

PŘÍRODOVĚDECKÁ FAKULTA UNIVERZITY PALACKÉHO
KATEDRA INFORMATIKY

DIPLOMOVÁ PRÁCE

Implementace expertního systému v jazyce Common Lisp



Anotace

Expertní systémy mají v praxi bohaté využití. Jejich smyslem je asistovat expertovi na danou problematiku, či jej plně nahradit. V příloze bakalářské práce implementuji prázdný expertní systém s dopředným řetězením inspirovaný systémem CLIPS jako knihovnu v programovacím jazyku Common Lisp tak, aby jej bylo možno plně integrovat do dalších programů.

Děkuji Mgr. Martinu Dostálovi, Ph.D. za vedení této diplomové práce.

Obsah

1. Úvod	4
2. Praktická část	5
2.1. Uživatelská příručka	5
2.1.1. Instalace	5
2.1.2. Common Lisp	5
2.1.3. Struktura programu	5
3. Teoretická část	8
3.1. Expertní systémy	8
Reference	9

Seznam obrázků

Seznam ukázek kódu

1	ExiL code example	7
---	-----------------------------	---

1. Úvod

Ve své bakalářské práci jsem implementoval základní knihovnu pro tvorbu expertních systémů (tzv. prázdný expertní systém) s dopředným řetězením v jazyce Common Lisp. Cílem této diplomové práce je tuto knihovnu rozšířit o následující:

- syntaktický režim pro zajištění přiměřené kompatibility se systémem CLIPS,
- možnost vrácení provedených změn včetně odvozovacích kroků,
- podpora pro ladění s jednoduchým grafickým uživatelským rozhraním pro prostředí LispWorksTM,
- rozšíření odvozovacího aparátu o základní zpětné řetězení.

Pojem expertního systému spadá do oblasti umělé inteligence. Jde o počítačový systém, který simuluje rozhodování experta nad zvolenou problémovou doménou. Expertní systém může experta zcela nahradit, nebo mu při rozhodování asistovat.

Jazyk Common Lisp¹ (případně jiné dialekty Lispu) je častou volbou pro implementaci umělé inteligence díky svým schopnostem v oblasti symbolických výpočtů (manipulace symbolických výrazů), na nichž řešení těchto problémů často staví. Navíc jde o velmi vysokoúrovňový, dynamicky typovaný jazyk, díky čemuž je programový kód stručný, snadno pochopitelný a tudíž jednoduše rozšiřitelný.

Syntax systému CLIPS² byla zvolena proto, že jde o reálně používaný systém³, jehož syntax je Lispu velmi blízká, takže není těžké ji v Lispu napodobit.

Přestože běžnou praxí je začínat diplomovou práci teoretickou částí, definovat jednotlivé pojmy a principy a ty poté v praktické části uplatnit, rozhodl jsem se postupovat opačně, tedy začít práci praktickou částí. Domnívám se totiž (také na základě zkušeností nabytých při vypracování bakalářské práce), že je podstatně snazší (minimálně v řešené problematice) pochopit příklady bez detailní znalosti teorie, než snažit se pochopit teorii bez příkladů, na nichž si lze popisované pojmy a principy představit. V praktické části tedy uvedu jen minimální množství teorie nutné pro pochopení aktuálního problému, načež se k ní v teoretické části textu vrátím, pojmy zadefinuji přesně a rozšířím o souvislosti.

¹http://en.wikipedia.org/wiki/Common_Lisp

²<http://clipsrules.sourceforge.net>

³<http://clipsrules.sourceforge.net/FAQ.html#Q6>

2. Praktická část

Tato sekce popisuje knihovnu ExiL¹, která je výsledkem této diplomové práce. Nejprve v uživatelské příručce popíšu její instalaci, základní možnosti a typickou strukturu programu, který ji využívá. Poté v referenční příručce projdu všechny možnosti, které knihovna poskytuje. Načež v části věnované implementaci popíšu architekturu jejího zdrojového kódu a zmíním zajímavé části kódu implementující jednotlivá rozšíření. Nakonec uvedu několik větších příkladů použití knihovny a rozeberu několik dalších možných rozšíření a co by obnášela z pohledu implementace.

2.1. Uživatelská příručka

2.1.1. Instalace

- instalace - slime, sbcl, quicklisp, asdf, lispworks, získání kódu, git

Zdrojový kód knihovny je přiložen k této diplomové práci a lze jej také získat zklonováním gitového² repozitáře na adrese `git@github.com:Incanus3/ExiL.git` (v *nixových systémech např. zadáním příkazu `git clone git@github.com:Incanus3/ExiL.git`).

2.1.2. Common Lisp

- úvod do lispu - odkaz na practical common lisp, clhs

2.1.3. Struktura programu

popsat jednotlivé sekce kódu, jejich význam (korespondence s fázemi návrhu ES, formulace problému, formát dat, vstupní znalosti, odvozovací krok, řízení odvozování, ladění)

- definice prostředí
- definice šablon - formát dat
- definice znalostní báze - vstupní znalost - deffacts, defrules
- (nastavení sledování průběhu inference - watchers)
- (úprava průběhu inference - strategie)
- spuštění / krokování inference - reset, run, step

¹TODO: původ názvu

²<http://git-scm.com/>

- dotazy nad working memory - facts, agenda
- úprava working memory - assert, retract, modify
- dotazy nad znalostní bází - fact-groups, rules
- cleanup - volatile vs durable sloty prostředí
- undo/redo
- zpětné řetězení
- GUI

```
1 (deftemplate goal action object from to)
2 (deftemplate in object location)
3
4 (def facts world
5   (in :object robot :location A)
6   (in :object box :location B)
7   (goal :action push :object box :from B :to A))
8
9 (defrule move
10  (goal :action push :object ?obj :from ?from)
11  (in :object ?obj :location ?from)
12  (- in :object robot :location ?from)
13  ?robot <- (in :object robot :location ?)
14  =>
15  (modify ?robot :location ?from))
16
17 (defrule push
18  (goal :action push :object ?obj :from ?from :to ?to)
19  ?object <- (in :object ?obj :location ?from)
20  ?robot <- (in :object robot :location ?from)
21  =>
22  (modify ?robot :location ?to)
23  (modify ?object :location ?to))
24
25 (defrule stop
26  ?goal <- (goal :action push :object ?obj :to ?to)
27  (in :object ?obj :location ?to)
28  =>
29  (retract ?goal)
30  (halt))
31
32 (reset)
33
34 ; (step)
35 (run)
```

Ukázka kódu 1: ExiL code example

3. Teoretická část

3.1. Expertní systémy

Reference

- [1] Jackson, P.: *Introduction to Expert Systems*. Addison Wesley, 1998, ISBN 0-201-87686-8.
- [2] Norvig, P.: *Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp*. Morgan Kaufmann Publishers, 1991, ISBN 1-55860-191-0.
- [3] Doorenbos, R. B.: *Production Matching for Large Learning Systems*. Dizertační práce, Carnegie Mellon University, 1995.
<http://reports-archive.adm.cs.cmu.edu/anon/1995/CMU-CS-95-113.pdf>
- [4] Siebel, P.: *Practical Common Lisp*. Apress, 2005, ISBN 1-59059-239-5.
<http://www.gigamonkeys.com/book/>
- [5] CLIPS: *Tool for Building Expert Systems*. 2013.
<http://clipsrules.sourceforge.net/OnlineDocs.html>
- [6] LispWorks Ltd.: *Common Lisp HyperSpec*. 2005.
<http://www.lispworks.com/documentation/HyperSpec/Front/>
- [7] Expert system — Wikipedia, The Free Encyklopedia. 2013.
http://en.wikipedia.org/wiki/Expert_system
- [8] Rete algorithm — Wikipedia, The Free Encyklopedia. 2013.
http://en.wikipedia.org/wiki/Rete_algorithm