

## 4.3.8. Rámce a skripty

Existuje mnoho dôkazov z oblasti psychológie, že ľudia využívajú veľké množstvo vedomostí z predchádzajúcich skúseností pri interpretácii nových situácií [1, 6, 7]. Napríklad pri návšteve reštaurácie, kde sme ešte neboli, máme množstvo očakávaní, predpokladov (tykajúce sa čašníkov, stola, jedla, ...) na základe návštev iných reštaurácií. Okrem týchto predpokladov o objektoch v iných reštauráciách máme aj predpoklady o postupnosti udalostí. Pre reprezentáciu takýchto objektov a udalostí boli navrhnuté rámce (M. Minsky, 1975) a skripty (Schank, Abelson, 1977). Obe dve metódy organizujú vedomosti tak, aby bolo možné usmernenie pozornosti a uľahčenie inferencie.

**Rámce** - nové údaje sú reprezentované pomocou pojmov z predchádzajúcich skúseností, okrem toho organizácia vedomostí umožňuje spracovanie na základe očakávaní - v rámci toho zodpovedajú rubriky (slot). Rámec je údajová štruktúra, ktorá zahŕňa deklaratívne aj procedurálne informácie v preddefinovanom vnútornom vzťahu. Napríklad rámec pre všeobecný (generický) pojem pes môže obsahovať rubriky typu POTRAVA, MAJITEĽ, MENO a pripojené procedúry pre určenie majiteľa, ak to nie je známe. Rámec pre konkrétneho psa má tie isté rubriky (zdedené zo všeobecného rámca pre psa), ale obsahy rubriek sú špecifické. V jazyku tohto typu KRL (Knowledge Representation Language) [1] môže rámec vyzeráť takto

Generický PES rámec

Druh: ZVIERA; OBLÚBENÉ\_DOMÁCE\_ZVIERA

Potrava:

Majiteľ: OSOBA

(if\_needed: najdi OSOBA s  
OBLÚBENÉ\_DOMÁCE\_ZVIERA = tento\_pes)

Meno: VLASTNÉ\_MENO

(default = Dunčo)

Rámec SUSEDŮV\_PES

Druh: PES

Potrava: baranina

Majiteľ: Jano

Meno: Adonis

Mechanizmus rubriek umožňuje aj uvažovanie - hľadanie potvrdenia pre dané očakávanie - zapínanie rubriek. Pre ilustráciu uvedieme rámec všeobecného pojmu reštaurácia (RESTAURANT - keďže rámce pochádzajú z USA, aj tento ilustračný príklad sa týka reštaurácií v USA, nie v ČSFR).

## Generický rámec REŠTAURÁCIA

Druh: Obchodné\_zariadenia

Typ:

rozsah: (Bufet\_automat, Reštaurácia\_sadni\_si\_sám,  
Reštaurácia\_čakaj\_kým\_ťa\_usadia)

default: Reštaurácia\_čakaj\_kým\_ťa\_usadia

if\_needed: IF zásobník\_táčok THEN Bufet\_automat

IF čakaj\_na\_znamenie\_čašníka OR

nutná\_rezervácia

THEN Reštaurácia\_čakaj\_kým\_ťa\_usadia

OTHERWISE Reštaurácia\_sadni\_si\_sám

Umiestnenie:

rozsah: Adresa

if\_needed: (POZRI Menu)

Názov:

if\_needed: (POZRI Menu)

Typ\_jedál:

rozsah: (Hamburger, Čínske, Americké, Morské,  
Francúzske)

default: Americké

if\_added: (Aktualizácia Alternatív\_Reštaurácií)

Otvorené:

rozsah: 24\_hodín\_dňa

default: každý\_večer\_okrem\_pondelka

Forma\_platenia:

rozsah: (Hotovosť, Kreditná\_karta, Šek,  
Skript\_umyvacie\_riadu)

Postupnosť\_udalostí:

default: Skript\_Návšteva\_reštaurácie

V tomto rámci je reprezentovaných viacero typov vedomostí: rubrika "Druh" sa používa na definovanie hierarchie zdedených vlastností (property inheritance hierarchy) medzi rámcami, vlastnosti otcovského rámca môžu byť zdedené synovským rámcom (obdoba ISA hrany u sémantických sietí); rubrika "Umiestnenie" má podrubriky (rubriky môžu obsahovať aj zložité štruktúry typu rámca). Rubrika "rozsah" (range) obsahuje očakávania, predpoklady, aké druhy reštaurácií môžu byť. Rubrika "if\_needed" ("ak\_je\_potrebné") obsahuje pripojenú procedúru, ktorú možno použiť na určenie obsahu rubriky, ak je to potrebné. Podobne existujú procedúry "if\_added" ("ak\_pridané") a "if\_deleted" ("ak\_vymazané"), ktoré sa spúšťajú pri pridaní resp. vymazaní obsahu rubriky a vykonávajú špecifikovanú činnosť. Dôležitým typom rubriky je "default" ("preddefinované") - navrhuje hodnotu obsahu rubriky, ak neexistuje protirečivé tvrdenie. V zložitejších rámcoch môžu existovať ešte ďalšie typy rubriek.

Ako je naznačené v rubrike "Postupnosť\_udalostí" ("Event\_se

quences"), vedomosti o tom, čo sa zvyčajne robí v reštaurácii môžu byť reprezentované skriptom typu

### Skript Návšteva\_reštaurác.

Oporné\_slová: (Reštaurácia, Peniaze, Jedlo, Menu, Stôl, Stolička)  
 Rola: (Hladný\_človek, Čašník, Kuchár)  
 Uhol\_pohľadu: Hladný\_človek  
 Čas\_udalosti: (Otváracie\_hodiny\_reštaurácie)  
 Miesto\_udalosti: (Umiestnenie\_reštaurácie)  
 Postupnosť\_udalostí:  
 najprv: Skript Vstup\_do\_reštaurácie  
 potom: IF (Čakaj\_kým\_ča\_usadia\_znamenie OR Rezervácia\_stola)  
 THEN (Skript Upútanie\_pozornosti\_čašníka)  
 potom: Skript Prosím\_posaďte\_sa  
 potom: Skript Objednanie\_jedla  
 potom: Skript Konzumácia\_jedla  
 UNLESS (Dlhé\_čakanie)  
 WHEN Skript Nahnevany\_odchod  
 potom: IF (Kvalita\_jedla\_lepšia\_ako\_prijateľná)  
 THEN Skript Kompliment\_kuchárovi  
 potom: Skript Platenie  
 nakoniec: Skript Odchod\_z\_reštaurácie

Takýto skript špecifikuje normálnu alebo preddefinovanú postupnosť udalostí, rovnako ako predpoklady (očakávaní) a možné chybové situácie; vyžaduje aj niekoľko deskriptorov typu "Oporné\_slová" (Props) a "Rola" (Role), ktoré sa odvolávajú na iné rámce. Konštrukcia typu "Skript1 UNLESS Podmienka WHEN Skript2" znamená, že sa vykoná "Skript1", s výnimkou prípadu (UNLESS-iba ak nie"), že je splnená "podmienka", kedy sa vykoná "Skript2".

Zaujímavou črtou rámcov je ich schopnosť zistiť, či sú aplikovateľné v danej situácii. Pre určenie danej situácie (dialóg, scéna, problém) sa vyberie určitý "pravdepodobný" rámec a jeho terminálne hodnoty sa porovnávajú s faktami danej situácie. Ak porovnanie (matching) nie je úspešné, odovzdá sa riadenie inému rámcu.

V niektorých systémoch, používajúcich rámce, sú pripojené procedúry (attached procedures) principiálnym mechanizmom pre usmerňovanie uvažovania, sú aktivované pri napíňaní rubriek (napr. if\_needed).

Ak zvolíme určitý rámec pre reprezentáciu konkrétnej situácie, najdôležitejšie je potom vyplniť rubriky tohto rámcu (upresniť details). Napríklad ak si zvolíme všeobecný rámec reštaurácie (Generický rámec Reštaurácia), musíme okrem iného určiť obsah rubriky "Typ". Tu je niekoľko možností - typ je zdedený (v

našom prípade je však niekoľko alternatív); použijeme preddefinovanú hodnotu, ak nedôjde k protirečeniu; použije sa procedúra `if_needed` - výpočtovo najnáročnejšie, avšak môže sa použiť aktuálny kontext - špecifické heuristiky rubriky (slot-specific heuristics). Čiže všeobecné metódy riešenia môžu byť rozšírené o vedomosti z problémovej oblasti.

Okrem usmernenia, získavania, zhromažďovania ďalšej informácie môže vypĺňanie rubrik rámca aj potvrdiť, či sme zvolili vhodný rámec pre danú situáciu, resp. scénu. Ak dôjde k odmietnutiu daného rámca, pripojené procedúry môžu spôsobiť odovzdaniu riadenia iným rámcom.

"Spúšťače procedúr" (triggers) - často používanou formou procedurálneho spracovania sú procedúry, ktoré sa spúšťajú (vyvolávajú) pri vyplnení danej rubriky, resp. pri zmene jej obsahu. Tieto "trigger" procedúry sú vlastne implementáciou spracovania riadeného údajmi (data driven processing) alebo spracovania riadeného udalosťami (event driven processing), pretože prevezmú riadenie, iba keď sa vyskytne určitý údaj, alebo nastane určitá udalosť. Napríklad procedúra `if_added` sa použije na modifikáciu alternatív typov reštaurácií.

Zo všeobecného pojmu rámca vychádza jazyk KRL (Knowledge Representation Language; Bobrow, Winograd, 1977) pre reprezentáciu vedomostí. Systém KRL sa pokúša integrovať procedurálne vedomosti so širokou bázou deklaratívnych foriem vedomostí.

Jazyk FRL (Frame Representaton Language) je experimentálny jazyk pre skúmanie použitia rámcov ako techniky reprezentácie vedomostí. Základným objektom FRL je pojem tzv. FRL rámca, ktorý vznikol formalizáciou všeobecného pojmu rámca tak, aby bol vhodný pre spracovanie na počítači. FRL rámec je údajová štruktúra nasledovného tvaru:

```
(rámec
  (rubrika1 (položka1 (hodnota1 (poznámka1 (návestie1
    správai ... viac správ)
    ... viac návestí)
    ... viac poznámok))
    (hodnota2 (poznámka1 (návestie1 ...)))
    ... viac hodnôt))
  (položka2 (hodnota1 (poznámka1 ...)))
    ... viac položiek))
  (rubrika2 (položka1 (hodnota1 (poznámka1 ...) ...) ...) ...)
    ... viac rubriék))
```

Formálnejšie možno pojem rámca vo FRL pomocou Backus-Naurovej formy vyjadriť takto:

```
<rámec> ::= ( <meno> {<rubrika>} )
```

```

<rubrika> ::= ( <meno> {<položka>} )
<položka> ::= ( <meno> {<hodnota>} )
<hodnota> ::= ( <S-výraz> ) | ( <meno> {<poznámka>} )
<poznámka> ::= ( <návestie> {<správa>} )
<návestie> ::= <atóm>
<správa> ::= <atóm>
<meno> ::= <atóm>

```

Zložené zátvorky { } vyjadrujú možnosť 0 až n-násobného opakovania obsahu týchto zátvoriek; S-výraz a atóm sú pojmy prevzaté z jazyka LISP [8].

FRL rámec je zoznam s piatimi možnými úrovňami vnorenia. Rubriky v FRL rámci možno chápať ako zovšeobecnenie dvojice atribút-hodnota z tradičnej lisповskej reprezentácie zoznamom vlastností [8]. Vzťah medzi rámcami, podľa ktorého sú údaje zdedené, definujú dve rubriky, nazývané systémové:

- AKO (A Kind Of - druh)
- INSTANCE (príklad, inštancia)

Relácia AKO je použitá na zavedenie pojmu hierarchie rámcov, v ktorej je všeobecná informácia, vložená vyššie do hierarchie, zdedená viacerými špeciálnymi pojmami nižšie v hierarchii. Relácia INSTANCE, definujúca reláciu výskytu daného rámca, je inverzná k rubrike AKO. Každá rubrika môže obsahovať niekoľko položiek, na úrovni položky môže byť bližšie špecifikovaný typ informácie, ktorú rubrika obsahuje. FRL systém umožňuje pracovať so šiestimi typmi položiek:

1. VALUE je položka, ktorá obsahuje hodnotu danej rubriky
2. DEFAULT je položka, ktorá v prípade potreby poskytuje preddefinovanú hodnotu pre danú rubriku
3. IF\_ADDED položka obsahuje procedúry, ktoré sú spúšťané vždy vtedy, keď je do položky VALUE v danej rubrike pridávaná hodnota
4. IF\_REMOVED položka obsahuje procedúry, ktoré sú spúšťané vtedy, keď je z rubriky hodnota vymazávaná
5. IF\_NEEDED je položka obsahujúca procedúry spúšťané pri vyčíslívaní hodnoty danej rubriky
6. REQUIRE položka obsahuje predikáty, ktoré opisujú a ohraničujú hodnotu v danej rubrike

Každá rubrika môže obsahovať niekoľko hodnôt. Pre bližšiu špecifikáciu hodnoty s ňou môže byť spojený komentár. Je tvorený návestím a jednou alebo viacerými správami.

Okrem toho boli rámce a skripty použité aj v týchto systémoch UI [9]:

- porozumenie prirodzeného jazyka (Bobrow, 1977) - systém GUS pre rezerváciu leteniek
- NUDGE (Goldstein, Roberts, 1977) - spracovanie neúplných,

nekonzistentných požiadaviek na plánovanie (management-scheduling requests)

- riešenie fyzikálnych problémov, formulovaných v prirodzenej reči - angličtine (Novak, 1977)
- UNITS (Stefik, 1980) - prenosný súbor programov
- KLONE (Brachman, 1978) - zahŕňa výskum v oblasti teórie návrhu systémov na báze rámcov
- SAM (Schank, Abelson, 1977) - porozumenie krátkych príbehov v prirodzenej reči (využíva skripty na výskum pojmu kauzality a porozumenie postupností udalostí), po porozumení textu môže robiť inferencie o podobných udalostiach

Metódy RV na báze rámcov sú pokladané za veľmi perspektívne v oblasti UI.

#### 4.3.9. Sémantické primitíva

Ak si už zvolíme techniku reprezentácie - logiku, procedurálne reprezentácie, sémantické siete, produkčné pravidlá, rámce - musíme riešiť ďalšiu otázku - otázku slovníka daného formalizmu (t.j. aké predikáty použijeme pri logickej reprezentácii, aké typy uzlov a hrán použijeme v sémantickej sieti apod.).

Pojem sémantické primitívum nemá presnú definíciu, voľne ho môžeme definovať ako ľubovoľný symbol, ktorý je použitý, ale nie definovaný v rámci systému, resp. ako významové prvky, na ktoré môže byť rozložené slovo alebo veta [1].

Yorick Wilks vyvinul systém pre strojový preklad z angličtiny do francúzštiny (anglický text --> vnútorná reprezentácia, t.j. údajová štruktúra, ktorej uzly reprezentujú sémantické primitíva --> francúzsky text). Wilks tvrdí, že takýto preklad si často vyžaduje povrchnejšie porozumenie textu, ako by si vyžadoval systém, schopný odpovedať na ľubovoľnú otázku, týkajúcu sa textu.

Medzi problémy, ktoré vznikajú pri takomto preklade, patrí dvoj- a viacznačnosť slov (slovo má viac významov) a problém rozšírenia významu slov, napr.: "Moje auto žerie 8 litrov".

Základná myšlienka Wilksovho prístupu je tzv. preferenčná sémantika (preference semantics) - vedomosti o možných významoch slov sa použijú na vyriešenie dvojznačnosti iných slov. Napríklad časť významu slova "žrať" preferuje (uprednostňuje) jedlé objekty a časť významu slova "mäso" je, že je to jedlý objekt. Ak význam slova nespĺňa všetky preferencie (napr. že subjektom pri slove "žrať" má byť živočích), potom sa môže prijať rozšírený význam slova. Formalizmus, v ktorom možno vyjadriť tieto preferencie, má bližšie k rámcom než k sémantickým sieťam.

Podrobnejšie je táto problematika diskutovaná v kapitole o spracovaní prirodzeného jazyka (podkapitola 7.3.3).