# Prague Dependency Treebank:
## Introduction – (Non-)Projectivity

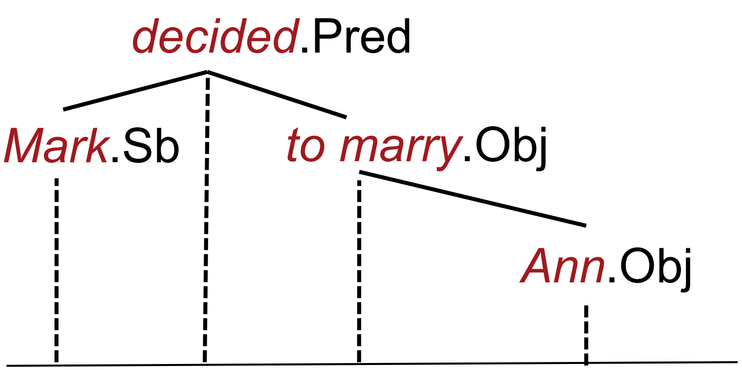

Markéta Lopatková, Jiří Mírovský

Institute of Formal and Applied Linguistics, MFF UK

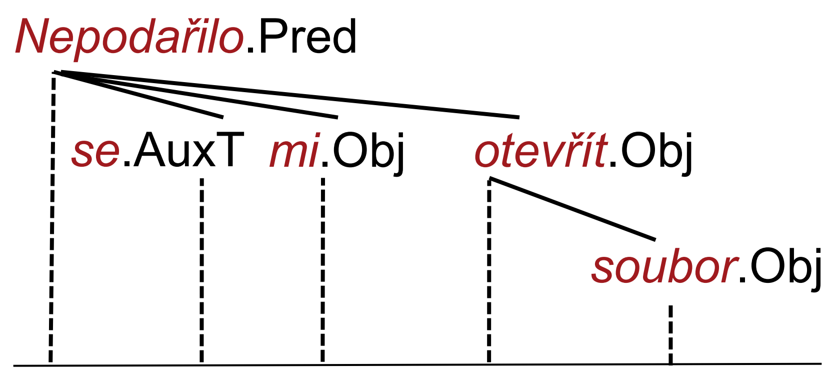
lopatkova@ufal.mff.cuni.cz

# Projectivity and non-projectivity (definition)

Mark decided to marry Ann.




*Nepodařilo se mi otevřít soubor.*



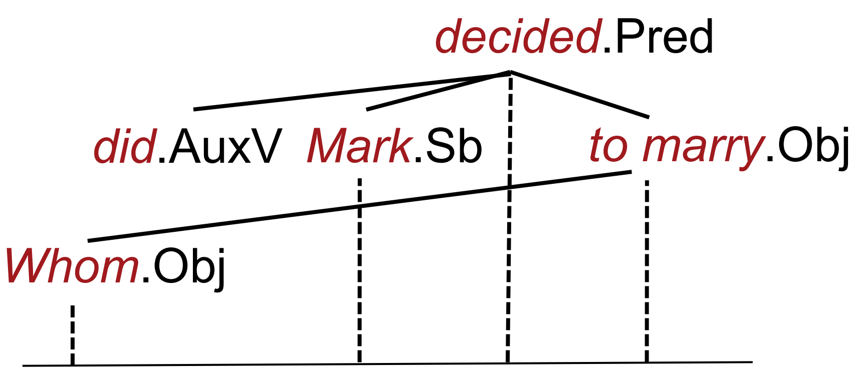
# Projectivity and non-projectivity (definition)

Whom did Mark decided to marry?

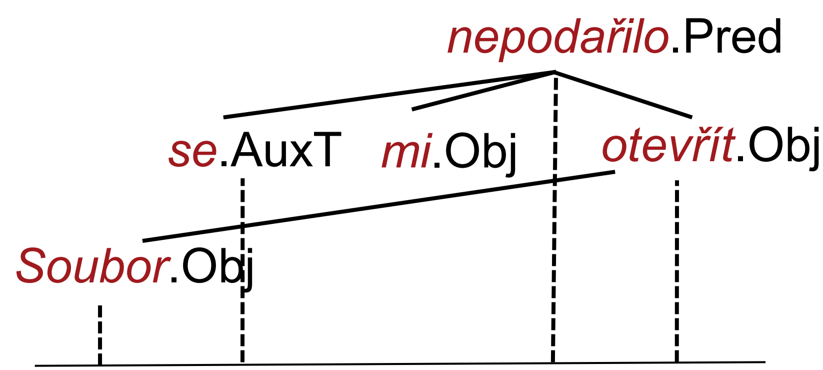
*Soubor se mi nepodařilo otevřít.* (Oliva)

# Projectivity and non-projectivity (definition)

Whom did Mark decided to marry?




*Soubor se mi nepodařilo otevřít.* (Oliva)



# Projectivity and non-projectivity (definition)

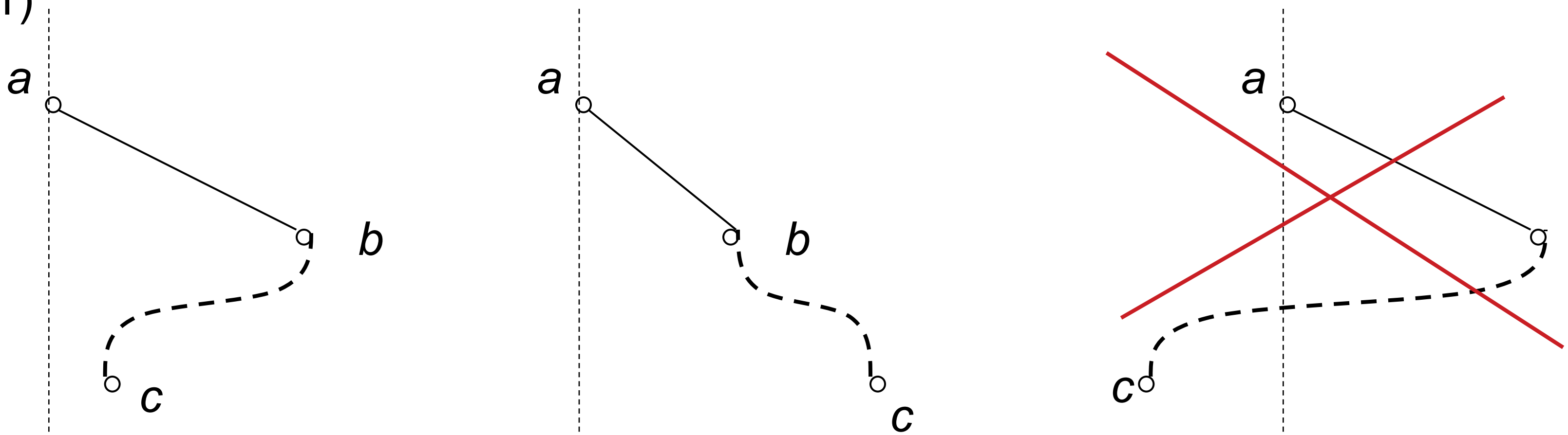
A subtree *S* of a rooted dependency tree *T* is *projective* iff for all nodes *a*, *b* and *c* of the subtree *S* the condition holds:

(1) $(a \leq_D b)\ \&\ (a <_{WO} b)\ \&\ (b \leq_D^* c) \Rightarrow (a <_{WO} c)$

<u>and</u>

(2) $(a \leq_D b)\ \&\ (b <_{WO} a)\ \&\ (b \leq_D^* c) \Rightarrow (c <_{WO} a)$

(1)



# Projectivity and free word order

## free word order:

- freedom of word order of dependents within a continuous 'head domain' (i.e., substring of head + its dependents)

# Projectivity and free word order

## free word order:

- freedom of word order of dependents within a continuous ‘head domain’ (i.e., substring of head + its dependents)
- relaxation of continuity of a head domain

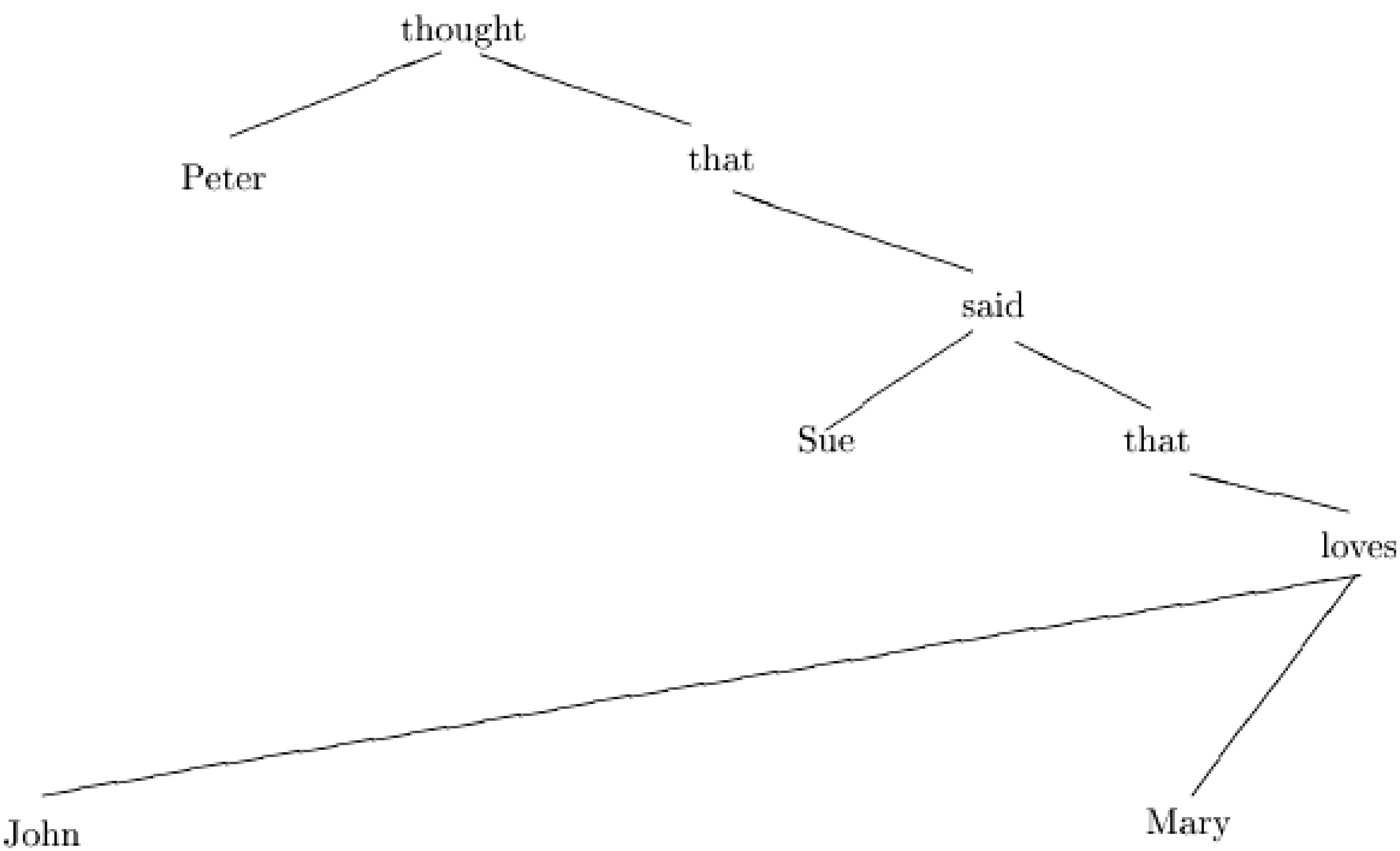
German:
*Maria hat einen Mann kennengelernt der Schmetterlinge sammelt.*
Mary - has - a man - met - the butteries - collects
Mary has met a man who collects butteries

# Projectivity and free word order

English: long-distance unbounded dependency
*John, Peter thought that Sue said that Mary loves.*

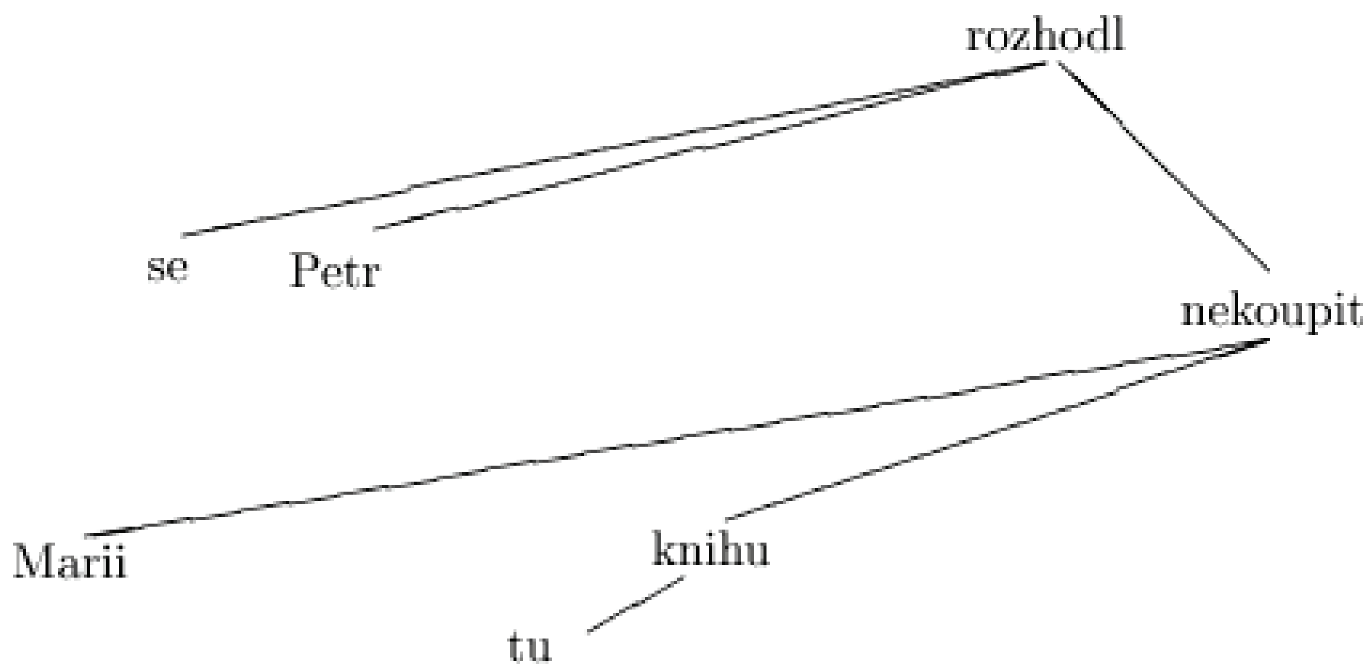
# Projectivity and free word order

English: long-distance unbounded dependency
*John, Peter thought that Sue said that Mary loves.*



# Projectivity and free word order

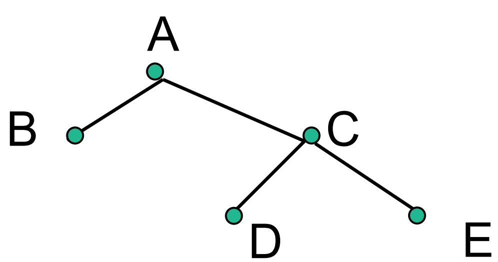
Czech:
*Marii se Petr tu knihu rozhodl nekoupit.*
to-Mary PART Peter that book decided not-buy
[Peter decided not to buy that book to Mary.]



# Projectivity and non-projectivity

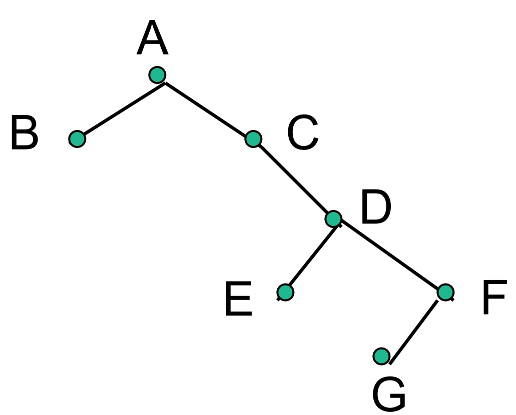
Projective dependency trees can be encoded by *linearization*:

- string of nodes, edges ~ brackets




A ( B C ( D) )  without WO ordering

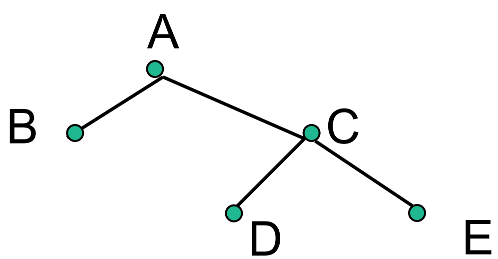
( B ) A ( ( D ) C  ( E ) )  with WO



# Projectivity and non-projectivity

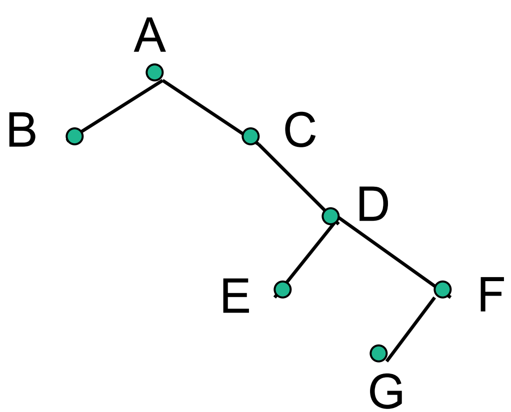
Projective dependency trees can be encoded by *linearization*:

- string of nodes, edges ~ brackets




A ( B C ( D E ) )   without WO ordering

( B ) A ( ( D ) C ( E ) )   with WO




A ( B C ( D ( E F ( G ) ) ) )   without WO

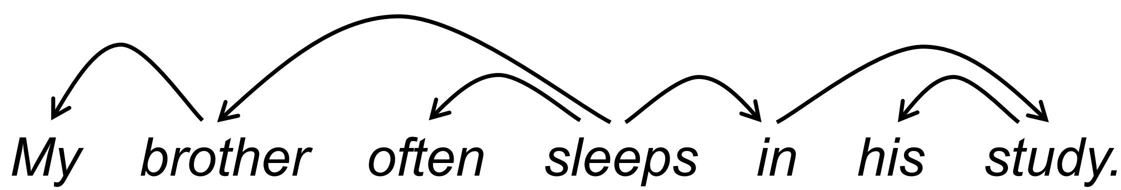
( B ) A ( C ( ( E ) D ( ( G ) F ) ) )   with WO

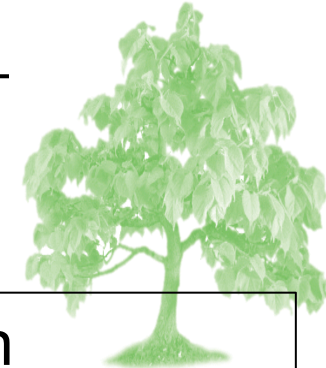
# Planarity

A dependency graph *T* is *planar*, if it does not contain nodes *a, b, c, d* such that:

$$linked(a,c) \;\;\&\;\; linked(b,d) \;\;\&\;\; a <_{WO} b <_{WO} c <_{WO} d$$

*linked(i,j)* … ‘there is an edge in *T* from *i* to *j*, or vice versa’



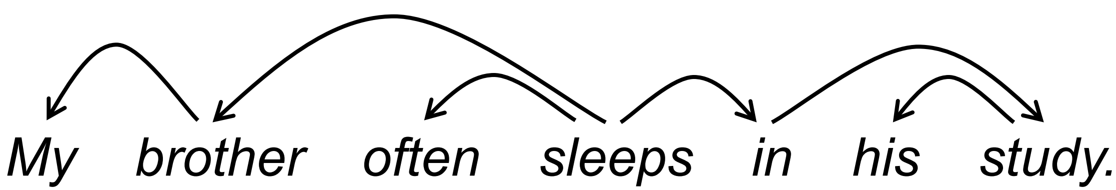


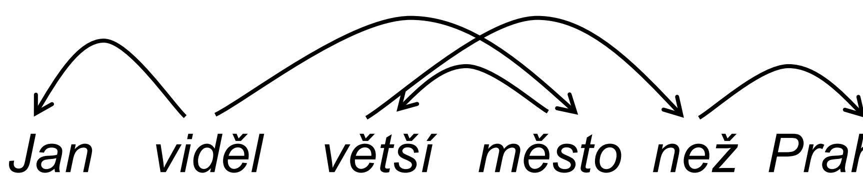
# Planarity

A dependency graph *T* is *planar*, if it does not contain nodes *a, b, c, d* such that:

$$linked(a,c) \ \& \ linked(b,d) \quad \& \ a <_{WO} b <_{WO} c <_{WO} d$$

*linked(i,j)* … ‘there is an edge in *T* from *i* to *j*, or vice versa’




*Jan - saw - bigger - city - than – Prague*

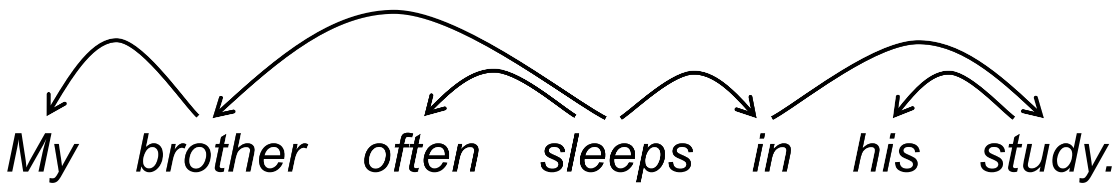
*Jan saw a city bigger than Prague.*

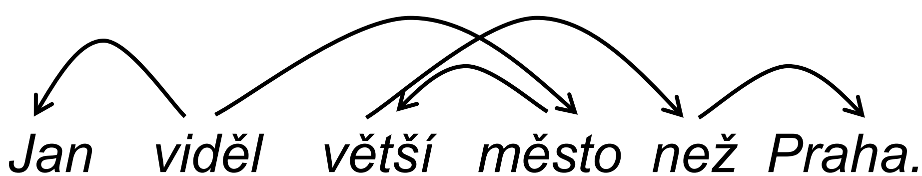
# Planarity

A dependency graph *T* is *planar*, if it does *not* contain nodes *a, b, c, d* such that:

$$linked(a,c) \;\; \& \;\; linked(b,d) \;\; \& \;\; a <_{WO} b <_{WO} c <_{WO} d$$

*linked(i,j)* … ‘there is an edge in *T* from *i* to *j*, or vice versa’

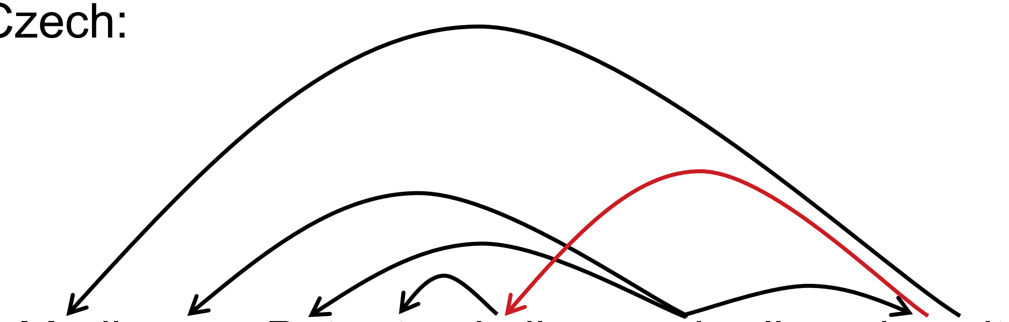



Informally, a dependency graph is planar, if its edges can be drawn above the sentence without crossing.

# Projectivity and free word order

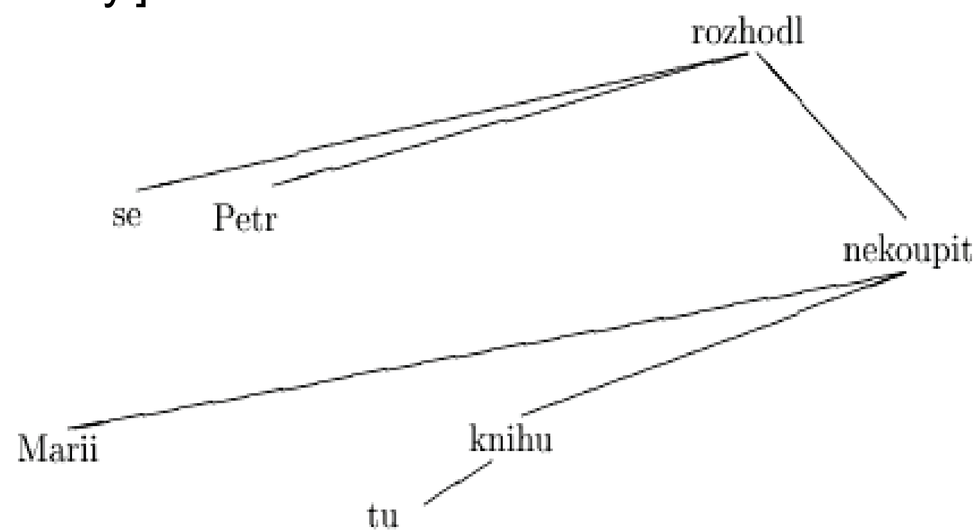
Czech:



*Marii se Petr tu knihu rozhodl nekoupit.*

to-Mary PART Peter that book decided not-buy

[Peter decided not to buy that book to Mary.]

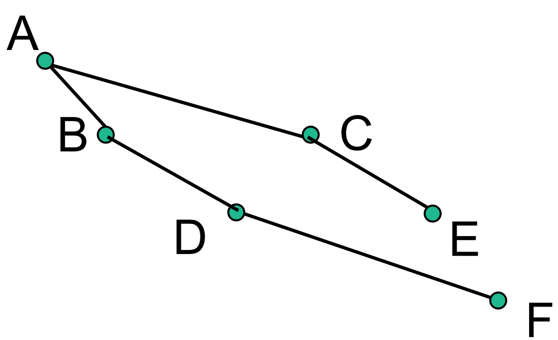


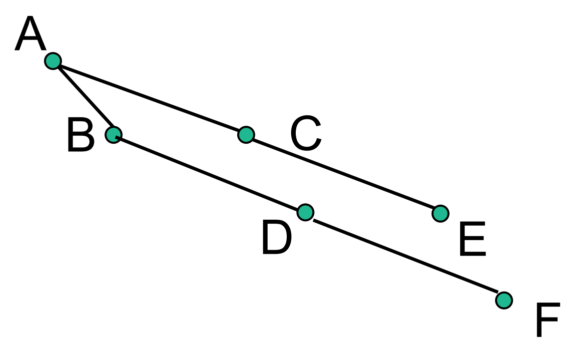
# ‘Well-Nestedness’

Two subtrees $T_1$, $T_2$ *interleave*, if there are nodes $l_1, r_1 \in T_1$ and $l_2, r_2 \in T_2$ such that

$$l_1 <_{WO} l_2 <_{WO} r_1 <_{WO} r_2$$

A dependency graph is *well-nested*, if no two of its disjoint subtrees interleave.’



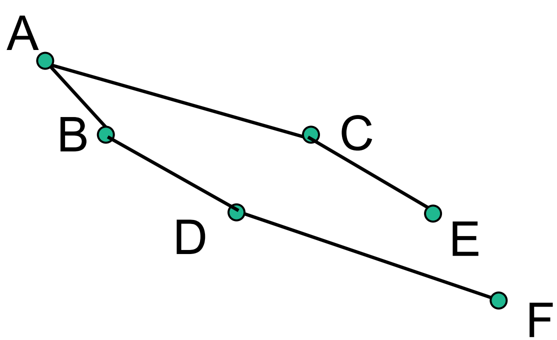



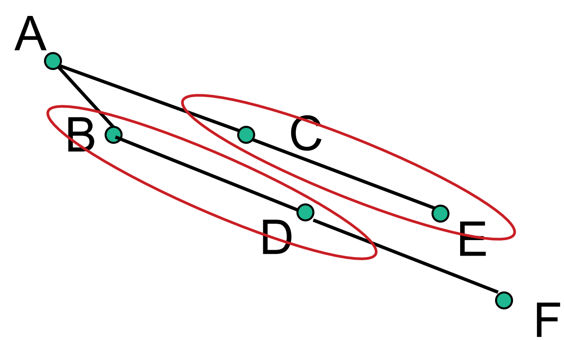
# 'Well-Nestedness'

Two subtrees $T_1$, $T_2$ *interleave*, if there are nodes $l_1, r_1 \in T_1$ and $l_2, r_2 \in T_2$ such that

$$l_1 <_{WO} l_2 <_{WO} r_1 <_{WO} r_2$$

A dependency graph is *well-nested*, if no two of its disjoint subtrees interleave.'



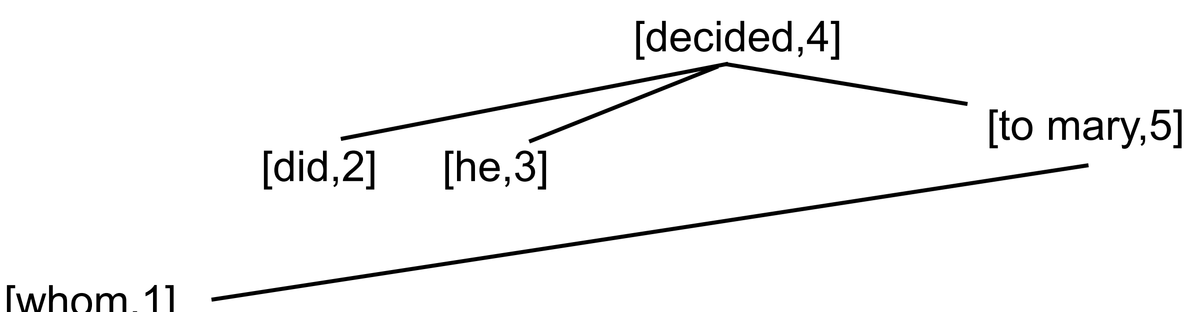



# Gap Degree *dNh(T)*

*Coverage* of a node $u \in T$

$Cov(u,T) = \{\, i \mid i$ - word order position of $v \in T$ such that, $u \leq_D v \,\}$

$Cov(u_1,T)=\{1\}$;   $Cov(u_2,T)=\{2\}$;   $Cov(u_3,T)=\{3\}$;   $Cov(u_4,T)=\{1,2,3,4,5\}$;   $Cov(u_5,T) = \{1,5\}$

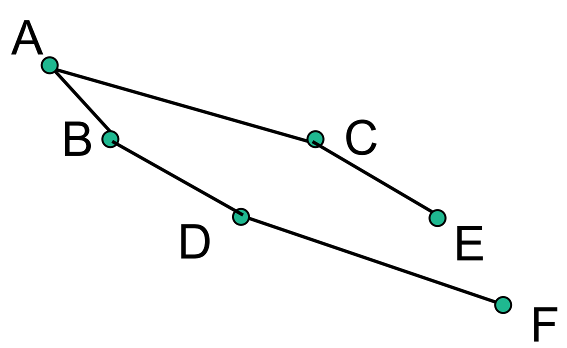


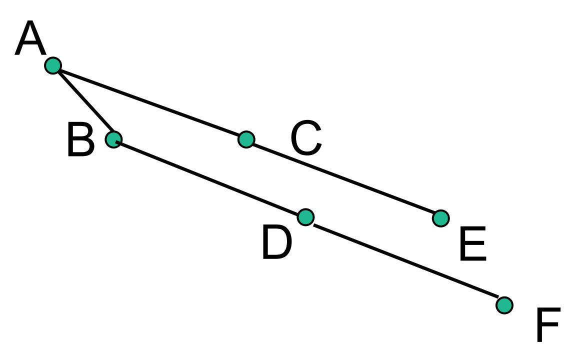
# Gap Degree *dNh(T)*

*Coverage* of a node $u \in T$

$Cov(u,T) = \{\ i \mid i$ - word order position of $v \in T$ such that, $u \leq_D v\ \}$

*Gap in Coverage* of a node $u \in T$ $\Leftrightarrow_{def}$ $Cov(u,T)$ is not an interval



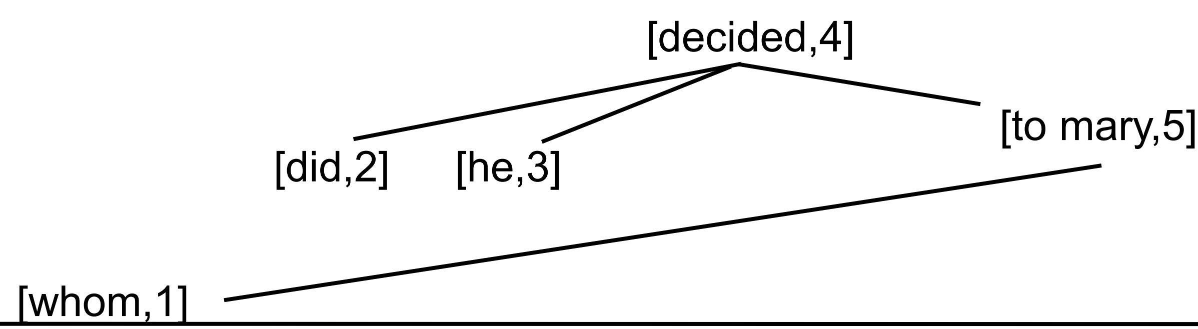



# Gap Degree *dNh(T)*

Coverage of a node $u \in T$

$Cov(u,T)$ = { $i$ | $i$ - word order position of $v \in T$ such that, $u \leq_D v$ }

$Cov(u_1,T)=\{ 1\}$; $Cov(u_2,T)=\{2\}$; $Cov(u_3,T)=\{3\}$; $Cov(u_4,T)=\{1,2,3,4,5\}$; $Cov(u_5,T) = \{1,5\}$



# Gap Degree *dNh(T)*

Coverage of a node $u \in T$

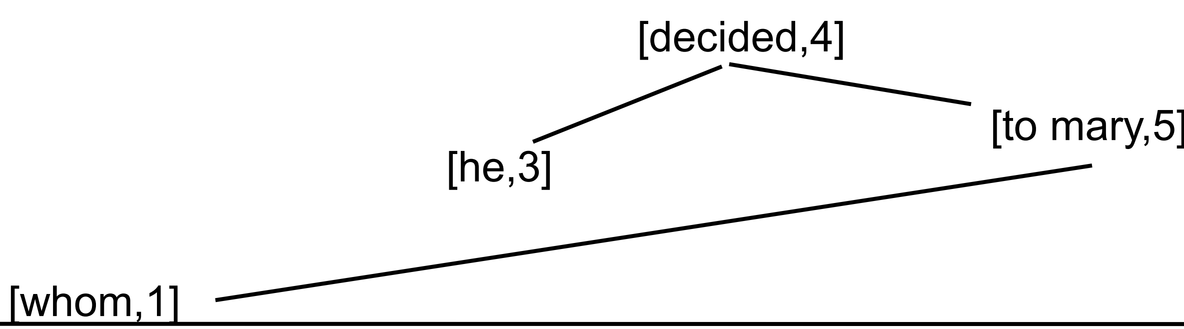
$Cov(u,T)$ = { $i$ | $i$ - word order position of $v \in T$ such that, u $\leq_D$ v }

Gap in Coverage of a node $u \in T$ $\Leftrightarrow_{def}$ $Cov(u,T)$ is not an interval

$dNh(u,T)$ … *number of Gaps* in $Cov(u,T)$

*Tree Gegree Degree* $dNh(T)$ = max {$dNh(u,T)$| $u \in T$ }

$Cov(u_1,T)=\{1\}$; $Cov(u_2,T)=\{2\}$; $Cov(u_3,T)=\{3\}$; $Cov(u_4,T)=\{1,2,3,4,5\}$; $Cov(u_5,T) = \{1,5\}$



# Gap Degree *dNh(T)*

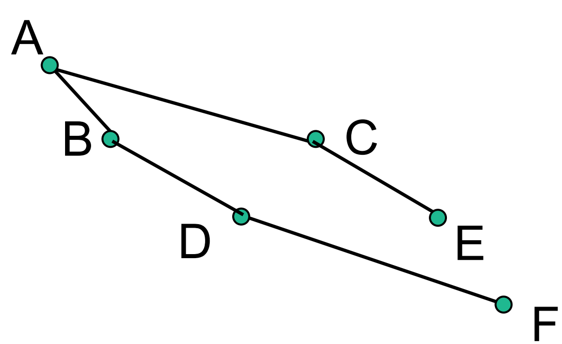
*Coverage* of a node $u \in T$

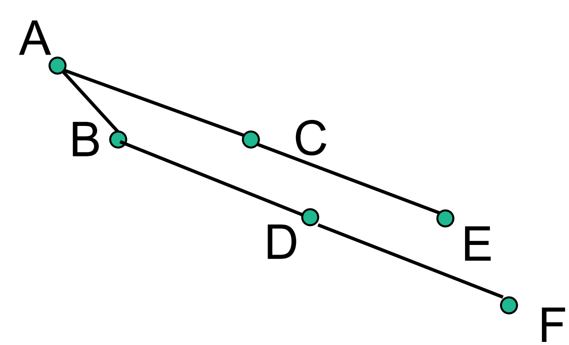
$Cov(u,T)$ = { $i$ | $i$ - word order position of $v \in T$ such that, u $\leq_D$ v }

*Gap in Coverage* of a node $u \in T$ $\Leftrightarrow_{def}$ $Cov(u,T)$ is not an interval

$dNh(u,T)$ … *number of Gaps* in $Cov(u,T)$

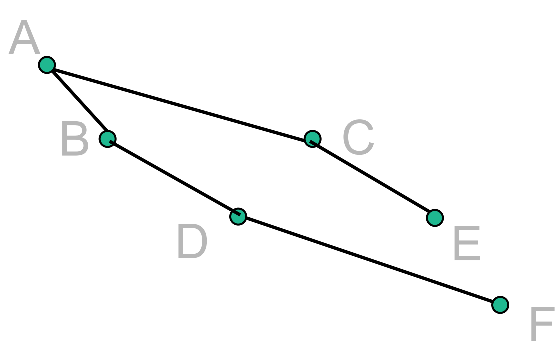
*Tree Gegree Degree* $dNh(T)$ = max $\{dNh(u,T) | u \in T\}$

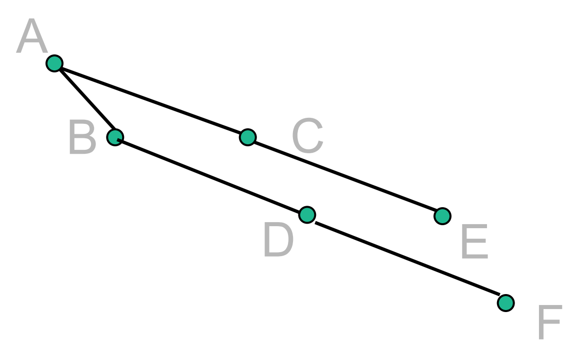


# Edge Degree

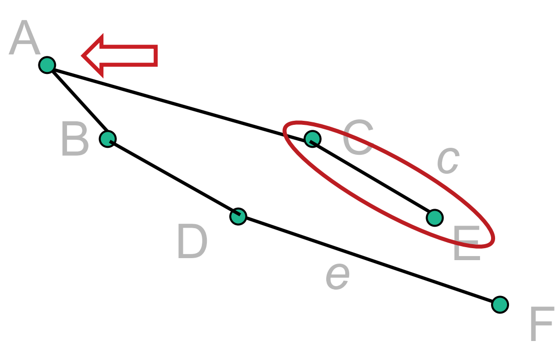
Let $T = (N, E)$ dependency tree, $e = [i, j]$ an edge in $E$,
$T_e$ the subgraph of $T$ induced by the nodes contained in the span of $e$.
***Degree of an edge*** $e \in E$, ***ed(e)***, is the number of connected components $c$ in $T_e$ such that the root of $c$ is not dominated by the head of $e$.
***Edge degree of T, ed(T)*** … max $\{ed(e) | e \in T\}$

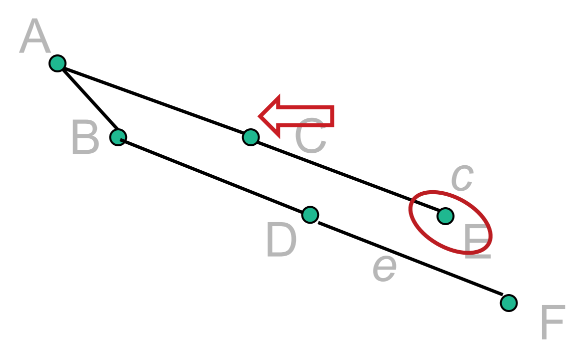


# Edge Degree

Let $T = (N, E)$ dependency tree, $e = [i, j]$ an edge in $E$,
$T_e$ the subgraph of $T$ induced by the nodes contained in the span of $e$.
***Degree of an edge*** $e \in E$, ***ed(e)***, is the number of connected components $c$ in $T_e$ such that the root of $c$ is not dominated by the head of $e$.
***Edge degree of T, ed(T)*** … max $\{ed(e) \mid e \in T\}$



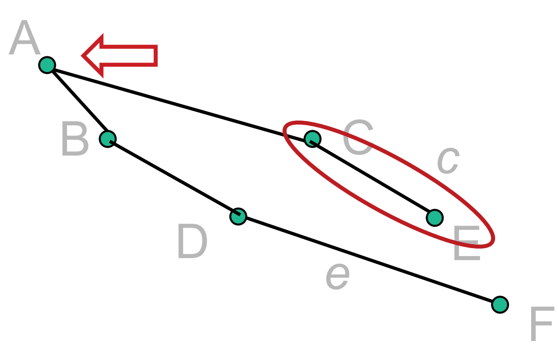



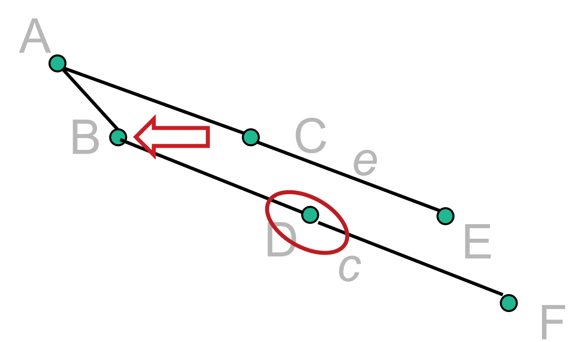
# Edge Degree

Let $T = (N, E)$ dependency tree, $e = [i, j]$ an edge in $E$, $T_e$ the subgraph of $T$ induced by the nodes contained in the span of $e$.

***Degree of an edge*** $e \in E$, ***ed(e)***, is the number of connected components $c$ in $T_e$ such that the root of $c$ is not dominated by the head of $e$.

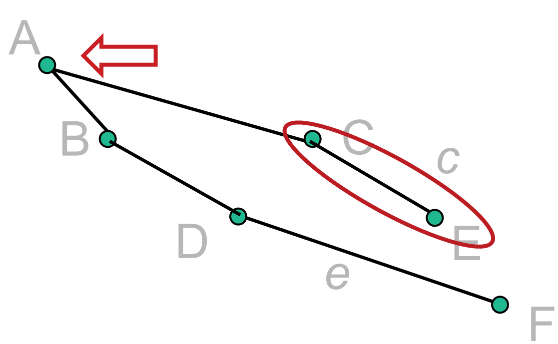
***Edge degree of T, ed(T)*** … max $\{ed(e) \mid e \in T\}$

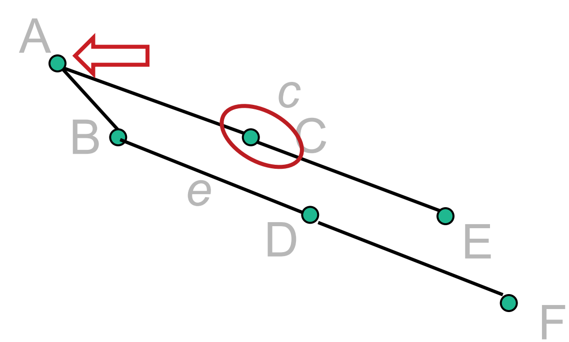


# Edge Degree

Let $T = (N, E)$ dependency tree, $e = [i, j]$ an edge in $E$,
$T_e$ the subgraph of $T$ induced by the nodes contained in the span of $e$.
***Degree of an edge*** $e \in E$, ***ed(e)***, is the number of connected components $c$ in $T_e$ such that the root of $c$ is not dominated by the head of $e$.
***Edge degree of T, ed(T)*** … max $\{ed(e) | e \in T\}$



| property | DDT | | PDT | |
|---|---|---|---|---|
| *all structures* | $n = 4393$ | | $n = 73088$ | |
| gap degree 0 | 3732 | 84.95% | 56168 | 76.85% |
| gap degree 1 | 654 | 14.89% | 16608 | 22.72% |
| gap degree 2 | 7 | 0.16% | 307 | 0.42% |
| gap degree 3 | – | – | 4 | 0.01% |
| gap degree 4 | – | – | 1 | < 0.01% |
| edge degree 0 | 3732 | 84.95% | 56168 | 76.85% |
| edge degree 1 | 584 | 13.29% | 16585 | 22.69% |
| edge degree 2 | 58 | 1.32% | 259 | 0.35% |
| edge degree 3 | 17 | 0.39% | 63 | 0.09% |
| edge degree 4 | 2 | 0.05% | 10 | 0.01% |
| edge degree 5 | – | – | 2 | < 0.01% |
| edge degree 6 | – | – | 1 | < 0.01% |
| projective | 3732 | 84.95% | 56168 | 76.85% |
| planar | 3796 | 86.41% | 60048 | 82.16% |
| well-nested | 4388 | 99.89% | 73010 | 99.89% |
| *non-projective structures only* | $n = 661$ | | $n = 16920$ | |
| planar | 64 | 9.68% | 3880 | 22.93% |
| well-nested | 656 | 99.24% | 16842 | 99.54% |

Kuhlmann, M., Nivre, J. (2006)

# References


- Hajičová, E., Havelka, J., Sgall, P., Veselá, K., Zeman, D. (2004) Issues of Projectivity in the Prague Dependency Treebank. *PBML,* vol. 81
- Holan, T., Kuboň, V., Oliva, K., Plátek, M. (2000) On Complexity of Word Order. *Les grammaires de dépendance – Traitement automatique des langues,* vol. 41, no. 1, 273-300
- Kuhlmann, M., Nivre, J. (2006) Mildly Non-Projective Dependency Structures. In COLING/ACL Main Conference Poster Sessions, 507–514.
- Mel'čuk, I. (1988) *Dependency Syntax: Theory and Practice.* State University of New York Press, Albany
- Partee, B. H.; ter Meulen, A.; Wall, R. E. (1990) *Mathematical Methods in Linguistics*. Kluwer Academic Publishers
- Petkevič, V. (1995) A New Formal Specification of Underlying Structure. *Theoretical Linguistics*, vol. 21, No.1
- Sgall, P., Hajičová, E., Panevová, J. (1986) *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. D. Reidel Publishing Company, Dordrecht/Academia, Prague
- Štěpánek, J. (2006) *Závislostní zachycení větné struktury v anotovaném syntaktickém korpusu*. PhD Thesis, MFF UK