

Gramatické formalismy pro ZPJ

Aleš Horák

E-mail: hales@fi.muni.cz

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

Obsah:

- Gramatické formalismy
- Kategoriální gramatiky
- Závislostní gramatiky
- Stromové gramatiky TAG a LTAG
- HPSG – Head-driven Phrase Structure Grammar

Gramatické formalismy

- existuje množství různých přístupů k formální specifikaci gramatik (přirozených jazyků), různé **gramatické formalismy**
- nejznámější formalismy:
 - *lexikální funkční gramatiky* – Lexical Functional Grammar, *LFG*
 - *kategoriální gramatiky* – categorial grammars, *CG*
 - *závislostní gramatiky* – dependency grammars
 - *stromové gramatiky* – (Lexicalized) Tree Adjoining Grammar, *(L)TAG*
 - *gramatiky příznakových struktur* – Head Phrase Structure Grammar, *HPSG*
- soustředíme se na *zápis gramatiky* (notaci)

Gramatické formalismy

- existuje množství různých přístupů k formální specifikaci gramatik (přirozených jazyků), různé **gramatické formalismy**
- nejznámější formalismy:
 - *lexikální funkční gramatiky* – Lexical Functional Grammar, *LFG*
 - *kategoriální gramatiky* – categorial grammars, **CG**
 - *závislostní gramatiky* – dependency grammars
 - *stromové gramatiky* – (Lexicalized) Tree Adjoining Grammar, **(L)TAG**
 - *gramatiky příznakových struktur* – Head Phrase Structure Grammar, **HPSG**
- soustředíme se na **zápis gramatiky** (notaci)

Gramatické formalismy

- existuje množství různých přístupů k formální specifikaci gramatik (přirozených jazyků), různé **gramatické formalismy**
- nejznámější formalismy:
 - *lexikální funkční gramatiky* – Lexical Functional Grammar, *LFG*
 - *kategoriální gramatiky* – categorial grammars, **CG**
 - *závislostní gramatiky* – dependency grammars
 - *stromové gramatiky* – (Lexicalized) Tree Adjoining Grammar, **(L)TAG**
 - *gramatiky příznakových struktur* – Head Phrase Structure Grammar, **HPSG**
- soustředíme se na **zápis gramatiky** (notaci)

Kategoriální gramatiky

- existuje několik různých variant notace

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>	
NP/N	N	$>$	$(S \setminus NP)/NP$	NP	$>$
NP		$S \setminus NP$			
S					$<$

- jiný rozšířený zápis – *výsledek na vrcholku* (result on top) Lambek 1958

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>	
NP/N	N	$>$	$(NP \setminus S)/NP$	NP	$>$
NP		$NP \setminus S$			
S					$<$

Kategoriální gramatiky

- existuje několik různých variant notace

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>	
NP/N	N	$>$	$(S \setminus NP)/NP$	NP	$>$
NP			$S \setminus NP$		$<$
S					

- jiný rozšířený zápis – **výsledek na vrcholku** (result on top) Lambek 1958

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>	
NP/N	N	$>$	$(NP \setminus S)/NP$	NP	$>$
NP			$NP \setminus S$		$<$
S					

Kategoriální gramatiky

- existuje několik různých variant notace

$$\begin{array}{c}
 \underline{\text{šikovní}} \quad \underline{\text{psi}} \quad \underline{\text{mají rádi}} \quad \underline{\text{kočky}} \\
 \underline{NP/N} \quad N > \quad \underline{(S \setminus NP)/NP} \quad NP > \\
 NP \quad \quad \quad S \setminus NP < \\
 \hline
 S
 \end{array}$$

- jiný rozšířený zápis – **výsledek na vrcholku** (result on top) Lambek 1958

$$\begin{array}{c}
 \underline{\text{šikovní}} \quad \underline{\text{psi}} \quad \underline{\text{mají rádi}} \quad \underline{\text{kočky}} \\
 \underline{NP/N} \quad N > \quad \underline{(NP \setminus S)/NP} \quad NP > \\
 NP \quad \quad \quad NP \setminus S < \\
 \hline
 S
 \end{array}$$

Kategoriální gramatiky

- existuje několik různých variant notace

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>
NP/N	N	$>$	$(S \setminus NP)/NP$	NP
NP			$S \setminus NP$	
			$<$	
S				

- jiný rozšířený zápis – **výsledek na vrcholku** (result on top) Lambek 1958

<u>šikovní</u>	<u>psi</u>		<u>mají rádi</u>	<u>kočky</u>
NP/N	N	$>$	$(NP \setminus S)/NP$	NP
NP			$NP \setminus S$	
			$<$	
S				

Kategoriální gramatiky

- existuje několik různých variant notace

$$\begin{array}{c}
 \underline{\text{šikovní}} \quad \underline{\text{psi}} \quad \quad \quad \underline{\text{mají rádi}} \quad \underline{\text{kočky}} \\
 \underline{NP/N} \quad \underline{N} > \quad \quad \quad \underline{(S \setminus NP)/NP} \quad \underline{NP} > \\
 \underline{\quad \quad \quad NP} \quad \quad \quad \quad \quad \quad \quad \quad \underline{S \setminus NP} < \\
 \hline
 S
 \end{array}$$

- jiný rozšířený zápis – **výsledek na vrcholku** (result on top) Lambek 1958

$$\begin{array}{c}
 \underline{\text{šikovní}} \quad \underline{\text{psi}} \quad \quad \quad \underline{\text{mají rádi}} \quad \underline{\text{kočky}} \\
 \underline{NP/N} \quad \underline{N} > \quad \quad \quad \underline{(NP \setminus S)/NP} \quad \underline{NP} > \\
 \underline{\quad \quad \quad NP} \quad \quad \quad \quad \quad \quad \quad \quad \underline{NP \setminus S} < \\
 \hline
 S
 \end{array}$$

Kategoriální gramatiky

- **kategoriální gramatika** (categorial grammar, CG) – skupina teorií syntaxe a sémantiky PJ s velkým důrazem na **lexikon**
- neobsahuje *pravidla* pro kombinování slov → **lexikální kategorie** slov tvoří **funkce**, které určují, jak se dané kategorie kombinují s jinými
výraz je výsledkem **aplikace podvýrazů na sebe**

pěkný := NP/N ... funkce, která má argument N a vrací NP

- **zakladatelé** generativních gramatik – Leśniewski (publ. 1929) a Ajdukiewicz (publ. 1935) ve vazbě na Husserlovu a Russellovu teorii kategorií a teorii typů
- první použití kategoriálních gramatik pro **popis přirozeného jazyka** – Jehošua Bar-Hillel, 1953

Kategoriální gramatiky

- **kategoriální gramatika** (categorial grammar, CG) – skupina teorií syntaxe a sémantiky PJ s velkým důrazem na **lexikon**
- neobsahuje *pravidla* pro kombinování slov → **lexikální kategorie** slov tvoří **funkce**, které určují, jak se dané kategorie kombinují s jinými výraz je výsledkem **aplikace podvýrazů na sebe**

pěkný := NP/N ... funkce, která má argument N a vrací NP

- **zakladatelé** generativních gramatik – Leśniewski (publ. 1929) a Ajdukiewicz (publ. 1935) ve vazbě na Husserlovu a Russellovu teorii kategorií a teorii typů
- první použití kategoriálních gramatik pro **popis přirozeného jazyka** – Jehošua Bar-Hillel, 1953

Kategoriální gramatiky

- **kategoriální gramatika** (categorial grammar, CG) – skupina teorií syntaxe a sémantiky PJ s velkým důrazem na **lexikon**
- neobsahuje *pravidla* pro kombinování slov → **lexikální kategorie** slov tvoří **funkce**, které určují, jak se dané kategorie kombinují s jinými výraz je výsledkem **aplikace podvýrazů na sebe**

pěkný := NP/N ... funkce, která má argument N a vrací NP

- **zakladatelé** generativních gramatik – Leśniewski (publ. 1929) a Ajdukiewicz (publ. 1935) ve vazbě na Husserlovu a Russellovu teorii kategorií a teorii typů
- první použití kategoriálních gramatik pro **popis přirozeného jazyka** – Jehošua Bar-Hillel, 1953

Kategoriální gramatiky

- **kategoriální gramatika** (categorial grammar, CG) – skupina teorií syntaxe a sémantiky PJ s velkým důrazem na **lexikon**
- neobsahuje *pravidla* pro kombinování slov → **lexikální kategorie** slov tvoří **funkce**, které určují, jak se dané kategorie kombinují s jinými
výraz je výsledkem **aplikace podvýrazů na sebe**

pěkný := NP/N ... funkce, která má argument N a vrací NP

- **zakladatelé** generativních gramatik – Leśniewski (publ. 1929) a Ajdukiewicz (publ. 1935) ve vazbě na Husserlovu a Russellovu teorii kategorií a teorii typů
- první použití kategoriálních gramatik pro **popis přirozeného jazyka** – Jehošua Bar-Hillel, 1953

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

- Σ je konečná množina **slov**
- C_{base} je konečná množina **základních kategorií** (funkčních typů)
- C je množina **kategorií** definovaná induktivně takto:
 - $C_{base} \subseteq C$
 - pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - C obsahuje pouze prvky dané výše uvedenými body a) a b)
- $Lex \subseteq \Sigma \times C$ je konečná množina – **lexikon** (zapisujeme v indexovém tvaru **slovo**_{kategorie})
- RS je množina následujících **schémat pravidel**:
 - $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
- $C_{complete} \subseteq C$ je množina **dokončených (kompletních) kategorií**

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

1. Σ je konečná množina **slov**
2. C_{base} je konečná množina **základních kategorií** (funkčních typů)
3. C je množina **kategorií** definovaná induktivně takto:
 - a) $C_{base} \subseteq C$
 - b) pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - c) C obsahuje pouze prvky dané výše uvedenými body a) a b)
4. $Lex \subseteq \Sigma \times C$ je konečná množina – **lexikon** (zapisujeme v indexovém tvaru **slovo**_{kategorie})
5. RS je množina následujících **schémat pravidel**:
 - a) $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - b) $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
6. $C_{complete} \subseteq C$ je množina **dokončených (kompletních) kategorií**

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

- Σ je konečná množina slov
- C_{base} je konečná množina základních kategorií (funkčních typů)
- C je množina kategorií definovaná induktivně takto:
 - $C_{base} \subseteq C$
 - pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - C obsahuje pouze prvky dané výše uvedenými body a) a b)
- $Lex \subseteq \Sigma \times C$ je konečná množina – lexikon (zapisujeme v indexovém tvaru slovo_{kategorie})
- RS je množina následujících schémat pravidel:
 - $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
- $C_{complete} \subseteq C$ je množina dokončených (kompletních) kategorií

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

- Σ je konečná množina **slov**
- C_{base} je konečná množina **základních kategorií** (funkčních typů)
- C je množina **kategorií** definovaná induktivně takto:
 - $C_{base} \subseteq C$
 - pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - C obsahuje pouze prvky dané výše uvedenými body a) a b)
- $Lex \subseteq \Sigma \times C$ je konečná množina – **lexikon** (zapisujeme v indexovém tvaru **slovo**_{kategorie})
- RS je množina následujících **schémat pravidel**:
 - $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
- $C_{complete} \subseteq C$ je množina **dokončených (kompletních) kategorií**

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

1. Σ je konečná množina **slov**
2. C_{base} je konečná množina **základních kategorií** (funkčních typů)
3. C je množina **kategorií** definovaná induktivně takto:
 - a) $C_{base} \subseteq C$
 - b) pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - c) C obsahuje pouze prvky dané výše uvedenými body a) a b)
4. $Lex \subseteq \Sigma \times C$ je konečná množina – **lexikon** (zapisujeme v indexovém tvaru **slovo**_{kategorie})
5. RS je množina následujících **schémat pravidel**:
 - a) $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - b) $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
6. $C_{complete} \subseteq C$ je množina **dokončených (kompletních) kategorií**

Notace kategoriálních gramatik

kategoriální gramatika je šestice $\langle \Sigma, C_{base}, C, Lex, RS, C_{complete} \rangle$, kde

- Σ je konečná množina **slov**
- C_{base} je konečná množina **základních kategorií** (funkčních typů)
- C je množina **kategorií** definovaná induktivně takto:
 - $C_{base} \subseteq C$
 - pokud $X, Y \in C$, potom i $(X/Y) \in C$ a $(X \setminus Y) \in C$
 - C obsahuje pouze prvky dané výše uvedenými body a) a b)
- $Lex \subseteq \Sigma \times C$ je konečná množina – **lexikon** (zapisujeme v indexovém tvaru **slovo**_{kategorie})
- RS je množina následujících **schémat pravidel**:
 - $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - $\beta_{(Y)} \circ \alpha_{(X \setminus Y)} \rightarrow \beta\alpha_{(X)}$,
 kde $\alpha, \beta \in \Sigma$ a $X, Y \in C$
- $C_{complete} \subseteq C$ je množina **dokončených (kompletních) kategorií**

Notace kategoriálních gramatik – pokrač.

- daná schémata umožňují 2 způsoby kombinace:
 - argument **vpravo** ($/$) – $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - argument **vlevo** (\backslash) – $\beta_{(Y)} \circ \alpha_{(X \backslash Y)} \rightarrow \beta\alpha_{(X)}$
- tento typ kategoriální gramatiky označoval Bar-Hillel jako **obousměrný** (bidirectional CG)
- Karel miluje Marii:
 - bázové kategorie = $\{NP, S\}$
 - kategorie z lexikonu: $Karel_{(NP)}$, $Marii_{(NP)}$, $miluje_{((S \backslash NP)/NP)}$
 - $C_{complete} = \{S\}$
- v tomto tvaru je odvození **ekvivalentní derivačním stromům** CFG
- existují ale **rozšíření kategoriálních gramatik**, která vedou k systémům s vyšší vyjadřovací silou, než mají standardní CFG

Notace kategoriálních gramatik – pokrač.

- daná schémata umožňují 2 způsoby kombinace:
 - argument **vpravo** (/) – $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - argument **vlevo** (\) – $\beta_{(Y)} \circ \alpha_{(X/Y)} \rightarrow \beta\alpha_{(X)}$
- tento typ kategoriální gramatiky označoval Bar-Hillel jako **obousměrný** (bidirectional CG)

- Karel miluje Marii:

- bázové kategorie = $\{NP, S\}$
- kategorie z lexikonu: $Karel_{(NP)}$, $Marii_{(NP)}$, $miluje_{((S \setminus NP)/NP)}$
- $C_{complete} = \{S\}$

- v tomto tvaru je odvození **ekvivalentní derivačním stromům** CFG
- existují ale **rozšíření kategoriálních gramatik**, která vedou k systémům s vyšší vyjadřovací silou, než mají standardní CFG

Notace kategoriálních gramatik – pokrač.

- daná schémata umožňují 2 způsoby kombinace:
 - argument **vpravo** ($/$) – $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - argument **vlevo** (\backslash) – $\beta_{(Y)} \circ \alpha_{(X \backslash Y)} \rightarrow \beta\alpha_{(X)}$
- tento typ kategoriální gramatiky označoval Bar-Hillel jako **obousměrný** (bidirectional CG)
- Karel miluje Marii:
 - bázové kategorie = $\{NP, S\}$
 - kategorie z lexikonu: $Karel_{(NP)}$, $Marii_{(NP)}$, $miluje_{((S \backslash NP)/NP)}$
 - $C_{complete} = \{S\}$
- v tomto tvaru je odvození **ekvivalentní derivačním stromům** CFG
- existují ale **rozšíření kategoriálních gramatik**, která vedou k systémům s vyšší vyjadřovací silou, než mají standardní CFG

Notace kategoriálních gramatik – pokrač.

- daná schémata umožňují 2 způsoby kombinace:
 - argument **vpravo** ($/$) – $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - argument **vlevo** (\backslash) – $\beta_{(Y)} \circ \alpha_{(X \backslash Y)} \rightarrow \beta\alpha_{(X)}$
- tento typ kategoriální gramatiky označoval Bar-Hillel jako **obousměrný** (bidirectional CG)
- Karel miluje Marii:
 - báze kategorie = $\{NP, S\}$
 - kategorie z lexikonu: $Karel_{(NP)}$, $Marii_{(NP)}$, $miluje_{((S \backslash NP)/NP)}$
 - $C_{complete} = \{S\}$
- v tomto tvaru je odvození **ekvivalentní derivačním stromům** CFG
- existují ale **rozšíření kategoriálních gramatik**, která vedou k systémům s vyšší vyjadřovací silou, než mají standardní CFG

Notace kategoriálních gramatik – pokrač.

- daná schémata umožňují 2 způsoby kombinace:
 - argument **vpravo** ($/$) – $\alpha_{(X/Y)} \circ \beta_{(Y)} \rightarrow \alpha\beta_{(X)}$
 - argument **vlevo** (\backslash) – $\beta_{(Y)} \circ \alpha_{(X \backslash Y)} \rightarrow \beta\alpha_{(X)}$
- tento typ kategoriální gramatiky označoval Bar-Hillel jako **obousměrný** (bidirectional CG)
- Karel miluje Marii:
 - bázové kategorie = $\{NP, S\}$
 - kategorie z lexikonu: $Karel_{(NP)}$, $Marii_{(NP)}$, $miluje_{((S \backslash NP)/NP)}$
 - $C_{complete} = \{S\}$
- v tomto tvaru je odvození **ekvivalentní derivačním stromům** CFG
- existují ale **rozšíření kategoriálních gramatik**, která vedou k systémům s vyšší vyjadřovací silou, než mají standardní CFG

Rozšíření kategoriálních gramatik

- klíčový problém – nespojité větné části, tzv. **neprojektivity**
- řešení pomocí rozšíření CG – přídavné **kombinatorické operátory** založené na **typech**
- dva možné přístupy:
 - pravidlově orientovaný přidává pravidla odpovídající jednoduchým operacím nad kategoriemi, jako jsou:
 - wrap – komutace argumentů
 - type-raising – aplikace typů podobná aplikaci tradičních pádů na jmenné fráze
 - comp – kompozice funkcí
 - k nejpropracovanějším systémům tohoto typu patří **kombinatorické kategoriální gramatiky (CCG)**.
 - deduktivní přístup vychází z Lambekova syntaktického kalkulu
 - pohled na kategoriální lomítko (slash) jako formu logické implikace
 - axiomy a inferenční pravidla potom definují teorii důkazu
např. *aplikace funkce* \approx pravidlo *modus ponens* $P \wedge (P \Rightarrow Q) \Rightarrow Q$

OpenCCG library – <http://openccg.sourceforge.net/>

Rozšíření kategoriálních gramatik

- klíčový problém – nespojité větné části, tzv. **neprojektivity**
- řešení pomocí rozšíření CG – přídavné **kombinatorické operátory** založené na **typech**
- dva možné přístupy:
 - pravidlově orientovaný přidává pravidla odpovídající jednoduchým operacím nad kategoriemi, jako jsou:
 - wrap – komutace argumentů
 - type-raising – aplikace typů podobná aplikaci tradičních pádů na jmenné fráze
 - comp – kompozice funkcí
 - k nejpropracovanějším systémům tohoto typu patří **kombinatorické kategoriální gramatiky (CCG)**.
 - deduktivní přístup vychází z Lambekova syntaktického kalkulu
 - pohled na kategoriální lomítko (slash) jako formu logické implikace
 - axiomy a inferenční pravidla potom definují teorii důkazu
např. *aplikace funkce* \approx pravidlo *modus ponens* $P \wedge (P \Rightarrow Q) \Rightarrow Q$

OpenCCG library – <http://openccg.sourceforge.net/>

Rozšíření kategoriálních gramatik

- klíčový problém – nespojité větné části, tzv. **neprojektivity**
- řešení pomocí rozšíření CG – přídavné **kombinatorické operátory** založené na **typech**
- dva možné přístupy:
 - **pravidlově orientovaný** přidává pravidla odpovídající jednoduchým operacím nad kategoriemi, jako jsou:

- **wrap** – komutace argumentů
- **type-raising** – aplikace typů podobná aplikaci tradičních pádů na jmenné fráze
- **comp** – kompozice funkcí

k nejpropracovanějším systémům tohoto typu patří **kombinatorické kategoriální gramatiky (CCG)**.

- **deduktivní přístup** vychází z Lambekova syntaktického kalkulu
 - pohled na kategoriální lomítko (slash) jako formu **logické implikace**
 - axiomy a inferenční pravidla potom definují **teorii důkazu**
např. *aplikace funkce* \approx pravidlo *modus ponens* $P \wedge (P \Rightarrow Q) \Rightarrow Q$

OpenCCG library – <http://openccg.sourceforge.net/>

Rozšíření kategoriálních gramatik

- klíčový problém – nespojité větné části, tzv. **neprojektivity**
- řešení pomocí rozšíření CG – přídavné **kombinatorické operátory** založené na **typech**
- dva možné přístupy:
 - **pravidlově orientovaný** přidává pravidla odpovídající jednoduchým operacím nad kategoriemi, jako jsou:
 - **wrap** – komutace argumentů
 - **type-raising** – aplikace typů podobná aplikaci tradičních pádů na jmenné fráze
 - **comp** – kompozice funkcí
 - k nejpropracovanějším systémům tohoto typu patří **kombinatorické kategoriální gramatiky (CCG)**.
 - **deduktivní přístup** vychází z Lambekova syntaktického kalkulu
 - pohled na kategoriální lomítko (slash) jako formu **logické implikace**
 - axiomy a inferenční pravidla potom definují **teorii důkazu**
např. *aplikace funkce* \approx pravidlo *modus ponens* $P \wedge (P \Rightarrow Q) \Rightarrow Q$

OpenCCG library – <http://openccg.sourceforge.net/>

Obsah

- 1 Gramatické formalismy
- 2 Kategoriální gramatiky
 - Notace kategoriálních gramatik
 - Rozšíření kategoriálních gramatik
- 3 Závislostní gramatiky
 - Universal Dependencies
- 4 Stromové gramatiky TAG a LTAG
 - operace
 - lexikalizace
 - generované jazyky
- 5 HPSG – Head-driven Phrase Structure Grammar
 - Syntaktické kategorie
 - Dobře utvořené příznakové struktury

Závislostní gramatiky

- blízko ke kategoriálním gramatikám – vztah **závislosti** mezi **řídícími** a **závislými** větnými členy
- vhodné pro popis jazyků s volným slovosledem
- používají výhradně **lexikalizovaných uzlů** (v závislostním stromu) – neexistují žádné neterminály
→ závislostní analýza se jeví *jednodušší*
- využívá **valence** či subkategorizace – vztah mezi jedním slovem a jeho argumenty
typicky vztah mezi slovesem a jeho možnými doplněními:

```
nosit  
= koho|co  
= komu & koho|co
```

Závislostní gramatiky

- blízko ke kategoriálním gramatikám – vztah **závislosti** mezi **řídícími** a **závislými** větnými členy
- vhodné pro popis jazyků s volným slovosledem
- používají výhradně **lexikalizovaných uzlů** (v závislostním stromu) – neexistují žádné neterminály
→ závislostní analýza se jeví *jednodušší*
- využívá **valence** či subkategorizace – vztah mezi jedním slovem a jeho argumenty
typicky vztah mezi slovesem a jeho možnými doplněními:

```
nosit  
= koho|co  
= komu & koho|co
```

Závislostní gramatiky

- blízko ke kategoriálním gramatikám – vztah **závislosti** mezi **řídícími** a **závislými** větnými členy
- vhodné pro popis jazyků s volným slovosledem
- používají výhradně **lexikalizovaných uzlů** (v závislostním stromu) – neexistují žádné neterminály
→ závislostní analýza se jeví *jednodušší*
- využívá **valence** či subkategorizace – vztah mezi jedním slovem a jeho argumenty
typicky vztah mezi slovesem a jeho možnými doplněními:

```
nosit  
= koho|co  
= komu & koho|co
```


Závislostní gramatiky

- blízko ke kategoriálním gramatikám – vztah **závislosti** mezi **řídícími** a **závislými** větnými členy
- vhodné pro popis jazyků s volným slovosledem
- používají výhradně **lexikalizovaných uzlů** (v závislostním stromu) – neexistují žádné neterminály
 - závislostní analýza se jeví *jednodušší*
- využívá **valence** či subkategorizace – vztah mezi jedním slovem a jeho argumenty
typicky vztah mezi slovesem a jeho možnými doplněními:

```
nosit  
= koho|co  
= komu & koho|co
```

Závislostní gramatiky – pokrač.

hlavní přístupy:

- navazuje na evropskou lingvistickou tradici – až k antice
- nejstarší užití – Tesnière 1959
- **funkční generativní popis** (*Functional Generative Description*, FGD) – jeden z nejpropracovanějších závislostních systémů, pražská lingvistická škola (Sgall, Hajičová, Panevová)
- **LG, Link Grammar** – Temperley, Carnegie Mellon University
<http://www.link.cs.cmu.edu/link/>
- **UDG, Unification Dependency Grammar** – Maxwell
- **MTT, Meaning-Text Theory** – Mel'čuk
- **WG, Word Grammar** – Hudson
- **Lexibase** – Starosta
- **FG, Functional Grammar** – Dik
- **DUG, Dependency Unification Grammar** – Halliday

Závislostní gramatiky – pokrač.

hlavní přístupy:

- navazuje na evropskou lingvistickou tradici – až k antice
- nejstarší užití – Tesnière 1959
- funkční generativní popis (*Functional Generative Description*, FGD) – jeden z nejpropracovanějších závislostních systémů, pražská lingvistická škola (Sgall, Hajičová, Panevová)
- LG, *Link Grammar* – Temperley, Carnegie Mellon University
<http://www.link.cs.cmu.edu/link/>
- UDG, *Unification Dependency Grammar* – Maxwell
- MTT, *Meaning-Text Theory* – Mel'čuk
- WG, *Word Grammar* – Hudson
- Lexicase – Starosta
- FG, *Functional Grammar* – Dik
- DUG, *Dependency Unification Grammar* – Halliday

Závislostní gramatiky – pokrač.

hlavní přístupy:

- navazuje na evropskou lingvistickou tradici – až k antice
- nejstarší užití – Tesnière 1959
- **funkční generativní popis** (*Functional Generative Description*, FGD) – jeden z nejpropracovanějších závislostních systémů, pražská lingvistická škola (Sgall, Hajičová, Panevová)
- **LG, Link Grammar** – Temperley, Carnegie Mellon University
<http://www.link.cs.cmu.edu/link/>
- UDG, *Unification Dependency Grammar* – Maxwell
- MTT, *Meaning-Text Theory* – Mel'čuk
- WG, *Word Grammar* – Hudson
- Lexicase – Starosta
- FG, *Functional Grammar* – Dik
- DUG, *Dependency Unification Grammar* – Halliday

Závislostní gramatiky – pokrač.

hlavní přístupy:

- navazuje na evropskou lingvistickou tradici – až k antice
- nejstarší užití – Tesnière 1959
- **funkční generativní popis** (*Functional Generative Description*, FGD) – jeden z nejpropracovanějších závislostních systémů, pražská lingvistická škola (Sgall, Hajičová, Panevová)
- **LG, Link Grammar** – Temperley, Carnegie Mellon University
<http://www.link.cs.cmu.edu/link/>
- **UDG, Unification Dependency Grammar** – Maxwell
- **MTT, Meaning-Text Theory** – Mel'čuk
- **WG, Word Grammar** – Hudson
- **Lexibase** – Starosta
- **FG, Functional Grammar** – Dik
- **DUG, Dependency Unification Grammar** – Halliday

Universal Dependencies

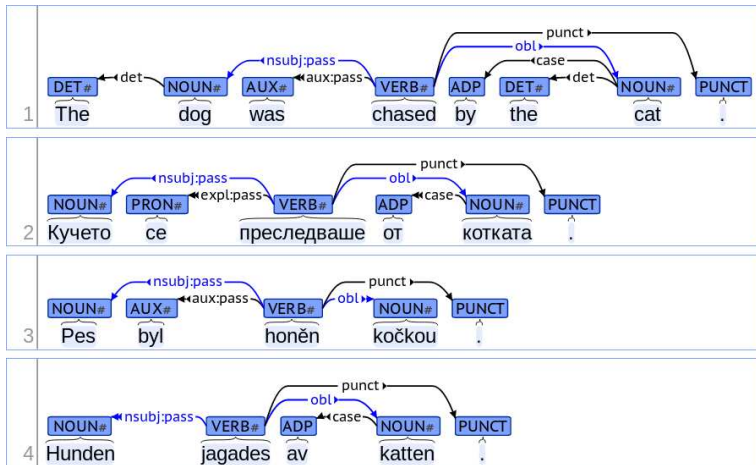
- www.universaldependencies.org, UD
- sjednocení závislostní anotace pro různé jazyky
- cca 200 stromových bank (*treebanks*) ve více než 100 jazycích

Universal Dependencies

- www.universaldependencies.org, UD
- sjednocení závislostní anotace pro různé jazyky
- cca 200 stromových bank (*treebanks*) ve více než 100 jazycích

Universal Dependencies

- www.universaldependencies.org, UD
- sjednocení závislostní anotace pro různé jazyky
- cca 200 stromových bank (*treebanks*) ve více než 100 jazycích



Google Universal Tagset

- gramatiky pro jednotlivé jazyky založené na podobných principech
- detaily značkování ale často nejsou převoditelné 1:1
- sjednocení – značkování v UD založené na minimalistické Google Universal Tagset

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Google Universal Tagset

- gramatiky pro jednotlivé jazyky založené na podobných principech
- detaily značkování ale často nejsou převoditelné 1:1
- sjednocení – značkování v UD založené na minimalistické Google Universal Tagset

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Google Universal Tagset

- gramatiky pro jednotlivé jazyky založené na podobných principech
- detaily značkování ale často nejsou převoditelné 1:1
- sjednocení – značkování v UD založené na minimalistické Google Universal Tagset

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Google Universal Tagset

- gramatiky pro jednotlivé jazyky založené na podobných principech
- detaily značkování ale často nejsou převoditelné 1:1
- sjednocení – značkování v UD založené na minimalistické Google Universal Tagset

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Universal Features

- značky z **Universal Tagset** vymezují základní třídy
- lexikální a gramatické vztahy popisují **Universal Features**

Lexical features	Inflectional features	
	Nominal	Verbal
PronType	Gender	VerbForm
NumType	Animacy	Mood
Poss	Number	Tense
Reflex	Case	Aspect
Foreign	Definite	Voice
Abbr	Degree	Evident
		Polarity
		Person
		Polite

Universal Features

- značky z **Universal Tagset** vymezují základní **třídy**
- lexikální a gramatické vztahy popisují **Universal Features**

Lexical features	Inflectional features	
	Nominal	Verbal
PronType	Gender	VerbForm
NumType	Animacy	Mood
Poss	Number	Tense
Reflex	Case	Aspect
Foreign	Definite	Voice
Abbr	Degree	Evident
		Polarity
		Person
		Polite

Universal Features

- značky z **Universal Tagset** vymezují základní **třídy**
- lexikální a gramatické vztahy popisují **Universal Features**

Lexical features	Inflectional features	
	Nominal	Verbal
PronType	Gender	VerbForm
NumType	Animacy	Mood
Poss	Number	Tense
Reflex	Case	Aspect
Foreign	Definite	Voice
Abbr	Degree	Evident
		Polarity
		Person
		Polite

Universal Dependencies

1	Správkyňě	Správkyňě	NOUN	Case=Nom Gender=Fem Number=Sing Polarity=Pos
2	dědictví	dědictví	NOUN	Case=Gen Gender=Neut Number=Sing Polarity=Pos
3	Nováková	Nováková	PROPN	Case=Nom Gender=Fem NameType=Sur Number=Sing Polarity=Pos
4	označila	označit	VERB	Aspect=Perf Gender=Fem,Neut Number=Plur,Sing Polarity=Pos Tense=Past VerbForm=Part Voice=Act
5	pondělní	pondělní	ADJ	Case=Acc Degree=Pos Gender=Neut Number=Sing Polarity=Pos
6	rozhodnutí	rozhodnutí	NOUN	Case=Acc Gender=Neut Number=Sing Polarity=Pos
7	za	za	ADP	AdpType=Prep Case=Acc
8	potěšující	potěšující	ADJ	Aspect=Imp Case=Acc Gender=Neut Number=Sing Polarity=Pos Tense=Pres VerbForm=Part Voice=Act
9	.	.	PUNCT	-

Jazykové instrukce pro Universal Dependencies

- každý jazyk má uvedené **instrukce** pro:
 - tokenizaci (hranice slov)
 - morfologické značky
 - syntax – základní a rozšířené závislosti
- např. pro **češtinu** – www.universaldependencies.org/cs/
- **cíl instrukcí** – **sjednocení** anotací napříč jazyky
- obsahuje i instrukce **netypické** pro daný jazyk – např. v češtině značkování některých zájmen jako **determiner** nebo expandování slov – **kdybych** = **když** + **bych**

SlidO

Jazykové instrukce pro Universal Dependencies

- každý jazyk má uvedené **instrukce** pro:
 - tokenizaci (hranice slov)
 - morfologické značky
 - syntax – základní a rozšířené závislosti
- např. pro **češtinu** – www.universaldependencies.org/cs/
- **cíl instrukcí** – **sjednocení** anotací napříč jazyky
- obsahuje i instrukce **netypické** pro daný jazyk – např. v češtině značkování některých zájmen jako **determiner** nebo expandování slov – **kdybych** = **když** + **bych**

SlidO

Jazykové instrukce pro Universal Dependencies

- každý jazyk má uvedené **instrukce** pro:
 - tokenizaci (hranice slov)
 - morfologické značky
 - syntax – základní a rozšířené závislosti
- např. pro **češtinu** – www.universaldependencies.org/cs/
- **cíl instrukcí** – **sjednocení** anotací napříč jazyky
- obsahuje i instrukce **netypické** pro daný jazyk – např. v češtině značkování některých zájmen jako **determiner** nebo expandování slov – **kdybych** = **když** + **bych**

SlidO

Jazykové instrukce pro Universal Dependencies

- každý jazyk má uvedené **instrukce** pro:
 - tokenizaci (hranice slov)
 - morfologické značky
 - syntax – základní a rozšířené závislosti
- např. pro **češtinu** – www.universaldependencies.org/cs/
- **cíl instrukcí** – **sjednocení** anotací napříč jazyky
- obsahuje i instrukce **netypické** pro daný jazyk – např. v češtině značkování některých zájmen jako **determiner** nebo expandování slov – **kdybych** = **když** + **bych**

Slido

Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- anotace (editor, statistiky, validace)
- vizualizace
- dotazování
- UDPipe – trénování a automatické anotace

Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- anotace (editor, statistiky, validace)
- vizualizace
- dotazování
- UDPipe – trénování a automatické anotace

Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- anotace (editor, statistiky, validace)
- vizualizace
- dotazování
- UDPipe – trénování a automatické anotace

Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

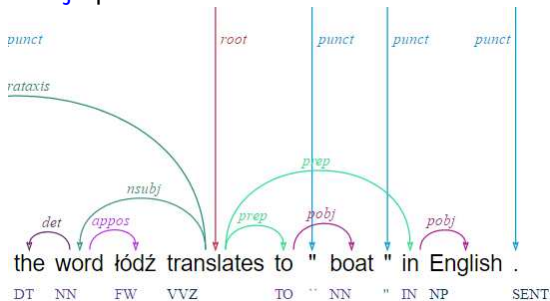
- anotace (editor, statistiky, validace)
- vizualizace
- dotazování
- UDPipe – trénování a automatické anotace

Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- **anotace** (editor, statistiky, validace)
- vizualizace
- dotazování
- **UDPipe** – trénování a automatické anotace

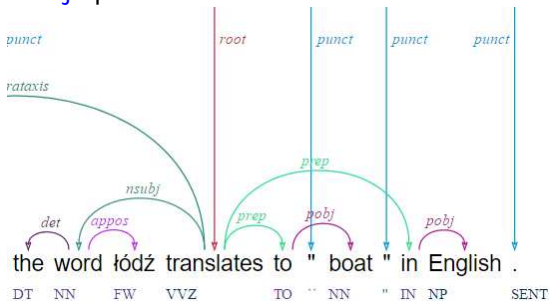


Využití Universal Dependencies

- **rovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- **anotace** (editor, statistiky, validace)
- **vizualizace**
- **dotazování**
- **UDPipe** – trénování a automatické anotace

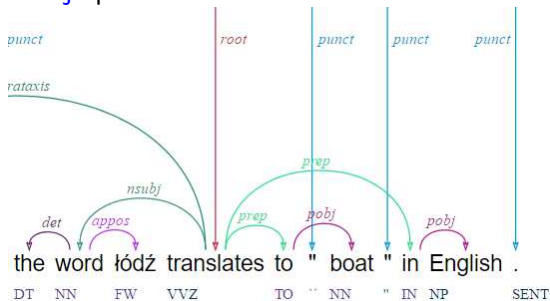


Využití Universal Dependencies

- **srovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- **anotace** (editor, statistiky, validace)
- **vizualizace**
- **dotazování**
- **UDPipe** – trénování a automatické anotace

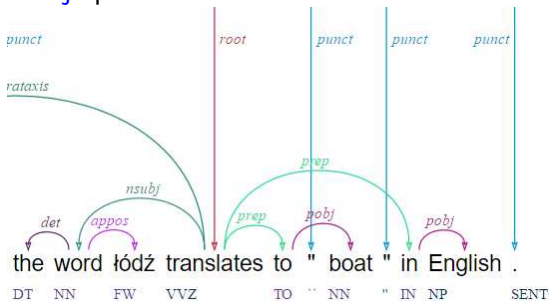


Využití Universal Dependencies

- **rovnání** lingvistických fenomenů **napříč jazyky**
- **testování** syntaktické analýzy na různých jazycích
- **vícejazyčná syntaktická analýza** – paralelní dokumenty
- snadné **porozumění** rozdílům v anotacích

UD poskytuje **univerzální nástroje** pro:

- **anotace** (editor, statistiky, validace)
- **vizualizace**
- **dotazování**
- **UDPipe** – trénování a automatické anotace



Obsah

- 1 Gramatické formalismy
- 2 Kategoriální gramatiky
 - Notace kategoriálních gramatik
 - Rozšíření kategoriálních gramatik
- 3 Závislostní gramatiky
 - Universal Dependencies
- 4 Stromové gramatiky TAG a LTAG
 - operace
 - lexikalizace
 - generované jazyky
- 5 HPSG – Head-driven Phrase Structure Grammar
 - Syntaktické kategorie
 - Dobře utvořené příznakové struktury

Stromové gramatiky TAG a LTAG

- Tree Adjoining Grammar – Joshi, Levy a Takahashi: *TAG Formalism*, 1975
- Lexicalized TAG – Joshi a Schabes: *Parsing with Lexicalized TAG*, 1991
- pracují přímo se **stromy** a ne s řetězcí slov
- množina **počátečních stromů** – základní stavební prvky
- složitější věty odvozovány s použitím **pomocných stromů**

počáteční (*initial*) strom:

pomocný (*auxiliary*) strom:

Stromové gramatiky TAG a LTAG

- Tree Adjoining Grammar – Joshi, Levy a Takahashi: *TAG Formalism*, 1975
- Lexicalized TAG – Joshi a Schabes: *Parsing with Lexicalized TAG*, 1991
- pracují přímo se **stromy** a ne s řetězcí slov
- množina **počátečních stromů** – základní stavební prvky
- složitější věty odvozovány s použitím **pomocných stromů**

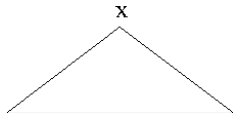
počáteční (*initial*) strom:

pomocný (*auxiliary*) strom:

Stromové gramatiky TAG a LTAG

- Tree Adjoining Grammar – Joshi, Levy a Takahashi: *TAG Formalism*, 1975
- Lexicalized TAG – Joshi a Schabes: *Parsing with Lexicalized TAG*, 1991
- pracují přímo se **stromy** a ne s řetězcí slov
- množina **počátečních stromů** – základní stavební prvky
- složitější věty odvozovány s použitím **pomocných stromů**

počáteční (*initial*) strom:

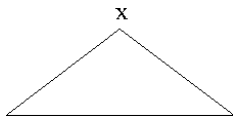


pomocný (*auxiliary*) strom:

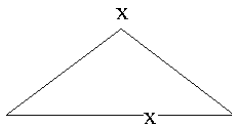
Stromové gramatiky TAG a LTAG

- Tree Adjoining Grammar – Joshi, Levy a Takahashi: *TAG Formalism*, 1975
- Lexicalized TAG – Joshi a Schabes: *Parsing with Lexicalized TAG*, 1991
- pracují přímo se **stromy** a ne s řetězcí slov
- množina **počátečních stromů** – základní stavební prvky
- složitější věty odvozovány s použitím **pomocných stromů**

počáteční (*initial*) strom:

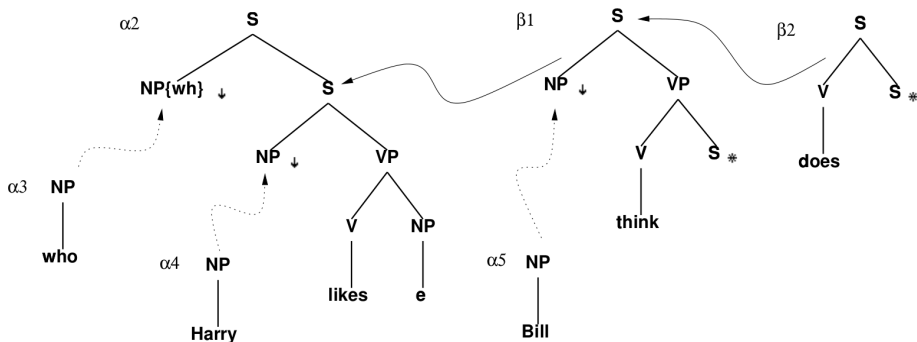


pomocný (*auxiliary*) strom:



XTAG Project

The XTAG Project – <http://www.cis.upenn.edu/~xtag/>



TuLiPA-frames

The screenshot displays the TuLiPA software interface. The main window, titled "Derivation Trees + Derived Trees", shows a derivation tree for the sentence "Timm liebt Feigen". The root node is TVnp2-26-74--liebt--1, which branches into TVnp2-26-75--liebt--1 and TVnp2-26-73--liebt--1. The left child further branches into TNoun-0-6--Timm1--0, and the right child branches into TNoun-0-6--Feigen3--0.

The configuration window, titled "TuLiPA", is open in the foreground. It contains the following settings:

- Mode:** TT-MCTAG/TAG (selected), RCG
- TT-MCTAG:** Max LPA length: []
- TT-MCTAG/TAG:** Show derivation steps in GUI (unchecked), Dependency output (unchecked)
- Misc:** Verbose mode (unchecked), XML output, no GUI (unchecked)
- Grammar:** /local/toy.xml
- Lemmas:** /local/lemma.xml
- Morphological entries:** /local/morph.xml
- Output file:** []
- Axiom:** []
- Sentence:** Timm liebt Feigen
- Buttons:** parse, Quit

At the bottom of the configuration window, the following information is displayed:

```

Grammar conversion time: 0.572578 sec.
Sentence "Timm liebt Feigen" parsed.
Parsing time: 0.199693 sec.
Total parsing time for sentence: Timm liebt Feigen: 0.79414 sec.

```

<https://github.com/spetitjean/TuLiPA-frames>

TAG – počáteční a pomocné stromy

- **počáteční stromy** – neobsahují rekurzi → popisují složkovou strukturu jednoduchých vět, jmenných skupin, předložkových skupin, ...
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci*

počáteční strom typu X = jeho kořen je označen termem X

- **pomocné stromy** – reprezentují *rekurzivní struktury* popisují větné členy, které se **připojují** k základním strukturám (např. příslovečné určení)
- charakterizace:
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci* kromě právě jednoho neterminálního uzlu (**patový uzel**, *foot node*)
 3. **patový uzel** má stejné označení jako kořenový uzel

patový uzel – slouží k připojení stromu k jinému uzlu

TAG – počáteční a pomocné stromy

- **počáteční stromy** – neobsahují rekurzi → popisují složkovou strukturu jednoduchých vět, jmenných skupin, předložkových skupin, ...
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci*

počáteční strom typu X = jeho kořen je označen termem X

- **pomocné stromy** – reprezentují *rekurzivní struktury* popisují větné členy, které se **připojují** k základním strukturám (např. příslovečné určení)
- charakterizace:
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci* kromě právě jednoho neterminálního uzlu (**patový uzel**, *foot node*)
 3. **patový uzel** má stejné označení jako kořenový uzel

patový uzel – slouží k připojení stromu k jinému uzlu

TAG – počáteční a pomocné stromy

- **počáteční stromy** – neobsahují rekurzi → popisují složkovou strukturu jednoduchých vět, jmenných skupin, předložkových skupin, ...
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci*

počáteční strom typu X = jeho kořen je označen termem X

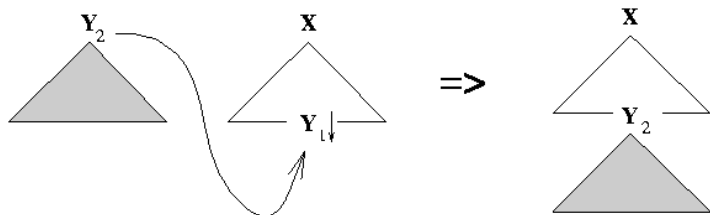
- **pomocné stromy** – reprezentují *rekurzivní struktury* popisují větné členy, které se **připojují** k základním strukturám (např. příslovečné určení)
- charakterizace:
 1. všechny **nelistové uzly** odpovídají *neterminálům*
 2. všechny **listové uzly** odpovídají *terminálům* nebo *neterminálním* uzlům určeným k *substituci* kromě právě jednoho neterminálního uzlu (**patový uzel**, *foot node*)
 3. **patový uzel** má stejné označení jako kořenový uzel

patový uzel – slouží k připojení stromu k jinému uzlu

TAG – operace

dvě operace – **substituce** a **připojení** (*adjunction*)

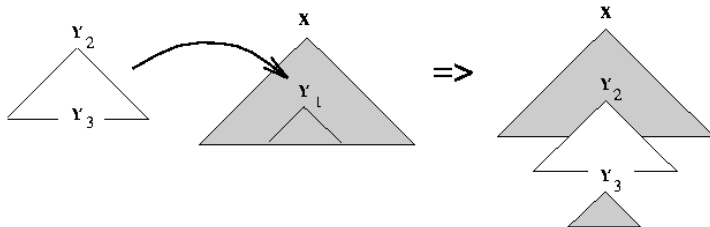
operace **substituce** – nahrazuje označený neterminál v listech nějakého stromu stromem, jehož kořen nese stejné označení



$Y_1 \downarrow$ – označený pro substituci

TAG – operace připojení

operace **připojení** – vložení pomocného stromu, popisujícího rekurzi neterminálu X , se stromem, který obsahuje uzel označený rovněž X



Definice TAG

- TAG $G = (I, A, S)$ je:
 - množina I konečných počátečních stromů
 - množina A pomocných stromů
 - typ stromu S – neterminál označující větu
- množina stromů $\mathcal{T}(G)$ TA gramatiky G = množina všech stromů odvoditelných z počátečních stromů typu S z I , jejichž spodní okraj sestává čistě z terminálních uzlů (všechny substituční uzly byly doplněny)
- jazyk řetězců $\mathcal{L}(G)$ generovaných TA gramatikou G = množina všech terminálních řetězců na spodním okraji stromů v $\mathcal{T}(G)$.

Definice TAG

- TAG $G = (I, A, S)$ je:
 - množina I konečných počátečních stromů
 - množina A pomocných stromů
 - typ stromu S – neterminál označující větu
- množina stromů $\mathcal{T}(G)$ TA gramatiky G = množina všech stromů odvoditelných z počátečních stromů typu S z I , jejichž spodní okraj sestává čistě z terminálních uzlů (všechny substituční uzly byly doplněny)
- jazyk řetězců $\mathcal{L}(G)$ generovaných TA gramatikou G = množina všech terminálních řetězců na spodním okraji stromů v $\mathcal{T}(G)$.

Definice TAG

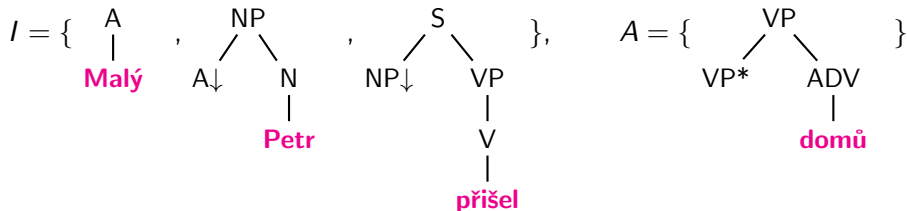
- TAG $G = (I, A, S)$ je:
 - množina I konečných počátečních stromů
 - množina A pomocných stromů
 - typ stromu S – neterminál označující větu
- množina stromů $\mathcal{T}(G)$ TA gramatiky G = množina všech stromů odvoditelných z počátečních stromů typu S z I , jejichž spodní okraj sestává čistě z terminálních uzlů (všechny substituční uzly byly doplněny)
- jazyk řetězců $\mathcal{L}(G)$ generovaných TA gramatikou G = množina všech terminálních řetězců na spodním okraji stromů v $\mathcal{T}(G)$.

LTAG – lexikalizace

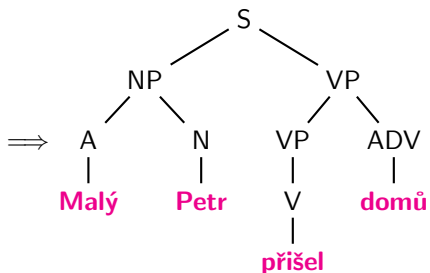
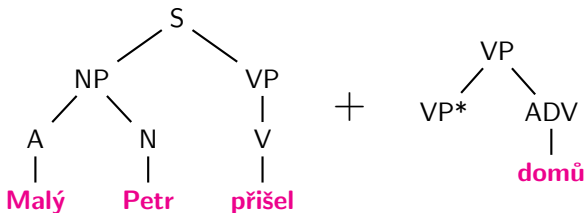
LTAG je **lexikalizovanou variantou** formalismu TAG

→ počáteční i pomocné stromy obsahují v listech jednu nebo více tzv. **lexikálních kotev** – uzly, které jsou přiřazeny (ukotveny) k určitým slovům lexikonu

lexikalizované stromy (*substituční uzly* – ↓, *patové uzly* – *):



LTAG – lexikalizované připojení



TAG a LTAG – generované jazyky

- díky použití operace připojení mají TAG a LTAG **větší generativní sílu** než bezkontextové gramatiky ($CFG \subset MCSL$) → generují **mírně kontextové jazyky** (*mildly context-sensitive languages*)

MCSL:

- vlastnost konstantního růstu – pokud uspořádáme řetězec jazyka vzestupně podle délky, potom rozdíl dvou po sobě jdoucích délek nemůže být libovolný (každá délka je lineární kombinací konečného počtu pevných délek).
- analyzovatelnost v polynomiálním čase $O(n^6)$ vzhledem k délce vstupu
- i jiné formalismy umí MCSL (jsou ekvivalentní s (L)TAG):
 - LF, *Linear Feature Grammars* – Carter, 1985
 - HG, *Head Grammars* – Pollard, 1984
 - CCG, kombinatorické kategoriální gramatiky

TAG a LTAG – generované jazyky

- díky použití operace připojení mají TAG a LTAG **větší generativní sílu** než bezkontextové gramatiky ($CFG \subset MCSL$) → generují **mírně kontextové jazyky** (*mildly context-sensitive languages*)

MCSL:

- vlastnost **konstantního růstu** – pokud uspořádáme řetězce jazyka vzestupně podle délky, potom rozdíl dvou po sobě jdoucích délek nemůže být libovolný (každá délka je lineární kombinací konečného počtu pevných délek).
 - analyzovatelnost v **polynomiálním čase** $O(n^6)$ vzhledem k délce vstupu
- i jiné formalismy umí MCSL (jsou ekvivalentní s (L)TAG):
 - *LIG, Linear Indexed Grammars* – Gazdar, 1985
 - *HG, Head Grammars* – Pollard, 1984
 - CCG, kombinatorické kategoriální gramatiky

TAG a LTAG – generované jazyky

- díky použití operace připojení mají TAG a LTAG **větší generativní sílu** než bezkontextové gramatiky ($CFG \subset MCSL$) → generují **mírně kontextové jazyky** (*mildly context-sensitive languages*)

MCSL:

- vlastnost **konstantního růstu** – pokud uspořádáme řetězce jazyka vzestupně podle délky, potom rozdíl dvou po sobě jdoucích délek nemůže být libovolný (každá délka je lineární kombinací konečného počtu pevných délek).
- analyzovatelnost v **polynomiálním čase** $O(n^6)$ vzhledem k délce vstupu
- i jiné formalismy umí MCSL (jsou ekvivalentní s (L)TAG):
 - *LIG, Linear Indexed Grammars* – Gazdar, 1985
 - *HG, Head Grammars* – Pollard, 1984
 - CCG, kombinatorické kategoriální gramatiky

TAG a LTAG – generované jazyky

- díky použití operace připojení mají TAG a LTAG **větší generativní sílu** než bezkontextové gramatiky ($CFG \subset MCSL$) → generují **mírně kontextové jazyky** (*mildly context-sensitive languages*)

MCSL:

- vlastnost **konstantního růstu** – pokud uspořádáme řetězce jazyka vzestupně podle délky, potom rozdíl dvou po sobě jdoucích délek nemůže být libovolný (každá délka je lineární kombinací konečného počtu pevných délek).
- analyzovatelnost v **polynomiálním čase** $O(n^6)$ vzhledem k délce vstupu
- i jiné formalismy umí MCSL (jsou ekvivalentní s (L)TAG):
 - LIG, *Linear Indexed Grammars* – Gazdar, 1985
 - HG, *Head Grammars* – Pollard, 1984
 - CCG, kombinatorické kategoriální gramatiky

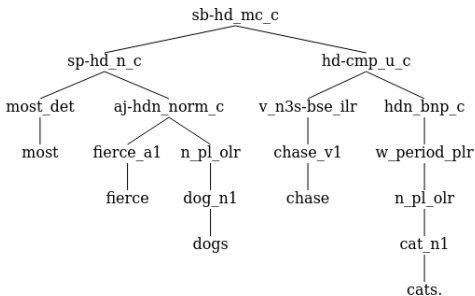
Obsah

- 1 Gramatické formalismy
- 2 Kategoriální gramatiky
 - Notace kategoriálních gramatik
 - Rozšíření kategoriálních gramatik
- 3 Závislostní gramatiky
 - Universal Dependencies
- 4 Stromové gramatiky TAG a LTAG
 - operace
 - lexikalizace
 - generované jazyky
- 5 HPSG – Head-driven Phrase Structure Grammar
 - Syntaktické kategorie
 - Dobře utvořené příznakové struktury

HPSG – Head-driven Phrase Structure Grammar

English Resource Grammar <https://github.com/delph-in/erg>

DELPH-IN demo <http://delph-in.github.io/delphin-viz/demo/>



TOP INDEX	<i>h0</i> <i>e2</i>				
RELS	$\left[\begin{array}{l} \text{most}_q(0:4) \\ \text{LBL } h4 \\ \text{ARG0 } x3 \\ \text{RSTR } h5 \\ \text{BODY } h6 \end{array} \right]$	$\left[\begin{array}{l} \text{fierce}_a_1(5:11) \\ \text{LBL } h7 \\ \text{ARG0 } e8 \\ \text{ARG1 } x3 \end{array} \right]$	$\left[\begin{array}{l} \text{dog}_n_1(12:16) \\ \text{LBL } h7 \\ \text{ARG0 } x3 \end{array} \right]$	$\left[\begin{array}{l} \text{chase}_v_1(17:22) \\ \text{LBL } h1 \\ \text{ARG0 } e2 \\ \text{ARG1 } x3 \\ \text{ARG2 } x9 \end{array} \right]$	$\left[\begin{array}{l} \text{undef}_q(23:28) \\ \text{LBL } h10 \\ \text{ARG0 } x9 \\ \text{RSTR } h11 \\ \text{BODY } h12 \end{array} \right]$
	$\left[\begin{array}{l} \text{cat}_n_1(23:28) \\ \text{LBL } h13 \\ \text{ARG0 } x9 \end{array} \right]$				
HCONS	$\left[\begin{array}{l} \text{qeq} \\ \text{HARG } h5 \\ \text{LARG } h7 \end{array} \right]$	$\left[\begin{array}{l} \text{qeq} \\ \text{HARG } h11 \\ \text{LARG } h13 \end{array} \right]$	$\left[\begin{array}{l} \text{qeq} \\ \text{HARG } h0 \\ \text{LARG } h1 \end{array} \right]$		

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar

- HPSG, **Head-driven Phrase Structure Grammar** – Pollard & Sag, 1994
- navazuje na Gazdar, **Generalized Phrase Structure Grammar**, 1985
- **lexikalizovaná** teorie generativní gramatiky přirozeného jazyka
- *neterminály* CFG jsou nahrazeny **příznakovými strukturami**
- založená na **omezeních** (constraints)
- modeluje jazyk pomocí **deklarativních omezení** typovaných struktur. Pro vyhodnocení omezení se používá **unifikace** mezi příznakovými strukturami.
- **příznaky** jsou propojeny pomocí **strukturního sdílení**, tedy předáváním proměnných mezi podstrukturami dané struktury
- HPSG je **nederivační**, na rozdíl od jiných formalismů, kde jsou různé úrovně syntaktické struktury sekvenčně odvozovány pomocí transformačních operací

HPSG – Head-driven Phrase Structure Grammar – pokrač.

- gramatika je v HPSG modelována pomocí **uspořádaných příznakových struktur**, které korespondují s typy výrazů přirozeného jazyka a jejich částmi
- cílem teorie je detailní specifikace, které příznakové struktury jsou **přípustné**
- příznakové struktury definují **omezení**
hodnoty příznaků mohou být jednoho ze čtyř typů
 - atomy
 - příznakové struktury
 - množiny příznakových struktur (**{...}**)
 - nebo seznamy příznakových struktur (**<...>**)

HPSG – Head-driven Phrase Structure Grammar – pokrač.

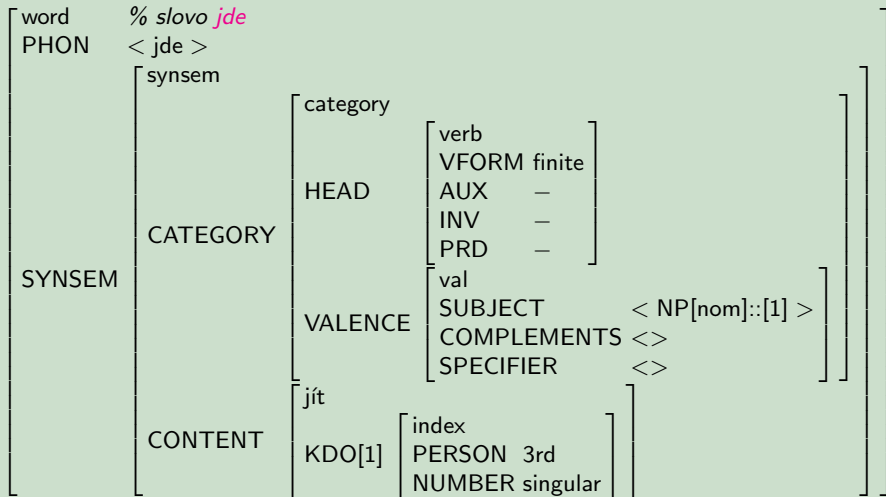
- gramatika je v HPSG modelována pomocí **uspořádaných příznakových struktur**, které korespondují s typy výrazů přirozeného jazyka a jejich částmi
- cílem teorie je detailní specifikace, které příznakové struktury jsou **přípustné**
- příznakové struktury definují **omezení**
hodnoty příznaků mohou být jednoho ze čtyř typů
 - atomy
 - příznakové struktury
 - množiny příznakových struktur (**{...}**)
 - nebo seznamy příznakových struktur (**<...>**)

HPSG – Head-driven Phrase Structure Grammar – pokrač.

- gramatika je v HPSG modelována pomocí **uspořádaných příznakových struktur**, které korespondují s typy výrazů přirozeného jazyka a jejich částmi
- cílem teorie je detailní specifikace, které příznakové struktury jsou **přípustné**
- příznakové struktury definují **omezení**
hodnoty příznaků mohou být jednoho ze čtyř typů
 - atomy
 - příznakové struktury
 - množiny příznakových struktur ($\{\dots\}$)
 - nebo seznamy příznakových struktur ($\langle\dots\rangle$)

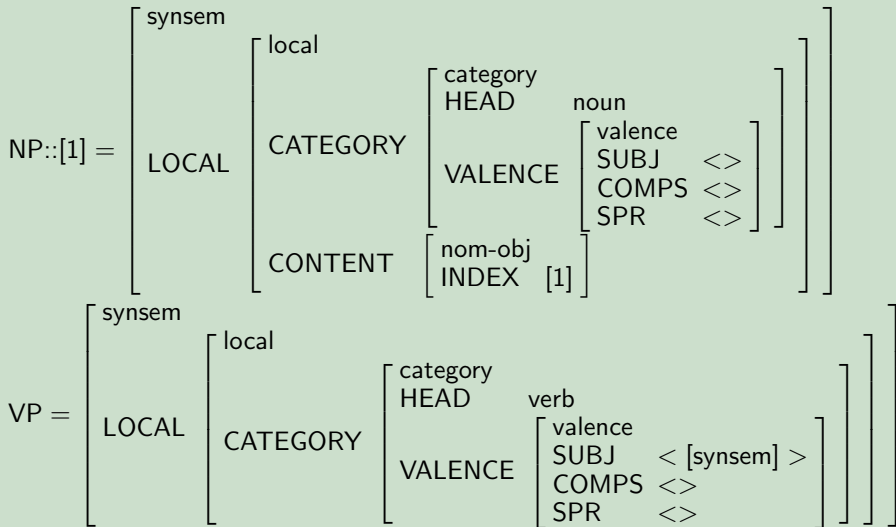
HPSG struktury

HPSG struktury jsou **typované příznakové struktury**
 zapisují se pomocí AVM – **příznaky** velkými písmeny, **typy** malými



Syntaktické kategorie

symbolsy **syntaktických kategorií** – zkratky určitých příznakových popisů:



Lexikální položky

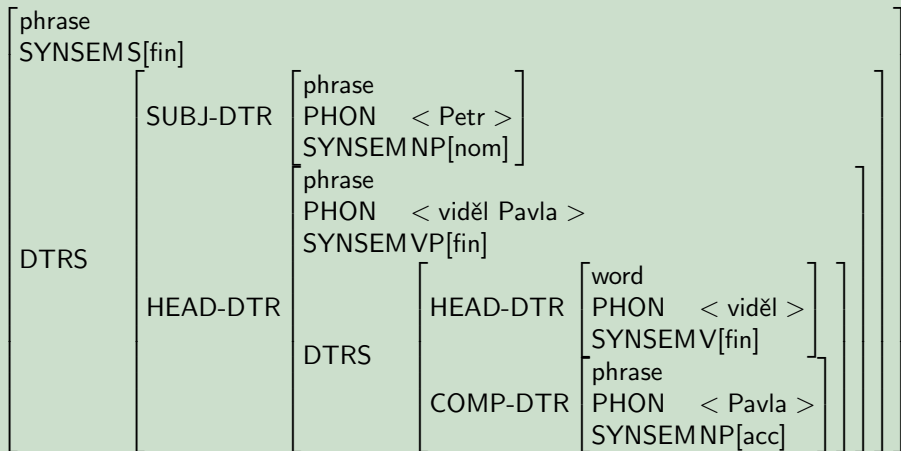
velké množství akcí je v **lexikonu**:

JÍT	[CATEGORY	HEAD	verb	
			VALENCE	[SUBJ < NP::[1] >
				COMPS	<>
	CONTENT	jít			
			KDO	[1]	

DÁT	[CATEGORY	HEAD	verb	
			VALENCE	[SUBJ < NP::[1] >
				COMPS	< NP::[2], NP::[3] >
	CONTENT	dát			
			KDO	[1]	
			CO	[2]	
			KOMU	[3]	

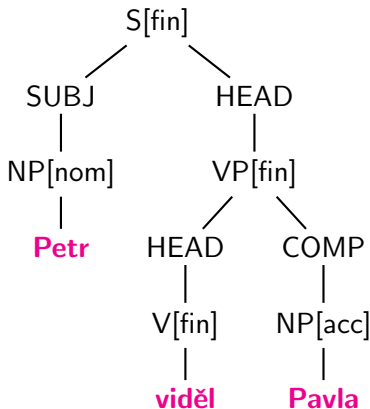
Fráze

reprezentace **frází** – v HPSG obdoba reprezentace **slov**
 navíc příznak **DAUGHTERS** – struktura členů fráze



Fráze – pokrač.

pro snazší čtení popisů frází používáme **stromový zápis**:



ve skutečnosti se ovšem jedná o **příznakovou strukturu**, ne strom!

SLiDo

Dobře utvořené příznakové struktury

dobře utvořené příznakové struktury musí splňovat **omezení daná gramatikou**

příznaková struktura je **dobře utvořená** \Leftrightarrow :

- každý uzel splňuje **omezení geometrie příznaku**
- každá uzel vstupního slova splňuje **omezení některé lexikální položky**
- každý frázový uzel splňuje **frázová omezení** – *omezení přímé dominance* (immediate dominance, viz dále), *omezení hlavových příznaků* (head feature), *valenční omezení*, ...

omezení geometrie příznaku specifikují:

- s jakými **typy** se pracuje
- jaká je použitá **typová hierarchie** – který typ je podtypem jiného typu
- pro každý typ – jaké příznaky přísluší tomuto typu
- pro každý typ a každý příznak – jakých typů mohou být hodnoty tohoto příznaku

Dobře utvořené příznakové struktury

dobře utvořené příznakové struktury musí splňovat **omezení daná gramatikou**

příznaková struktura je **dobře utvořená** \Leftrightarrow :

- každý uzel splňuje **omezení geometrie příznaku**
- každá uzel vstupního slova splňuje **omezení** některé **lexikální položky**
- každý frázový uzel splňuje **frázová omezení** – *omezení přímé dominance* (immediate dominance, viz dále), *omezení hlavových příznaků* (head feature), *valenční omezení*, ...

omezení geometrie příznaku specifikují:

- s jakými **typy** se pracuje
- jaká je použitá **typová hierarchie** – který typ je podtypem jiného typu
- pro každý typ – jaké příznaky přísluší tomuto typu
- pro každý typ a každý příznak – jakých typů mohou být hodnoty tohoto příznaku

Dobře utvořené příznakové struktury

dobře utvořené příznakové struktury musí splňovat **omezení daná gramatikou**

příznaková struktura je **dobře utvořená** \Leftrightarrow :

- každý uzel splňuje **omezení geometrie příznaku**
- každá uzel vstupního slova splňuje **omezení** některé **lexikální položky**
- každý frázový uzel splňuje **frázová omezení** – *omezení přímé dominance* (immediate dominance, viz dále), *omezení hlavových příznaků* (head feature), *valenční omezení*, ...

omezení geometrie příznaku specifikují:

- s jakými **typy** se pracuje
- jaká je použitá **typová hierarchie** – který typ je podtypem jiného typu
- pro každý typ – jaké příznaky přísluší tomuto typu
- pro každý typ a každý příznak – jakých typů mohou být hodnoty tohoto příznaku

HPSG – deklarace typu

pro popis omezení geometrie příznaku se používají **typové deklarace**:

```
category: [HEAD: head, VALENCE: valence]
```

```
head      # příznaková struktura složená z příznakových struktur
```

```
  noun: [CASE: case]
```

```
  verb: [VFORM: vform, AUX: boolean, INV: boolean]
```

```
  prep: [PFORM: pform]
```

```
  ...
```

```
vform     # jednoduchý příznak, forma slovesa – možné hodnoty:
```

```
  fin     # určitý tvar slovesa
```

```
  inf     # neurčitý tvar slovesa – infinitive
```

```
  ...
```

```
case      # jednoduchý příznak, gramatický pád
```

```
  nom     # 1. pád, nominativ
```

```
  acc     # 4. pád, akuzativ
```

```
  ...
```

HPSG – dobře utvořená slova a fráze

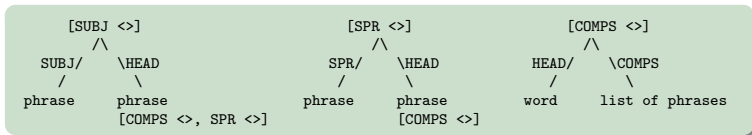
- každé vstupní **slovo** musí splňovat některou **lexikální položku**
- fráze** musí splňovat **frázová omezení** (constraints):
 - omezení přímé dominance** – každá fráze musí odpovídat jednomu ze schémat – schéma *head-subject*, schéma *head-specifier*, schéma *head-complement*, ...



- omezení hlavových příznaků** – pro každou frázi, která má hlavu, musí být hlavové příznaky fráze shodné s hlavovými příznaky potomka, který je hlavou
- valenční omezení** – pro každý z valenčních příznaků (SUBJECT, COMPLEMENTS, ...) – hodnota příznaku na hlavové frázi musí odpovídat hodnotě na potomku, který je hlavou, minus ty příznaky, které jsou splněny některým z nehlavových potomků

HPSG – dobře utvořená slova a fráze

- každé vstupní **slovo** musí splňovat některou **lexikální položku**
- fráze** musí splňovat **frázová omezení** (constraints):
 - omezení přímé dominance** – každá fráze musí odpovídat jednomu ze schémat – schéma *head-subject*, schéma *head-specifier*, schéma *head-complement*, ...



- omezení hlavových příznaků** – pro každou frázi, která má hlavu, musí být hlavové příznaky fráze shodné s hlavovými příznaky potomka, který je hlavou
- valenční omezení** – pro každý z valenčních příznaků (SUBJECT, COMPLEMENTS, ...) – hodnota příznaku na hlavové frázi musí odpovídat hodnotě na potomku, který je hlavou, minus ty příznaky, které jsou splněny některým z nehlavových potomků

Dobře utvořené příznakové struktury

omezení ve větě 'Petr viděl Pavla.':

