fenoménu/javu, jeho vlastnostiach, štruktúre (zloženia) a správaní,

⁸ Kap. 6 v Rybár J., Beňušková E., Kvasnička V. (eds.): *Kognitívne vedy*, Kalligram, Bratislava, 2002.

⁹ Nekladie si otázky typu: Prečo nedokážeme súčasne rozmýšľať o viacerých záležitostiach, aký je proces porozumenia významu slov, čo sa odohráva v našej myslí keď násobíme 7 so 4, čo je podstatou procesu zapamätávania a neskoršieho vybavovania zapamätaného, prečo rovnako znejúcu úlohu odborníci z príslušnej profesie riešia rozličnými postupmi, rôznou rýchlosťou a s líšiacou sa úrovňou kvality?

o jeho príčinách, dôsledkoch, pôsobení, implikáciách, o jeho ovplyvniteľnosti, riadení, vyvolaní, znemožnení, o jeho vzťahu a podobnosti k iným záležitostiam, o predvídateľnosti jeho vzniku, zániku a správania.

Porozumenie

Na základe mentálneho modelu má človek očakávania, aspirácie, volí si ciele, tvorí plány, selektívne vyberá hypotézy a informácie, ktoré vyhodnocuje a spracúva, prijíma, vytvára premisy a propozície a aj na ich základe hypotézy dokazuje či zamietá, potvrdzuje alebo sponchybňuje.

Mentálny model je fenomén, na základe ktorého v interakcii individua s okolím (s inými individuami, spoločenstvami, so živou aj neživou prírodou, s fyzickými či duchovnými dielami človeka) dochádza k procesom **porozumenia**. Sú to procesy, v ktorých individuum

- (a) prijímané informácie dokáže bezprostredne (významovo a propozične) **interpretovať v rozsahu svojho mentálneho modelu - teórie a/alebo**
- (b) vďaka prijímaným informáciám **modifikuje svoj mentálny model** (doplňuje, mení, vylučuje niektoré jeho prvky, vzťahy medzi nimi, premisy a propozície), čím im priradzuje (významovú a propozičnú) interpretáciu. Možno vysloviť tvrdenie

Porozumenie niečoho, hoci len čiastočné, predpokladá vytvorenie zodpovedajúceho mentálneho modelu, t.j. určitého vzájomne previazaného (začleneného) súboru súvisiacich poznatkov vytvárajúcich znalosť daného – teóriu, resp. presvedčenie o danom.

Model

Spôsobilosť čosi *vysvetliť* predpokladá, že danému bolo porozumené. Zároveň úspešné vysvetlenie je potvrdením porozumenia. Proces vysvetľovania sa odohráva najmenej medzi dvoma účastníkmi **spôsobilými porozumieť téme**. *To znamená, že príjemca vysvetľovania musí už disponovať určitými poznatkami – mentálnym modelom, aby výklad mohol pochopiť.*

Model, nezávisle na materiálnej podstate jeho realizácie, by mal reflektovať štruktúru modelovanej predlohy a musí byť spôsobilý uskutočňovať analogické procesy (chovanie) ako nim simulovaná (imitovaná) predloha (vzor-systém).

V ohnisku našej pozornosti je teda principiálna otázka: Dajú sa vytvoriť modely, ktoré funkčne umožňujú simuláciu myslenia? Otázka je zrozumiteľná a vari aj dobre formulovaná. Odpoveď však nie je ani jednoduchá, ani jednoznačná¹⁰.

V ďalšom, bez dokazovania, vychádzame z predpokladu, že **pre modely a simuláciu myslenia** nemusí byť podstatným ľudský mozog (prirodzený nositeľ – realizátor myslenia) so svojimi mechanizmami vnímania a produkovania akcií. Namä vtedy nie, keď uvažujeme o takých procesoch v mysli, ktoré nie sú tvarované

¹⁰ Problém je už s jednoznačným vymedzením obsahu pojmu 'myslenie'.

¹¹ Záujemcom o hlbšie preniknutie do tejto problematiky možno odporučiť rad literárnych prameňov. Spomedzi oponentov strojového myslenia treba siahnuť najmä po dielach autorov Searla a Penrosea. Pozornosť si zasluhuje súhrn náhľadov na túto problematiku ako ju uvádzajú Russell, Norvig, 1995, kapitola 26. Medzi výrazných oponentov dá sa radiť J. R. Searle. Vyplýva to aj z jeho *diela Minds, Brains and Science*, Harward University Press, Cambridge, Ma, 1984, český preklad *Mysl, mozek a věda*, Mladá fronta, Praha 1994.

¹² Nimi sa podrobne zaoberáme v ďalších kapitolách.

Abstrakcia:
symbolové
operácie

emóciami, intenciami, či inými možnými psycho-fyziologickými faktormi, teda keď procesy v mysli majú racionálnu *mentálnu* povahu pri produkovani **kompetentných** riešení úloh a dosahovania cieľov. Ide pritom o *hypotézu symbolizmu*, ktorá si nekladie za cieľ zahrňať aj procesy, ktoré či už na biologickej alebo fyzikálnej úrovni vedú k vzniku symbolov a ich významov. Podľa tejto hypotézy

operácie so symbolmi, t.j. ich vytváranie/rušenie, spájanie/rozčleňovanie, štruktúrovanie, menenie, porovnávanie, zaraďovanie/preraďovanie/prirad'ovanie a podobne, umožňujú realizáciu modelov a sú prostriedkom simulácie procesov myslenia - prirodzene iba vtedy, keď je "svet" vhodne reprezentovaný zodpovedajúcimi symbolmi.

Táto hypotéza - spočívajúca okrem iného na skutočnosti, že aktivity neurónov v mozgu sa prejavujú procesmi ich *excitácie* a *inhibície* (*zapnuté-vypnuté; môžu byť teda iba v dobre známych binárnych stavoch, hoci prechod medzi nimi nie je úplne jednoduchý proces*) – by mohla implikovať možnosť vytvárať prostriedkami počítača modely myslenia. Hypotéza korešponduje so snahami UI počítačom realizovať pracovné modely (kompetentného) myslenia a jeho simulovanie.

Uvádzame niekoľko ďalších hypotéz, ktoré prispievajú k rozvíjaniu a približovaniu predstáv o spôsobilosti počítača *myslieť*, resp. vytvárať v niektorých oblastiach *zdanie myslenia*.

Treba však zdôrazniť, že jestvuje nezanedbateľný počet renomovaných odborníkov, ktorí tieto hypotézy prijímajú s výhradami, prípadne ich aj s implikovanými dôsledkami odmietajú. Myslenie považujú za prejav výlučne ľudského (prípadne ešte živočíšneho) mozgu (*vetvér*). Pokiaľ ide o hardvér (*drajvér*), jeho spôsobilosť myslieť až radikálne vylučujú, hoci pripúšťajú, že zdanie myslenia dokážu vyvolať.¹¹

Prvá z nasledujúcich hypotéz je reflexiou materiálnej podstaty myslenia a druhá je vlastne reflexiou existencie psychológie.

Mozog
realizuje
rekurzívne
funkcie

Skúmanie
myslenia

H-1 Materiálna podstata neurofyziologického substrátu (mozgu), bez toho, aby kládla ohraničenia, poskytuje biofyzikálne základy procesov myslenia tým, že má schopnosť realizovať rekurzívne funkcie - teda to čo sa dá uskutočňovať Turingovým strojom.

H-2 Myslenie sa dá skúmať nezávisle od skúmania mozgu, t.j. nezávisle od neurofyziológie.

Psychologická
teória a TS

Aj nasledujúcich 6 hypotéz o myslení pochádza z "myšlienkovvej dielne" významného kognitívneho psychológa Johnson-Lairda z rokov 1987-89. Ako psychológ hlboko prenikol do teórie informatiky aj praktických otázok programovania. Vďaka tomu jeho psychologicko-informatické hypotézy nepochybne poskytujú množstvo motivujúcich podnetov.

Psychologická
teória a
vypočítateľné
funkcie

H-3 Psychologická teória sa dá považovať za schopnú čosi vysvetliť iba vtedy, keď je ju možné formulovať ako efektívnu procedúru (Turingov stroj - TS).

Produkčné

H-4 Psychologické teórie (pokrývajúce rozsiahle oblasti myslenia) sa dajú po-

pravidlá a
možnosť
tvorby teórií
kognície

nímať ako vypočítateľné funkcie. Jestvujú však také aspekty ľudskej mentality, ktoré sa nedajú vystihnúť žiadnou teóriou. Zároveň však nemožno vylúčiť rozšírenie konceptu efektívnej procedúry (TS) tak, aby bol schopný zahrnúť viac než množinu súčasne vypočítateľných funkcií a tým pokryť aj ďalšie mentálne fenomény.

Mentálne a
výpočtové
procesy

H-5 Produkčné pravidlá¹² vytvárajú gramatiku, ktorá je ekvivalentná univerzálnemu TS, pričom sú považované za základňu psychologicky orientovaných teórií vypočítateľnosti, ktoré umožňujú tvorbu špecifických teórií kognície.

O možnosti
simulovať
teórie myslenia

H-6 Funkčná organizácia mentálnych procesov môže byť charakterizovateľná pojmom efektívnych procedúr, pretože schopnosť mysle vytvárať pracovné modely zodpovedá výpočtovému procesu.

Myslenie a vy-
počítateľnosť

H-7 Akceptovanie hypotézy H-6 (funkcionalistický pohľad) implikuje (a) možnosť vhodnými programami simulovať teórie myslenia, (b) principiálne aj možnosť nimi (aspoň z časti) realizovať racionálne myslenie.

H-7 hovorí o tom, že myslenie, pokiaľ je v určitom rozsahu stotožniteľné s výpočtovým procesom, dá sa realizovať aj počítačom.

H-8 Ak Turingové tézy a funkcionálny pohľad sú správne, tak vznikajúce teórie myslenia budú vyjadriteľné v termínoch vypočítateľnosti.

K hypotéze H-8: Problém spočíva v tom, že k danej vypočítateľnej funkcii existuje mnoho efektívnych procedúr, ktoré sa môžu vzájomne až principiálne líšiť. Z toho plynie, že aj keby sme presne poznali funkcie výpočtovo zodpovedajúce jednotlivým mentálnym procesom, aj tak by vznikli problémy s nachádzaním (identifikáciou) procedúr, ktoré sú ich schopné vypočítať (Marr, 1977). Napriek tomu tvorba programových systémov **schopných vytvárať pracovné modely myslenia** je oprávneným predmetom teoretického aj praktického záujmu UI. Opiera sa o zásadné myšlienky o možnostiach využívania počítačov v úlohe **prostriedku modelovania procesov operácií so symbolmi**, ktoré zodpovedajú najcharakteristickejšej ľudskej činnosti - mysleniu. Základy týchto myšlienok vznikli už v priebehu *Dartmouth Summer Conference on Artificial Intelligence v roku 1956*. Následný vývoj vhodných programovacích prostriedkov a experimentovania s nimi vytvorili predpoklady na vyslovenie základnej hypotézy motivujúcej rozvoj UI. Vyslovili ju A. Newell a H.A. Simon v r. 1976 a nazýva sa **hypotéza fyzikálneho symbolového systému**. Jej formulácia je nasledovná:

Hypotéza
fyzikálneho
symbolového
systému

H-9 Číslíkový počítač splňuje nutné a postačujúce podmienky pre vykonávanie inteligentnej činnosti.

E. A. Feigenbaum, vedúca osobnosť kolektívu tvorcov systému DENDRAL zo Stanfordskej university, obohacuje H-9 tzv. **znalostnou hypotézou**:

Znalostná
hypotéza

H-10 Systém (počítač) dokáže vykonávať inteligentnú činnosť na vysokej úrovni kompetencie vďaka znalostiam, ktoré má z oblasti svojich aktivít

Propozičná
hypotéza

(znalosti zodpovedajúce oblasti, v ktorej riešený problém vzniká).

a jej implikácie

Pokračujúci rozvoj poznávania rozvíja a ďalej obohacuje základné myšlienky. Príkladom je B. Smith z Massachusetts Institute of Technology, ktorý v roku 1982 významne rozvinul hypotézu H-10, tým že formuloval nasledovnú hypotézu (nazvime ju **propozičná**):

H-11 Každý (určitým mechanizmom) realizovaný inteligentný proces je tvorený štruktúrnymi prvkami, ktoré

- (a) **sa vonkajšiemu pozorovateľovi správania systému javia ako propozičný prejav znalostí,**
- (b) **nezávisle od vonkajšieho (sémantického) hodnotenia zabezpečujú (sú príčinou) správania systému, ktoré je manifestáciou jeho znalostí.**

Táto hypotéza osvetľuje významné stránky (mechanickej) inteligencie:

- (a) **Nevyhnutnosť symbolovej reprezentácie znalostí propozične interpretovateľnými prvkami** (možno im priradiť pravdivostné hodnoty) **vytvárajúcimi konzistentne interpretovateľné vety určitého jazyka.**
- (b) **Inkorporovanie zodpovedajúcich viet do systému** (mechanickej inteligencie) **je nevyhnutným predpokladom jeho inteligentného správania.**

Fenoméni
inteligentného
procesu
a človeka

Uvedené hypotézy rozvíjajú a spresňujú výroky týkajúce sa počítačom realizovaných procesov s atribútom inteligentného správania. Hovoria o tom, že také procesy

nevystačia s realizáciou iba vopred daných jednoznačných deterministických návodov.

Schopnosť manifestovať fenomén inteligencie spočíva na

nemonotónnom (revidovateľnom, odvolateľnom) situačnom uplatňovaní prehľadávacích procesov, ktorých postupnosť nie je vopred deterministicky daná.

Je to v súlade s poznatkami o myslení človeka. Ten

- *spravidla dokáže svoje špecifické, t.j. problémovo zamerané, a všeobecné znalosti efektívne situačne používať aj v prípade problémov, pre riešenie ktorých nemá vopred daný presný deterministický predpis; **dokáže teda vykonať viac, než na čo je "naprogramovaný"**; navyše dokáže svoje znalosti rozširovať, prehĺbovať, zdokonaľovať,*
- *keď vôbec nemá k dispozícii predpis na riešenie daného problému, alebo len taký, ktorý je neúplný, nedeterministický, alebo inak nezaručujúci rezultatívnosť, používa rôzne (mentálne) techniky vyhľadávania využiteľných riešacích krokov k vytváraniu takej revokovateľnej postupnosti z nich, ktoré mu napokon umožnia dospieť k želanému výsledku - teda postup riešenia si vytvára, produkuje.*

a počítač

Podstaným je
realizovateľnosť
funkcií

Obdobné spôsobilosti sa očakávajú aj od fyzikálneho symbolového systému pokiaľ má patriť do kategórie informačných technológií majúcich oprávnenie na atribút *inteligentná*. V predošlom uvedené hypotézy podporujú presvedčenie, že by

to malo byť v určitom (doteraz definitívne nevymedzenom) rozsahu možné:

počítač by mal byť spôsobilý vykonať viac, než mu je programom predpísané!

Deklaratívny
štýl programovania

Fyzikálne princípy (pneumatické, hydraulické, reléové, elektrónkové, polovodičové, integrované obvody vysokej hustoty) a funkcie (registre, organizácia ALU, pamäte, vykonávateľné kódy) konkrétneho počítača nie sú z tohto pohľadu relevantné. Podstatné sú **funkcie** počítačom vykonávaného **programu**.

Ten istý program môže byť preložený do úplne iného výkonávateľného kódu zbiehateľného na konštrukčne inom počítači, ktorý uskutočňuje svoje činnosti (aj fyzikálne) odlišným spôsobom a napriek tomu realizuje tú istú funkciu (konformné s H-2). Teda pre dokonalé pochopenie činností, ktoré môže vykonávať *fyzikálny symbolový systém*, je podstatné pochopiť funkcie programu a princípy umožňujúce realizáciu želaných funkcií.

Mnohé z týchto princípov sú už známe. Spoznalo sa, že ich programová realizácia vyžaduje uplatňovať ďalší **štýl programovania**. Jednou z podstatných možností je doplnenie (nahradenie) **procedurálneho** programovania **deklaratívnym**. To je téma, ktorej je venovaná väčšina z nasledujúceho výkladu.

1.4 Procedurálne a deklaratívne programovanie

Neurónové (konekcionistické) siete, genetické algoritmy, objektovo-orientovaná a deklaratívna paradigma programovania vznikli ako dôsledok úsilia riešiť triedy problémov, ktoré sa stali a stávajú predmetom pozornosti a potreby až v pomerne nedávnom období. Je to aj dôsledok výrazného rozvoja technických prostriedkov (hardvér aj softvér) a vedeckých poznatkov.

Skutočnosť, že formulácie v UI uvažovaných tried problémov neimplikujú zreteľne formulované riešiacie postupy v klasickej algoritmickej podobe, vedie k potrebe rozšíriť predchádzajúce ponímanie prístupov k programovaniu. V tomto kontexte je dôležité ozrejmiť predovšetkým odlišnosti a vzťah medzi **klasickým procedurálnym** a iným (novším) **deklaratívnym štýlom programovania**.

Princíp klasického **procedurálneho** spôsobu **programovania** je tvorba postupnosti príkazov, pre ktorú platí:

Princíp procedurálneho programu

Ak z práve vykonávaného kroku priamo nevyplýva krok, ktorý má následne vykonať, tak sa vykoná ten, ktorý bezprostredne nasleduje v poradí.

Podstatou uvedeného je veľmi **jednoduchý princíp riadenia** zabezpečujúci zbiehanie procedurálneho programu: Nech **n** je prirodzené číslo označujúce poradie inštrukcie v programe. Nech **m** je celé číslo a NIL zodpovedá prázdnej inštrukcii. Princíp riadenia možno potom formulovať v nasledujúcom **funkčnom vyjadrení**

$$f(n) = \begin{cases} n+m & \text{(v prípade odskoku)} \\ n & \text{(cyklus)} \\ \text{NIL} & \text{(keď sa proces zastaví)} \\ n+1 & \text{(implicitné riadenie)} \end{cases}$$

Jednoduchosť riadiaceho algoritmu má za následok nevyhnutnosť domyslieť do všetkých podrobností situácie, do ktorých sa spracúvanie údajov môže dostať, a spôsoby, ako sa v nich má výpočtový proces správať. Vedie to k potrebe tvorby **riadiacej infraštruktúry výpočtových procesov**. Zodpovedajúce príkazy sú spravidla roztrúsené v tele programov, čo spôsobuje, že riadiaca infraštruktúra je často neprehľadná a ťažko zrozumiteľná dokonca aj samotnému autorovi procedurálneho programu.

Expertné systémy sa vytvárajú na riešenie problémov, **pre ktoré nie sú vopred dané deterministické algoritmy**. Nimi realizované riešiacie postupy sú založené **iba na situačne uplatňovaných predpisoch spočívajúcich na poznatkoch** umožňujúcich **vyhľadávať revokovateľné postupnosti využiteľných riešiacich krokov potenciálne vedúcich k želanému výsledku**. A preto programová realizácia zodpovedajúcich procesov vyvoláva potrebu odlišného štýlu programovania v porovnaní s klasickým procedurálnym štýlom, vedie k

Potreba zmeny spôsobu programovania

tvorbe vhodne organizovaných a štrukturovaných zoskupení symbolových konštruktov reprezentujúcich zákonitosti (pravidlá, vzťahy, väzby, ohraničenia, predpisy, výnimky atď.) danej problémovej oblasti.

Deklaratívny štýl programovania

To tvorí princíp deklaratívneho (situačného) štýlu programovania, v ktorom sa **zdôrazňujú sa nim zákonitosti riešenia jednotlivých druhov problémov, pričom sa do značnej miery ich symbolové vyjadrenie (reprezentácia) separuje od vlastných (univerzálnych) riešiacich procesov.**¹³

Riešiteľnosť

Pri deklaratívnom programovaní riešiteľnosť problému nezávisí od žiadnej konkrétnej riadiacej stratégie. **Riešiteľnosť** problému je **závislá** iba od (vzájomne nezávislých alebo len málo závislých) **symbolových výrazov reprezentujúcich zákonitosti, ktorým podlieha riešenie problému.**

Kvalita deklaratívneho programu je podmienená kvalitou

- **analýzy predvídateľných situácií (stavov) vyskytujúcich sa v danej triede riešených problémov,**
- **zodpovedajúcich akcií, ktoré sú im priradené, a**
- **symbolovej reprezentácie takých situačno-akčných prvkov.**

Kvalita deklaratívnych programov je nezávislá, prípadne len veľmi málo závislá od vlastných výpočtových procesov. **To, či a kedy jednotlivé situácie vzniknú sa nemusí v nich predvídať.**

Vznik jednotlivých situácií sa nemusí predvídať, ale

Jednotlivé situácie zodpovedajú zostavám

- **známych údajov** – tie môžu byť *dodané* (používateľom, prostredím), získané (dotazovaním používateľa alebo vonkajšieho prostredia), *odvodené* (použitou metódou odvodzovania, vypočítané, dedené), *predpokladané* a ešte nepochybné (pokiaľ inými dostupnými prostriedkami sa v danom stave nedajú zistiť),
- **chýbajúcich a nedostupných údajov,**

¹³ Univerzálne riešiacie procesy majú nezanedbateľné limity. Ich prekonávanie postupov špecifickými generickými makro-operáciami je predmetom výkladu mnohých ďalších častí týchto textov.

- **údajov korešpondujúcich s obsahom infraštruktúry** riešiacich procesov – zodpovedajú prebiehajúcim stavom riešeného problému.

Zostavy týchto údajov charakterizujú riešený problém aj jeho priebeh. *Kedže zbiehanie riešiaceho programu je podmienené výskytom situácií a tie sú určované údajmi, hovoríme o situačnom alebo o údajmi riadenom vykonávaní programu.*

s ich možným výskytom sa musí rátať

Pri deklaratívnom programovaní teda stačí, keď sa vhodne reprezentujú zákonitosti problémovej oblasti. Následne, pri riešení problémov sa nemusí formulovať ako sa má postupovať, **stačí uviesť ČO sa má riešiť**.

Interpretátor

Postup riešenia je zabezpečovaný nevyhnutným komplementom - **AKO** komponentom – deklaratívnych programov, ktorý sa nazýva **INTERPRETÁTOR**. To je klasický v širokom rozsahu univerzálny program slúžiaci **inferovaniu (odvodzovaniu)**, čoho integrálnou súčasťou je aj **koordinácia procesov** – inferenčná infraštruktúra - dosahovania požadovaných cieľov. Tým, že stelesňuje všeobecné (univerzálne) čiastkové riešiace procedúry,

automaticky – na základe interpretovania symbolovo vyjadrených zákonitostí problémovej oblasti a z nich plynúcich dôsledkov - vyvoláva a koordinuje ich činnosť a tak zabezpečuje realizáciu procesov potrebných pre riešenie problému.

Vďaka deklaratívnemu programu nie je potrebné vopred deterministicky predpisovať jednotlivé akcie (operácie) ani ich postupnosť.

Interpretátor - napr. na základe špecifikovaných riadiacich operácií - vyhladá, vyberie a zabezpečí vykonanie operácií vtedy, keď ním detekovaná situácia akciu vyžaduje alebo umožňuje.

Interpretátor je prostriedkom, ktorý umožňuje vytváranie deklaratívnych programov. Odbremeňuje tvorcu takých programov od nevyhnutnosti predvídať postupnosť vykonávania jednotlivých krokov deklaratívneho programu, teda starosť o riadenie jeho zbiehania a vytvárania zodpovedajúcej infraštruktúry.

Zbiehanie deklaratívneho programu

Vyhľadávanie a vyberanie riešiacich operácií (riešiacich krokov) je **nedeterministický proces**. Keďže iba niektoré postupnosti operácií potenciálne vedú k požadovanému výsledku, vopred však nie je však známe ktoré, stáva sa nevyhnutným zabezpečiť možnosť od nich **odstúpiť, tým spravidla aj revidovať (odvolať, zrušiť) dovtedy vykonané akcie a následne voliť iný postup.**

Zbiehanie deklaratívneho programu je riadené (vo všeobecnosti) netriviálnou procedúrou. Na jej priblíženie použijeme túto symboliku:

- s_k – **stav (stuácia) riešenia problému,**

¹⁴ Množina aplikovateľných operácií implikuje vznik nedeterminizmu !

¹⁵ V stuáciach, pre ktoré absentujú potrebné poznatky, táto množina môže byť aj prázdna. Taký prípad vyžaduje prostriedok ošetrenia vzniknutého stavu.

¹⁶ Napríklad usudzovanie na základe dedukcie, indukcie, abdukcie, asociovania, zovšeobecňovania, podobnosti, analógií, vylučovania a komplementovania, generalizácie alebo špecializácie, príčin alebo dôsledkov, nezáväzných hypotéz, dynamického zjednodušovania, rozkladu, syntézy a interakcií problémov, uplatňovaných ohraničení, substituovania, unifikácie, zohľadňovania neurčitostí, kvalitatívnych úvah, spoľahlivosti, vierohodnosti, presnosti, špecificity, senzitivity, počtu a konzistencie získaných aj odvodených údajov, či na základe ich diskriminačnej účinnosti, aplikovateľnosti, kvality a spoľahlivosti metód a postupov, nevyhnutných chronologických či cielených revízií, rezolšovania protirečení a konfliktov, zameriavania pozornosti atď.

- **g** - zobrazenie stavu na množinu aplikovateľných (prípustných) operácií $\{o_k\}$,¹⁴ t.j. situačno-akčných pravidiel¹⁵,
 - **f** – funkcia výberu vhodného operátora (pravidla, povelu, inštrukcie) - ktorá vo všeobecnosti môže byť pomerne netriviálna,
 - **n** – poradové číslo vybratého operátora, ktoré pri ukončení činnosti je nahradené príslušným symbolom NIL.
- Teda riadiaca procedúra zodpovedá **zobrazeniu** a následnej **aplikácii výberovej funkcie**. Možno to vyjadriť v nasledujúcej podobe

$$f(g(s_k)) = f(\{o_k\}) = \begin{cases} \mathbf{n} \\ \mathbf{NIL} \end{cases}$$

Realizácia
výberovej
funkcie

Výberová funkcia **f**, v závislosti na dômyselnosti jej realizácie, môže na jednej strane zodpovedať triviálnym, na druhej strane až veľmi zložitým a sofistikovaným, **heuristikami** podmieneným, procesom. Uvádzame niektoré možnosti:

- mechanický výber najmenšieho (prvej v poradí), či najväčšieho (poslednej v poradí) **n**, alebo náhodný výber operácie (akcie) z množiny tých, ktoré sa javia uplatniteľné,
- výber operácie (akcie), ktorá
 - ✓ bola posledne prerušená, teda nedokončená,
 - ✓ je buď zo všetkých prípustných najvšeobecnejšia alebo najšpecializovanejšia,
 - ✓ posledne vykonanú najlepšie doplňuje (rozvíja ju), alebo je práve jej opakom,
 - ✓ sa posledne vykonanej a iba čiastočne úspešnej najviac podobá,
 - ✓ má pre uskutočnenie k dispozícii najviac údajov alebo najlepšie údaje (spoľahlivosť, presnosť, kategorickosť, špecifickosť, senzitivnosť, diskriminačná účinnosť a pod.),
- výber operácie, ktorá
 - ✓ je buď najčastejšie alebo najzriedkavejšie používaná,
 - ✓ spôsobí aktiváciu buď najväčšieho alebo najmenšieho počtu nadväzujúcich akcií,
 - ✓ spôsobí aktiváciu najlacnejších (napr. v zmysle výpočtovej zložitosti, nárokov na doplnenie chýbajúcich údajov a pod.) nadväzujúcich akcií,
 - ✓ má potenciál získať najviac nových alebo najdiferencujúcejších informácií,
 - ✓ sa v analogickej situácii najčastejšie osvedčovala,
 - ✓ vzhľadom na dané kritéria, pokiaľ ich splňuje, zabezpečuje najrýchlejšie dosiahnutie cieľového stavu,

a ďalšie.

Nezávisle od uplatnenej výberovej stratégie, poslaním výberovej funkcie je určiť buď jednoznačne jedinou nadväzujúcu operáciu, alebo voľbu jedinej **makrooperácie** tvorenej vhodnou deterministickou kombináciou zodpovedajúcich elementárnych operácií. V prvom prípade ide o pomerne dobre zvládnuté procesy elementárnej úrovne, t.j. takej, ktorá má *mechanickú povahu*. V druhom prípade ide zväčša o zatiaľ málo spoznané riešiacie procesy vyššej úrovne, t.j. také čo aproximujú obdoby kognitívnych procesov¹⁶. Pokiaľ nie sú k dispozícii

Nedeter-
minizmus

jednoznačné krytéria je proces výberu nedeterministický, iba nezáväzne sa skúša aplikovateľnosť a úspešnosť vybraných operácií!

Nedeterminizmus, ako charakteristický rys problémov riešených ES, spravidla implikuje mnoho riešiacich krokov, v ktorých nastáva aspoň jedna z nasledujúcich situácií

Heuristické pravidlá

- **z formulácie problému ani z predchádzajúceho priebehu jeho riešenia nie je zrejmé, ktorý krok sa má v ďalšom vykonať,**
- **výber ďalšieho postupu riešenia je podmienený výsledkom testovania podmienky, o ktorej však nie je možné s určitosťou rozhodnúť, či je alebo nie je splnená.**

Pri alternatívnom prekleňovaní nedeterminizmov sa uplatňujú **heuristické pravidlá**. Sú to nezdôvodnené - skúsenosťou nadobúdané - návody na rozhodovanie v nedeterministických situáciách riešiaceho postupu. Hoci sú často úspešné, predsa len nie sú spoľahlivé, môžu viesť aj k nezdaru. Ťažkosť spočíva v tom, že

heuristika neumožňuje vopred rozpoznať, či jej použitie je vhodné; nie je vyjadriteľná v tvare kritéria použiteľnosti,

preto nie je návodom k **deterministickému** riešiacemu postupu.

1.5 Paradigma deklaratívneho programovania a produktívne riešiace postupy

Na základe doterajších úvah sa o aplikovateľnosti procedurálneho, resp. deklaratívneho programovania, dá vysloviť nasledovné

MUSÍ
MÔŽE

a Keď zákonitosti platné pre riešenie problému umožňujú hovoriť o tom, ČO A KEDY MUSÍ nastať, vtedy je výhodný štýl PROCEDURÁLNEHO PROGRAMOVANIA – vtedy dokážeme k daným problémom vytvoriť jednoznačný a efektívny riešiaci proces.

Keď však zákonitosti platné pre riešenie problému poznáme iba natoľko, že dokážeme len predvídať ČO A KEDY MÔŽE nastať, vtedy je potrebné vytvárať také riešiace prostriedky, ktoré umožňujú potenciálne odvodenie postupu riešenia. A to je to prípad, keď je potrebné orientovať pozornosť na deklaratívne programovanie.

Paradigma deklaratívneho programovania

Paradigma deklaratívneho programovania kladie dôraz na **zákonitosti (pravidelnosti), ktorým podliehajú problémy danej aplikačnej oblasti**. Vyjadruje ich prostriedkami formalizmu, ktoré sú **propozičnými prejavmi znalostí**. Tie považuje za nezávislé od samotných riešiacich procesov. Deklaratívny program

- je tvorený štruktúrovanými zložkami explicitne symbolovo reprezentujúcimi deklarácie zákonitostí platných pre určitú triedu problémov;
- postup jeho zbiehania (riešenia problému) vyplýva z dôsledkov špecifikovaných (logických) formúl reprezentujúcich zodpovedajúce zákonitosti;
- postup je generovaný na základe riadenej - formálnej logike neodporujúcej inferencie, t.j. interpretácie príslušných formúl - vhodným (dodaným, vytvoreným) interpretátorom.

Tvorba deklaratívneho programu podlieha

- syntaktickým pravidlám zvoleného reprezentačného formalizmu a
- korešpondujúcim princípom interpretácie formúl a riadeného priebehu interpretácie.

Odčlenenie riešiacich procesov - a teda súvisiacich riadiacich konštruktov - od vlastného deklaratívneho programu výrazne uľahčuje

- tvorbu jednotlivých prvkov programu,
- porozumenie ich úlohám a
- účinkom ich použitia.

Deklaratívny program a jeho interpretátor vo "vlastnej réžii" zabezpečujú odvodenie riešiaceho postupu a generujú ho.

Súhrn.

Súhrn	Produktívne riešiace postupy
- prehľadávanie	<ul style="list-style-type: none"> ✓ vyžadujú prehľadávanie, odhadovanie, voľbu, skúšanie, či inak nedeterministické výber niektorých z alternatívnych možností riešenia problému, ✓ sú pri výbere flexibilné a na výber sa nefixujú: ak sa v nadväzujúcom riešiacom postupe <ul style="list-style-type: none"> ▪ narazí na protirečenie či konflikt, ▪ zistí, že nevedie k požadovanému výsledku, ▪ odhalí možnosť iného, výhodnejšieho postupu,
- nezáväznosť	<ul style="list-style-type: none"> tak sa od časti alebo celého prebiehajúceho postupu odstúpi, revidujú sa vybrané (odhadnuté, zvolené, skusmé, či inak podmienené, alebo kontextuálne závislé) operácie/údaje a pokračuje sa iným, vzhľadom na dosiahnutý stav vhodnejším variantným postupom,
- vratnosť	<ul style="list-style-type: none"> ✓ sú <i>vratné</i> – čo ich principiálne odlišuje od nevratných <i>reproduktívnych</i> postupov uskutočňovaných procedurálnymi programami, ✓ <i>nemajú metriku</i>, čo znemožňuje hodnotenie vzdialenosti od požadovaného výsledku alebo zisťovanie odchýlky od správneho "smeru" postupu k riešeniu - <i>súvisí to s absentujúcou možnosťou analytického opisu riešiaceho postupu, a teda absenciou algoritmu, ktorý by jednoznačne viedol k správnejmu výsledku.</i>

1.6 Princípy prehľadávacích procesov

Vnímanie riešiacich procesov	<p>Prehľadávacie procesy sa dajú vnímať z rôznych pohľadov. Zdôrazňované aspekty najčastejšie hovoria o (riešiacich) postupoch v terminológii:</p> <p>(a) prehľadávania stavového priestoru problému,</p> <p>(b) procesov splňovania ohraničujúcich podmienok,</p> <p>(c) logickej inferencie.</p> <p>Ide o vzájomne sa nevylučujúce poňatia, ktoré sa odlišujú prístupmi k reprezentovaniu a riešeniu problémov. Každé z nich má svoje oprávnenie aj poslanie.</p>
Stavový priestor	<p>Predstava stavového priestoru problému vedie k úvahám o grafe, ktorého vrcholy zodpovedajú stavom riešenia problému a hrany zodpovedajú operáciám meniacim stav na iný: <i>produktívne riešenie problému má potom povahu prehľadávania možných ciest medzi vrcholom zodpovedajúcim počiatočnému stavu riešenia problému a niektorým z vrcholov zodpovedajúcich podmienkám cieľového stavu.</i></p>
Splňovanie ohraničení	<p>Predstava procesov splňovania ohraničení (ohraničujúcich podmienok) vníma riešenie problému ako postupné splňovanie explicitne zadaných ohraničujúcich podmienok: <i>V riešiacom postupe sa postulovaná počiatočná množina objektov, o ktorých sa predpokladá, že môžu vyhovovať daným ohraničeniam, postupne ohraničuje. Ak ich vlastnosti nie sú v súlade s danými a</i></p>

odvoditeľnými ohraničeniami,¹⁷ tak sa z množiny odstráni, čím v nej zostávajú objekt(y) potenciálne najlepšie vyhovujúci(e) všetkým ohraničujúcim podmienkám. Taká množina zodpovedá cieľovému stavu.

Predstava **logickej inferencie** vníma riešenie problému ako logické rezolovanie: *Predpokladá (postuluje) určitý logický systém, ktorý umožňuje z daných axióm a už odvodených tvrdení odvodzovať ďalšie. Postup riešenia spočíva v prehľadávaní a vyberaní použiteľných axióm a viet (tvrdení), následne z vybraných logickej inferenciou produkuje nové tvrdenia. Postup pokračuje kým sa nenájde odpoveď na otázku zodpovedajúcu riešenému problému – cieľový stav.*

Sú to to v širokom rozsahu univerzálne symbolové procesy. Ich špecilizované uplatňovanie je určované symbolovými štruktúrami reprezentujúcimi špecifické poznatky.

1.7 Formálna logika a jej postavenie pri tvorbe ES

Interpretácia deklaratívneho programu súvisí s realizáciou univerzálnych procesov uplatňovania reprezentovaných poznatkov. Zodpovedajúci proces musí spočívať (byť "ukotvený") na teoreticky podložených princípoch. V tom majú formálne logiky nezastupiteľné postavenie. Súvisiacej problematike je venovaný tento článok.

Z pohľadu formálnej logiky sústava znalostí **Z** (axiómami a teorémami vyjadrená teória) reprezentovaná príslušnými symbolovými prostriedkami má v konkrétnych prípadoch umožniť nájdenie všetkých pravdivých viet (klauzúl), ktoré z nej vyplývajú. To znamená, že keď sa v rámci teórie **Z** hľadá odpoveď na konkrétnu otázku **Q**, je nevyhnutné, aby sa dalo zistiť, či **Z** umožňuje túto odpoveď určiť (odvodiť). V symbolovom vyjadrení píšeme

$$\mathbf{Z} \Rightarrow \mathbf{Q},$$

kde \Rightarrow je symbolom pre 'určuje' (odpoveď na otázku **Q**).

Na rozdiel od bežných databáz obsahujúcich iba konkrétne inštalácie údajov, odvodzovanie odpovede na otázku **Q** môže vyžadovať prehľadávanie sústavy explicitných poznatkov v **Z**, z ktorých odpoveď iba **implicitne** vyplýva. Uskutočňuje sa to na základe odvodzovacích (inferenčných) pravidiel tvoriacich komplement k reprezentovaným poznatkom **Z**. Sú to napríklad pravidlá typu

$$\forall x(\text{vlastnosť}_a(x) \rightarrow \text{vlastnosť}_c(x)) \quad (1.1)$$

ktoré na základe platnosti predikátu **vlastnosť_a(K)** - t.j. z tvrdenia, že **'K má vlastnosť a'**,

na základe **Z** - obsahujúcej pravidlo (1.1) hovoriace, že *pre každé individuum majúce vlastnosť a platí, že má aj vlastnosť c*, odvodiť pre individuum konštantu **K** platnosť predikátu **'vlastnosť_c(K)'**, čo tvorí odpoveď na príslušnú otázku **Q**.

Symbolovo píšeme

$$\text{vlastnosť}_a(\mathbf{K}) \rightarrow \text{vlastnosť}_c(\mathbf{K})$$

¹⁷ Odvoditeľné ohraničenia explicitne alebo implicitne vyplývajú z reprezentovaných zákonitostí problémovej oblasti.

(1.2)

kde \rightarrow je symbolom pre *odvoditeľné*.

Logika
jej pravidlá sú
permisívne

Kým vzťah (1.1) je **deklarovaním určitého poznatku** z danej problémovej oblasti, vzťah (1.2) zodpovedá **procedurálnej stránke použitia daného poznatku**.

Logika vytvára základňu - **syntaktických** - mechanizmov algoritmickeho exhaustívneho (zo sémantického hľadiska nedeterministického) **prehľadávania všetkých dôsledkov**, ktoré vyplývajú z množiny splnených predpokladov. Jej **pravidlá sú PERMISÍVNE**, určujú dôsledky priamo odvoditeľné z množiny predpokladov, umožňujú usudzovať o **možnostiach**, ktoré však nemusia nastať alebo nie sú plauzibilné (vierohodné) ani relevantné, pričom však **neurčujú tie dôsledky, ktoré sa musia odvodiť, ani poradie ich odvodzovania**.

Výpočtová
zložitosť

Táto skutočnosť má veľmi vážne implikácie. Dá sa to ilustrovať na dokazovaní teorém: Jazyk prvorádovej predikátovej logiky (skrátke JPL) umožňuje tvorbu procedúr **dokazovania teorém** – uplatňujúc procesy substituovania, unifikácie a rezolvovania - na základe daných axióm a daných či odvodených teorém. Nedochoádza však pritom k **redukcií exhaustívnych prehľadávacích procesov** na taký zvládnuteľný rozsah, ktorý by adekvátne a účinne imitoval efektívne formulovanie problému (otázku, ktorá by sa mala zodpovedať) ani riešiace postupy skúseného odborníka. Dokazovanie teorém má **exponenciálnu výpočtovú zložitosť**. Hoci sa dôsledky prejavujú iba v pesimálnych prípadoch, skutočnú výpočtovú zložitosť riešenia konkrétneho problému, ani jej priemerný odhad, nevieme predvídať. Musí sa teda rátať aj s najhoršou alternatívou, čo z praktických hľadísk je neprijateľné.

Problémy s
monotónnosťou

Všimnime si ďalšie ťažkosti.

V priebehu riešenia konkrétnych problémov môže sa meniť pravdivosť tvrdení o realite (napr. kvôli dynamickým zmenám v nej). Preto je nevyhnutné sústavne sledovať bezospornosť všetkých daných a odvodených tvrdení a podľa potreby ich aktualizáciou zabezpečovať konzistentnosť riešiaceho postupu. Avšak **mechanizmy logickej inferencie nad klasickými JPL sú kumulatívne (monotónne)**: pri odvodzovaní nových tvrdení všetky v predošlom odvodené zostávajú v platnosti, nie sú aktualizované. Reálne systémy usudzovania nie sú také, môžu porušovať pravdivosť už odvodených výrokov - **sú nemonotónne**.

Lokálne
versus
globálne

Pri riešení reálnych problémov, v počiatočom stave riešenia spravidla nie sú k dispozícii všetky potenciálne použiteľné údaje. Tie sa v priebehu riešenia postupne, podľa potreby, získavajú, zhromažďujú. Príliv nových údajov môže viesť k situácii, ktorá vyžaduje (dramatickú) **preformuláciu pôvodného problému** a tým riešenie úplne nového problému. Zodpovedajúce procesy preformulovania nie sú založené na **formalizova(tel')ných postupoch**, majú predovšetkým empirickú povahu. Aj to svedčí proti názorom zužujúcim a stotožňujúcim uvažovanie výhradne na mechanizmy klasickej logickej dedukcie.

Fuzzy
kvantifikátory

Prostriedky inferencie v JPL vyberajú niektorý z možných alternatívnych riešiacich postupov len vzhľadom na riešenie práve aktuálnej čiastkovej úlohy nezávisle od riešenia ostatných podproblémov, či celkového cieľa. Realizujú teda len tzv. **lokálne riešiace procesy**. Absentuje v nich možnosť uskutočňovať **globálne riešiace postupy (makrokroky)**, t.j. spôsobilosť z určitého nadhľadu

syntetizovať čiastkové úlohy. Napríklad využívať určitý plánovací mechanizmus analyzujúci stavový priestor problému na výber akcie, ktorý umožňuje súčasne rozhodovať medzi väčším počtom alternatívnych postupov (výbere hypotéz).

Riešenie praktických problémov nezriedka vedie k potrebe reprezentovať, sémanticky interpretovať a procedurálne využívať aj iné ako bežné logické kvantifikátory, napríklad NAJMENEJ, PRÁVE, NAJVIAC, MÁLO, MNOHO, SPRAVIDLA, ČASTO, ZRIEDKA a mnoho iných. Podobne je často žiaduce reprezentovať nemálo **výnimiek** (napr. "Každý (?) vták lieta.", "Každý (?) kov je kujný."). Prostriedky JPL neposkytujú k tomu priamočiare možnosti ani na deklaratívnej ani na procedurálnej úrovni. (Kým prvé tri z uvedených kvantifikátorov sú jednoznačné a dajú sa vyjadriť prostredníctvom univerzálneho a existenčného kvantifikátora, ostatné sú problematickejšie, hoci len kvôli ich kontextovej podmienenosti).

Psychológia tvrdí (nie sú známe protiargumenty), že: inferenčné pravidlá (dokazovania teorémov) nie sú totožné s prirodzeným (spontánnym) usudzovaním človeka. Ten si vytvára, osvojuje, preberá systémy pravidiel a mechanizmov ich používania, ktoré sa odlišujú od formálnych pravidiel logiky, napr. jednoduchý princíp substituovania pravdivostných hodnôt do premís (napr. *hradlovanie*, *komplementovanie*, *asociovanie*, ako o tom pojednáme v ďalšom) a sledovanie dôsledkov, ktoré z toho vyplývajú.

Potreba
mimologickýc
h konštruktov

JPL umožňuje pri reprezentácii znalostí vynechávať množstvo špecifikácií skutočností, ktoré spôsobujú neúplnosť reprezentácie. Tým sa prednosť JPL, ktorá spočíva v jeho všeobecnosti, stáva zároveň jeho nedostatkom. Východisko sa hľadá a nachádza v používaní jednoduchšieho, či slabšieho formalizmu, resp. v jeho obohacovaní mimologickými konštruktami a komplementárnymi procedurálnymi zložkami.

Schodnosť tohto postupu vyplýva z existencie viacerých prostriedkov tvorby znalostných (expertných) systémov. Každý z nich je výsledkom snahy hľadať kompromis medzi požiadavkami logickej koherentnosti a prijateľnej výpočtovej zložitosti, ako aj medzi vlastnosťami symbolickej (syntaktickej) a mentálnej (sémantickej) logiky.

Hľadajú a nachádzajú sa neštandardné inferenčné procesy (a prirodzene zodpovedajúce formalizmy reprezentácie znalostí), ktoré dopĺňujú štandardné vhodnými riešiacimi schémami pre **situačné uprednostňované a selektívne odvodzovanie iba cielenej podmnožiny logických dôsledkov**. Prostriedkom toho sú napríklad

- ✓ **neklasické logiky** (modálne, viachodnotové, t.j. stupňovanie "pravdivosti", resp. "nepravdivosti" až po spojité (prípadne nelineárne) kontinuum medzi pravdou a nepravdou - fuzzy logika),
- ✓ nezáväznú používanie **očakávateľných (predpokladateľných) údajov** nahradzujúcich chýbajúce,
- ✓ udržiavanie **konzistentnosti** odvodených faktov - **nemonotónna logická inferencia**,
- ✓ **oceňovanie "nákladov"** za vyhodnocovanie podmienok (získavanie údajov) a vykonanie operácií (ich informačný prínos),
- ✓ **sémantické odlíšenie** poslania operácií (pravidiel), či častí stavového priestoru riešenia problému (napr. rozlišovanie hlavných a vedľajších inferenčných

schém, rozlišovanie fenomenologických a kauzálných závislostí, rešpektovanie ohraničení a pod.),

- ✓ špecifikovanie, generovanie a používanie **podporných riadiacich údajových štruktúr a čiastkových (generických makro-) procesov** umožňujúcich obohacovať prostriedky odvodzovania, ktoré vyplývajú z vlastností JPL a iné.

Vo všeobecnosti jestvujú dve alternatívy používania mimologických konštruktov a možnosť ich kombinovania:

- (1) **rozšírenie systému logických klauzúl o riadiace údajové štruktúry, ktoré umožňujú explicitne vyjadrovať situačne podmienené riadiace akcie a**
- (2) **rozšírenie funkčných spôsobilostí interpretátora logických formúl o situačne riadené spúšťanie procedurálnych riešiacich mechanizmov.**

1.7 Poznatky a znalosti

Na záver tejto uvádzajúcej kapitoly venujme pozornosť ešte významu, v ktorom používame kľúčové pojmy súvisiace s fenoménom (nazvaným) expertný systém: **poznatok** a **znalosť**. Sú to pojmy, ktorých význam sa nedá jednoducho definovať, najmä vtedy nie, keby sa mali zohľadniť všetky súvisiace aspekty. Pre potreby výkladu v nadväzujúcich kapitolach (*a kvôli vytvoreniu neprázdneho prieniku prvkov 'mentálneho modelu' čitateľa a autora*) uvádzame vymedzenia, ktoré sú nepochybne primeranou aproximáciou týchto fenoménov, pričom ich považujeme za otvorené, teda nenárokujúce si definitívnosť.

Poznatky

Poznatky sú reprodukciou určitej vymedzenej časti objektívneho sveta a v ňom platných zákonitostí.

Je pre ne charakteristické, že

sú transformáciou rozptýlených (aj nejasných) predstáv a tušení do takej komunikateľnej formy, v ktorej sa z nich zachováva to, čo možno oznámiť iným ako ustálený základ racionálneho konania.

Vznikajú ako produkt myšlienkovvej činnosti najmä v súvislosti s pracovnými a spoločenskými aktivitami.

Znalosti

Znalosti sú meniteľné a doplniteľné štruktúry vzájomne previazaných poznatkov, ktoré sú podmienené ich uplatniteľnosťou v interakcii so svetom:

- ✓ **znalosť čohosi znamená vlastniť tomu zodpovedajúcu reprezentáciu v podobe dostatočne verného a presného mentálneho modelu, vrátane spôsobilosti vykonávať s tým, čo je reprezentované, rôzne kognitívne (mentálne) operácie,**
- ✓ **na základe a v rozsahu týchto operácií dokáže človek (potenciálne aj počítač) predvídať a predpovedať to, čo sa musí alebo môže v reálnom svete odohrať**

súvisia s konceptualizáciou entít (objektov, javov, ich vlastností) a v nich, či s nimi prebiehajúcich procesov:

- ✓ **konceptualizácia spočíva - okrem iného - v schopnosti pomenovať, opisovať a definovať entity a ich vzájomné vzťahy, ako aj**

Deskripcie (opisy)	<p>kategorizovať ich, t.j. zaradiť ich do rôznych tried, medzi ktorými sú špecifikovateľné vzájomné vzťahy, ✓ súčasťou konceptualizácie je aj formovanie výrokov o entitách, skladanie a vyhodnocovanie (interpretovanie) výrokov a v tom spočívajúce odvodzovanie dôsledkov, ktoré z nich vyplývajú.</p>
Relácie (vzťahy)	<p>Termín znalosť zahrňuje a kombinuje</p> <ul style="list-style-type: none"> ■ deskripcie (<i>opisy, definície</i>) identifikujúce a rozlišujúce entity aj ich triedy - sú vyjadrované vetami určitého jazyka, ktorých prvky sú tvorené <i>primitívnymi konceptami</i>, ■ relácie, ktoré vyjadrujú vzťahy (závislosti, podmienenosti) entít - zodpovedajú definičným, asociatívnym (empirickým), príčinným (kauzálnym), taxonomickým, štrukturálnym, kontextovým, priestorovým, časovým, funkčným, úlohovým, ohraničujúcim, modelovým a riadiacim/regulačným väzbám (súvislostiam), ■ procedúry (<i>preskripcie</i>), ktoré postupnosťami operácií v priebehu riešenia problémov (situačne) vymedzujú používanie a interpretovanie deskripcií a relácií.
Procedúry (predpisy)	<p>Uvedené členenie - kategorizácia – má svoje opodstatnenie pri tvorbe báz znalostí znalostných systémov. V tejto súvislosti uvádzame ešte ďalšie pojmy spolu s významom, v ktorom sú v týchto textoch používané:</p> <ul style="list-style-type: none"> • Definícia – presné, jednoznačné a úplné vymedzenie/určenie objektov, javov, procesov. • Asociácia - obvyklé, pozorovateľné, či očakávateľné spájanie aspoň dvoch entít spravidla na základe skúseností, prípadne analógie, nie nevyhnutne na základe zdôvodniteľných vzťahov. • Kauzalita - známe zdôvodniteľné súvislosti medzi dvomi alebo viacerými entitami, ktoré sú vo vzťahu príčiny a následku. • Taxonómia – usporiadané orientované vzťahy typu všeobecný-špeciálny (generalizačno-špecializačný vzťah), celok-časť, nadriadený-podriadený (širšie-uzšie právomoci) – prostriedok umožňujúci účinnú štruktúrovanú reprezentáciu entít. • Štruktúra - vzájomný vzťah prvkov celku, t.j. ich skladba, priestorové vzťahy, väzby (prepojenie), funkčné podmieňovanie, prípadne iné. • Kontext - okolnosti, súvislosti, za ktorých sa uplatňujú určité špecifické vzťahy medzi entitami podmieňujúce ich líšiac sa postavenie či úlohy, teda aj rôzne prípustné významy (pori tiež <i>úlohy</i>). • Priestorové vzťahy - v tomto prípade sa spravidla jedná o pozičné, rozmerové a smerové vzťahy medzi entitami (susedstvo, vnútro-vonkajšok, veľkosť, vzájomné vzdialenosti/poloha/orientácia v priestore, tvar, smer pohybu atď.). • Časové (temporálne) vzťahy - spravidla sa uvažujú súvislosti medzi entitami, ktoré vzhľadom na ich výskyt alebo zmeny v určitých časových okamihoch/intervaloch podmieňujú okamžité, súčasné a časovo následné výskyty, či zmeny iných entít - umožňujú uvažovať o javoch minulých, súčasných a budúcich. • Funkčné vzťahy - správanie, poslanie, či použitie entít. • Úlohy - známe či predpokladané polymorfizmy (polyfunkčnosti) entít (tá ista entita môže mať rôzne (situačne podmienené) úlohy a poslanie s odlišnými kvalitami, vlastnosťami - (pozri tiež <i>kontext</i>)).

- **Ohraničenia** - predpisy a zákony (spoločenské a prírodné, vrátane vzťahov k fyzikálnym, chemickým, biologickým, ekonomickým konštantám) vzťahujúce sa k entitám a ich vzájomným funkčným, štrukturálnym, ekonomickým, sociálnym a iným väzbám (podmienenosťami).
- **Modely** - známe či predpokladané dynamické súvislosti určitým spôsobom vzájomne previazaných a ovplyvňujúcich sa entít, ktoré reprezentujú štruktúru a správanie sa modelovanej predlohy.
- **Regulačné/riadiace vzťahy** - špecifické závislosti, na základe ktorých je možné ovplyvňovať správanie sa entít.

Termy

Uvedené typy poznatkov a znalostí v bázach znalostí - v závislosti na konceptualizácii reprezentácie uvažovaného výseku sveta – môžu vystupovať ako prvok deskripcí, teda v roli **termu**, alebo ako prvok tvrdenia - pravidla, teda v roli **asercie**. V súlade s uvažovanou rolou príslušného kvanta znalostí je potrebné voliť odlišný formalizmus symbolovej reprezentácie. Sú témou nasledujúcich kapitol.

Kapitolu uzatvárame vymedzením dvoch vyššie uvedených pojmov rozlišujúcich poznatky a znalosti z hľadiska ich postavenia v znalostných systémoch: **term** a **asercia**.

Asercie

TERMOM zodpovedajú opisy/definície entít, ich vlastností a im zodpovedajúce (skutočné alebo predpokladané) hodnoty, ich (kontextuálne podmienené) úlohy, zaradenie a vzťahy, ich štruktúry a prípustná organizácia (vzťahy časť-celok), hierarchická klasifikácia (taxonomizácia), (podmienené) príčinnno-dôsledkové závislosti a úlohy (role), čo všetko možno s výhodou situačne použiť pri efektívnom využívaní asertívnych znalostí a umožňuje odvodzovať konkrétne implicitné poznatky a fakty.

ASERCIE sú tvrdenia o entitách, o ich konkrétnych stavoch, vzťahoch a vlastnostiach. Zodpovedajú propozíciám (možno im priradiť pravdivostnú hodnotu), ktoré umožňujú explicitne odvodzovať konkrétne poznatky, t.j. skutočnosti súvisiace s riešeným problémom. V deklaratívnej podobe majú najčastejšie tvar izolovaných produkčných ("AK-TAK", "KEĎ-POTOM") pravidiel reprezentujúcich vzťahy a súvislosti medzi entitami. Môžu však mať aj tvar analytických alebo iných matematických závislostí reprezentujúcich ohraničenia, funkcie a (v znalostných systémoch najmä *kvalitatívne*) modely. Dajú sa tiež považovať buď za jedinečné preskripcie, alebo také čo parametrizujú všeobecné procedúry odvodzovania (najčastejší prípad).

2. SYMBOLOVÁ REPREZENTÁCIA POZNATKOV A ZNALOSTÍ

2.1 Prostriedky symbolovej reprezentácie

Hypotézy uvedené v predošlej kapitole, najmä koncept *mentálneho modelu*¹⁸, sú východiskom realistického predpokladu o uvažovaní človeka. **Sú to v mysli uskutočňované procesy (operácie, manipulácie) s entitami mentálneho modelu, teda s mentálnou reprezentáciou týchto entít.** Existencia príslušných entít v mysli človeka je nevyhnutnou podmienkou uskutočňovania takýchto procesov a teda myslenia.

¹⁸ Koncept mentálneho modelu nespochybnuje ani Searle, hoci je jedným z výrazných oponentov fenoménu *strojového myslenia*.

Podstata procesov *odvodzovania/usudzovania* uskutočňovaných počítačom je tiež proces operácií (manipulácií) s entitami v jeho aritmeticko-logickej jednotke. Nevyhnutnou podmienkou ich uskutočňovania je existencia symbolovej reprezentácie príslušných entít v pamäti počítača.

Z tejto evidentnej analógie vyplýva pozornosť, ktorá sa v súvislosti s **UI** a tvorbou **ES**, venuje *symbolovej reprezentácii poznatkov a znalostí*. Menovite

Symboly

☞ aspektom, ktoré sú **spoločné pre myslenie a spracúvanie symbolov,**
☞ **spoznávaníu** takých **procesov vytvárania a spracúvania symbolových štruktúr, ktoré v prípade, keď ich vykonáva človek, nazývame myslením.**

Symbol je jednoznačne identifikovateľný a manipulovateľný fenomén, ktorý môže označovať (zastupovať, reprezentovať) iné entity. Základným elementárnym (nerozložiteľným) symbolom hovoríme **symbolové primitíva**. Umožňujú **vytváranie zložitejších symbolových štruktúr**. A práve v súvislosti s takou tvorbou sa dá hovoriť nielen o procesoch **vytvárania, rušenia, spájania, rozčleňovania, rozdeľovania**, ale aj **vzájomného prepájania, viazania** či iného **menenia** takýchto štruktúr.

Účinnosť symbolovej reprezentácie

Z hľadiska **symbolovej reprezentácie znalostí** je veľmi významné rozoznávať a vo vzájomných súvislostiach uvažovať o vlastnostiach reprezentačných prostriedkov, o ich

- **vyjadrovacej účinnosti (expresivite)**, t.j. **ČO A V AKOM ROZSAHU** umožňujú **REPREZENTOVAŤ,**
- **odvodzovacej účinnosti (použitelnosti)**, t.j. **ČO A AKO** umožňujú **ODVODZOVAŤ,**
- **výpočtovej účinnosti (efektivite)** t.j. o **VÝPOČTOVEJ A PAMÄŤOVEJ ZLOŽITOSTI ODVODZOVANIA.**

ČO, ČÍM A PROSTREDNÍCTVOM ČOHO reprezentovať

Predmet reprezentácie je to, **ČO** sa v určitom **rozsahu** má a môže reprezentovať. *Ten istý predmet reprezentácie sa dá zastúpiť (reprezentovať) rôznymi symbolovými prostriedkami.* Preto je potrebné rozhodnúť **ČÍM** reprezentovať, t.j. voliť **reprezentujúci** (zastupujúci) **symbol reprezentovaného**. Naviac, dôležitý je aj **prostriedok reprezentácie**, t.j. prostredníctvom **čoho** sa reprezentuje (symbolom priradené operácie, procesy) tak, aby bolo možné zachytiť všetky - z istých hľadísk podstatné - črty predmetu reprezentácie. Symboly a nimi vykonávateľné operácie determinujú **expresivitu** symbolovej reprezentácie. Zvýraznime, že ide o rozhodovanie

Expresivita

- **čím** reprezentovať, t.j. voliť **symbol reprezentovaného,**
- **prostredníctvom čoho** reprezentovať, t.j. voliť **prostriedok reprezentácie** tak, aby bolo možné zachytiť čo najviac - z istých hľadísk podstatné - črt predmetu reprezentácie.

Príslušné rozhodnutia ovplyvňujú **expresivitu** symbolovej reprezentácie.

ČO a AKO

Úlohu expresivity osvetlíme na dvoch príkladoch. Najprv uvažujme problém identifikovania stavu (diagnózy) určitého technického zariadenia, povedzme televízora. Na riešenie problému je potrebné reprezentovať jeho *štruktúru, funkcie a správanie (prejav)*. Tým je špecifikované, **čo** sa má reprezentovať aj **rozsah** reprezentovaného (nespomíname napr. fyzikálne zákony podmieňujúce funkcie zariadenia). V tabuľke na nasledujúcej strane sú uvedené ilustračné možnosti toho, **čím** a prostredníctvom **čoho** reprezentovať. Ďalšia tabuľka uvádza jednoduchší a priezračnejší príklad – týka sa reprezentovania kvantít - ilustrujúci a ozrejmujujúci význam pojmu **expresivita**.

Odvodzovací a účinnosť

Voľba **symbolov (čím)** a **prostriedkov reprezentácie (prostredníctvom čoho)** podmieňuje aj **použitelnosť reprezentanta**, teda **procesy**, ktoré umožňujú

produkovat' riešiacie postupy, t.j. to **ČO a AKO** sa dá odvodzovať. V tejto súvoslosti hovoríme o **odvodzovacej účinnosti**.

Význam pojmu *prostriedok reprezentácie* ilustrujeme na prípade reprezentácie *celočíselných množstiev* (druhá z uvedených tabuliek). Tabuľka zreteľne poukazuje na výrazné rozdiely odvodzovacej účinnosti v závislosti na voľbe prostriedku reprezentácie.

Slovná reprezentácia je nesporne prostriedkom zaznamenávania a prenosu informácií o množstvách. Keby sa však mala vykonať operácia súčtu medzi výrazmi *štyristosedemdesiatdeväť* a *deväťstosedemdesiatštyri*, vznikli by určité ťažkosti. To isté možno tvrdiť o prípade, v ktorom by sa spomínané množstvá mali reprezentovať prostredníctvom Morzeovej abecedy. Ani jeden z týchto prostriedkov reprezentácie neposkytuje totiž účinné prostriedky na vykonávanie príslušných výpočtových operácií. O málo výhodnejšie by bolo používanie vhodných zoskupení sybolov III..., aspoň z hľadiska operácie súčtu. Binárna reprezentácia množstiev, najmä s ohľadom na rôzne možnosti pravidiel kódovania a nimi podmienených operácií, poskytuje (predovšetkým pre elektronické zariadenia) rozsiahle možnosti produkovať zodpovedajúce výpočtové procesy. Keď sa napr. kvantily 479 a 974 vyjadria v tvare

IIIOIIII a IIIIOIIIO,

operácia násobenia, ako je to dobre známe, sa vykoná zodpovedajúcimi posuvmi a sčítovaním. V prípade BCD kódu nadobúdajú uvedené čísla podobu

OIOO OIII IOOI, resp. IOOI OIII OIOO

a čitateľ ľahko zistí pravidlá, na základe ktorých vznikne proces násobenia. O prednostiach dekadickéj číselnej sústavy (najmä vzhľadom na to, že väčšina ľudstva je dobre "naprogramovaná" na zodpovedajúce operácie, hoci konštruktér zodpovedajúcich elektronických obvodov zrejme sleduje iné hľadiská) sa vari netreba zmieňovať.

ČO A V AKOM ROZSAHU SA MÁ REPREZENTOVAŤ	ŠTRUKTÚRA, FUNKCIE A SPRÁVANIE (POZOROVATELNÉ PREJAVY) SYSTÉMU (súvisiace fyzikálne zákony správania sa neuvažujú)
ČÍM REPREZENTOVAŤ (SYMBOLY REPREZENTOVANÉHO) fotografie vnútornej konštrukcie a záber obrazovky <ul style="list-style-type: none"> ■ bloková schéma elektrického zapojenia (konštrukcie) s uvedením náležitých hodnôt fyzikálnych veličín, ■ podrobná schéma elektrického zapojenia s požadovanými hodnotami jednotlivých parametrov 	PROSTREDNÍCTVOM ČOHO (PROSTRIEDKY REPREZENTÁCIE) bitmapa zodpovedajúca fotografiám ikonické symboly zaužívaných geometrických útvarov: trojuholníky, štvoruholníky, úsečky, kružnice, špirálovité krivky..., (prípadne aj vo farbe) s priradenými znakmi/symbolmi (čísla, označenia fyzikálnych jednotiek); v pamäti reprezentované napr. kódmi ikonických útvarov alebo bitmapami
slovný opis prvkov systému, ich vzájomných väzieb a funkcií korešpondujúcich so schémou systému	výrazové prostriedky prirodzeného jazyka podliehajúce zodpovedajúcim gramatickým pravidlám
formalizovaný (symbolový) prepis schémy či slovného opisu daného zariadenia	symbolové prostriedky a pravidlá zvoleného formalizmu (výroková/predikátova logika, produkčné pravidlá, rámce)

ČO A V AKOM ROZSAHU SA MÁ REPREZENTOVAŤ	CELOČÍSELNÉ MNOŽSTVÁ
ČÍM REPREZENTOVAŤ (SYMBOLY REPREZENTOVANÉHO)	PROSTREDNÍCTVOM ČOHO PROSTRIEDKY REPREZENTÁCIE
slovné formulácie	prirodzený jazyk
bodkami a čiarkami	Morzeova abeceda
zoskupeniami znakov III... kombináciami symbolov O a I	päťprvkové zoskupovanie znakov (III III) binárna číselná sústava, alebo BCD kód
arabskými číslicami	dekadická pozičná číselná sústava
arabskými číslicami v kombinácii s písmenami	hexadecimálna pozičná číselná sústava
vymedzenými písmenami	nepozičná sústava rím skych čísiel

Obdobnesa dá hovoriť o hexadecimálnejčíselnej sústave.

Napokon sa prichodí zmieniť o reprezentácii celočíselných množstiev rímskymi číslami. Porovnaním vzájomne sa odlišujúcich reprezentácií nasledujúcich čísiel

$$479 = IIIIOIIII = CDLXXIX$$

$$974 = IIIIOOIIIO = CMLXXIV$$

je zrejme, že vzhľadom na expresivitu sú si ekvivalentné, významne sa však líši ich odvodzovacia účinnosť. Porovnaním aritmetických operácií sa ľahko zistia zvýšené nároky na odvodzovanie aj výpočtovú zložitosť.

Výpočtová účinnosť

Výpočtová účinnosť¹⁹ je synonymom výpočtovej zložitosti, resp. efektivity procedúr vykonávaných (situačne) so vzájomne previazanými REPREZENTAN-TAMI REPREZENTOVANÉHO. Voľba reprezentanta dokáže aj z hľadiska efektivity výrazne ovplyvniť použiteľnosť navrhnutého systému. Preto je žiaduce voliť prostriedky reprezentácie, ktoré sú vyhovujúce aj z pohľadu výpočtovej účinnosti.

Predmetom záujmu pri skúmaní prostriedkov reprezentácie poznatkov a znalostí sú vhodné symboly a symbolové štruktúry. A práve pri ich voľbe je veľmi dôležité ozrejmiť si uvedené tri charakteristiky a ich optimálne vyváženie.

Vyjadriteľnosť a použiteľnosť poznatkov/znalostí sú dôležitými vlastnosťami všetkých prostriedkov reprezentácie. Každý z takých prostriedkov disponuje určitým systémom symbolových štruktúr, ktoré sú nositeľmi spôsobilosti čosi vyjadriť a použiť.

Pri uplatnení hociktorého programovacieho jazyka, použiteľné štruktúry reprezentácie sú determinované prípustnými syntaktickými konštrukciami jazyka ako aj prípustnými spôsobmi organizovania údajov pre procesy, ktoré sú realizovateľné programami vytvorenými v danom jazyku. To je skutočnosť, ktorá aj v súvislosti s rozvojom metód UI, ale aj z mnohých iných (aj teoreticky motivovaných) dôvodov viedli a vedú k vzniku

- programovacích jazykov obsahujúcich a rozširujúcich predchádzajúce prostriedky programovania (napr. LISP, Prolog, FRL, KRL, C++, SmallTalk a iné),
- nových metód programovania (napr. volanie procedúr cieľom alebo operáciami, ktoré sú schopné vykonať, namiesto volania procedúr menom,

¹⁹ Čitateľ, povedzme vynásobením dvoch čísiel (napr. vyššie uvedených) reprezentovaných dekadicky, binárne, rímskymi číslami, sa môže pragmaticky presvedčiť o líšiacom sa počte nevyhnutných elementárnych operácií vynútených použitou reprezentáciou a ňou implikovanými prostriedkami odvodzovania. Aj v prípade realizácie zodpovedajúcich výpočtových procesov počítačom narazíme na takéto rozdielne počty operácií.

situačná aktivácia procedúr nazývaných démon, objektovo-orientované metódy a programovanie, multiagentové systémy a množstvo iných),

- prostredí na vytváranie aplikácií (uviesť treba predovšetkým vývojové prostredia tvorby problémovo zameraných ES, napr. Egeria, ART-IM, Clips, NexpertObject, G2 a pod.).

Uvedené prostriedky – niektoré menej, iné výrazne - prispievajú k zvyšovaniu expresivity reprezentácie a použiteľnosti reprezentovaného.

Pre zvýraznenie a ilustrovanie témy pristavme sa pri produkčných pravidlách a prvorádovom predikátovom počte. Vyjadrovacia účinnosť v oboch prípadoch je daná aserciami typu AK-TAK či KEĎ-POTOM. Predikátový počet navyše umožňuje reprezentovať aj termy a vlastnosti reprezentovaných entít. Použiteľnosť (odvodzovacia účinnosť) v prípade produkčných pravidiel je daná možnosťami odvodzovať na základe modus ponens, modus tollens, techník prehľadávania A-ALEBO grafov. V prípade prvorádového predikátového počtu je daná dokazovaní teorém uplatňovaním substitúcií, unifikácií a rezolovania.

Ešte poznámka k ohraničenej *expresivite* JPL: jestvujú ťažkosti pri priamočiarom a jednoznačnom reprezentovaní vzťahov (v prirodzenom jazyku bežne používaných) ako napr. "vzájomne", "respektíve" (vyjadrenie vzťahu medzi prvkami z dvoch alebo viacerých zoznamov), "spoločne", "musí", "môže" a pod. Nie je tomu inak keď je potrebné vyjadriť vierohodnosť predikátov, či nekategoricky kvalifikovať platnosť propozície výrazmi ako "zväčša", "takmer", "spravidla", "často", "zriedka", "súčasne", "následne", ako aj iných, pre uvažovanie človeka veľmi významných kvalifikátorov.²⁰

2.2 Znalosti: začlenené poznatky

Znalosti:
začlenené,
vzájomne
previazané
poznatky

Ak hovoríme o *znalostiach*, nemáme na mysli navzájom *izolované poznatky*, ale aj ich **vzájomné súvislosti, previazanosť**. Preto pri reprezentácii znalostí je významná **požiadavka začleniteľnosti poznatkov**. Z toho plynie požiadavka na prostriedky reprezentácie: **mali by poskytovať možnosti začleňovať poznatky**.

Sústavu symbolovo reprezentovaných začlenených poznatkov nazývame **báza znalostí (symbolovo BZ)**.

Nasledujúce články tejto kapitoly stručne uvádzajú najčastejšie uplatňované reprezentačné prostriedky, s ktorými sa stretávame pri tvorbe znalostných systémov. S **JPL** sa samostatne nezaobráame,²¹ (napokon dá sa dá predpokladať, že čitateľ jeho vplyv, resp. jeho rigorózne princípy rozpozna v pozadí ostatných reprezentačných prostriedkov aj keď bez aplikovania metód rezolvenencie). Ppoznamenávame však, že v prípade **JPL** začleniteľnosť poznatkov vyplýva z **implicitných** vzájomných súvislostí medzi formulami v dôsledku **jednotnej interpretácie identických predikátových a individuových symbolov**.

Príklad: Všetky formuly, v ktorých sa vyskytuje predikát reprezentovaný povedzme symbolom tvoreným reťazcom **OTEC**, napr. v atóme **OTEC(x,y)**, jeho význam zostáva zachovaný pre ľubovoľné premenné rovnakej kategórie (sorty)

²⁰ Ako už bolo spomenuté, to je dôvod pre obohacovanie pôvodného formalizmu **JPL** mimologickými konštruktami a pre používanie (nie bezproblémovo) prostriedkov neklasických alebo logík vyššieho rádu.

²¹ Pokiaľ čitateľ sa mieni zahľbiť do témy JPL, z bežne dostupnej literatúry odporúčame predovšetkým knihu **J. Šafránek: Inteligencia ako výpočet []**.

ako to ilustruje aj výraz

$$\text{STARÝ_OTEČ}(x,z) \equiv \text{OTEČ}(x,y) \ \& \ \text{OTEČ}(y,z).$$

Pokiaľ sa v procese odvodzovania symboly premenných nahrádzajú menami individuí, každá identická premenná sa substituuje tou istou individuovou konštantou. A tak prostredníctvom rovnakých symbolov sú implicitne zabezpečené vzájomné súvisiace poznatky a tým je zabezpečená ich vzájomná začleniteľnosť.

2.2.1 Reprezentácia znalostí produkčnými pravidlami

Produkčné pravidlá	<p>Produkčné pravidlo je</p> <ul style="list-style-type: none">➤ propozičný výrok typu AK-TAK, KEĎ-POTOM➤ vyjadrením poznatku asertívneho typu➤ vyjadrením určitej dávky (kvanta) znalostí.
Štruktúra pravidiel	<p>Uplatňovanie produkčných pravidiel je motivané tým, že sa jedná o teoreticky dobre zdôvodnený a prebádaný formalizmus, ktorý vznikol v oblasti <i>výpočtových výskumov logiky</i>. V matematickej informatike, v podobe <i>prepísavacieho systému</i>, je používaný ako jeden z modelov výpočtov. Svoje uplatnenie nachádza aj v oblasti <i>kognitívnej psychológie pri výskumoch myslenia človeka</i> (Newell-Simon [], Johnson-Laird []).</p>
Ľavá strana	<p>Štandardná štruktúra produkčného pravidla je zložená z dvoch častí:</p> <ul style="list-style-type: none">◆ z <i>ľavej strany (LS)</i>, zodpovedá predpokladu či situácii, označuje sa tiež ako AK-časť, v logike sa hovorí antecedent, v anglickej odbornej literatúre sa často stretávame s výrazom left [hand] side (L[H]S),
Pravá strana	<ul style="list-style-type: none">◆ z <i>pravej strany (PS)</i>, zodpovedá dôsledku či akcii, tiež akčná-časť zodpovedá alebo TAK-časť, v logike sa hovorí konzekvent, v anglickej odbornej literatúre sa často stretávame s výrazom right [hand] side (R[H]S).
Štruktúra predpokladovej časti	<p>Predpokladová časť (<i>propozičný výraz AK-časti pravidla</i>) je tvorená formulou v syntakticky stanovenej podobe. Elementárnym prvkom formuly je podmienka (atóm). Formula je tvorená jednou alebo viacerými podmienkami vzájomne viazanými logickými operátormi. Formula tvoriaca predpoklad pravidla je splnená, t.j. nadobúda propozičnú hodnotu PRAVDA, vtedy a len vtedy, keď sú splnené podmienky v požadovanej štruktúre. Platnosť, resp. neplatnosť predpokladu implikuje priradenie propozičnej hodnoty dôsledkovej časti pravidla, t.j. oprávnenosť vyvodit' určitý záver (platnosť, resp. neplatnosť referencovaného faktu) alebo vykonať istú akciu.</p>
Prednosti formalizmu produkčných pravidiel	<p><i>Akčná (dôsledková) časť</i> pravidla môže byť tvorená konjunkciou viacerých zložiek. To či sa im priradí propozičná hodnota a aká, resp. či sa vykonajú a ktoré, okrem propozičnej hodnoty predpokladu vo všeobecnosti môže byť ešte podmieňované ďalšími lokálnymi podmienkami. (Podrobne v 4. kapitole venovanej reprezentácii asertívnych znalostí.)</p> <p>Formalizmus produkčných pravidiel je najfrekvencovanejším prostriedkom reprezentácie asertívnych znalostí pretože</p> <ul style="list-style-type: none">◆ umožňuje prirodzeným spôsobom vyjadrovať poznatky asertívneho typu (<i>vyjadriteľnosť</i>),◆ ich interpretácia umožňuje prirodzeným spôsobom imitovať široký rozsah usudzovacích postupností (úvah) človeka (<i>použiteľnosť</i>),

- ◆ umožňuje pomerne priamočiara a teda aj jednoducho vysvetľovať a zdôvodňovať riešiacie postupnosti (vysvetlovacím mechanizmom expertného systému²²),
 - ◆ umožňuje postupnú modulárnu tvorbu a upravovanie báz znalostí,
 - ◆ dá sa použiť ako nositeľ **mimologických reprezentačných konštrukcií**:
 - najmä pri reprezentovaní rôznych druhov neurčitostí,
 - situačne sa uplatňujúcich riadiacich príznakov (semafórov, návěstí, priorit a pod.).
- Symbolovo píšeme:

$$s; p \rightarrow d; a,$$

- Semafór kde
- ◆ $p \rightarrow d$ tvorí **jadro** produkčného pravidla - vyjadruje propozičný poznatok tvrdiaci, že z platnosti predpokladu **p** vyplýva platnosť dôsledku **d**;
- Akčná časť
- ◆ symbol **s** tvorí **semafór, návestie**, alebo **strážcu** pravidla – je to vlastne jedna alebo niekoľko **riadiacich podmienok**, ktoré situačne pripúšťajú alebo zabráňujú interpretáciu (použitelnosť) pravidla;
 - ◆ symbol **a** zodpovedá (**riadiacim**) **akciám**, ktoré sa *situačne* vykonajú po vyhodnotení pravidla. (Podrobnosti najmä v 4. kapitole.)
- Nedostatky reprezentácie
- Nedostatky produkčných pravidiel (*systémov produkčných pravidiel*), najmä vzhľadom na odvodzovaciu účinnosť, spočívajú v tom, že
- ◆ že reprezentujú relatívne **izolované poznatky**, čo bez ďalších symbolizmov, najmä reprezentovania poznatkov termového typu, vedie spravidla len k *lokálne ponímaným riešiacim postupom* nepostačujúcim na uplatňovanie účelných *globalizujúcich nadhľadov*; je to dôsledok toho, že začleniteľnosť poznatkov reprezentovaných produkčnými pravidlami, ako aj formulami **JPL**, vyplýva iba z *implicitných* súvislostí plynúcich z jednotnej interpretácie rovnakých symbolov,
 - ◆ produkčný systém bez vhodného štruktúrovania a prípadných riadiacich konštruktov (napr. semaforey alebo mimologické konštrukty reprezentujúce neurčitosti) vedie k riešiacim procesom, ktoré nie je možné považovať za adekvát usudzovacích procesov človeka.

2.2.2 Reprezentácia znalostí asociatívnou sieťou

- Asocionistický prístup k reprezentácii
- Problematika **začleniteľnosti** poznatkov zvyrazňuje *asocionistické predstavy*. Tie vo vývoji reprezentačných prostriedkov viedli k vzniku **asociatívnych (sémantických) sietí**. Motiváciou bola snaha o prekonávanie ťažkostí s izolovanou reprezentáciou poznatkov, ktoré sa prejavujú v prípade formalizmov **JPL** a produkčných systémov. Išlo teda o snahu nájsť formalizmus umožňujúci reprezentovať **konceptuálne jednotky - TERMS**. *Konceptuálnou jednotkou* to je čosi, čo *možno opísať*, čo *má určité vlastnosti*, o čom *môžeme čosi vedieť a čosi predpokladať*, o čom *môžeme mať informácie*.
- Konceptuálna jednotka
- Asociatívne siete** v úlohe reprezentačného prostriedku mali umožniť združovanie všetkých poznatkov a informácií o danej konceptuálnej entite tak, aby boli
- Združovanie

²² Avšak pre mnohé aplikácie vysvetľovanie produkované iba na báze znalostí reprezentovaných produkčnými pravidlami nie je postačujúce.

informácií a poznatkov	<p>čo najbezprostrednejšie sprístupiteľné z miesta, kde je symbolová reprezentácia tejto entity uložená. Myšlienka spočíva v nájdení účelnej a efektívnej organizácie údajov reprezentujúcich znalosti typu <i>termov</i> zodpovedajúcich</p>
Vymedzenie asociatívnej siete	<ul style="list-style-type: none"> ▲ konceptom (pojmom, entitám), s ktorými sú združené okrem ich opisov a vzťahov aj spôsoby zachádzania s nimi, hoci len implicitne, čo vplýva z konštrukcie funkcií komplementárneho interpretátora, ▲ možnostiam vyjadrovať len čiasočnú znalosť, ktorá sa dá postupne doplňovať aj modifikovať, ▲ metódam umožňujúcim porovnávať daný pojem s ostatnými v sieti a tým spoznať kontext, v ktorom sa vyskytuje. <p>Asociatívne siete sa dajú vo všeobecnosti charakterizovať takto:</p> <ul style="list-style-type: none"> ▲ konkrétne (individuové) aj všeobecné pojmy sú reprezentované samostatnými vrcholmi v sieti, ▲ elementárne poznatky sú reprezentované podgrafmi siete, ktoré s každým javovým vrcholom obsahujú všetky jeho povinné atribúty (vlastnosti), ▲ každému pojmu zodpovedá najviac jeden vrchol v sieti a všetky s ním súvisiace informácie sa dajú vyčítať z jeho väzieb s okolitými vrcholmi, menovite ide najmä o väzby na <ul style="list-style-type: none"> ▪ všeobecnejšie pojmy ktoré ho obsahovo vymedzujú, ▪ na špeciálnejšie pojmy - až po individua, ktoré sú jeho inštanciami, ▪ javy, na ktorých sa daný pojem zúčastňuje.
Prednosti asociatívnych sietí: taxonómie, zahrňovanie, exkluzivita	<p>Prednosti týchto sietí spočívajú v</p> <ul style="list-style-type: none"> ✓ možnosti vytvárať taxonómie entít – vzťahy generalizácia-špecializácia, ✓ úspornosti reprezentácie - zahrňovanie, bez potreby opakovať vlastnosti získavané z nadtriedy, ✓ reprezentácii vzťahov exkluzivity - vzájomného vylučovania sa entít tej istej nadtriedy.
Lineárny zápis siete	<p>Praktická symbolová reprezentácia asociatívnych sietí sa dá realizovať prostriedkami vhodného JPL napríklad s predikátmi typu</p>
Nedostatky	<p>JE_PODMNOŽINOU (<i>trieda1, trieda2</i>), JE_PRVKOM (<i>trieda, individum</i>), JE_ČASŤOU (<i>trieda1:individum_a, trieda2:individum_b</i>) JE_ČASŤOU (<i>individum1, individum2</i>), JE_VLASTNÍKOM (<i>trieda1:vlastník, trieda2:vlastnené, čas:začiatok, čas:koniec</i>), JE_PRÍPADOM (<i>vlastnenie:vlastníctvo_i</i>), VLASTNÍCTVO_i (<i>vlastník:Ivan, vlastnený:Dunčo, čas:1991, čas:1992</i>).</p> <p>Použitelnosť asociatívnych sietí vyžaduje vytvoriť k nim príslušnú komplementárnu interpretačnú zložku, lebo explicitne neurčujú procesy produkujúce postupy usudzovania. Neobsahujú vlastné prostriedky generovania príslušných procesov.</p>

2.2.3 Procedurálna reprezentácia poznatkov

Procedúry deklaratívneho programu

V súvislosti so znalostnými systémami sa štandardne stretávame s **proceduralitou interpretátora**. Jedná sa o klasický procedurálny program, ktorý viacnásobne používa **všeobecné parametrizovateľné procedúry** na báze vopred deterministicky predpísaných postupností. V jednom behu sa opakovane uplatňujú procedúry ako napr. hľadanie komplementárnych literálov, rezolovanie, modus ponens atď. Uplatňovanie takých procedúr je determinované syntaxou prostriedkov reprezentácie znalostí.

Procedúry procedurálneho programu

Okrem toho aj v rámci deklaratívnych programov sa dá stretnúť s **procedurálnou reprezentáciou znalostí - tvoria súčasť deklaratívneho programu**. To je predmet záujmu tohto článku.

Operacionistické znalosti

Deklaratívny program môže obsahovať množstvo vzájomne odlišných **partikulárnych – konkrétnych čiastkových - procedúr** reprezentujúcich poznatky, ktoré sa týkajú spracúvania symbolov (hoci aj týkajúcich sa kvantít). Môže pritom ísť aj o ich opakovateľné použitie. Ich potenciálne použitie nie je deterministicky predpísané, sú iba definované podmienky, za ktorých by mohli byť použité. Teda ich uplatňovanie je **situačne podmienené**. Keď teda v tomto článku hovoríme o procedurálnej reprezentácii poznatkov, máme na mysli

operacionistické znalosti - poznatky majúce povahu určitého procesu, ktoré vykonávajú špecifické problémovo zamerané operácie.

Vznik tohto druhu reprezentácie bol motivovaný zdôrazňovaním **použitelnosti** poznatkov prostredníctvom súboru procedúr, ktorými sa reprezentujú **partikulárne** (parametrizovateľné) **poznatky**.

Volanie procedúr cieľmi

Medzi významné charakteristiky tohto spôsobu reprezentácie môže patriť **volanie procedúr cieľmi**, t.j. na základe **ich funkcie**, či **ich obsahu**. Je to odlišný princíp v porovnaní s volaním programu menom! Ilustrujme to na príklade.

Uvažujme triviálny prípad: identifikácia mnohouholníka. Konkrétne, v danej situácii treba zistiť, či skúmaný objekt je štvorcom. Symbolova podoba formulácie cieľa úlohy by mohla mať nasledujúci tvar

Cieľ: **ŠTVOREC(objekt:X)**

Daná úloha, či otázka

Takáto formulácia požiadavky vedie k zisťovaniu toho, či objekt **X** vyhovuje požiadavkám predikátu **ŠTVOREC**, alebo inými vyjadrením, či je za reťazcom "Cieľ:" uvedená atomárna **propozícia** pravdivá. Z druhej časti predošlej vety vyplýva, že ide vlastne o **zodpovedanie otázky: Je objekt X štvorcom?**

Odpoveď sa má získať na základe nasledovných symbolovo vyjadrených faktov

Dostupné fakty

JE_ČASŤOU (objekt:X, objekt:AB) & ÚSEČKA(objekt:AB)
JE_ČASŤOU (objekt:X, objekt:BC) & ÚSEČKA(objekt:BC)
JE_ČASŤOU (objekt:X, objekt:CD) & ÚSEČKA(objekt:CD)
JE_ČASŤOU (objekt:X, objekt:DA) & ÚSEČKA(objekt:DA)

Na zistenie odpovede možno použiť nasledujúce dve iba uvádzané procedúry (sú uvádzané iba ich fragmenty v rozsahu potrebnom na ilustrovanie princípu volania procedúry cieľom).

PROCEDÚRA_1 (zisťuje počet úsečiek tvoriacich skúmaný objekt X)

Cieľom volané procedúry	<p>...</p> <p>JE_ČASŤOU(objekt:v, objekt:w) (propozícia, test)</p> <p>...</p> <p>POČET_ÚSEČIEK(objekt:w) := POČET_ÚSEČIEK(objekt:w) + 1 (počítadlo a pmätanie výsledku)</p> <p>...</p> <p>+</p> <p>PROCEDÚRA_2 (zistuje druh objektu v)</p> <p>...</p> <p>ÚSEČKA(objekt:w) (propozícia)</p>
a priebeh ich volaní	<p>...</p> <p>POČET_ÚSEČIEK(objekt:v) = 4 (propozícia)</p> <p>...</p> <p>ŠTVOREC(objekt:v) (cieľová propozícia)</p> <p>...</p> <p>Špecifikovaný cieľ riešenia problému ŠTVOREC(objekt:X) vedie k tomu, že interpretátor (po substitúcii) identifikuje použiteľnosť druhej z uvedených procedúr - zodpovedajúci literál sa nachádza v tele druhej z uvedených procedúr. Pri jej zbiehaní sa narazí na propozíciu POČET_ÚSEČIEK(objekt:V)=4. Jej vyhodnotenie sa stáva podcieľom. Interpretátor hľadá procedúru, v ktorej sa vyskytuje premenná POČET_ÚSEČIEK(objekt:V) na ľavej strane priradovacieho príkazu. V danom prípade najde ju v prvej z uvedených procedúr a preto ju inicializuje.</p> <p>Je zrejmé, že interpretátor musí stelesňovať prostriedky substituovania, unifikácie a príslušného prehľadávania procedúr. Na základe stanoveného cieľa, resp. aktuálneho podcieľa, prehľadáva procedurálne reprezentované poznatky a vyberá z nich tie, ktoré môžu svojim účinkom zabezpečiť dosiahnutie výsledku. Tým realizuje postupné, vopred presne nedetrminované reťazenie partikulárnych procedúr a teda produkuje riešiacu postupnosť.</p> <p><i>Odvodzovací proces</i> nahradzujúci slepé prehľadávanie procedúr, ktoré sú súčasťou deklaratívneho programu, cielenými postupmi sa zabezpečuje zodpovedajúcimi prostriedkami. K ním patria stavové a riadiace údajové štruktúry, ktoré slúžia informovanému rozhodovaniu v odvodzovacom procese. Ide napríklad o štruktúry, ktoré</p> <ul style="list-style-type: none"> ✓ zjednodušujú identifikáciu procedúr potenciálne zabezpečujúce dosiahnutie práve aktuálnych (pod)cieľov, ✓ registrujú aktuálny stav jednotlivých procedúr, t.j. ich stav aktivovania, zamietnutia, prerušenia, opustenia²³ a pod., ✓ zaznamenávajú situačné použitie jednotlivých procedúr a ich úspešnosť v jednotlivých situáciách <p>atď.</p> <p>Medzi nevýhody procedurálnej reprezentácie poznatkov sa radí strata možnosti postupného, ľahko modifikovateľného a nezávislého reprezentovania jednotlivých poznatkov, potenciálne ťažkosti so zrozumiteľnosťou deklaratívneho programu (môže sa v ňom vyskytovať neprehľadné rekurzívne volanie procedúr) a nebezpečenstvo, že sa doň zavedú neľahko identifikovateľné nekonzistentnosti.</p> <p>O začleniteľnosti procedurálne reprezentovaných poznatkov možno vysloviť</p>
Informované rozhodovanie	
Nevýhody procedurálne j reprezentácie	
Začleniteľ- nosť pro- cedurálne reprezen- tovaných poznatkov	

²³ Význam týchto pojmov sa bližšie osvetľuje v neskorších častiach týchto textov.

nasledovné:

- ✓ *V prípade bežných programovacích jazykov sa spravidla jedná o **explicitné** odvolávanie na procedúry ich menami. V takom prípade súvislosť medzi dvomi zložkami programu vyplýva z výskytu mena procedúry v roli argumentu operácie volania na jednom mieste a na druhom mieste v roli identifikátora určitej sekvencie kódov.*
- ✓ *V prípade deklaratívnych programov sa vlastne tiež jedná o explicitné vzťahy. Rozdiel je iba v tom, že tie sú špecifikované symbolovými konštruktami, ktoré v prípade referencovania tvoria **zložku propozície (podmienky)** a v prípade volaného prvok prirad'ovacieho príkazu v tele určitého programu.*

2.2.4 Reprezentácia znalostí rámcami

Motivácia
rámcovej
reprezentácie
znalostí

Motiváciu vzniku tejto formy reprezentácie **znalostí termového typu** je potrebné vidieť v snahách motivovaných hľadiskami

- ◆ **INFORMATCKÝMI: účelná a účinná syntéza atribútov vyjadriteľnosti, použiteľnosti, efektivity a začleneniteľnosti.**
- ◆ **PSYCHOLOGICKÝMI: reprezentácia predstáv a očakávaní (mentálny model) súvisiacich s určitou entitou a procesov ich zlad'ovania so skutočnosťou.**

Ilustračný
príklad

Majme ilustračný príklad na osvetlenie podstaty reprezentačného formalizmu rámcového typu. Uvažujeme v ňom prípad osoby, ktorá pre ňu v nie dostatočne známej budove má nájsť zasadaciu miestnosť, v ktorej sa má zúčastniť rokovania:

1. Nachádzajúc sa osamotene (nemá sa koho opýtať) v priestrannej hale (chodbe) budovy, v ktorej je rad číselne (nie funkčne, t.j. nápismi alebo ikonami) označených dverí, medzi ktorými sa dajú predpokladať aj tie, čo vedú do cieľovej miestnosti.
2. Očakáva, že po ich otvorení uvidí určité charakteristiky - atribúty miestnosti: povedzme štyri istým spôsobom upravené steny vyzdobené obrazmi, kdesi v stenách okná a prípadne ďalšie dvere, podlahu vyloženú parketami čiastočne pokrytú kobercom, strop so svetidlom, dlhší konferenčný stôl obklopený pohodlnejšími stoličkami, nevylučuje kávu, či osviežujúce nápoje na stole, prítomnosť tabule, premietacieho plátna, spätného projektora a pod.
3. Dá sa predpokladať, že v mysli má pripravenú predstavu, určité očakávanie, t.j. akýsi (zrejme všeobecný a pomerne flexibilný) rámec (v psychologizujúcom poňatí mentálny model) zodpovedajúci konceptu primeranej zasadačej miestnosti.
4. V závislosti od toho, čo po otvorení dverí naozaj uvidí, pôvodné predpoklady/očakávania sa buď naplnia, alebo sú v rozpore s realitou. Ak by otvorené dvere viedli do záhrady, k schodom na povalu, či do pivnice, dôjde k nezhode reality s danou predstavou. Ak by za otvorenými dverami nebodaj zazrela šíry oceán, nezhoda by mohla dosiahnuť stupeň šoku. V prípade, že by išlo o kancelársku miestnosť, predsieň, klasickú učebňu, čítareň a pod. môže situačne dôjsť k čiastočnej zhode s rámcovou predstavou miestnosti.
5. Iba ak by to bola naozaj typická zasadačka alebo miestnosť, ktorá podľa všeobecných predstáv by mohla slúžiť ako zasadačka, došlo by k dostatočnej zhode medzi očakávanými a skutočnými vlastnosťami reality, pričom sa modifikujú a konkretizujú hlavné charakteristické predstavy o očakávaných vlastnostiach miestnosti (napr. tvar a pozícia okien, druh, kvalita, rozmery a umiestnenie nábytku, veľkosť a kvalita koberca a pod.) a to nezávisle od toho, či je alebo nie je v miestnosti tabuľa, spätný projektor s premietacím plátnom, či na stole je nejaké občerstvenie, alebo či miestnosť má ešte ďalšie dvere.

Tento príklad hovorí o určitých predstavách/očakávaniach a naznačuje proces ich konfrontácie a zlad'ovania s realitou. Treba pritom zvýrazniť, že niektoré z očakávaní, aj keď sú modifikovateľné, sú považované za **nevyhnutné podmienky zhody** (ak by namiesto stola/stolov boli v miestnosti posteľ, kuchynské zariadenie, rad vaní, či bazén, nebola by splnená nevyhnutná podmienka), iné tvoria iba fakultatívne podmienky (absencia koberca, občerstvenia, tabule, ...).

Nevyhnutné a
nepovinné
očakávania

Špecifikácia rámcov	<p>Čo sú teda rámce ako prostriedky reprezentácie? Rámce sú údajové štruktúry, ktoré integrujú prednosti</p> <ul style="list-style-type: none"> ◆ asociatívnych sietí, vrátane možností reprezentovať vlastnosti entít a hodnôt, ktoré sú pre ne povinné, prípustné, očakávateľné (predpokladateľné), vrátane spôsobov ich nadobúdania - výhodné na reprezentovanie termových znalostí, ◆ reprezentácie asertívnych poznatkov v podobe produkčných pravidiel, resp. formúl logických formalizmov (vrátane JPL), ◆ procedurálnej reprezentácie znalostí, teda akciami, cieľmi, či menom volaných procedúr.
Charakteristiky rámcov	<p>O čosi podrobnejšie sa dajú rámce špecifikovať nasledujúcim prehľadom charakteristík:</p>
Identifikovateľná, vnútorne štrukturovaná sústava údajových položiek	<p>RÁMCE</p> <ul style="list-style-type: none"> ✓ sú svojim menom identifikovateľné údajové štruktúry, ktoré sú vzájomne prepojitelné (môžu na seba odkazovať, odvolávať), ✓ ich štruktúra obsahuje (aj hierarchicky) usporiadanú postupnosť rôznych pomenovaných a typovo charakterizovateľných údajových položiek, nazývaných rubrika, ktoré sa jednak v rozsahu rámca môžu na seba odvolávať a jednak sú referencovateľné aj z iných rámcov alebo referencujú iné rámce či ich rubriky (menom rubriky a menom rámca, v ktorom sa nachádzajú),
Rubriky	<ul style="list-style-type: none"> ✓ rubriky zodpovedajú vlastnostiam reprezentovanej entity, sú to rezervované pamäťové miesta, v ktorých buď už sú požadované údaje, alebo sa tieto do nich, na základe pozorovania skutočnosti, postupne kladajú ako ich hodnoty v súlade so špecifikovaným typom rubriky,
a ich dátový typ	<ul style="list-style-type: none"> ✓ (dátový) typ rubriky (ich šablóna) určuje požiadavky, resp. ohraničenia vzťahujúce sa k povoleným hodnotám rubriky (napr. kardinalitu, t.j. či je prípustná iba jediná alebo aj viacnásobná hodnota, prípustné vlastnosti hodnoty - napr. celočíselná, reálna, či komplexná hodnota kvantity, pravdivostná (propozičná) hodnota, reťazcová hodnota, meno procedúry, telo procedúry, produkčné pravidlo atď.),
spôsob nadobúdania a prehľadávaní a hodnôt	<ul style="list-style-type: none"> ✓ (aj dynamicky meniteľnými) predpismi prioritného spôsobu nadobúdania hodnôt jednotlivých rubriky resp. prioritného prehľadávania a vyhodnocovania obsahu rubriky,
povinné a nepovinné hodnoty	<ul style="list-style-type: none"> ✓ povinnými (obligatónymi) rubrikami s povinnými hodnotami (nevyhnutnými vlastnosťami reprezentovanej entity, napr. <i>minimálny počet (nezaoblených) stien v miestnosti s mnohouholníkovou podlahou: 3, počet okien v obývacej miestnosti ≥ 1</i>, [okno je reprezentovateľné vlastným rámcom]), fakultatívnymi rubrikami s voliteľnými hodnotami (zodpovedajúce možným, avšak nepovinným vlastnostiam entity, napr. <i>rozmer steny, farba steny, prítomnosť okna v stene, rozmery miestnosti</i> [rubrika môže obsahovať aj situačne aktivovanú procedúru, ktorá vypočíta túto hodnotu z rozmerov stien]) a iné,
fixné a dynamické hodnoty	<ul style="list-style-type: none"> ✓ obsahom (hodnotou) rubriky, ktorý môže byť vopred definovaný a fixný, v priebehu odvodzovania získaný fixný, alebo dočasný a modifikovateľný (dynamický), náhradný (zástupný, predpokladateľný, 'default')
špecifikovanie zovše-	

bečení, špecializácii a	nahradzujúci nepoznanú, len očakáateľnú, domnelú skutočnosť, ✓ hodnotami rubriék (spravidla obligatónnymi) sa špecifikujú <ul style="list-style-type: none"> ▪ zovšeobecnia reprezentovanej entity, ich nadtriedy, ktoré sú tiež reprezentované rámcami (napr. miestnosť je prvkom triedy <i>uzavretý priestor</i>), ▪ špecializácie, t.j. buď ich podtriedy alebo individuové prípady, ktoré sú tiež reprezentované rámcami (napr. miestnosť má podtriedy: obývačka, spáľňa, sála, trieda, hala, a iné, pričom zasadačka č. 5 na konkrétnej adrese je individuová entita), ▪ súčasti reprezentovanej entity ako odkazy na zodpovedajúce rámce (stena, okno, podlahu, spätný projektor, ...), ▪ príčiny a dôsledky reprezentovanej entity odkazujúce na príslušné rámce, ▪ procedúry (identifikátor alebo telo procedúry môže byť obsiahnutý v rubrike),
částí entit	
kauzálit, procedúr hodnotami rubriék	
procesy priradené rubrikám	✓ s rubrikami je možné spájať aj explicitnú alebo implicitnú podmienku spustenia im zodpovedajúcich procedúr pri nadobudnutí, modifikovaní, zrušení ich hodnoty (napr. vyhľadanie najbližšej voľnej stoličky, pozdraviť prítomných, ak sa už nachádzajú v miestnosti, keď je potrebné vypočítať plochu podlahy, zistenie rozmerov a ich uplatnenie v zodpovedajúcej výpočtovej procedúre).

2.3 Prostriedky reprezentácie znalostí

Prístup k tvorbe ES

Všetky spomenuté prostriedky možno realizovať na základe vlastnej koncepcie v niektorom z bežných programovacích jazykov. Prirodzene všetky majú vzhľadom na problematiku reprezentácie svoje výhody aj nedostatky. V aplikáciách metód umelej inteligencie pôvodne bolo možné pomerne často sa stretávať s programovacími jazykmi ako je **LISP** alebo **PROLOG**. Tie sa však neskôr nahradzovali objektovo-orientovanými programovacími jazykmi, napr. **C++** alebo **SMALLTALK**.

Otázka optimálneho výberu programovacieho jazyka tvorí mnohodimenzionálnu problematiku. Každý možno hodnotiť z viacerých hľadísk a tie sú spravidla protichodné. Navyiac, pri hodnotení sa uplatňujú aj požiadavky aplikácií. Komplexný rozbor celej problematiky, vzhľadom na nevyhnutné predpoklady, rozsah a náročnosť presahuje zámer týchto textov. Preto len značne všeobecná úvaha:

Odporúčanie v prvom priblížení

výhodné je použiť jazyk, ktorý je výkonný pri uskutočňovaní rozmanitých reťazcových operácií (operácie so znakmi) a poskytuje dostatok možností na tvorbu rozmanitých údajovo-riadicích štruktúr, ktoré sú nevyhnutné pre efektívne riadenie interpretácie deklaratívneho programu.

Keď sa jedná o tvorbu netriviálnych reprezentačných prostriedkov, realizácia zodpovedajúcich produktov je intelektuálne, rozsahom a prácnosťou zvyčajne veľmi náročná. V snahe racionalizovať príslušné programátorske činnosti, boli vytvorené viaceré špeciálne programovacie jazyky, napr. **KRL** (Knowledge Representation Language), **FRL** (Frame Representation Language), **OWL** (One World Language) a iné. Tieto jazyky mali predovšetkým teoretický význam, v praxi nenašli príliš veľké uplatnenie.

V súčasnosti sa uplatňujú predovšetkým **vývojové prostredia** tvorby expertných systémov. Sú to programové prostriedky, ktoré umožňujú vytvárať

- **deklaratívny program**, čo v kontexte expertných systémov sa nazýva **báza znalostí**,
- **interpretátor deklaratívneho programu**, čo v kontexte ES sa nazýva **inferenčný mechanizmus**,
- **používateľské rozhrania** - v kontexte ES nazývané **komunikačný modul**, a iné.

Príkladmi takýchto (komerčne dostupných) prostredí sú napr. Egeria, M4, Leonardo, ART-IM, KEE, Kappa, NexpertObject, RTworks, G2 a rastúci počet ďalších. V novších verziách, či variantoch týchto prostredí sa uplatňujú princípy objektovo-orientovaných programovacích technológií.

Hoci rôzne prostredia poskytujú rozmanité možnosti, ktorými sa dá vyhovieť požiadavkám *expresivity, odvodzovacej účinnosti, výpočtovej účinnosti a začleniteľnosti*, aj tie majú svoje vzájomne sa líšiace prednosti a ohraničenia. Sú tiež nositeľmi rôznych implementačných náhodností, ktorých neželaný dopad sa môže prejaviť až po dlhšom používaní. Vzhľadom na ich vysokú nadobúdaciu cenu (cena niektorých vývojových prostredí sa pohybuje rádomo na úrovni miliónov SK), pred ich zakúpením je potrebné veľmi starostlivo zvažovať a preskúšavať, či vyhovujú požiadavkám aplikácií. Ich vhodnosť sa spravidla preukáže až pri konkrétnom a opakovanom použití. Preto je výhodné pred ich potenciálnym zakúpením zabezpečiť ich skúšobné používanie.

Tak ako pri programovacích jazykoch, ani pri vývojových prostrediách nie je zatiaľ možné poskytnúť dostatočne všeobecné návody na ich optimálny výber. Rozhodne však treba uprednostňovať prostredia, ktoré vzhľadom na aplikácie poskytujú čo najpružnejšie možnosti reprezentácie a používania znalostí. Menovite také, čo umožňujú

- ✓ prístup k tomu istému poznatku z rôznych aspektov,
- ✓ realizáciu rôznych reprezentačných formalizmov,
- ✓ vytvárať rozmanité problémovo zamerané makro-operácie.

Najmä pokiaľ ide o makro-operácie (stretne sa s nimi ešte v ďalších kapitolách) je dôležité, aby ich bolo možné vytvárať na základe vhodnej konceptualizácie riešiacich postupov danej aplikačnej oblasti bez toho, aby ich tvorba príliš podliehala vlastnostiam daného programového prostredia.