

# Reprezentace a vyvozování znalostí

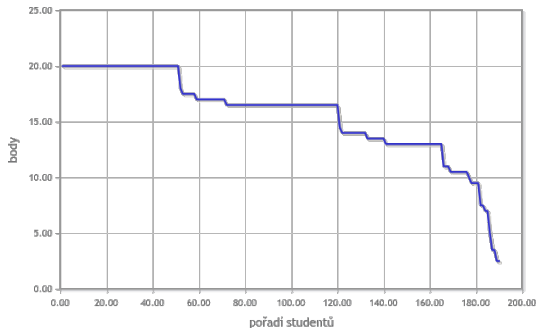
Aleš Horák

E-mail: [hales@fi.muni.cz](mailto:hales@fi.muni.cz)  
<http://nlp.fi.muni.cz/uui/>

Obsah:

- Statistické výsledky průběžné písemky
- Reprezentace a vyvozování znalostí
- Extralogické informace
- Pravidlové systémy
- Nejistota a pravděpodobnost

# Statistické výsledky průběžné písemky



průběžná písemka PB016  
190 studentů

Body	Počet studentů
20	51
18–17	20
16.5	49
14.5–13	45
11–10	12
9.5–2.5	13

Medián: 16.5 bodů

# Reprezentace a vyvozování 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**?

# 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 sní)
- během tohoto procesu člověk **zjistí** a **uloží si** všechny základní vlastnosti předmětu
- 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 nebo rozsáhlé informace – zadané v **symbolickém jazyce**

# Volba reprezentace znalostí

kteřá reprezentace znalostí je **nejlepší**?

# Volba reprezentace znalostí

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

*Pro řešení skutečně obtížných problémů musíme používat několik různých reprezentací. Každý konkrétní typ datových struktur má totiž své klady a zápory a žádný se sám o sobě nezdá adekvátní pro všechny funkce zahrnuté v tom, čemu říkáme "selský rozum" (common sense).*

*– Marvin Minsky, spoluzakladatel MIT AI lab*

# Limity reprezentace znalostí

***Všechny modely se mýlí, některé jsou ale užitečné.***

*(All models are wrong, but some are useful)*

– *George Box, významný britský statistik*



# Limity reprezentace znalostí

***Všechny modely se mýlí, některé jsou ale užitečné.***

*(All models are wrong, but some are useful)*

– *George Box, významný britský statistik*

Kompletní reprezentace  
znalostí má řešit:

- kategorie
- míry a hodnoty
- složené objekty
- čas, prostor a změny
- události a procesy
- fyzické objekty
- látky/substance
- mentální objekty a postoje
- ...

Často je nutné se kvůli **efektivitě modelu** vzdát **přesnosti** nebo **úplnosti**.

# Reifikace – abstrakce jako objekt

**reifikace** (zvěcnění) – zjednodušení (logické) analýzy  
abstrakci/objekt vyššího řádu modelujeme jako **objekt**

**vlastnost** objektu  $\approx$  *predikát pomeranč*

“držím pomeranč”  $\approx$  držet(*Já*, *x*)  $\wedge$  pomeranč(*x*)

“mám rád chuť pomeranče”  $\approx$  mít\_rád(*Já*, *x*)  $\wedge$  *x* = chuť(?pomeranč?)

... v PL1 nelze

# Reifikace – abstrakce jako objekt

**reifikace** (zvěcnění) – zjednodušení (logické) analýzy  
abstrakci/objekt vyššího řádu modelujeme jako **objekt**

**vlastnost** objektu  $\approx$  **predikát pomeranč**

“držím pomeranč”  $\approx$   $\text{držet}(Já, x) \wedge \text{pomeranč}(x)$

“mám rád chuť pomeranče”  $\approx$   $\text{mít\_rád}(Já, x) \wedge x = \text{chuť}(\text{?pomeranč?})$

... v PL1 nelze

s **reifikací** predikátu **pomeranč** (konstanta):

“mám rád chuť pomeranče”  $\approx$   $\text{mít\_rád}(Já, x) \wedge x = \text{chuť}(\text{pomeranč})$

“držím pomeranč”  $\approx$   $\text{držet}(Já, x) \wedge \text{is\_instance}(x, \text{pomeranč})$

# Reifikace – abstrakce jako objekt

**reifikace** (zvěcnění) – zjednodušení (logické) analýzy  
**abstrakci**/objekt vyššího řádu modelujeme jako **objekt**

**vlastnost** objektu  $\approx$  **predikát pomeranč**

“držím pomeranč”  $\approx$   $\text{držet}(\text{Já}, x) \wedge \text{pomeranč}(x)$

“mám rád chuť pomeranče”  $\approx$   $\text{mít\_rád}(\text{Já}, x) \wedge x = \text{chuť}(\text{?pomeranč?})$

... v PL1 nelze

s **reifikací** predikátu **pomeranč** (konstanta):

“mám rád chuť pomeranče”  $\approx$   $\text{mít\_rád}(\text{Já}, x) \wedge x = \text{chuť}(\text{pomeranč})$

“držím pomeranč”  $\approx$   $\text{držet}(\text{Já}, x) \wedge \text{is\_instance}(x, \text{pomeranč})$

Z čistě logického hlediska vznikají **neexistující objekty** (spory při tvrzení o *neexistenci*)

**ale** při **korektním** zpracování – stačí **jednodušší formalismus**

# Obsah

- 1 Statistické výsledky průběžné písemky
- 2 Repräsentace a vyvozování znalostí
  - Repräsentace znalostí
- 3 Extralogické informace**
  - Třídy objektů
  - Ontologie
  - Sémantické sítě
  - Rámce
- 4 Pravidlové systémy
  - Pravidlová báze znalostí
  - Expertní systémy
- 5 Nejistota a pravděpodobnost
  - Nejistota
  - Pravděpodobnost
  - Vyvozování z nejistých znalostí

# Extralogické informace – třídy, sémantické sítě, rámce

## co jsme dosud ignorovali:

- objekty reálného světa mají mezi sebou **vztahy**
  - třídy/kategorie, podtřídy × nadtřídy
  - hierarchie vztahů části/celku
  - dědění vlastností v hierarchiích
- stav světa se může **měnit** v čase
  - explicitní reprezentace času
  - nemonotónní uvažování (pravdivost se může měnit v čase)
- ne každá informace je “černobílá”
  - **nejistota**
  - statistika, fuzzy logika

# Třídy objektů

- “Chci si koupit fotbalový míč.”
  - *Chci si koupit FM27341* – špatně
  - *Chci si koupit objekt, který je prvkem třídy fotbalových míčů* – správně
- objekty jsou organizovány do **hierarchie tříd**
  - $FM27341 \in \text{fotbalové\_míče}$
  - $\text{fotbalové\_míče} \subset \text{míče}$
- **fakta** (objekty)  $\times$  **pravidla** (třídy)
  - *Všechny míče jsou kulaté.*
  - *Všechny fotbalové míče mají X cm v průměru.*
  - *FM27341 je červenomodrobílý.*
  - *FM27341 je fotbalový míč.*
  - (Proto: *FM27341 je kulatý a má X cm v průměru.*)

# Ontologie

- **ontologie** ve filozofii – nauka o **existenci** a typech existencí
- **ontologie** v informatice – **formální popis znalostí**, pojmy a vztahy mezi pojmy, hierarchie
- ontologie **obecné** (*upper level*) × **doménové**
- různé dostupné obecné, žádná standardem (zatím):  
(Open)Cyc, SUMO/MILO, Dublin Core, DOLCE, ...  
<http://archivo.dbpedia.org>, <http://www.ontologyportal.org>



# Ontologie

- **ontologie** ve filozofii – nauka o **existenci** a typech existencí
- **ontologie** v informatice – **formální popis znalostí**, pojmy a vztahy mezi pojmy, hierarchie
- ontologie **obecné** (*upper level*) × **doménové**
- různé dostupné obecné, žádná standardem (zatím):  
(Open)Cyc, SUMO/MILO, Dublin Core, DOLCE, ...  
<http://archivo.dbpedia.org>, <http://www.ontologyportal.org>

```
(=>
  (and
    (instance ?KILL Killing)
    (patient ?KILL ?OBJ))
  (exists (?DEATH)
    (and
      (instance ?DEATH Death)
      (experiencer ?DEATH ?OBJ)
      (causes ?KILL ?DEATH))))
```

*Pokud nějaký proces (?KILL) je instancí zabíjení (Killing) a nějaký agent (?OBJ) je předmětem toho procesu*

=>

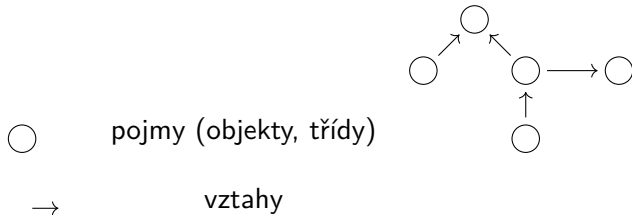
*pak existuje jiný proces (?DEATH) takový, že tento jiný proces je instancí smrti (Death) a agent ?OBJ se účastní tohoto jiného procesu ?DEATH a původní proces ?KILL je příčinou tohoto jiného procesu ?DEATH*

**SLiDo**

# Sémantické sítě

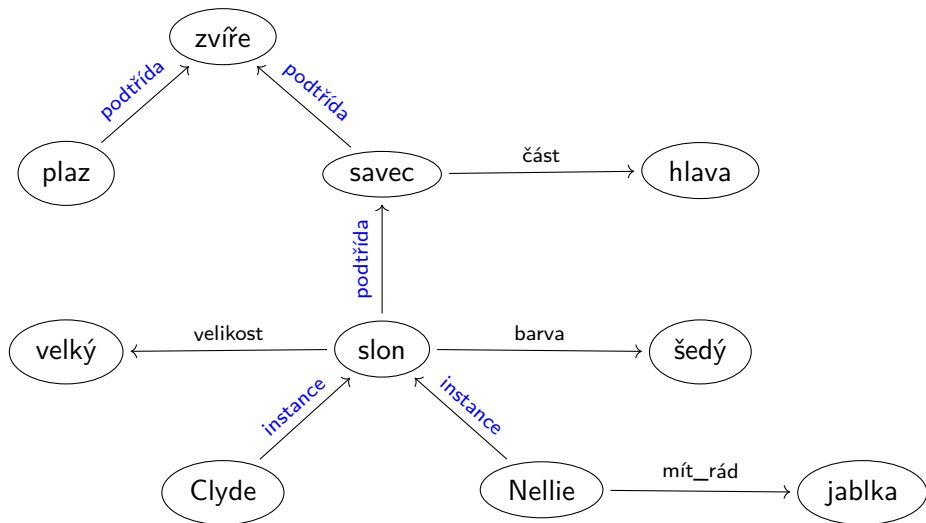
**sémantické sítě** – reprezentace faktových znalostí (pojmy + vztahy)

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



- nejdůležitější vztahy – **taxonomie**:
  - 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

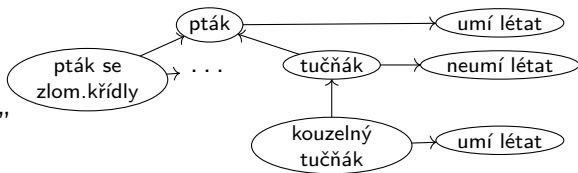
# Dědičnost vztahů část/celek

- “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

# Dědičnost vztahů část/celek

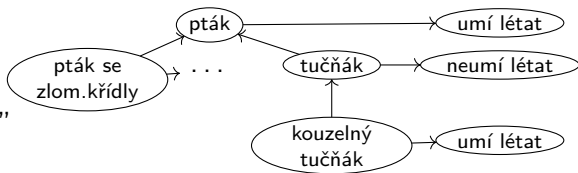
- “Krávy mají 4 nohy.”
  - každá noha je částí krávy
- “Na poli je (konkrétní) kráva.”
  - všechny části krávy jsou taky na poli
- “Ta kráva (na poli) je hnědá (celá).”
  - všechny části té krávy jsou hnědé
- “Ta kráva je šťastná.”
  - ~~všechny části té krávy jsou šťastné~~ – neplatí
- lekce: některé 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.”
- “**Tučňáci** jsou ptáci, ale **neumí** létat.”
- “**Kouzelní tučňáci** jsou tučňáci, kteří **umí** létat.”
- kdo umí létat:
  - “Penelope je pták.”
  - “Penelope je tučňák.”
  - “Penelope je kouzelný tučňák.”

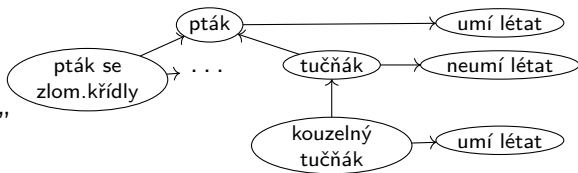
# Vzory a výjimky – příklad



- “Všichni ptáci mají křídla.”
  - “Všichni **ptáci** **umí** létat.”
  - “Ptáci se zlomenými křídly jsou ptáci, ale **neumí** létat.”
  - “**Tučňáci** jsou ptáci, ale **neumí** létat.”
  - “**Kouzelní tučňáci** jsou tučňáci, kteří **umí** létat.”
  - kdo umí létat:
    - “Penelope je pták.”
    - “Penelope je tučňák.”
    - “Penelope je kouzelný tučňák.”
- ⇒ “Penelope **umí** létat.”

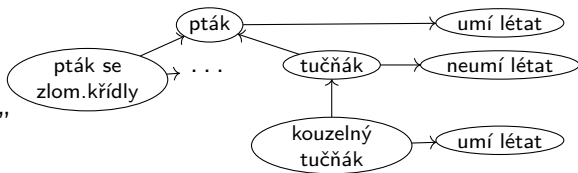


# Vzory a výjimky – příklad



- “Všichni ptáci mají křídla.”
- “Všichni **ptáci** **umí** létat.”
- “Ptáci se zlomenými křídly jsou ptáci, ale **neumí** létat.”
- “**Tučňáci** jsou ptáci, ale **neumí** létat.”
- “**Kouzelní tučňáci** jsou tučňáci, kteří **umí** létat.”
- kdo umí létat:
  - “Penelope je pták.”  $\Rightarrow$  “Penelope **umí** létat.”
  - “Penelope je tučňák.”  $\Rightarrow$  “Penelope **neumí** létat.”
  - “Penelope je kouzelný tučňák.”

# Vzory a výjimky – příklad

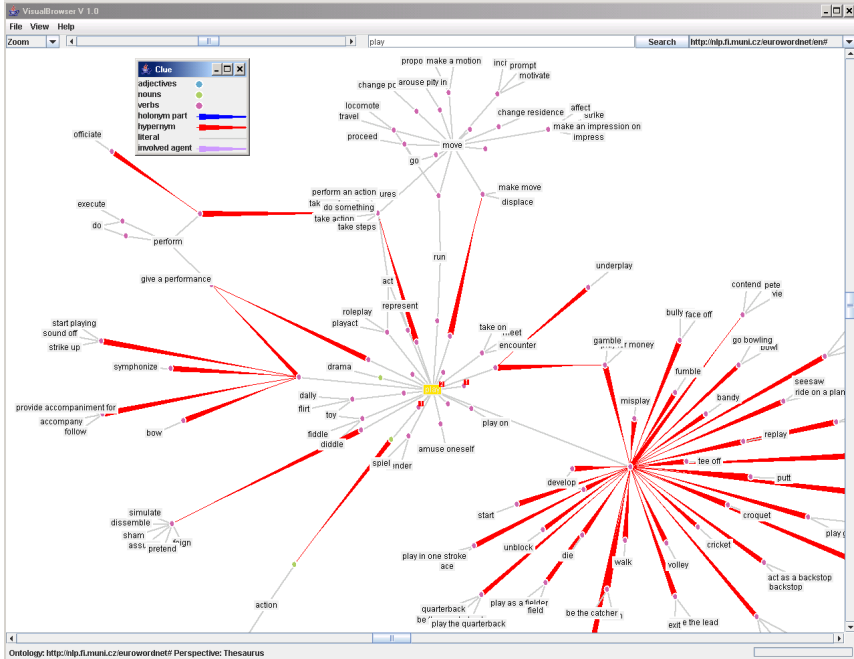


- “Všichni ptáci mají křídla.”
- “Všichni **ptáci** **umí** létat.”
- “Ptáci se zlomenými křídly jsou ptáci, ale **neumí** létat.”
- “**Tučňáci** jsou ptáci, ale **neumí** létat.”
- “**Kouzelní tučňáci** jsou tučňáci, kteří **umí** létat.”
- kdo umí létat:
  - “Penelope je pták.”  $\Rightarrow$  “Penelope **umí** létat.”
  - “Penelope je tučňák.”  $\Rightarrow$  “Penelope **neumí** létat.”
  - “Penelope je kouzelný tučňák.”  $\Rightarrow$  “Penelope **umí** létat.”
- všimněte si, že **znalost** hodnoty vlastnosti objektu **se může měnit** s příchodem nových informací o klasifikaci objektu

# Aplikace sémantických sítí

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

- **sémantická síť** 150.000 (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 tvorbu a **editaci** národních wordnetů – **DEBVisDic/VisDic**,  
vyvinutý na FI MU – <http://deb.fi.muni.cz/>
- **VisualBrowser** –  
<http://nlp.fi.muni.cz/projekty/visualbrowser/>  
nástroj na **vizualizaci** (sémantických) sítí, vznikl jako DP na FI MU



Ontology: <http://nlp.fl.muni.cz/eurowordnet/en/#> Perspective: Thesaurus

The screenshot displays the DEBVisDic application with several overlapping windows and a context menu.

- DEBVisDic (Main Window):** Contains a menu bar with 'User', 'Settings', 'Tools', 'Windows', and 'Help'. The 'Tools' menu is open, showing options: 'Dictionary - SSČ', 'dictionary - SSČ', 'Morph. analyzer ajka', and 'Google'.
- English Wordnet:** Search for 'dog'. Results include:
  - [n] andiron:1, fire-dog:1, dog:7, dog-
  - [n] frump:1, dog:2
  - [n] cad:1, boulder:1, black-guard:1, dog:4, houn-
  - [n] dog:1, domestic dog:1, Canis familiaris:1
  - [n] frank:2
- Greek Wordnet:** Search for 'σοῦ ἰκό'. Results include:
  - [n] περ τοῦ ἰκό:1
  - [n] περ τοῦ ἰκό:0
- Russian Wordnet:** Search for 'журнал'. Results include:
  - [n] журнал:1
- Czech Wordnet:** Search for 'pes'. Results include:
  - [n] zakopaný pes:1
  - [n] policejní pes:1
  - [n] hlídač:4, hlídač pes:1
  - [n] pes:1
  - [n] slepecký pes:1, vodící pes:1
- Context Menu (over Russian Wordnet):**
  - Show in Czech Wordnet
  - Take key from Czech Wordnet
  - AutoLookUp in
  - Copy entry to Czech Wordnet
  - Import IDs from file
- Preview Panel (Czech Wordnet):**

```

- <SYNONYM>
<LITERAL Inote="" sense="1">pes</L
<WORD>pes</WORD>
</SYNONYM>
<ILR type="hypernym">ENG20-020005
<ILR type="holo_member">ENG20-020
<ILR type="holo_member">ENG20-075
<STAMP>xcapek1 2003/06/25</STAMP
<BCS>3</BCS>
<RILR type="hypernym">ENG20-02002
<RII.R tvne="hynernym">ENG20-02027
      
```
- Preview Panel (Russian Wordnet):**

```

POS: n ID: RUS-1234560515
Synonyms: книга:1
Definit
сброш
Usage
библи
Usage
театр?
--> [has_hypernym] печатное издание:1
      
```

# Rámce – příklad

**Rámce** (*frames*) – varianta sémantických sítí představená Marvinem Minskim

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

# Rámce

- **rámec** – univerzální struktura všech informací relevantních pro daný pojem
- velice populární pro reprezentaci znalostí v **expertních systémech**
- stejně jako sémantické sítě, rámce podporují **dědičnost**
- OO programovací jazyky vycházejí z teorie rámců

Sémantické sítě × rámce:

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

**deskripční logika** (*description logic*) – logický systém, který manipuluje přímo s rámci

# Rámce – využití v praxi

příklad využití rámců – ontologie **Friend of A Friend (FOAF)**

- popisuje **osoby**, jejich **činnosti** a **vztahy** k jiným osobám a objektům
- decentralizovaný přístup, <http://xmlns.com/foaf/spec/>
- využívány komunitami (blogovací platformy, MediaWiki, ...)
- základ standardu W3C **WebID 1.0** (2014 draft)



# Rámce – využití v praxi

příklad využití rámců – ontologie **Friend of A Friend (FOAF)**

- popisuje **osoby**, jejich **činnosti** a **vztahy** k jiným osobám a objektům
- decentralizovaný přístup, <http://xmlns.com/foaf/spec/>
- využívány komunitami (blogovací platformy, MediaWiki, ...)
- základ standardu W3C **WebID 1.0** (2014 draft)

```
<link rel="meta" type="application/rdf+xml" title="FOAF"
      href="http://example.com/~vangogh/foaf.rdf" />

<foaf:Person rdf:nodeID="VincentvanGogh"
              xmlns:foaf="http://xmlns.com/foaf/0.1/">
  <foaf:name>Vincent van Gogh</foaf:name>
  <foaf:homepage rdf:resource="https://www.vangoghgallery.com/" />
  <foaf:img rdf:resource="/images/vangogh.jpg" />
  <foaf:knows rdf:resource="#PaulGauguin"/>
  <foaf:made rdf:resource="https://www.vangoghgallery.com/painting/sunflowerindex.html" />
</foaf:Person>
```

# Obsah

- 1 Statistické výsledky průběžné písemky
- 2 Repräsentace a vyvozování znalostí
  - Repräsentace znalostí
- 3 Extralogické informace
  - Třídy objektů
  - Ontologie
  - Sémantické sítě
  - Rámce
- 4 Pravidlové systémy
  - Pravidlová báze znalostí
  - Expertní systémy
- 5 Nejistota a pravděpodobnost
  - Nejistota
  - Pravděpodobnost
  - Vyvozování z nejistých znalostí

# Pravidlová báze znalostí – příklad

pravidla pro **oblékání**:

- pravidlo 1 IF X je **seriózní**  
AND X bydlí ve **městě**  
THEN X by měl nosit **sako**
- pravidlo 2 IF X je **akademik**  
AND X je společensky **aktivní**  
AND X je **seriózní**  
THEN X by měl nosit **sako** a  
**kravatu**
- pravidlo 3 IF X bydlí ve **městě**  
AND X je **akademik**  
THEN X by měl nosit **kravatu**
- pravidlo 4 IF X je **podnikatel**  
AND X je společensky **aktivní**  
AND X je **seriózní**  
THEN X by měl nosit **sako**, ale  
**ne kravatu**

**společenská** pravidla:

- pravidlo 5 IF X je **podnikatel**  
AND X je **ženatý**  
THEN X je společensky  
**aktivní**
- pravidlo 6 IF X je **akademik**  
AND X je **ženatý**  
THEN X je **seriózní**

**profesní** pravidla:

- pravidlo 7 IF X učí na **univerzitě**  
OR X učí na **vysoké škole**  
THEN X je **akademik**
- pravidlo 8 IF X vlastní **firmu**  
OR X je **OSVČ**  
THEN X je **podnikatel**

# Pravidlová báze znalostí

- 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, ...
- důležitá vlastnost – strukturování znalostí do **modulů**
  - systém může být snadno rozšířen přidáním nových pravidel beze změny zbytku systému

# Expertní systémy

## Aplikace pravidlových systémů

- zaměřeny na specifické oblasti – medicínská **diagnóza**, **návrh konfigurace počítače**, **expertíza** pro těžbu nafty, ...
- snaha zachytit **znalosti experta** pomocí pravidel ale znalosti experta zahrnují – postupy, strategie, odhady, ...
- expertní systém musí pracovat s **procedurami**, **nejistými znalostmi**, různými formami vstupu
- vhodné oblasti pro nasazení expertního systému:
  - **diagnóza** – hledání řešení podle symptomů
  - **návrh konfigurace** – složení prvků splňujících podmínky
  - **plánování** – posloupnost akcí splňujících podmínky
  - **monitorování** – porovnání chování s očekávaným chování, reakce na změny
  - **řízení** – ovládání složitého komplexu
  - **předpovědi** – projekce pravděpodobných závěrů z daných skutečností
  - **instruktáž** – inteligentní vyučování a zkoušení studentů

# Expertní systémy v praxi

**CLIPS**, pravidlový jazyk a prostředí pro **expertní systémy**

- původně **NASA** Johnson Space Center – řízení kontroly raket (1985–1996)
- aktivně vyvíjený (v. 6.41 z 21.4.2023, 50 tisíc stažení), tisíce projektů
- pracuje s **objekty**, **fakty** a IF-THEN **pravidly**

# Expertní systémy v praxi

**CLIPS**, pravidlový jazyk a prostředí pro **expertní systémy**

- původně **NASA** Johnson Space Center – řízení kontroly raket (1985–1996)
- aktivně vyvíjený (v. 6.41 z 21.4.2023, 50 tisíc stažení), tisíce projektů
- pracuje s **objekty**, **fakty** a IF-THEN **pravidly**

```
(deftemplate emergency (slot type))      ; objekt
(deftemplate response (slot action))    ; objekt
(defrule fire-emergency (emergency (type fire))      ; pravidlo
 => (assert (response (action activate-sprinkler))))

(reset)
(assert (emergency (type fire)))          ; fakt
(run)
(facts)
  f-0   (initial-fact)
  f-1   (emergency (type fire))
  f-2   (response (action activate-sprinkler))
For a total of 3 facts.
```

# Obsah

- 1 Statistické výsledky průběžné písemky
- 2 Repräsentace a vyvozování znalostí
  - Repräsentace znalostí
- 3 Extralogické informace
  - Třídy objektů
  - Ontologie
  - Sémantické sítě
  - Rámce
- 4 Pravidlové systémy
  - Pravidlová báze znalostí
  - Expertní systémy
- 5 **Nejistota a pravděpodobnost**
  - **Nejistota**
  - **Pravděpodobnost**
  - **Vyvozování z nejistých znalostí**



# Nejistota

definujme akci  $A_t$  jako “Vyzít na letiště  $t$  hodin před odletem letadla.”  
jak najít odpověď na otázku “Dostanu se akcí  $A_t$  na letiště včas k odletu letadla?”

problémy:

1. částečná **pozorovatelnost** (stav vozovky, záměry ostatních řidičů, ...)
2. **nejistota** výsledků akcí (píchnutí kola, ...)
3. obrovská **složitost** modelování a předpovědi dopravní situace

čistě **logický přístup** tedy:

- riskuje **chybu** – “ $A_5$  mě na letiště dostane včas.”
- vede k závěrům, které jsou **příliš slabé** pro rozhodování: “ $A_5$  mě tam dostane včas, pokud nebude na dálnici nehoda a pokud nebude přšet a jestli nepíchnu kolo a jestli nebude fronta na odbavovacích přepážkách a jestli nebudou problémy při kontrole zavazadel ...”

# Metody pro práci s nejistotou

- defaultní/nemonotónní logika

Předpokládejme, že nepíchnu cestou kolo.

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

- logická pravidla s faktory nejistoty (problém příčiny a následku)

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

zalévání  $\mapsto_{0.99}$  mokrý trávník

mokrý trávník  $\mapsto_{0.7}$  déšť

- pravděpodobnost (míra předpokladu, že hodnota bude *true*)

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

poznámka: fuzzy logika se zabývá **mírou pravdivosti**, NE **pravděpodobností**

# Pravděpodobnost

Pravděpodobnost **sumarizuje** nejistotu pocházející z

- **lenosti** – nepodařilo se vypočítat všechny výjimky, podmínky, ...
- **neznalosti** – nedostatek relevantních údajů, počátečních podmínek, ...

**subjektivní** × **Bayesovská** pravděpodobnost:

- pravděpodobnostní vztah mezi tvrzením a jeho pravdivosti vzhledem k podmínkám:

$$P(A_4 | \text{žádné hlášené nehody}) = 0.5$$

nejedná se o vyjádření **pravděpodobnostní tendence** (ale může se získat ze znalostí podobných případů v minulosti)

- pravděpodobnost tvrzení se může měnit s novými (vstupními) podmínkami:

$$P(A_4 | \text{žádné hlášené nehody, je 4:00 ráno}) = 0.63$$

# Podmíněná pravděpodobnost

pravidlo pro **podmíněnou pravděpodobnost**

$$P(a|b) = \frac{P(a \wedge b)}{P(b)} \text{ pokud } P(b) \neq 0$$

neboli **pravidlo násobení** (*product rule*)

$$P(a \wedge b) = P(a, b) = P(a|b)P(b)$$

$$P(a \wedge b \wedge c) = P(a, b, c) = P(a|b, c)P(b, c) = P(a|b, c)P(b|c)P(c)$$

Dva jevy  $a$  a  $b$  jsou navzájem **nezávislé**

$$\Leftrightarrow P(a|b) = P(a) \text{ a } P(b|a) = P(b)$$

# Vyvozování z nejistých znalostí

- **náhodná proměnná** (*random variable*) – funkce, která vzorkům přiřazuje hodnoty → vrací výsledky měření sledovaného jevu  
**distribuce pravděpodobnosti** náhodné proměnné = (vektor) pravděpodobností, že daná proměnná bude mít konkrétní hodnotu

např.:

*náhodná proměnná **Odd**  $\approx$  výsledek hodu kostkou bude lichý*

*náhodná proměnná **Weather**  $\approx$  jaké bude počasí (slunce, déšť, mraky, sníh)*

$$\text{Odd}(1) = \text{true} \quad \text{Weather}(21.11.2005) = \text{déšť}$$

distribuce pravděpodobností proměnných **Odd** a **Weather**

$$P(\text{Odd} = \text{true}) = 1/6 + 1/6 + 1/6 = 1/2$$

$$P(\text{Odd}) = \langle 1/2, 1/2 \rangle$$

$$P(\text{Weather}) = \langle 0.72, 0.1, 0.08, 0.1 \rangle$$

- pravidla pro výpočet pravděpodobnosti logicky souvisejících událostí

$$P(a \vee b) = P(a) + P(b) - P(a \wedge b)$$

# Bayesovské pravidlo pro vývozování

z pravidla pro **podmíněnou pravděpodobnost** –  $P(a|b) = P(a \wedge b)/P(b)$   
 lze odvodit **Bayesovské pravidlo** pro určení **diagnostické**  
 pravděpodobnosti ze znalosti **příčinné** pravděpodobnosti:

$$P(\text{Příčina}|\text{Následek}) = \frac{P(\text{Následek}|\text{Příčina})P(\text{Příčina})}{P(\text{Následek})}$$

např. ZMB zánět mozkových blan, ZK ztuhlý krk:

$$P(zmb|zk) = \frac{P(zk|zmb)P(zmb)}{P(zk)} = \frac{0.8 \times 0.0001}{0.1} = 0.0008$$

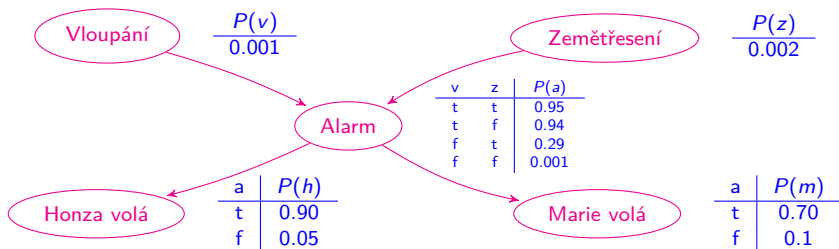
- vývozování** =
1. rozdělení akce na **atomické události**
  2. zjištění **pravděpodobností** atomických událostí
  3. výpočet/odvození pravděpodobností pomocí **složených distribucí pravděpodobností** (*joint probability distribution*)

$$P(\text{Odd, Weather}) = \left\langle \begin{array}{cccc} 0.36 & 0.05 & 0.04 & 0.05 \\ 0.36 & 0.05 & 0.04 & 0.05 \end{array} \right\rangle$$

# Bayesovské sítě

## Bayesovská síť:

- acyklický orientovaný graf
- uzly obsahují tabulky podmíněných pravděpodobností rodičů
- síť reprezentuje složenou distribuci pravděpodobností všech proměnných
- umožňuje efektivní přesné nebo přibližné (Monte Carlo) vyvozování
- nejčastěji používaný aparát pro vyvozování z nejistých znalostí



$$P(v|h, m) = 1/P(h, m) P(v) \sum_z P(z) \sum_a P(a|v, z) P(h|a) P(m|a)$$