

Syntaxe – gramatiky a syntaktické struktury

Aleš Horák

E-mail: hales@fi.muni.cz

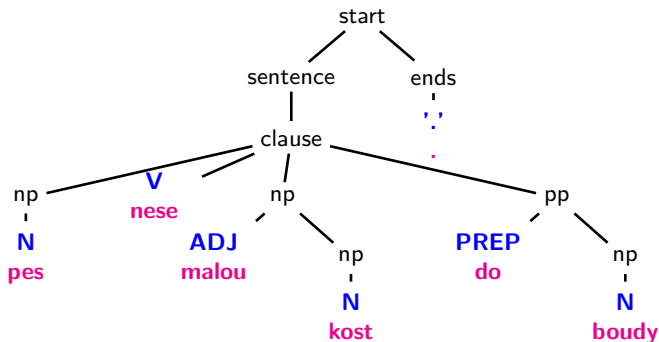
http://nlp.fi.muni.cz/nlp_intro/

Obsah:

- Syntaxe, syntaktická analýza
- Specifikace gramatik
- Chomského teorie syntaxe
- Východiska syntaktické analýzy

Syntaxe, syntaktická analýza

- **syntaxe** – charakterizace dobře utvořených kombinací slovních tvarů do **věty** nebo **fráze**
- pomocí **gramatických pravidel**
- výstup ze syntaktické analýzy (např. derivační strom) tvoří často **vstup pro analýzu sémantickou**



Základní termíny

- **fráze** (*phrase*) – jednotka jazyka větší než slovo, ale menší než věta
např. *jmenná fráze*, *slovesná fráze*, *adjektivní fráze* nebo *přísllovečná fráze*
- **lexikální symbol**, **lexikální kategorie** (*lexical category*) – tzv. **preterminál**
speciální neterminál gramatiky, který se přímo přepisuje na terminálový
řetězec znaků, tj. pravidla tvaru $X \rightarrow w$

N	→	pes		člověk		dům ...
V	→	nese		chodit		psal ...
ADJ	→	...				
PREP	→	...				
ADV	→	...				

označuje všechny slova, která odpovídají určitému lexikálnímu symbolu
(všechna podstatná jména, přídavná jména, ...)

Základní termíny

- **fráze** (*phrase*) – jednotka jazyka větší než slovo, ale menší než věta
např. *jmenná fráze*, *slovesná fráze*, *adjektivní fráze* nebo *přísllovečná fráze*
- **lexikální symbol, lexikální kategorie** (*lexical category*) – tzv. **preterminál**
speciální neterminál gramatiky, který se přímo přepisuje na terminálový řetězec znaků, tj. pravidla tvaru $X \rightarrow w$

N	→	pes		člověk		dům ...
V	→	nese		chodit		psal ...
ADJ	→	...				
PREP	→	...				
ADV	→	...				

označuje všechny slova, která odpovídají určitému lexikálnímu symbolu (všechna podstatná jména, přídavná jména, ...)

Základní termíny – pokrač.

- **frázová kategorie** (*phrasal category*)
neterminální symbol gramatiky, který nevyjadřuje lexikální kategorii

ADJP	→	ADJP	ADJ
NP	→	ADJP	N
VP	→	V	NP
S	→	NP	VP

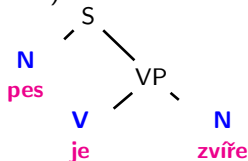
- **větný člen** (*constituent*) – lexikální nebo frázová kategorie

Základní termíny – pokrač.

- **větná struktura** (*sentence structure*) – strukturovaný popis větných členů

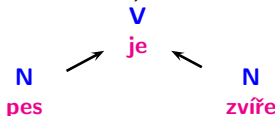
- **povrchová struktura** (*surface structure*)

derivační/složkový strom jako
výsledek bezkontextové (CF)
analýzy



- **závislostní struktura** (*dependency structure*)

zobrazuje závislosti mezi
větnými členy



- **hloubková struktura** (*deep structure*) – sémantická interpretace fráze. Popisuje **role větných členů** (agens, patiens, donor, cause, ...)

Syntaktická analýza programovacích × přirozených jazyků

- počítačové programy a přirozené jazyky sdílí **teorii formálních jazyků** a praktický zájem o **efektivní algoritmy** analýzy
- **ALGOL 60** – první programovací jazyk popsán pomocí **Backus-Naurovy formy** (BNF)

```
<if_statement> ::= if <boolean_expression> then  
                    <statement_sequence>  
                    [ else  
                      <statement_sequence> ]  
                    end if ;
```

- dokázalo se, že BNF je **ekvivalentní** CFG (1962) → podnítilo výzkum formálních jazyků z hlediska jazyků přirozených

Syntaktická analýza programovacích × přirozených jazyků

- počítačové programy a přirozené jazyky sdílí **teorii formálních jazyků** a praktický zájem o **efektivní algoritmy** analýzy
- **ALGOL 60** – první programovací jazyk popsán pomocí **Backus-Naurovy formy** (BNF)

```
<if_statement> ::= if <boolean_expression> then  
                    <statement_sequence>  
                    [ else  
                      <statement_sequence> ]  
                    end if ;
```

- dokázalo se, že BNF je **ekvivalentní** CFG (1962) → podnítilo výzkum formálních jazyků z hlediska jazyků přirozených

Syntaktická analýza programovacích × přirozených jazyků

- počítačové programy a přirozené jazyky sdílí **teorii formálních jazyků** a praktický zájem o **efektivní algoritmy** analýzy
- **ALGOL 60** – první programovací jazyk popsán pomocí **Backus-Naurovy formy** (BNF)

```
<if_statement> ::= if <boolean_expression> then  
                    <statement_sequence>  
                    [ else  
                      <statement_sequence> ]  
                    end if ;
```

- dokázalo se, že BNF je **ekvivalentní** CFG (1962) → podnítilo výzkum formálních jazyků z hlediska jazyků přirozených

Typy gramatik

gramatiky:

- **regulární** (regular) **neterminál** → **terminál**[neterminál]

$$S \rightarrow aS$$

$$S \rightarrow b$$

ekvivalentní síle **konečných automatů**,
neumí $a^n b^n$

- **bezkontextové** (context-free) **neterminál** → **cokoliv**
ekvivalentní síle **zásobníkových**

$$S \rightarrow aSb$$

automatů, umí $a^n b^n$, neumí $a^n b^n c^n$

- **kontextové** (context-sensitive) – víc termů na levé straně (*kontext* neterminálu)

$$\underline{ASB} \rightarrow \underline{AaBcB}$$

ekvivalentní síle **lineárně ohra-
ničeného Turingova stroje**,
umí $a^n b^n c^n$

- **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení
ekvivalentní síle **Turingova stroje**

přirozený jazyk byl dlouho pokládán za bezkontextový → nyní prokázáno,
že obsahuje **kontextové prvky**

Typy gramatik

gramatiky:

- **regulární** (regular) **neterminál** → **terminál**[neterminál]

$$\begin{array}{l} S \rightarrow aS \\ S \rightarrow b \end{array}$$

ekvivalentní síle **konečných automatů**,
neumí $a^n b^n$

- **bezkontextové** (context-free) **neterminál** → **cokoliv**

$$S \rightarrow aSb$$

ekvivalentní síle **zásobníkových automatů**, umí $a^n b^n$, neumí $a^n b^n c^n$

- **kontextové** (context-sensitive) – víc termů na levé straně (*kontext neterminálu*)

$$\underline{ASB} \rightarrow \underline{AaBcB}$$

ekvivalentní síle **lineárně ohraničeného Turingova stroje**,
umí $a^n b^n c^n$

- **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení ekvivalentní síle **Turingova stroje**

přirozený jazyk byl dlouho pokládán za bezkontextový → nyní prokázáno, že obsahuje **kontextové prvky**

Typy gramatik

gramatiky:

- **regulární** (regular) **neterminál** → **terminál**[neterminál]

$$\begin{array}{l} S \rightarrow aS \\ S \rightarrow b \end{array}$$

ekvivalentní síle **konečných automatů**,
neumí $a^n b^n$

- **bezkontextové** (context-free) **neterminál** → **cokoliv**

$$S \rightarrow aSb$$

ekvivalentní síle **zásobníkových automatů**, umí $a^n b^n$, neumí $a^n b^n c^n$

- **kontextové** (context-sensitive) – víc termů na levé straně (*kontext* neterminálu)

$$\underline{A}S\underline{B} \rightarrow \underline{A}aBc\underline{B}$$

ekvivalentní síle **lineárně ohraničeného Turingova stroje**,
umí $a^n b^n c^n$

- **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení ekvivalentní síle **Turingova stroje**

přirozený jazyk byl dlouho pokládán za bezkontextový → nyní prokázáno, že obsahuje **kontextové prvky**

Typy gramatik

gramatiky:

- **regulární** (regular) **neterminál** → **terminál**[neterminál]

$$S \rightarrow aS$$

$$S \rightarrow b$$

ekvivalentní síle **konečných automatů**,
neumí $a^n b^n$

- **bezkontextové** (context-free) **neterminál** → **cokoliv**
ekvivalentní síle **zásobníkových**

$$S \rightarrow aSb$$

automatů, umí $a^n b^n$, neumí $a^n b^n c^n$

- **kontextové** (context-sensitive) – víc termů na levé straně (*kontext* neterminálu)

$$\underline{A}S\underline{B} \rightarrow \underline{A}aBc\underline{B}$$

ekvivalentní síle **lineárně ohraničeného Turingova stroje**,
umí $a^n b^n c^n$

- **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení
ekvivalentní síle **Turingova stroje**

přirozený jazyk byl dlouho pokládán za bezkontextový → nyní prokázáno,
že obsahuje **kontextové prvky**

Typy gramatik

gramatiky:

- **regulární** (regular) **neterminál** \rightarrow **terminál**[neterminál]

$$\begin{array}{l} S \rightarrow aS \\ S \rightarrow b \end{array}$$

ekvivalentní síle **konečných automatů**,
neumí $a^n b^n$

- **bezkontextové** (context-free) **neterminál** \rightarrow **cokoliv**

$$S \rightarrow aSb$$

ekvivalentní síle **zásobníkových automatů**, umí $a^n b^n$, neumí $a^n b^n c^n$

- **kontextové** (context-sensitive) – víc termů na levé straně (*kontext* neterminálu)

$$\underline{A}S\underline{B} \rightarrow \underline{A}aBc\underline{B}$$

ekvivalentní síle **lineárně ohraničeného Turingova stroje**,
umí $a^n b^n c^n$

- **rekurzivně vyčíslitelné** (recursively enumerable) – bez omezení ekvivalentní síle **Turingova stroje**

přirozený jazyk byl dlouho pokládán za bezkontextový \rightarrow nyní prokázáno, že obsahuje **kontextové prvky**

Gramatiky přirozeného jazyka

- konkrétní popis **gramatiky přirozeného jazyka** je velmi složitým úkolem
- kontrast s faktem, že rodilí mluvčí nemívají potíže s pochopením významu vět
- asi **nejstarší formální popis jazyka** – gramatika sanskrtu od indického učenice Paniniho
 - vznikla cca 400 př.n.l.
 - dochovaná v rituálních védických textech
 - gramatika podobná BNF (Backus-Naurově formě)
 - používala bezkontextových i kontextových pravidel, obsahovala asi 1700 termů
 - zabývala se z větší části morfologií, nikoliv syntaxí, neboť pořádek slov je v sanskrtu dosti volný
 - toto dílo bylo evropské škole obecné lingvistiky, která má kořeny v řecké a římské tradici, neznámé až do 19. století

Gramatiky přirozeného jazyka

- konkrétní popis **gramatiky přirozeného jazyka** je velmi složitým úkolem
- kontrast s faktem, že rodilí mluvčí nemívají potíže s pochopením významu vět
- asi **nejstarší formální popis jazyka** – gramatika sanskrtu od indického učenice Paniniho



संस्कृत भारती

- vznikla cca 400 př.n.l.
- dochovaná v rituálních védických textech
- gramatika podobná BNF (Backus-Naurově formě)
- používala bezkontextových i kontextových pravidel, obsahovala asi 1700 termů
- zabývala se z větší části morfologií, nikoliv syntaxí, neboť pořádek slov je v sanskrtu dosti volný
- toto dílo bylo evropské škole obecné lingvistiky, která má kořeny v řecké a římské tradici, neznámé až do 19. století

Obsah

- 1 Syntaxe, syntaktická analýza
 - Základní termíny
 - Analýza programovacích a přirozených jazyků
 - Gramatiky přirozeného jazyka
- 2 Specifikace gramatik
 - Složkový a závislostní přístup
 - Uzly syntaktického stromu
 - Pořádek slov ve větě
 - Možnosti zadávání gramatik
- 3 Chomského teorie syntaxe
 - Standardní teorie syntaxe
- 4 Východiska syntaktické analýzy
 - Návrh podkladů a datových struktur
 - Grammatical Framework

Složkový a závislostní přístup

dva základní způsoby zadávání gramatik

složkový přístup:

- skupiny slov tvoří větné jednotky, které jsou označovány jako **fráze**, a jako **větné členy** (*složky*, *constituents*) formují **větu**

- např.

podstatné jméno – součást jmenné fráze (noun phrase – NP)
jmenná fráze spolu s předložkou – tvoří předložkovou frázi (prepositional phrase – PP)

- syntaktická struktura věty je zachycována jako **složkový strom**

Složkový a závislostní přístup

dva základní způsoby zadávání gramatik

složkový přístup:

- skupiny slov tvoří větné jednotky, které jsou označovány jako **fráze**, a jako **větné členy** (*složky, constituents*) formují **větu**

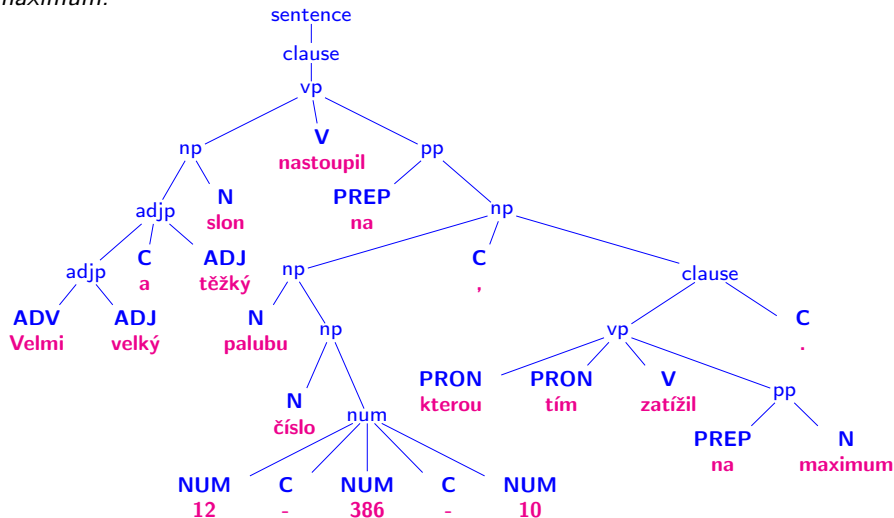
- např.

podstatné jméno – součást jmenné fráze (noun phrase – NP)
jmenná fráze spolu s předložkou – tvoří předložkovou frázi (prepositional phrase – PP)

- syntaktická struktura věty je zachycována jako **složkový strom**

Složkový a závislostní přístup – složkové stromy

Velmi velký a těžký slon nastoupil na palubu číslo 12-386-10, kterou tím zatížil na maximum.



Složkový a závislostní přístup – pokrač.

závislostní přístup:

- jeden člen vazby je označován jako **řídící**, druhý jako **závislý**

- např.

přídavné jméno závisí na řídícím podstatném jménu

- syntaktická struktura věty je zachycována pomocí **závislostního stromu**:
 - *uzly* odpovídají elementárním jednotkám vstupu (často slovům)
 - *hrany* označují vztahy závislosti mezi elementárními jednotkami
- závislost není relací mezi jednotlivými slovy, ale obecně relací mezi jedním **slovem a frází** řízenou druhým slovem. např.

vazba mezi konkrétním slovesem a podmětem
nebo vazba mezi slovesem a předmětem věty

technicky vzato, závislostní relace je vztahem mezi uzly a podstromy (uzlem a všemi uzly, které na tomto uzlu závisí)

Složkový a závislostní přístup – pokrač.

závislostní přístup:

- jeden člen vazby je označován jako **řídící**, druhý jako **závislý**
- např.

přídavné jméno závisí na řídícím podstatném jménu

- syntaktická struktura věty je zachycována pomocí **závislostního stromu**:
 - *uzly* odpovídají elementárním jednotkám vstupu (často slovům)
 - *hrany* označují vztahy závislosti mezi elementárními jednotkami
- závislost není relací mezi jednotlivými slovy, ale obecně relací mezi jedním **slovem a frází** řízenou druhým slovem. např.

vazba mezi konkrétním slovesem a podmětem
nebo vazba mezi slovesem a předmětem věty

technicky vzato, závislostní relace je vztahem mezi uzly a podstromy (uzlem a všemi uzly, které na tomto uzlu závisí)

Složkový a závislostní přístup – pokrač.

závislostní přístup:

- jeden člen vazby je označován jako **řídící**, druhý jako **závislý**

- např.

přídavné jméno závisí na řídícím podstatném jménu

- syntaktická struktura věty je zachycována pomocí **závislostního stromu**:

- *uzly* odpovídají elementárním jednotkám vstupu (často slovům)
- *hrany* označují vztahy závislosti mezi elementárními jednotkami

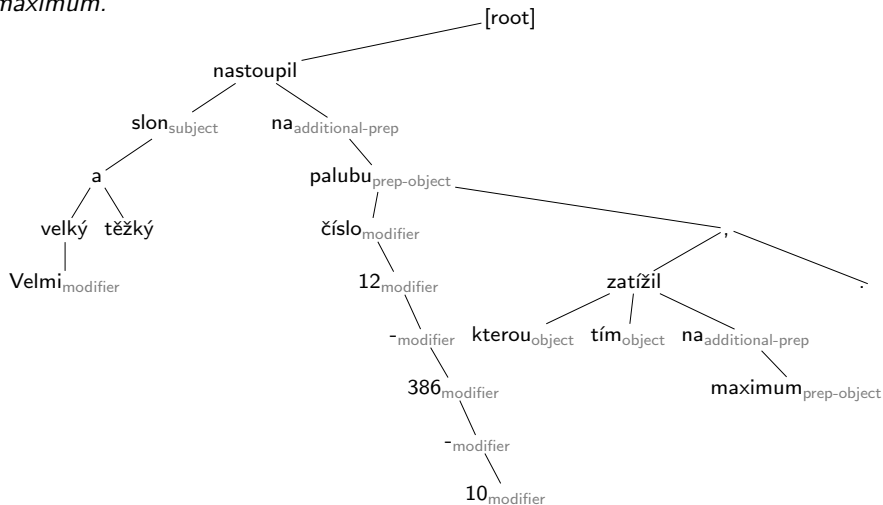
- závislost není relací mezi jednotlivými slovy, ale obecně relací mezi jedním **slovem a frází** řízenou druhým slovem. např.

vazba mezi konkrétním slovesem a podměttem
nebo vazba mezi slovesem a předmětem věty

technicky vzato, závislostní relace je vztahem mezi uzly a podstromy (uzlem a všemi uzly, které na tomto uzlu závisí)

Složkový a závislostní přístup – závislostní stromy

Velmi velký a těžký slon nastoupil na palubu číslo 12-386-10, kterou tím zatížil na maximum.



Složkový a závislostní přístup – pokrač.

- jen zřídka se používá **čistě** složkový či striktně závislostní přístup
- ve složkovém jsou závislosti zpravidla vyjádřeny přidáním označení, která složka je **řídící** pro danou frázi
- závislostní strom bývá doplněn o informaci určující **lineární precedenci**
- je možné pak mezi těmito přístupy **výsledek převádět**

Složkový a závislostní přístup – pokrač.

- jen zřídka se používá čistě složkový či striktně závislostní přístup
- ve složkovém jsou závislosti zpravidla vyjádřeny přidáním označení, která složka je řídicí pro danou frázi
- závislostní strom bývá doplněn o informaci určující lineární precedenci
- je možné pak mezi těmito přístupy výsledek převádět

Složkový a závislostní přístup – pokrač.

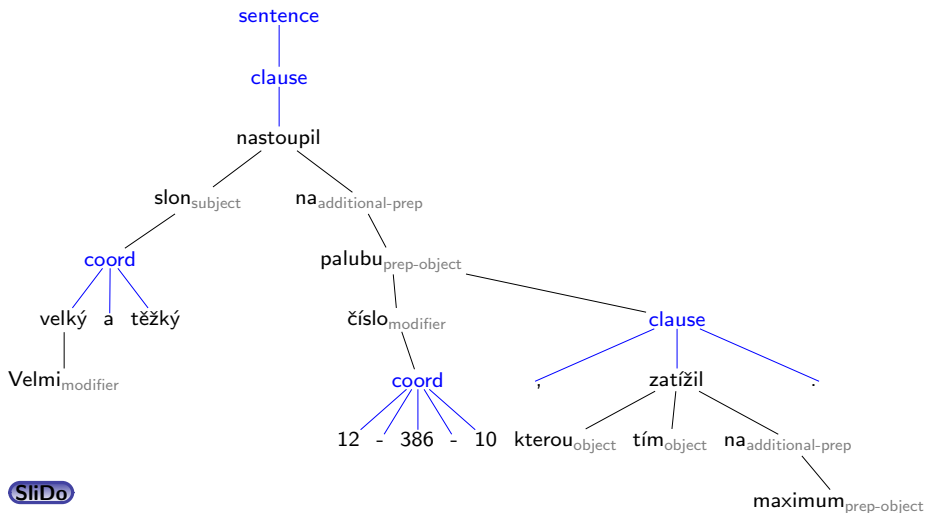
- jen zřídka se používá **čistě** složkový či striktně závislostní přístup
- ve složkovém jsou závislosti zpravidla vyjádřeny přidáním označení, která složka je **řídící** pro danou frázi
- závislostní strom bývá doplněn o informaci určující **lineární precedenci**
- je možné pak mezi těmito přístupy **výsledek převádět**

Složkový a závislostní přístup – pokrač.

- jen zřídka se používá **čistě** složkový či striktně závislostní přístup
- ve složkovém jsou závislosti zpravidla vyjádřeny přidáním označení, která složka je **řídící** pro danou frázi
- závislostní strom bývá doplněn o informaci určující **lineární precedenci**
- je možné pak mezi těmito přístupy **výsledek převádět**

Složkový a závislostní přístup – hybridní stromy

Velmi velký a těžký slon nastoupil na palubu číslo 12-386-10, kterou tím zatížil na maximum.

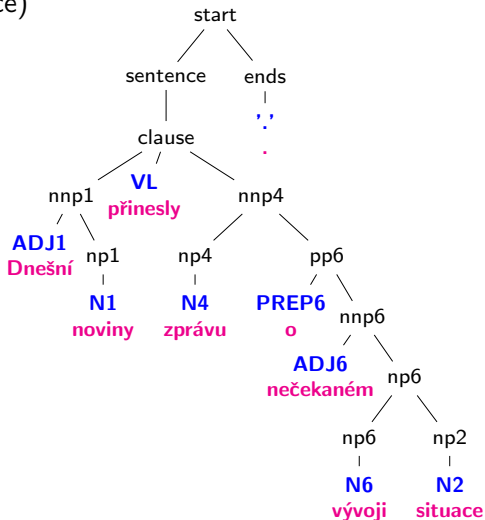


Uzly syntaktického stromu

označení uzlu (název neterminálu) podle zvoleného přístupu reprezentuje:

- **gramatická role** (gramatická funkce)

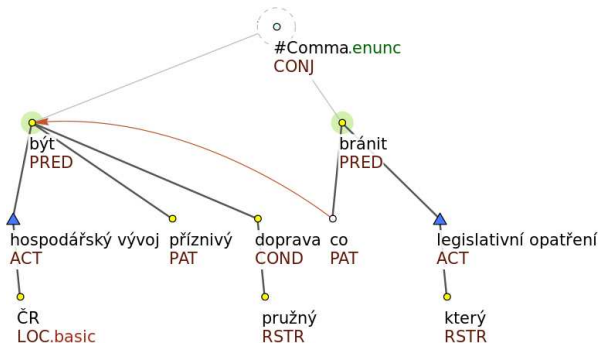
- charakterizují vztahy mezi větnými složkami na povrchové úrovni
- určujeme, zda daný větný člen je NP v roli **podmětu** (1.pád), NP v roli **předmětu** (4.pád), ADVP určující **lokaci** (předl. skupina se 6.pádem) atd.
- v češtině (a jazycích se systémem gramatických pádů) pomáhá k určení gramatické role právě **informace o pádu**
- ovšem přiřazení gramatických rolí ke gramatickým pádům a naopak není zdaleka jednoznačné.



Uzly syntaktického stromu – pokrač.

- **tematická role** (též hloubkový/sémantický pád)
 - na rozdíl od gramatické role se jedná o **sémantickou kategorii**
 - určujeme např.:
 - **Agens** – kdo je životným *původcem* nějaké cílevědomé činnosti
 - **Patiens** – co hraje roli entity, na kterou se *působí*
 - **Donor** – osoba, která *dává*
 - **Cause** – entita, která *způsobuje*, že je něco děláno

Hospodářský vývoj v ČR by mohl být příznivější při pružnější dopravě, v čemž brání některá legislativní opatření.



Příznaky a příznakové struktury

informace v uzlu syntaktického stromu:

- **příznaky/rysy** (*features*) – zaznamenávají **syntaktické nebo sémantické informace** o slovu nebo frázi.

např. **test na shodu**:

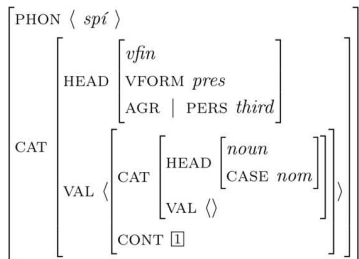
Malý Petr přišel domů.

podmět (Petr) je ve shodě s přísudkem (přišel) v **čísle** a **rodě**
přídavné jméno (malý) a podstatné jméno (Petr) se shodují v **pádě**,
čísle a **rodě**

S(n, g) → NP(-, n, g) VP(n, g)
NP(c, n, g) → ADJ(c, n, g) N(c, n, g)

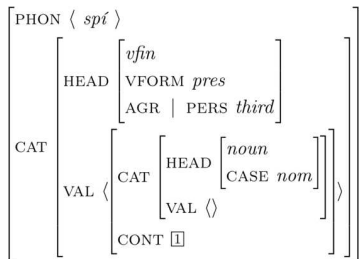
Příznaky a příznakové struktury – pokrač.

- gramatické znaky (slovní druh, gramatický pád, rod, číslo, osoba, ...) je výhodné začlenit do gramatiky ve formě dvojic **atribut–hodnota**
- potom je možné **zobecňovat**, např. vyjádřit shodu v pádě, čísle a rodě výhradně pomocí atributů
- aplikace – v mnoha gramatických formalismech jazykové objekty jsou zde modelovány jako **příznakové struktury** (*feature structures*), tedy právě **matice** dvojic atribut–hodnota.
- u složitějších struktur – nestačí pak běžné porovnání instanciace jde oběma směry → použije se **unifikace**



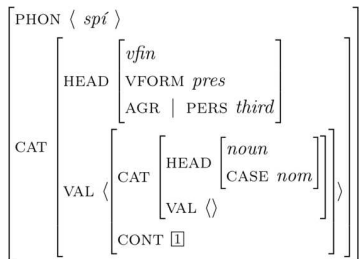
Příznaky a příznakové struktury – pokrač.

- gramatické znaky (slovní druh, gramatický pád, rod, číslo, osoba, ...) je výhodné začlenit do gramatiky ve formě dvojic **atribut–hodnota**
- potom je možné **zobecňovat**, např. vyjádřit shodu v pádě, čísle a rodě výhradně pomocí atributů
- aplikace – v mnoha gramatických formalismech jazykové objekty jsou zde modelovány jako **příznakové struktury** (*feature structures*), tedy právě **matice** dvojic atribut–hodnota.
- u složitějších struktur – nestačí pak běžné porovnání instanciace jde oběma směry → použije se **unifikace**



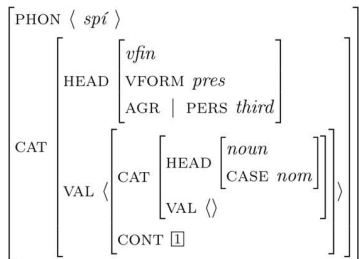
Příznaky a příznakové struktury – pokrač.

- gramatické znaky (slovní druh, gramatický pád, rod, číslo, osoba, ...) je výhodné začlenit do gramatiky ve formě dvojic **atribut–hodnota**
- potom je možné **zobecňovat**, např. vyjádřit shodu v pádě, čísle a rodě výhradně pomocí atributů
- aplikace – v mnoha gramatických formalismech jazykové objekty jsou zde modelovány jako **příznakové struktury** (*feature structures*), tedy právě **matice** dvojic atribut–hodnota.
- u složitějších struktur – nestačí pak běžné porovnání instanciace jde oběma směry → použije se **unifikace**



Příznaky a příznakové struktury – pokrač.

- gramatické znaky (slovní druh, gramatický pád, rod, číslo, osoba, ...) je výhodné začlenit do gramatiky ve formě dvojic **atribut–hodnota**
- potom je možné **zobecňovat**, např. vyjádřit shodu v pádě, čísle a rodě výhradně pomocí atributů
- aplikace – v mnoha gramatických formalismech jazykové objekty jsou zde modelovány jako **příznakové struktury** (*feature structures*), tedy právě **matice** dvojic atribut–hodnota.
- u složitějších struktur – nestačí pak běžné porovnání instanciace jde oběma směry → použije se **unifikace**



Pořádek slov ve větě

syntaktická pozice – standardní pozice větných členů ve větě

angličtina: **S V O M P T**

Subject, Verb, Object, Modus, Place, Temp

- avšak např. předmět se může přesunout na první pozici – **topikalizace**

The book I read.

- v češtině – téměř libovolné přesuny syntaktických elementů souvisí s tzv. **aktuálním větným členěním**

Slido

Pořádek slov ve větě

syntaktická pozice – standardní pozice větných členů ve větě

angličtina: **S V O M P T**

Subject, Verb, Object, Modus, Place, Temp

- avšak např. předmět se může přesunout na první pozici – **topikalizace**

The book I read.

- v češtině – téměř libovolné přesuny syntaktických elementů souvisí s tzv. **aktuálním větným členěním**

Slido

Možnosti zadávání gramatik

- nejčastější formát specifikace gramatik – **produkční pravidla**
gramatika se skládá z pravidel generujících **správně utvořené řetězce**

```
S → NP VP  
NP → Člověk  
VP → myslí
```

- cíl analyzátoru – najít **odvození** vstupního řetězce ze zadaného **neterminálu** (označovaného obvykle velkým písmenem **S** z anglického *sentence* – věta) na základě daných pravidel
- pokud je tohoto cíle dosaženo, vstup je **akceptován** a je mu přiřazena odpovídající **struktura**

Možnosti zadávání gramatik

- nejčastější formát specifikace gramatik – **produkční pravidla**
gramatika se skládá z pravidel generujících **správně utvořené řetězce**

```
S → NP VP  
NP → Člověk  
VP → myslí
```

- cíl analyzátoru – najít **odvození** vstupního řetězce ze zadaného **neterminálu** (označovaného obvykle velkým písmenem **S** z anglického *sentence* – věta) na základě daných pravidel
- pokud je tohoto cíle dosaženo, vstup je **akceptován** a je mu přiřazena odpovídající **struktura**

Možnosti zadávání gramatik

- nejčastější formát specifikace gramatik – **produkční pravidla**
gramatika se skládá z pravidel generujících **správně utvořené řetězce**

```
S → NP VP  
NP → Člověk  
VP → myslí
```

- cíl analyzátoru – najít **odvození** vstupního řetězce ze zadaného **neterminálu** (označovaného obvykle velkým písmenem **S** z anglického *sentence* – věta) na základě daných pravidel
- pokud je tohoto cíle dosaženo, vstup je **akceptován** a je mu přiřazena odpovídající **struktura**

Obsah

- 1 **Syntaxe, syntaktická analýza**
 - Základní termíny
 - Analýza programovacích a přirozených jazyků
 - Gramatiky přirozeného jazyka
- 2 **Specifikace gramatik**
 - Složkový a závislostní přístup
 - Uzly syntaktického stromu
 - Pořádek slov ve větě
 - Možnosti zadávání gramatik
- 3 **Chomského teorie syntaxe**
 - Standardní teorie syntaxe
- 4 **Východiska syntaktické analýzy**
 - Návrh podkladů a datových struktur
 - Grammatical Framework

Standardní teorie syntaxe

- 50. léta 20. stol. – **Noam Chomsky** vytvořil **formální teorii syntaxe**
- jedna ze základních tezí – **autonomie syntaxe**
 - ⇐ k ověření **syntaktické správnosti** věty nepotřebujeme znát její význam

Bezbarvé zelené myšlenky zuřivě spí.

vs.

Spí myšlenky zelené zuřivě bezbarvé.

resp. v angličtině

Colorless green ideas sleep furiously.

vs.

Furiously sleep ideas green colorless.

- syntaktické principy mají **univerzální platnost** pro různé přirozené jazyky

Standardní teorie syntaxe

- 50. léta 20. stol. – **Noam Chomsky** vytvořil **formální teorii syntaxe**
- jedna ze základních tezí – **autonomie syntaxe**
 - ⇐ k ověření **syntaktické správnosti** věty nepotřebujeme znát její význam

Bezbarvé zelené myšlenky zuřivě spí.

vs.

Spí myšlenky zelené zuřivě bezbarvé.

resp. v angličtině

Colorless green ideas sleep furiously.

vs.

Furiously sleep ideas green colorless.

- syntaktické principy mají **univerzální platnost** pro různé přirozené jazyky

Standardní teorie syntaxe

- 50. léta 20. stol. – **Noam Chomsky** vytvořil **formální teorii syntaxe**
- jedna ze základních tezí – **autonomie syntaxe**
 - ⇐ k ověření **syntaktické správnosti** věty nepotřebujeme znát její význam

Bezbarvé zelené myšlenky zuřivě spí.

vs.

Spí myšlenky zelené zuřivě bezbarvé.

resp. v angličtině

Colorless green ideas sleep furiously.

vs.

Furiously sleep ideas green colorless.

- syntaktické principy mají **univerzální platnost** pro různé přirozené jazyky

Chomského standardní teorie syntaxe

znalost jazyka = gramatika

Chomského předpoklady o rozumu:

- rozum má *vrozenou strukturu*
- rozum je *modulární*
- rozum obsahuje speciální modul pro *jazyk*
porozumění jazyku je oddělitelné od jiných aktivit
- syntaxe je *formální*
nezávislá na významu a komunikačních funkcích
- znalost jazyka je *modulární*
obsahuje moduly pro jednotlivé fáze analýzy jazyka

Chomského standardní teorie syntaxe

znalost jazyka = gramatika

Chomského předpoklady o rozumu:

- rozum má *vrozenou strukturu*
- rozum je *modulární*
- rozum obsahuje speciální modul pro *jazyk*
porozumění jazyku je oddělitelné od jiných aktivit
- syntaxe je *formální*
nezávislá na významu a komunikačních funkcích
- znalost jazyka je *modulární*
obsahuje moduly pro jednotlivé fáze analýzy jazyka

Chomského standardní teorie syntaxe

znalost jazyka = gramatika

Chomského předpoklady o rozumu:

- rozum má *vrozenou strukturu*
- rozum je *modulární*
- rozum obsahuje speciální modul pro *jazyk*
porozumění jazyku je oddělitelné od jiných aktivit
- syntaxe je *formální*
nezávislá na významu a komunikačních funkcích
- znalost jazyka je *modulární*
obsahuje moduly pro jednotlivé fáze analýzy jazyka

Chomského standardní teorie syntaxe

znalost jazyka = gramatika

Chomského předpoklady o rozumu:

- rozum má *vrozenou strukturu*
- rozum je *modulární*
- rozum obsahuje speciální modul pro *jazyk*
porozumění jazyku je oddělitelné od jiných aktivit
- syntaxe je *formální*
nezávislá na významu a komunikačních funkcích
- znalost jazyka je *modulární*
obsahuje moduly pro jednotlivé fáze analýzy jazyka

Chomského standardní teorie syntaxe

znalost jazyka = gramatika

Chomského předpoklady o **rozumu**:

- rozum má *vrozenou strukturu*
- rozum je *modulární*
- rozum obsahuje speciální modul pro *jazyk*
porozumění jazyku je oddělitelné od jiných aktivit
- syntaxe je *formální*
nezávislá na významu a komunikačních funkcích
- znalost jazyka je *modulární*
obsahuje moduly pro jednotlivé fáze analýzy jazyka

Standardní teorie syntaxe – pokrač.

- Noam Chomsky, **Aspects of the Theory of Syntax**, 1965 – standardní teorie syntaxe – **transformační generativní gramatika** (TGG)
- snaží se řešit i zachycení sémantických vztahů v **hloubkové struktuře**
- postupně se vyvinula:
 - v **rozšířenou standardní teorii** (1968)
 - později tzv. **Government & Binding Theory** (teorie nadřazení a vázání, 1981), která zakládá na pojmu *univerzální gramatiky*
 - 90. léta – teorie **minimalismu** (snaha po úspornosti popisného aparátu)

Standardní teorie syntaxe – pokrač.

- Noam Chomsky, **Aspects of the Theory of Syntax**, 1965 – standardní teorie syntaxe – **transformační generativní gramatika** (TGG)
- snaží se řešit i zachycení sémantických vztahů v **hloubkové struktuře**
- postupně se vyvinula:
 - v **rozšířenou standardní teorii** (1968)
 - později tzv. **Government & Binding Theory** (teorie nadřazení a vázání, 1981), která zakládá na pojmu *univerzální gramatiky*
 - 90. léta – teorie **minimalismu** (snaha po úspornosti popisného aparátu)

Standardní teorie syntaxe – pokrač.

- Noam Chomsky, **Aspects of the Theory of Syntax**, 1965 – standardní teorie syntaxe – **transformační generativní gramatika** (TGG)
- snaží se řešit i zachycení sémantických vztahů v **hloubkové struktuře**
- postupně se vyvinula:
 - v **rozšířenou standardní teorii** (1968)
 - později tzv. **Government & Binding Theory** (teorie nadřazení a vázání, 1981), která zakládá na pojmu *univerzální gramatiky*
 - 90. léta – teorie **minimalismu** (snaha po úspornosti popisného aparátu)

Standardní teorie syntaxe – pokrač.

základní části standardní teorie:

- **bázová komponenta**
 - bezkontextová **pravidla** a schémata pravidel generují základní strukturu větných členů
 - **lexikon** popisuje lexikální kategorie a syntaktické rysy lexikálních položek
- **transformační pravidla** – vložení, smazání, přesun, změna-rysu, kopie-rysu
transformace převádí hloubkové struktury na struktury povrchové

Příklad bázevých komponentů

pravidla:

$S \rightarrow NP VP$

$NP \rightarrow (D) A^* N PP^*$

$VP \rightarrow V (NP) (PP)$

$PP \rightarrow P NP$

lexikon:

D: ten, ta

A: velký, hnědý, starý

N: pták, psem, lovec, já, lesa

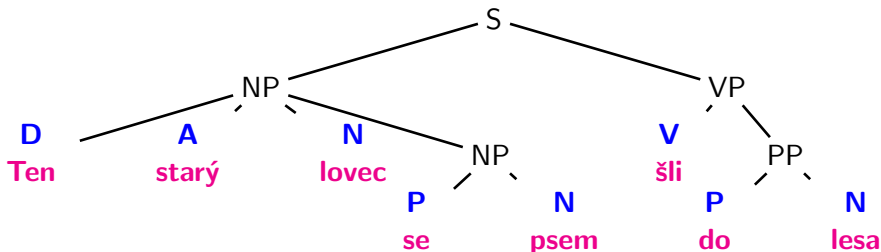
V: loví, jí, šli

P: se, do

věta:

Ten starý lovec se psem šli do lesa.

syntaktický strom:



Příklad transformačních pravidel

např. **pasivizace** (v angličtině):

John chose a book.

NP1 – V – NP2

1 – 2 – 3 → 3 – 2+be+en – by+1

přesuny + vložení + změny-rysu

- transformace:
 - obligatorní – např. přesun slovesné koncovky za sloveso
 - fakultativní – např. pasivizace, tvorba otázek, negace (změna významu)
- pravidla bázevých komponenty – popisují strom hloubkové struktury v obvyklém pořadí
- transformace umožňují jeho změny na různé povrchové varianty (trpný rod, otázka, ...)
- stopa (*trace*) – ukazuje, kde byl prvek před přemístěním

Příklad transformačních pravidel

např. **pasivizace** (v angličtině):

John chose a book.

NP1 – V – NP2

1 – 2 – 3 → 3 – 2+be+en – by+1

přesuny + vložení + změny-rysu

- transformace:
 - **obligatorní** – např. přesun slovesné koncovky za sloveso
 - **fakultativní** – např. pasivizace, tvorba otázek, negace (změna významu)
- pravidla báze komponenty – popisují strom hloubkové struktury v obvyklém pořadí
- transformace umožňují jeho změny na různé povrchové varianty (trpný rod, otázka, ...)
- **stopa** (*trace*) – ukazuje, kde byl prvek před přemístěním

Příklad transformačních pravidel

např. **pasivizace** (v angličtině):

John chose a book.

NP1 – V – NP2

1 – 2 – 3 → 3 – 2+be+en – by+1

přesuny + vložení + změny-rysu

- transformace:
 - **obligatorní** – např. přesun slovesné koncovky za sloveso
 - **fakultativní** – např. pasivizace, tvorba otázek, negace (změna významu)
- pravidla bázevých komponenty – popisují strom hloubkové struktury v obvyklém pořadí
- transformace umožňují jeho změny na různé povrchové varianty (trpný rod, otázka, ...)
- **stopa** (*trace*) – ukazuje, kde byl prvek před přemístěním

Příklad transformačních pravidel

např. **pasivizace** (v angličtině):

John chose a book.

NP1 – V – NP2

1 – 2 – 3 → 3 – 2+be+en – by+1

přesuny + vložení + změny-rysu

- transformace:
 - **obligatorní** – např. přesun slovesné koncovky za sloveso
 - **fakultativní** – např. pasivizace, tvorba otázek, negace (změna významu)
- pravidla bázevých komponenty – popisují strom hloubkové struktury v obvyklém pořadí
- transformace umožňují jeho změny na různé povrchové varianty (trpný rod, otázka, ...)
- **stopa** (*trace*) – ukazuje, kde byl prvek před přemístěním

Příklad transformačních pravidel

např. **pasivizace** (v angličtině):

John chose a book.

NP1 – V – NP2

1 – 2 – 3 → 3 – 2+be+en – by+1

přesuny + vložení + změny-rysu

- transformace:
 - **obligatorní** – např. přesun slovesné koncovky za sloveso
 - **fakultativní** – např. pasivizace, tvorba otázek, negace (změna významu)
- pravidla bázevých komponenty – popisují strom hloubkové struktury v obvyklém pořadí
- transformace umožňují jeho změny na různé povrchové varianty (trpný rod, otázka, ...)
- **stopa** (*trace*) – ukazuje, kde byl prvek před přemístěním

Obsah

- 1 Syntaxe, syntaktická analýza
 - Základní termíny
 - Analýza programovacích a přirozených jazyků
 - Gramatiky přirozeného jazyka
- 2 Specifikace gramatik
 - Složkový a závislostní přístup
 - Uzly syntaktického stromu
 - Pořádek slov ve větě
 - Možnosti zadávání gramatik
- 3 Chomského teorie syntaxe
 - Standardní teorie syntaxe
- 4 Východiska syntaktické analýzy
 - Návrh podkladů a datových struktur
 - Grammatical Framework

Návrh podkladů a datových struktur

- **syntaktický strom** – kompletní **hierarchický popis struktury** věty
- **úkol syntaktické analýzy** = pro danou gramatiku a daný vstup (větu) dát **všechny syntaktické stromy**
- existují techniky pro kompaktní uložení **lesa** takových stromů (chart parsing)
- jelikož se zabýváme výhradně syntaktickou strukturou a nevylučujeme a priori strukturní stromy s absurdní interpretací, má většina vět mnoho různých syntaktických stromů

Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.

Pocet uspesnych stromu = 3 610 464

Návrh podkladů a datových struktur

- **syntaktický strom** – kompletní **hierarchický popis struktury** věty
- **úkol syntaktické analýzy** = pro danou gramatiku a daný vstup (větu) dát **všechny syntaktické stromy**
- existují techniky pro kompaktní uložení **lesa** takových stromů (chart parsing)
- jelikož se zabýváme výhradně syntaktickou strukturou a nevylučujeme a priori strukturní stromy s absurdní interpretací, má většina vět mnoho různých syntaktických stromů

Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.

Pocet uspesnych stromu = 3 610 464

Návrh podkladů a datových struktur

- **syntaktický strom** – kompletní **hierarchický popis struktury** věty
- **úkol syntaktické analýzy** = pro danou gramatiku a daný vstup (větu) dát **všechny syntaktické stromy**
- existují techniky pro kompaktní uložení **lesa** takových stromů (chart parsing)
- jelikož se zabýváme výhradně syntaktickou strukturou a nevylučujeme a priori strukturní stromy s absurdní interpretací, má většina vět mnoho různých syntaktických stromů

Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.

Pocet uspesnych stromu = 3 610 464

Návrh podkladů a datových struktur

- **syntaktický strom** – kompletní **hierarchický popis struktury** věty
- **úkol syntaktické analýzy** = pro danou gramatiku a daný vstup (větu) dát **všechny syntaktické stromy**
- existují techniky pro kompaktní uložení **lesa** takových stromů (chart parsing)
- jelikož se zabýváme výhradně syntaktickou strukturou a nevylučujeme a priori strukturní stromy s absurdní interpretací, má většina vět mnoho různých syntaktických stromů

Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.

Pocet uspesnych stromu = 3 610 464

Návrh podkladů a datových struktur – pokrač.

Automatická analýza syntaxe musí vždy projít třemi fázemi:

1. musí být zvolena notace pro zápis gramatiky – **gramatický formalismus**
2. musí být ve zvoleném formalismu napsána **gramatika** pro každý jazyk, který bude zpracováván
3. musí být vybrán nebo navržen **algoritmus**, který určí, zda daný vstup odpovídá gramatice, a pokud ano, jaký popis mu odpovídá

Návrh podkladů a datových struktur – pokrač.

Automatická analýza syntaxe musí vždy projít třemi fázemi:

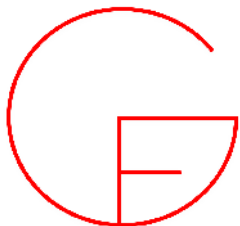
1. musí být zvolena notace pro zápis gramatiky – **gramatický formalismus**
2. musí být ve zvoleném formalismu napsána **gramatika** pro každý jazyk, který bude zpracováván
3. musí být vybrán nebo navržen **algoritmus**, který určí, zda daný vstup odpovídá gramatice, a pokud ano, jaký popis mu odpovídá

Návrh podkladů a datových struktur – pokrač.

Automatická analýza syntaxe musí vždy projít třemi fázemi:

1. musí být zvolena notace pro zápis gramatiky – **gramatický formalismus**
2. musí být ve zvoleném formalismu napsána **gramatika** pro každý jazyk, který bude zpracováván
3. musí být vybrán nebo navržen **algoritmus**, který určí, zda daný vstup odpovídá gramatice, a pokud ano, jaký popis mu odpovídá

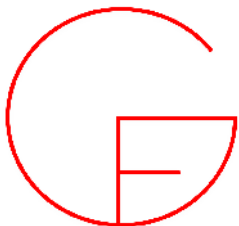
Grammatical Framework



www.grammaticalframework.org

- odděluje **abstraktní** a **konkrétní** gramatiky
 - návrh gramatik **desítek jazyků**
 - popis gramatiky využívá pro **analýzu** i **generování** (tzv. *linearizace*)
 - abstraktní gramatika může sloužit jako **interlingua** při překladu desítky navazujících aplikací
- <http://www.grammaticalframework.org/demos/>

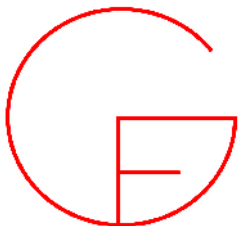
Grammatical Framework



www.grammaticalframework.org

- odděluje **abstraktní** a **konkrétní** gramatiky
 - návrh gramatik **desítek jazyků**
 - popis gramatiky využívá pro **analýzu** i **generování** (tzv. *linearizace*)
 - abstraktní gramatika může sloužit jako **interlingua** při překladu desítky navazujících aplikací
- <http://www.grammaticalframework.org/demos/>

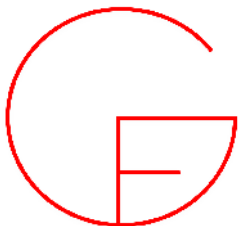
Grammatical Framework



www.grammaticalframework.org

- odděluje **abstraktní** a **konkrétní** gramatiky
 - návrh gramatik **desítek jazyků**
 - popis gramatiky využívá pro **analýzu** i **generování** (tzv. *linearizace*)
 - abstraktní gramatika může sloužit jako **interlingua** při překladu desítky navazujících aplikací
- <http://www.grammaticalframework.org/demos/>

Grammatical Framework



www.grammaticalframework.org

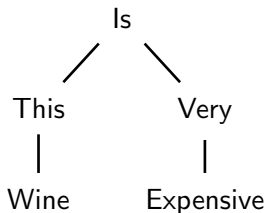
- odděluje **abstraktní** a **konkrétní** gramatiky
- návrh gramatik **desítek jazyků**
- popis gramatiky využívá pro **analýzu** i **generování** (tzv. *linearizace*)
- abstraktní gramatika může sloužit jako **interlingua** při překladu desítky navazujících aplikací

<http://www.grammaticalframework.org/demos/>

Grammatical Framework – abstraktní gramatika

```
1 abstract Food = {  
2  
3   cat Kind;  
4   fun Wine : Kind;  
5   fun Cheese : Kind;  
6   fun Fish : Kind;  
7  
8   cat Item;  
9   fun The : Kind -> Item;  
10  fun This : Kind -> Item;  
11  
12  cat Quality;  
13  fun Delicious : Quality;  
14  fun Expensive : Quality;  
15  fun Fresh : Quality;  
16  fun Very : Quality -> Quality;  
17  
18  cat Phrase;  
19  fun Is : Item -> Quality -> Phrase;  
20  
21  flags startcat = Phrase;  
22 }
```

Is (This Wine) (Very Expensive)



Grammatical Framework – konkrétní gramatika Eng

```
1 concrete FoodEng of Food = {
2
3   lincat Kind = {s : Str};
4   lin Wine = {s = "wine"};
5   lin Cheese = {s = "cheese"};
6   lin Fish = {s = "fish"};
7
8   lincat Item = {s : Str};
9   lin The kind = {s = "the" ++ kind.s};
10  lin This kind = {s = "this" ++ kind.s};
11
12  lincat Quality = {s : Str};
13  lin Delicious = {s = "delicious"};
14  lin Expensive = {s = "expensive"};
15  lin Fresh = {s = "fresh"};
16  lin Very quality = {s = "very" ++ quality.s};
17
18  lincat Phrase = {s : Str};
19  lin Is item quality = {s = item.s ++ "is" ++ quality.s};
20
21 }
```

Grammatical Framework – konkrétní gramatika CZ

```
1 concrete FoodCze of Food = {
2
3   param Gender = Masc | Fem | Neut;
4
5   lincat Kind = {s : Str; g : Gender};
6   lin Wine = {s = "vino"; g = Neut};
7   lin Cheese = {s = "sýr"; g = Masc};
8   lin Fish = {s = "ryba"; g = Fem};
9
10  lincat Item = {s : Str; g : Gender};
11  lin The kind = {
12    s = case kind.g of {Masc => "ten"; Fem => "ta"; Neut => "to"} ++ kind.s;
13    g = kind.g
14  };
15  lin This kind = {
16    s = case kind.g of {Masc => "tento"; Fem => "tato"; Neut => "toto"} ++ kind.s;
17    g = kind.g
18  };
19
20  lincat Quality = {s : Gender => Str};
21  lin Delicious = {
22    s = table {Masc => "dobrý"; Fem => "dobrá"; Neut => "dobré"}
23  };
24  lin Expensive = {
25    s = table {Masc => "drahý"; Fem => "drahá"; Neut => "drahé"}
26  };
27  lin Fresh = {
28    s = table {Masc => "čerstvý"; Fem => "čerstvá"; Neut => "čerstvé"}
29  };
30  lin Very quality = {
31    s = table {g => "velmi"} ++ quality.s!g}
32  };
33
34  lincat Phrase = {s : Str};
35  lin Is item quality = {s = item.s ++ "je" ++ quality.s!item.g};
```

Grammatical Framework – překlad

```
> import Food.gr
linking ... OK

Food> import FoodEng.gf
linking ... OK

Languages: FoodEng
0 msec
Food> import FoodCze.gf
linking ... OK

Languages: FoodCze FoodEng
4 msec
Food> linearize Is (This Cheese) Delicious
tento sýr je dobrý
this cheese is delicious

4 msec
Food> parse -lang=Eng "this wine is expensive" | linearize -lang=Cze
toto víno je velmi drahé
```

viz možnosti např. na www.phrasomatic.net