

Reprezentace znalostí a základní sémantické struktury

Aleš Horák

E-mail: hales@fi.muni.cz

http://nlp.fi.muni.cz/poc_lingv/

Obsah:

- Reprezentace znalostí
- Sémantické datové struktury
- Slovníky a specializované lexikony

Obsah

- 1 Reprezentace znalostí
 - Reprezentace znalostí
- 2 Sémantické datové struktury
 - Logika a sémantické datové struktury
 - Sémantické sítě
 - Rámce
 - Pravidlové systémy
 - Nejistota a pravděpodobnost
- 3 Slovníky a specializované lexikony
 - DEB – platforma pro vývoj slovníků
 - České valenční lexikony
 - Valeční lexikon VerbaLex

otázka:

Jak zapíšeme znalosti o problému/doméně?

Když je zapíšeme, můžeme z nich mechanicky odvodit nová fakta?

- **reprezentace znalostí** (*knowledge representation*) – hledá způsob vyjádření znalostí počítačově zpracovatelnou formou (za účelem odvozování)
- **vyvozování znalostí** (*reasoning*) – zpracovává znalosti uložené v **bázi znalostí** (*knowledge base, KB*) a provádí **odvození** (*inference*) nových závěrů:
 - odpovědi na dotazy
 - zjištění faktů, které vyplývají z faktů a pravidel v KB
 - odvodit akci, která vyplývá z dodaných znalostí, ...

otázka:

Jak zapíšeme znalosti o problému/doméně?

Když je zapíšeme, můžeme z nich mechanicky odvodit nová fakta?

- **reprezentace znalostí** (*knowledge representation*) – hledá způsob vyjádření znalostí počítačově zpracovatelnou formou (za účelem odvozování)
- **vyvozování znalostí** (*reasoning*) – zpracovává znalosti uložené v **bázi znalostí** (*knowledge base, KB*) a provádí **odvození** (*inference*) nových závěrů:
 - odpovědi na dotazy
 - zjištění faktů, které vyplývají z faktů a pravidel v KB
 - odvodit akci, která vyplývá z dodaných znalostí, ...

otázka:

Jak zapíšeme znalosti o problému/doméně?

Když je zapíšeme, můžeme z nich mechanicky odvodit nová fakta?

- **reprezentace znalostí** (*knowledge representation*) – hledá způsob vyjádření znalostí počítačově zpracovatelnou formou (za účelem odvozování)
- **vyvozování znalostí** (*reasoning*) – zpracovává znalosti uložené v **bázi znalostí** (*knowledge base, KB*) a provádí **odvození** (*inference*) nových závěrů:
 - odpovědi na dotazy
 - zjištění faktů, které vyplývají z faktů a pravidel v KB
 - odvodit akci, která vyplývá z dodaných znalostí, ...

Reprezentace znalostí

proč je potřeba speciální **reprezentace znalostí**?

vnímání lidí × *vnímání počítačů*

- člověk

- když dostane novou věc (třeba pomeranč) – **prozkoumá** a **zapamatuje** si ho (a třeba sni)
- během tohoto procesu člověk zjistí a uloží všechny základní vlastnosti
- později, když se **zmíní** daná věc, vyhledají se a připomenou uložené informace

- počítač

- musí se spolehnout na informace od lidí
- jednodušší informace – přímé *programování*
- složité informace – zadané v **symbolickém jazyce**

Reprezentace znalostí

proč je potřeba speciální **reprezentace znalostí**?

vnímání lidí × *vnímání počítačů*

- **člověk**

- když dostane novou věc (třeba pomeranč) – **prozkoumá** a **zapamatuje** si ho (a třeba sni)
- během tohoto procesu člověk zjistí a uloží všechny základní vlastnosti
- později, když se **zmíní** daná věc, vyhledají se a připomenou uložené informace

- **počítač**

- musí se spolehnout na informace od lidí
- jednodušší informace – přímé *programování*
- složité informace – zadané v **symbolickém jazyce**

Reprezentace znalostí

proč je potřeba speciální **reprezentace znalostí**?

vnímání lidí × *vnímání počítačů*

- **člověk**

- když dostane novou věc (třeba pomeranč) – **prozkoumá** a **zapamatuje** si ho (a třeba sni)
- během tohoto procesu člověk zjistí a uloží všechny základní vlastnosti
- později, když se **zmíní** daná věc, vyhledají se a připomenou uložené informace

- **počítač**

- musí se spolehnout na informace od lidí
- jednodušší informace – přímé *programování*
- složité informace – zadané v **symbolickém jazyce**

Volba reprezentace znalostí

kteřá **reprezentace znalostí** je **nejlepší**?

Pro řešení skutečně obtížných problémů musíme použít několik různých reprezentací. Důvodem pro to je to, že každý typ datových struktur má své přínosy i nedostatky a žádná z nich není adekvátní pro všechny různé funkce používané v tom, čemu říkáme "zdravý rozum" (common sense).

– Marvin Minsky

Volba reprezentace znalostí

která **reprezentace znalostí** je **nejlepší**?

Pro řešení skutečně obtížných problémů musíme použít několik různých reprezentací. Důvodem pro to je to, že každý typ datových struktur má své přínosy i nedostatky a žádná z nich není adekvátní pro všechny různé funkce používané v tom, čemu říkáme “zdravý rozum” (common sense).

– Marvin Minsky

Obsah

- 1 Re prezentace znalostí
 - Re prezentace znalostí
- 2 Sémantické datové struktury
 - Logika a sémantické datové struktury
 - Sémantické sítě
 - Rámce
 - Pravidlové systémy
 - Nejistota a pravděpodobnost
- 3 Slovníky a specializované lexikony
 - DEB – platforma pro vývoj slovníků
 - České valenční lexikony
 - Valeční lexikon VerbaLex

Reprezentace znalostí pomocí logiky nebo datových struktur

Logika:

- znalosti uloženy ve formě **logických formulí**
- vyvozování nových znalostí = hledání **důkazu**

Specializované datové struktury:

- sémantické sítě
- rámce
- pravidlové systémy
- struktury pro práci s nejistotou a pravděpodobností

Reprezentace znalostí pomocí logiky nebo datových struktur

Logika:

- znalosti uloženy ve formě **logických formulí**
- vyvozování nových znalostí = hledání **důkazu**

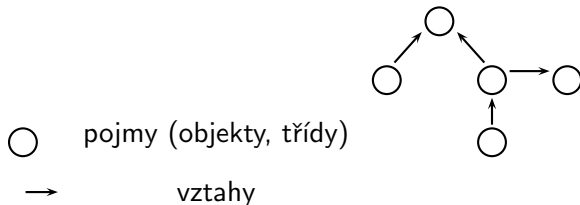
Specializované datové struktury:

- sémantické sítě
- rámce
- pravidlové systémy
- struktury pro práci s nejistotou a pravděpodobností

Sémantické sítě

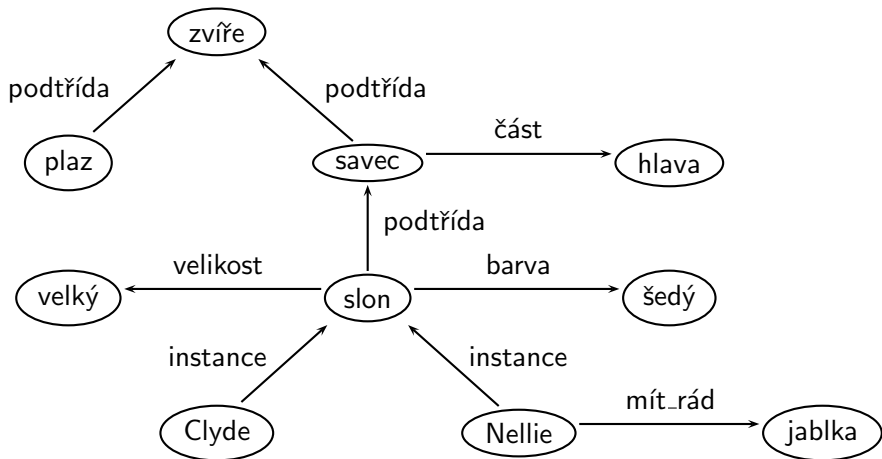
sémantické sítě – reprezentace faktových znalostí (pojmy + vztahy)

- vznikly kolem roku 1960 pro reprezentaci významu anglických slov
- znalosti jsou uloženy ve formě grafu



- nejdůležitější vztahy:
 - **podtřída** (*subclass*) – vztah mezi třídami
 - **instance** – vztah mezi konkrétním objektem a jeho rodičovskou třídou
- jiné vztahy – část (*has-part*), barva, ...

Sémantické sítě – příklad



Dědičnost v sémantických sítích

- pojem sémantické sítě *předchází* OOP
- **dědičnost:**
 - jestliže určitá vlastnost platí pro třídu → platí i pro všechny její podtřídy
 - jestliže určitá vlastnost platí pro třídu → platí i pro všechny prvky této třídy
- určení hodnoty vlastnosti – rekurzivní algoritmus
- potřeba specifikovat i výjimky – mechanismus **vzorů** a **výjimek** (*defaults and exceptions*)
 - vzor – hodnota vlastnosti u třídy nebo podtřídy, platí ta, co je blíže objektu
 - výjimka – u konkrétního objektu, odlišná od vzoru

Dědičnost vztahů část/celek

- “krávy mají 4 nohy.”
 - každá noha je částí krávy
- “Na poli je (konkrétní) kráva.”
 - všechny části krávy jsou taky na poli
- “Ta kráva (na poli) je hnědá (celá).”
 - všechny části té krávy jsou hnědé
- “Ta kráva je šťastná.”
 - všechny části té krávy jsou šťastné – *neplatí*
- lekce: některé vlastnosti jsou děděny částmi, některé nejsou explicitně se to vyjadřuje pomocí pravidel jako

$$part-of(x, y) \wedge location(y, z) \Rightarrow location(x, z)$$

Dědičnost vztahů část/celek

- “krávy mají 4 nohy.”
 - každá noha je částí krávy
- “Na poli je (konkrétní) kráva.”
 - všechny části krávy jsou taky na poli
- “Ta kráva (na poli) je hnědá (celá).”
 - všechny části té krávy jsou hnědé
- “Ta kráva je šťastná.”
 - všechny části té krávy jsou šťastné – neplatí
- lekce: některé vlastnosti jsou děděny částmi, některé nejsou explicitně se to vyjadřuje pomocí pravidel jako

$$part-of(x, y) \wedge location(y, z) \Rightarrow location(x, z)$$

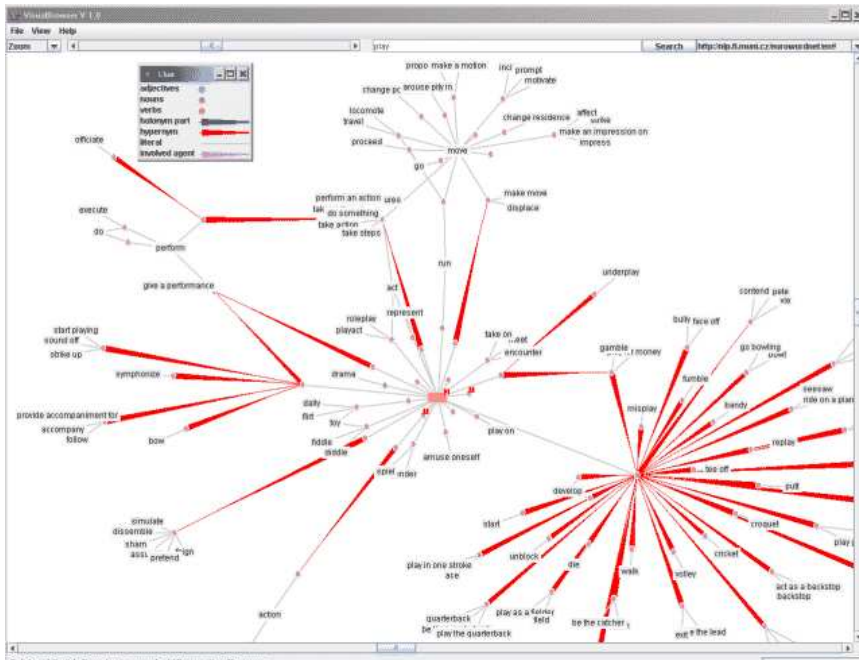
Vzory a výjimky – příklad

- “všichni ptáci mají křídla.”
- “všichni ptáci umí létat.”
- “ptáci se zlomenými křídly jsou ptáci, ale neumí létat.”
- “tučnáci jsou ptáci, ale neumí létat.”
- “kouzelní tučňáci jsou tučňáci, kteří umí létat.”
- kdo umí létat:
 - “Penelope je pták.” \Rightarrow “Penelope **umí** létat”
 - “Penelope je tučnák.” \Rightarrow “Penelope **neumí** létat”
 - “Penelope je kouzelný tučnák.” \Rightarrow “Penelope **umí** létat”
- všimněte si, že víra v hodnotu vlastnosti objektu se může měnit s příchodem nových informací o klasifikaci objektu

Aplikace sémantických sítí

(Princeton) [WordNet](http://wordnet.princeton.edu/) – <http://wordnet.princeton.edu/>

- sématická síť 100.000 (anglických) pojmů, zachycuje:
 - synonyma, antonyma (významově stejná/opačná)
 - hyperonyma, hyponyma (podtřídy)
 - odvozenost a další jazykové vztahy
- tvoří se [národní wordnety](#) (navázané na anglický WN)
český wordnet – cca 30.000 pojmů
- nástroj na editaci národních wordnetů – DEBVisDic, vyvinutý na FI MU
- VisualBrowser –
<http://nlp.fi.muni.cz/projekty/visualbrowser/>
nástroj na vizualizaci (sémantických) sítí, vznikl jako DP na FI MU



The screenshot displays the DEBVisDic application interface. At the top, a menu bar includes 'User', 'Settings', 'Tools', 'Windows', and 'Help'. The 'Tools' menu is open, showing options: 'Dictionary - SSJČ', 'dictionary - SSČ', 'Morph. analyzer ajka', and 'Google'. Below the menu, several wordnet windows are visible:

- English Wordnet:** Search for 'dog'. Results include: [n] andiron:1, firedog:1, dog:7, dog:8; [n] frump:1, dog:2; [n] cad:1, bouncer:1, blackguard:1, dog:4, hound:1; [n] dog:1, domestic dog:1, Canis familiaris:1; [n] frank:2.
- Greek Wordnet:** Search for 'σοῦ ἰκὸ'. Results include: [n] περισσοῦ ἰκὸ:1; [n] περισσοῦ ἰκὸ:0.
- Czech Wordnet:** Search for 'pes'. Results include: [n] zakopaný pes:1; [n] policejní pes:1; [n] hlíďač:4, hlíďací pes:1; [n] pes:1; [n] slepecký pes:1, vodící pes:1.
- Russian Wordnet:** Search for 'журнал'. Results include: [n] журнал:1.

The Czech Wordnet window is selected, showing a detailed view of the entry for 'pes'. It includes a tree view, a list of synonyms, and XML-like markup for semantic information. A context menu is open over the Czech Wordnet window, with the following options:

- Show in Czech Wordnet
- Take key from Czech Wordnet
- AutoLookUp in
- Copy entry to Czech Wordnet
- Import IDs from file

The detailed view of the Czech entry for 'pes' shows the following XML-like structure:

```

<SYNONYM>
  <LITERAL Inote="" sense="1">pes</LITERAL>
</SYNONYM>
<ILR type="hypernym">ENG20-020005
<ILR type="holo_member">ENG20-020006
<ILR type="holo_member">ENG20-075
<STAMP>xcapek1 2003/06/25</STAMP>
<BCS>3</BCS>
<RILR type="hypernym">ENG20-02002
<RILR type="hypernym">ENG20-02027
  
```

Number of entries: 12

Rámce

Rámce (*frames*):

- varianta sémantických sítí
- velice populární pro reprezentaci znalostí v expertních systémech
- všechny informace relevantní pro daný pojem se ukládají do univerzálních struktur – **rámců**
- stejně jako sémantické sítě, rámce podporují dědičnost
- OO programovací jazyky vycházejí z teorie rámců

Rámce – příklad

rámec obsahuje **objekty**, **sloty** a **hodnoty slotů**
příklady rámců:

savec:

<i>podtřída:</i>	zvíře
<i>část:</i>	hlava
* <i>má_kožich:</i>	ano

slon:

<i>podtřída:</i>	savec
* <i>barva:</i>	šedá
* <i>velikost:</i>	velký

Nellie:

<i>instance:</i>	slon
<i>mít_rád:</i>	jablka

'*' označuje **vzorové hodnoty**, které mohou měnit hodnoty u podtříd a instancí

Sémantické sítě × rámce

sémantické sítě	rámce
uzly	objekty
spoje	sloty
uzel na druhém konci spoje	hodnota slotu

deskripční logika – logický systém, který manipuluje přímo s rámci

Pravidlové systémy

- snaha zachytit **produkčními pravidly** znalosti, které má expert
- obecná forma pravidel

IF *podmínka*
THEN *akce*

- podmínky – booleovské výrazy, dotazy na hodnoty **proměnných**
- akce – nastavení hodnot proměnných, příznaků, ...
- důležité vlastnosti:
 - znalosti mohou být strukturovány do modulů
 - systém může být snadno rozšířen přidáním nových pravidel beze změny zbytku systému

Pravidlové systémy

- snaha zachytit **produkčními pravidly** znalosti, které má expert
- obecná forma pravidel

IF *podmínka*
THEN *akce*

- podmínky – booleovské výrazy, dotazy na hodnoty **proměnných**
- akce – nastavení hodnot proměnných, příznaků, ...
- důležité vlastnosti:
 - znalosti mohou být strukturovány do modulů
 - systém může být snadno rozšířen přidáním nových pravidel beze změny zbytku systému

Metody pro práci s nejistotou

definujme akci A_t jako “**Vyrazit na letiště t hodin před odletem letadla.**”
jak najít odpověď na otázku “*Dostanu se akcí A_t na letiště včas?*”

- defaultní/nemonotónní logika

Předpokládejme, že nepíchnu cestou kolo.

Předpokládejme, že A_5 bude OK, pokud se nenajde protipříklad.

- pravidla s faktory nejistoty

$A_5 \mapsto_{0.3}$ dostat se na letiště včas.

zalévání $\mapsto_{0.99}$ mokrý trávník

mokrý trávník $\mapsto_{0.7}$ déšť

- pravděpodobnost

Vzhledem k dostupným informacím, A_3 mě tam dostane včas s pravděpodobností 0.05.

Použití **náhodných proměnných** a pravidel pro výpočet pravděpodobnosti logicky souvisejících událostí (podmíněná pravděpodobnost, bayesovské pravidlo, ...)

Metody pro práci s nejistotou

definujme akci A_t jako “**Vyrazit na letiště t hodin před odletem letadla.**”
jak najít odpověď na otázku “*Dostanu se akcí A_t na letiště včas?*”

- defaultní/nemonotónní logika

Předpokládejme, že nepíchnu cestou kolo.

Předpokládejme, že A_5 bude OK, pokud se nenajde protipříklad.

- pravidla s faktory nejistoty

$A_5 \mapsto_{0.3}$ dostat se na letiště včas.

zalévání $\mapsto_{0.99}$ mokrý trávník

mokrý trávník $\mapsto_{0.7}$ déšť

- pravděpodobnost

Vzhledem k dostupným informacím, A_3 mě tam dostane včas s pravděpodobností 0.05.

Použití **náhodných proměnných** a pravidel pro výpočet pravděpodobnosti logicky souvisejících událostí (podmíněná pravděpodobnost, bayesovské pravidlo, ...)

Metody pro práci s nejistotou

definujme akci A_t jako “**Vyrazit na letiště t hodin před odletem letadla.**”
jak najít odpověď na otázku “*Dostanu se akcí A_t na letiště včas?*”

- **defaultní/nemonotónní logika**

Předpokládejme, že nepíchnu cestou kolo.

Předpokládejme, že A_5 bude OK, pokud se nenajde protipříklad.

- **pravidla s faktory nejistoty**

$A_5 \mapsto_{0.3}$ dostat se na letiště včas.

zalévání $\mapsto_{0.99}$ mokrý trávník

mokrý trávník $\mapsto_{0.7}$ déšť

- **pravděpodobnost**

Vzhledem k dostupným informacím, A_3 mě tam dostane včas s pravděpodobností 0.05.

Použití **náhodných proměnných** a pravidel pro výpočet pravděpodobnosti logicky souvisejících událostí (podmíněná pravděpodobnost, bayesovské pravidlo, ...)

Obsah

- 1 Re prezentace znalostí
 - Re prezentace znalostí
- 2 Sémantické datové struktury
 - Logika a sémantické datové struktury
 - Sémantické sítě
 - Rámce
 - Pravidlové systémy
 - Nejistota a pravděpodobnost
- 3 Slovníky a specializované lexikony
 - DEB – platforma pro vývoj slovníků
 - České valenční lexikony
 - Valeční lexikon VerbaLex

Slovníky a specializované lexikony

Slovníky typicky obsahují:

- specifikace **formy**:
 - grafická podoba – alternativy, dělení, velká počáteční písmena
 - zvuková podoba – výslovnost a její alternativy, slabiky, přízvuk, výška
- **gramatické** (morfo-syntaktické) **informace** – slovní druh a příslušné gramatické kategorie, morfologický vzor?
- specifikace **významu** – hierarchie

slovník uvádí významy listémů, **encyklopedie** informace o jejich denotátech

specializované lexikony a encyklopedie (znalost odborníků a rozdílné předpoklady a pohledy)

Slovníky a specializované lexikony

Slovníky typicky obsahují:

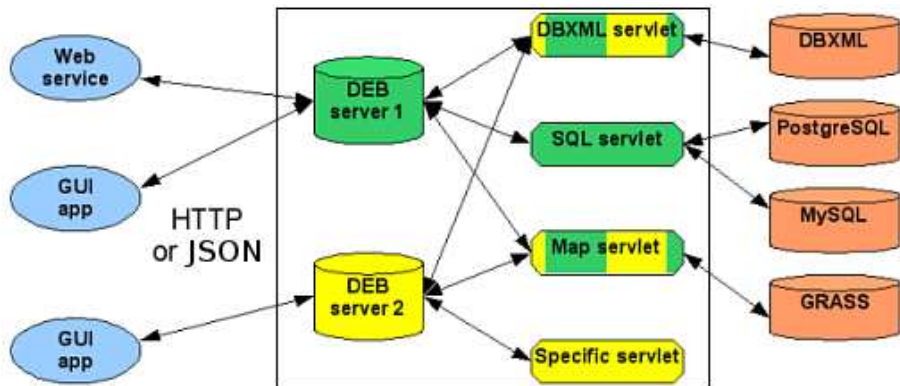
- specifikace **formy**:
 - grafická podoba – alternativy, dělení, velká počáteční písmena
 - zvuková podoba – výslovnost a její alternativy, slabiky, přízvuk, výška
- **gramatické** (morfo-syntaktické) **informace** – slovní druh a příslušné gramatické kategorie, morfologický vzor?
- specifikace **významu** – hierarchie

slovník uvádí významy listémů, **encyklopedie** informace o jejich denotátech

specializované lexikony a encyklopedie (znalost odborníků a rozdílné předpoklady a pohledy)

DEB – platforma pro vývoj slovníků

- platforma pro vývoj systémů na psaní slovníků
 - <http://deb.fi.muni.cz/>
 - pracuje s hesly ve formě XML struktury
- striktní klient-server architektura
- server
 - specializované moduly – *servlety*
 - databázové úložiště
- klient
 - jen jednoduchá funkcionalita
 - GUI i web rozhraní – postavený na *Mozilla Engine*

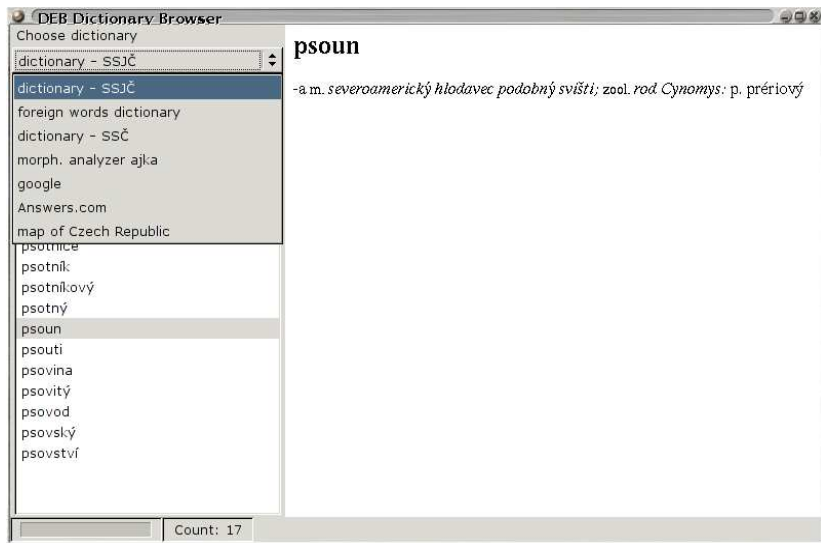


DEB používá komunikaci typu AJAX

DEBDict – příklad DEB klienta

jednoduchý klient původně určený pro demo základních funkcí

- dostupný jako instalovatelné rozšíření Firefoxu i jako vzdálená webová služba
- vícejazyčné uživatelské rozhraní (angličtina, čeština, další lze snadno doplnit)
- dotazy do několika XML slovníků s různou strukturou, výsledky jsou zpracovány XSLT transformací
- napojení na český morfologický analyzátor
- napojení na externí webové stránky (Google, Answers.com, Wikipedia)
- napojení na geografický informační systém – zobrazení geografických odkazů přímo na mapě



České valenční lexikony

specializované lexikony slovesných valencí:

- syntaktické valenční rámce **Brief** (FI MU, od 1997) cca 15,000 sloves:

```
lámat <v>hPTc4,hPTc4-hTc7,hPc3-hTc4
```

- valenční rámce v **českém wordnetu** (FI MU 2000), cca 3,000 slovesných literálů (sloveso+význam):

```
synset: lámat:3, dobývat:1, těžít:2
```

```
valence: kdo1*AG(person:1)=co4*SUBS(substance:1)
```

```
valence: co1*AG(institution:1)=co4*SUBS(substance:1)
```

- pražský lexikon **Vallex 1.0**, na začátku roku 2005 cca 1,000 sloves (teď snad až 4,000):

```
~ impf: lámat
```

```
+ ACT(1;obl) PAT(4;obl)
```

České valenční lexikony

specializované lexikony slovesných valencí:

- syntaktické valenční rámce **Brief** (FI MU, od 1997) cca 15,000 sloves:

lámat <v>hPTc4,hPTc4-hTc7,hPc3-hTc4

- valenční rámce v **českém wordnetu** (FI MU 2000), cca 3,000 slovesných literálů (sloveso+význam):

synset: lámat:3, dobývat:1, těžít:2

valence: kdo1*AG(person:1)=co4*SUBS(substance:1)

valence: co1*AG(institution:1)=co4*SUBS(substance:1)

- pražský lexikon **Vallex 1.0**, na začátku roku 2005 cca 1,000 sloves (teď snad až 4,000):

~ impf: lámat

+ ACT(1;obl) PAT(4;obl)

České valenční lexikony

specializované lexikony slovesných valencí:

- syntaktické valenční rámce **Brief** (FI MU, od 1997) cca 15,000 sloves:

lámat <v>hPTc4,hPTc4-hTc7,hPc3-hTc4

- valenční rámce v **českém wordnetu** (FI MU 2000), cca 3,000 slovesných literálů (sloveso+význam):

synset: lámat:3, dobývat:1, těžít:2

valence: kdo1*AG(person:1)=co4*SUBS(substance:1)

valence: co1*AG(institution:1)=co4*SUBS(substance:1)

- pražský lexikon **Vallex 1.0**, na začátku roku 2005 cca 1,000 sloves (teď snad až 4,000):

~ impf: lámat

+ ACT(1;obl) PAT(4;obl)

Valeční lexikon VerbaLex

- vznikl na začátku roku 2005, využívá všech dostupných zdrojů
- edituje se v jednoduchém textovém formátu, který se pro další zpracování převádí do XML
- vlastnosti:
 - dvouúrovňové sémantické role
 - odkazy na hypero/hyponymickou hierarchii v českém wordnetu
 - odlišení životnosti a neživotnosti větných členů
 - implicitní pozice slovesa
 - valenční rámce se odkazují na číslované významy sloves
- exporthy z XML do HTML pro prohlížení a PDF pro tisk

VerbaLex v HTML

[alphabet](#)
[wn link](#)
[verb class](#)
[functors](#)
[forms](#)
[aspect](#)
[complexity](#)
[miscel.](#)

[search](#)
[home](#)
[help ?](#)

- A (18)
- B (101)
- C (11)
- Č (18)
- D (457)
- E (6)
- F (11)
- H (68)
- CH (34)
- I (8)
- J (14)
- K (70)
- L (24)
- M (64)
- N (249)
- O (315)
- P (572)
- R (84)
- Ř (42)
- S (217)
- Š (33)
- T (25)
- U (160)
- V (469)
- Z (368)
- Ž (29)

- tahat₁
- tahat₂
- táhnout₃
- táhnout₆
- táhnout se₁
- téci₁
- téci₁
- téct₁
- téct₁
- teoretizovat₁
- testovat₁
- **těžít₂**
- těžít₃
- tisknout₂
- tlačit₂
- tlačit₂
- tlačit₃
- tlouct se₁
- toulat se

dobývat¹ / **těžít²** / **lámat³**
1 dobývat₁ / těžít₂ / lámat₃ =
 -frame: **AG**<person:1>_{kdo1} **VERB**^{obl} **SUBS**<substance:1>_{obl co4}
 -example: **ned**: *lámal v dolech kámen*
 -synonym:
 -use: prim
2 dobývat₁ / těžít₂ / lámat₃ =
 -frame: **AG**<institution:1>_{co1} **VERB**^{obl} **SUBS**<substance:1>_{obl co4}
 -example: **ned**: *tato společnost těží mramor*
 -synonym:
 -use: prim

Využití valencí v sémantické analýze

reprezentace **slovesného rámce**:

1. syntaktické rysy:

dávat něco_{neživ.NP, 4.pád, bez předložky}

někomu_{živ.NP, 3.pád, bez předložky}

2. sémantické rysy:

dávat Patiens Addressee

3. funkce významu:

dávat $x y \dots (o(o\pi)(o\pi))_{\omega}$, slovesný objekt

dávat / $(o(o\pi)(o\pi))_{\omega ll} \quad x \dots l \quad y \dots l : s_{wt}y, s \dots (ol)_{\tau w}$

překlad z valenčního výrazu do funkce významu:

typ argumentu = typ {

- jmenné skupiny
- příslovečné fráze
- vedlejší věty
- infinitivu

Využití valencí v sémantické analýze

reprezentace **slovesného rámce**:

1. syntaktické rysy:

dávat něco_{neživ.NP, 4.pád, bez předložky}

někomu_{živ.NP, 3.pád, bez předložky}

2. sémantické rysy:

dávat Patiens Addressee

3. funkce významu:

dávat $x y \dots (o(o\pi)(o\pi))_{\omega}$, slovesný objekt

dávat / $(o(o\pi)(o\pi))_{\omega ll} \quad x \dots l \quad y \dots l : s_{wt}y, s \dots (ol)_{\tau w}$

překlad z valenčního výrazu do funkce významu:

typ argumentu = typ {

- jmenné skupiny
- příslovečné fráze
- vedlejší věty
- infinitivu

Využití valencí v sémantické analýze

reprezentace **slovesného rámce**:

1. syntaktické rysy:

dávat něco_{neživ.NP, 4.pád, bez předložky}

někomu_{živ.NP, 3.pád, bez předložky}

2. sémantické rysy:

dávat Patiens Addressee

3. funkce významu:

dávat $x y \dots (o(o\pi)(o\pi))_{\omega}$, slovesný objekt

dávat / $(o(o\pi)(o\pi))_{\omega ll} \quad x \dots l \quad y \dots l : s_{wt}y, s \dots (ol)_{\tau\omega}$

překlad z valenčního výrazu do funkce významu:

typ argumentu = typ

- jmenné skupiny
- příslovečné fráze
- vedlejší věty
- infinitivu

Využití valencí v sémantické analýze

reprezentace **slovesného rámce**:

1. syntaktické rysy:

dávat něco_{neživ.NP, 4.pád, bez předložky}

někomu_{živ.NP, 3.pád, bez předložky}

2. sémantické rysy:

dávat Patiens Addressee

3. funkce významu:

dávat $x y \dots (o(o\pi)(o\pi))_{\omega}$, slovesný objekt

dávat / $(o(o\pi)(o\pi))_{\omega ll} \quad x \dots l \quad y \dots l : s_{wt}y, s \dots (ol)_{\tau\omega}$

příklad z valenčního výrazu do funkce významu:

typ argumentu = typ {

- jmenné skupiny
- příslovecné fráze
- vedlejší věty
- infinitivu