Netgraph

A Tool for Easy Searching in Prague Dependency Treebank 2.0

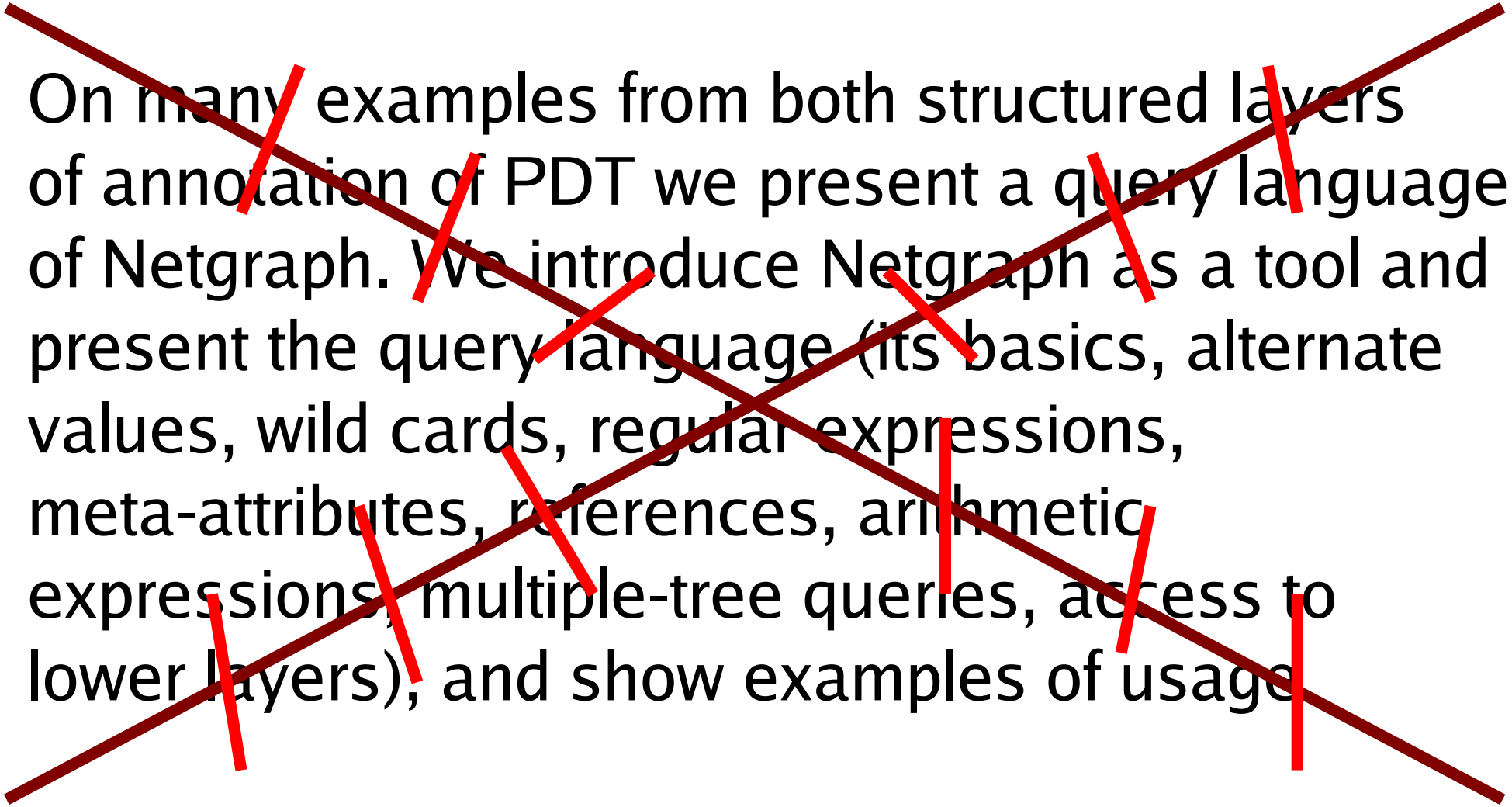
Jiří Mírovský

Charles University in Prague

Institute of Formal and Applied Linguistics

Prague, December 2007

Abstract (original)



On many examples from both structured layers of annotation of PDT we present a query language of Netgraph. We introduce Netgraph as a tool and present the query language (its basics, alternate values, wild cards, regular expressions, meta-attributes, references, arithmetic expressions, multiple-tree queries, access to lower layers), and show examples of usage.

Abstract (original)

On many examples from both structured layers of annotation of PDT we present a query language of Netgraph. We introduce Netgraph as a tool and present the query language (its basics, alternative values, wild cards, regular expressions, meta-attributes, references, arithmetic expressions, multiple-tree queries, access to lower layers), and show examples of usage.

Abstract

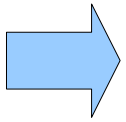
- Linguistic requirements on the query language
- Introduction to the query language
- Using the query language (examples in Netgraph)

Abstract

- Linguistic requirements on the query language
- Introduction to the query language
- Using the query language (examples in Netgraph)

Linguistic Requirements
The Basics

T-manual, page 1: The basic unit of annotation on the tectogrammatical layer of PDT 2.0 is a sentence.
The representation of the tectogrammatical annotation of a sentence is a rooted dependency tree. It consists of a set of nodes and a set of edges. Each node is a complex unit consisting of a set of pairs attribute-value. The edges express dependency relations between nodes.



The query language should be able to express node evaluation and tree dependency among nodes richly and in the most direct way.

Linguistic Requirements
Valency

To study valency, the query language should be able to:

- control a presence of a particular type of son (both in positive and negative meaning)
- control number of sons

Linguistic Requirements
Coordination etc.

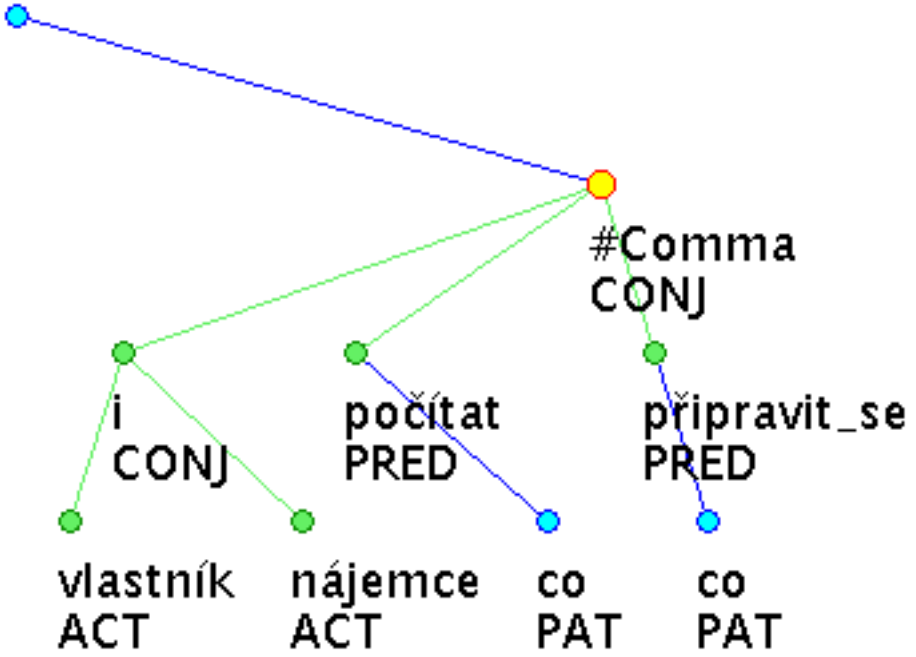
Tree dependency is not always linguistic dependency.

- skip a node (etc. coordination, apposition)
- even better: set a linguistic dependency

Linguistic Requirements
Complex Example of Coordination

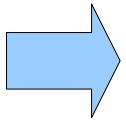
Czech: S čím mohou vlastníci i nájemci počítat, na co by se měli připravit?

English (lit.): What can owners and tenants expect, what they should get ready for?



Linguistic Requirements
Predicative Complement

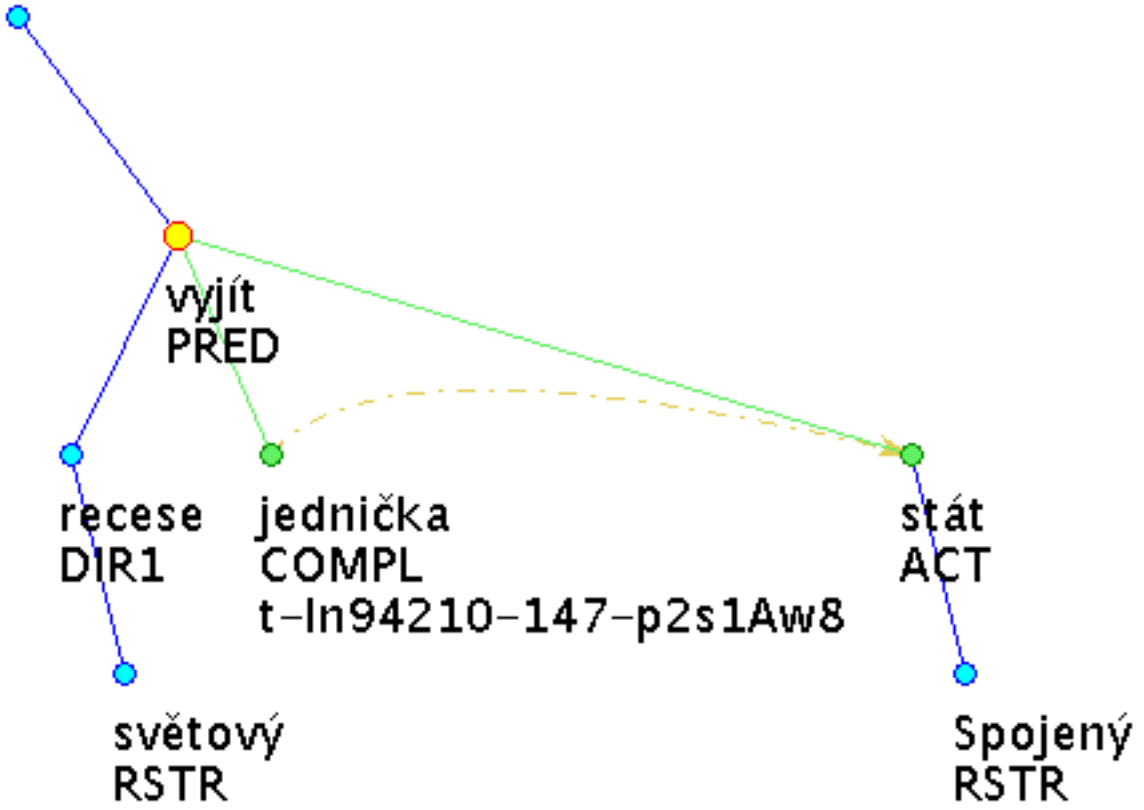
Dual dependency is represented by means of a reference to another node (attributes `compl.rf` and `id`).



match a value unknown at the time of creating the query

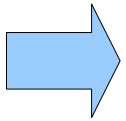
Linguistic Requirements
Predicative Complement

Czech: Ze světové recese vyšly jako jednička Spojené státy.
English (lit.): The United States emerged from the world recession as number one.



Linguistic Requirements
Coreferences

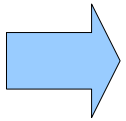
Represented by means of references (attributes `coref_gram.rf` and `coref_text.rf` (and `id`))



as before, match a value unknown at the time of creating the query

Linguistic Requirements
Topic-Focus Articulation

- *Contextual boundness* – attribute tfa
- *Communicative dynamism* – deep word order (attribute `deepord`)



set references to other nodes with other relations than “equal to”

Linguistic Requirements
Topic-Focus Articulation

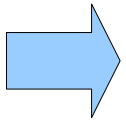
Focus proper – the rightmost contextually non-bound node in the tectogrammatical tree

⇨ define that there is no node (contextually non-bound) with bigger `deepord` in the whole tree

⇨ combine references, non-existence of a node and transitive closure of dependency

Linguistic Requirements
Topic-Focus Articulation

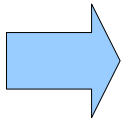
Rhematizers – closest left brother of its scope (or closest left son if the governing predicate belongs to its scope)



define distance between nodes in deep word order

Linguistic Requirements
Topic-Focus Articulation

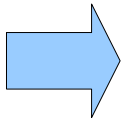
(Non-)Projectivity – between a father and its son there can only be direct or indirect sons of the father.



define multiple-tree query to combine several one-tree queries representing different orientations of non-projective edges

Linguistic Requirements
Idioms etc.

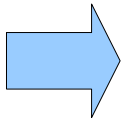
Not everything is annotated, not everything is easily accessible in the tree.



search in the linear form of the sentence

Linguistic Requirements
Layers with non-1:1 Relation

There are multiple layers of annotation, the relation among nodes on the analytical and tectogrammatical layers is not 1:1.



have special means of accessing lower layers

Linguistic Requirements
Summary

Evaluation of a node

- multiple attributes evaluation
- alternative values
- alternative nodes (alternative evaluation of the whole set of attributes)
- wild cards (regular expressions)
- negation, relations other than “equal to”

Linguistic Requirements
Summary

Dependencies between nodes
(vertical relations)

- direct, transitive (existence, non-existence)
- vertical distance (from root, from one another)
- number of sons (zero for lists)

Linguistic Requirements
Summary

Horizontal relations

- precedence, immediate precedence, distance
- negation of it

Secondary relations

- secondary dependencies, coreferences

Linguistic Requirements
Summary

Other features

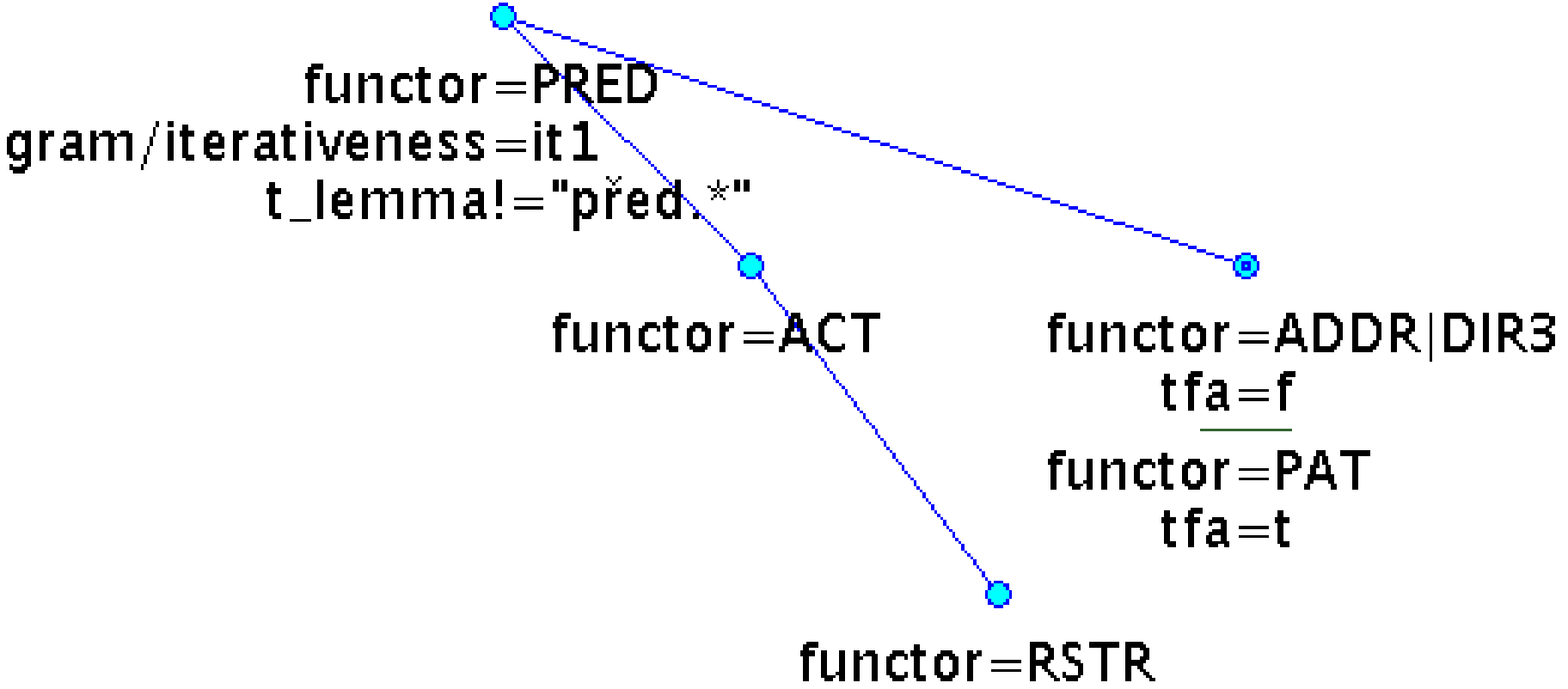
- multiple-tree queries
- accessing several layers of annotation at the same time
- searching in the linear form of the sentence

Abstract

- Linguistic requirements on the query language
- Introduction to the query language
- Using the query language (examples in Netgraph)

Introduction to the Query Language

The Basics



Introduction to the Query Language
Meta-Attributes

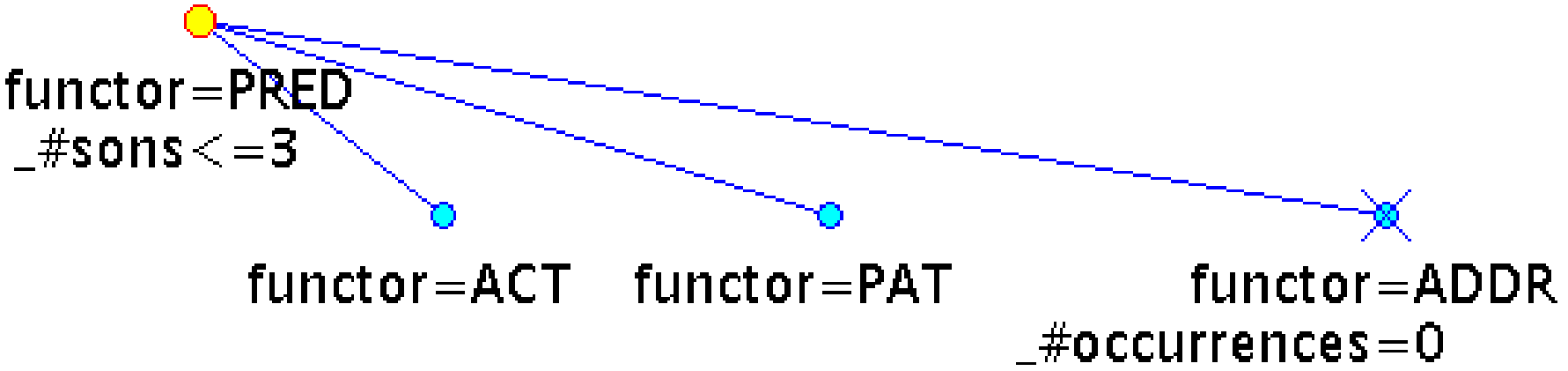
Attributes not present in the corpus, treated like normal attributes:

- **_transitive** *(transitive edge)*
- **_optional** *(optional node(s))*
- **_#sons** *(number of sons)*
- **_#hsons** *(number of hidden sons)*
- **_#descendants** *(number of nodes in the subtree)*

Introduction to the Query Language
Meta-Attributes

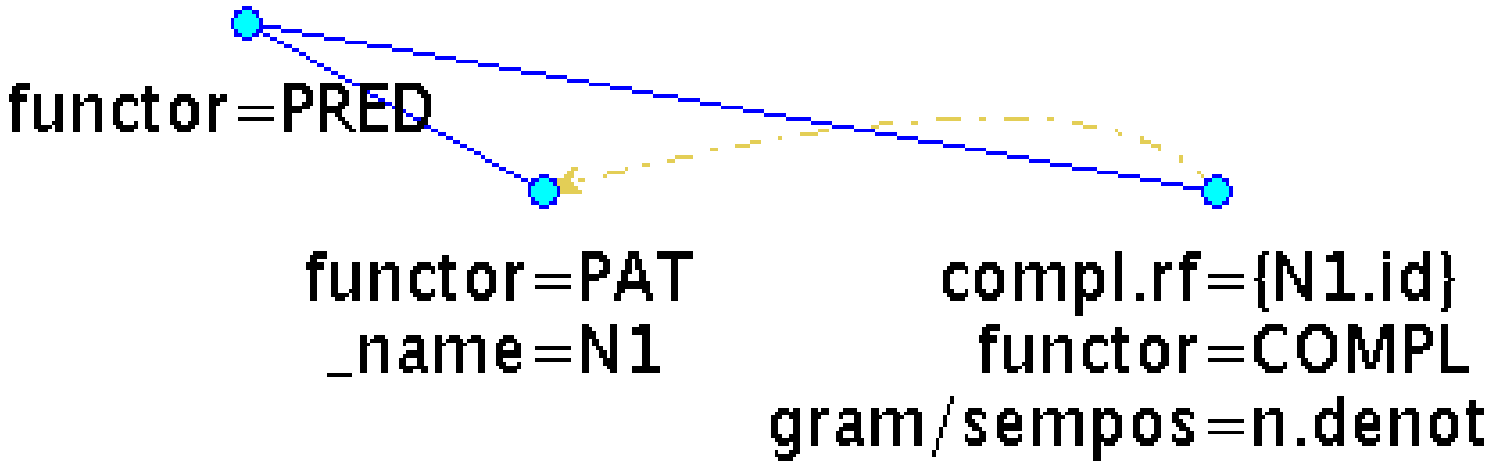
- **_#lbrothers** *(number of left brothers)*
- **_#rbrothers** *(number of right brothers)*
- **_depth** *(distance from the root)*
- **_#occurrences** *(exact number of a particular type of sons/descendants)*
- ***_name*** *(label of a node for references)*
- ***_sentence*** *(linear form of the sentence)*

Introduction to the Query Language
An Example Query



- No more than three sons
- One ACTor, one PATient, no ADDRessee

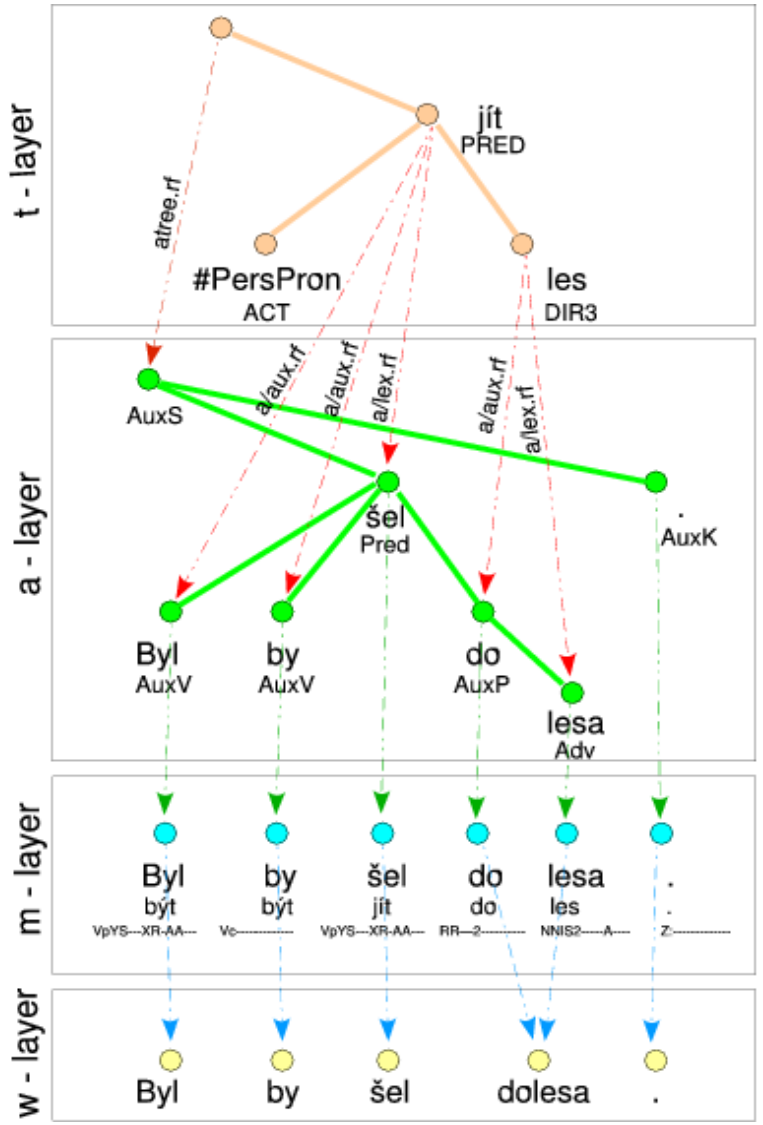
Introduction to the Query Language
References



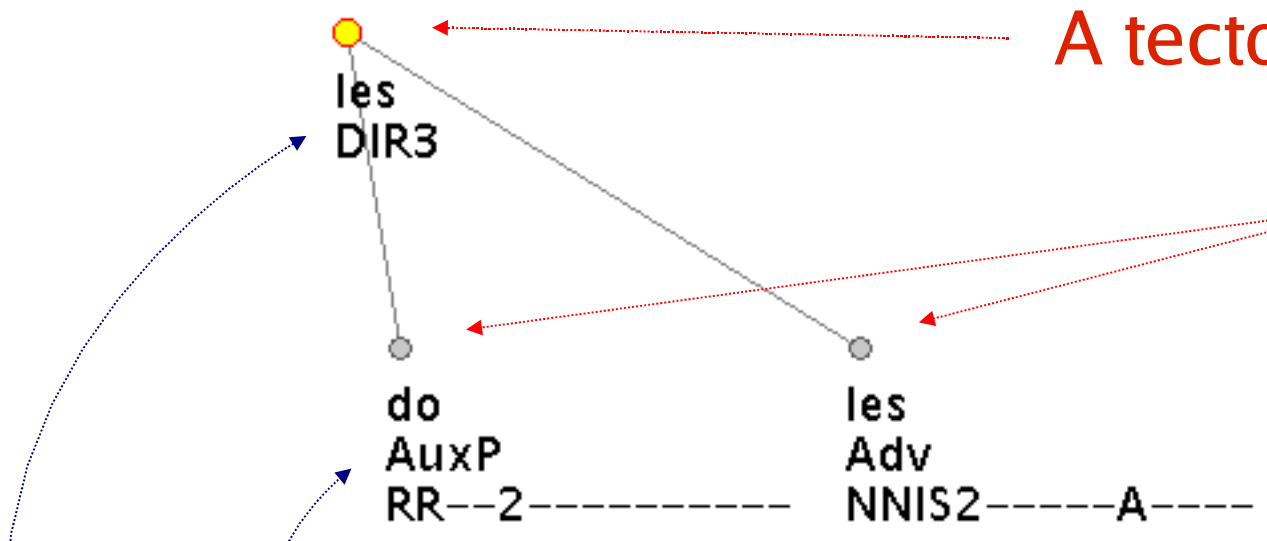
- Predicative complement expressed by a noun
- Second dependency on a PATient

Introduction to the Query Language

Layers in PDT 2.0



Introduction to the Query Language
Hidden Nodes



A tectogrammatical node

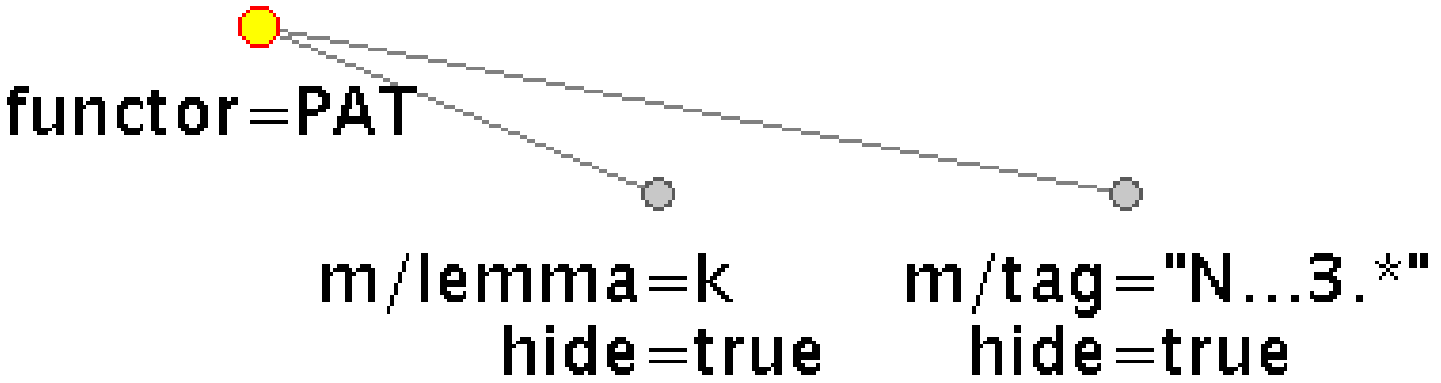
Hidden nodes

Morphological information

Analytical information

Tectogrammatical information

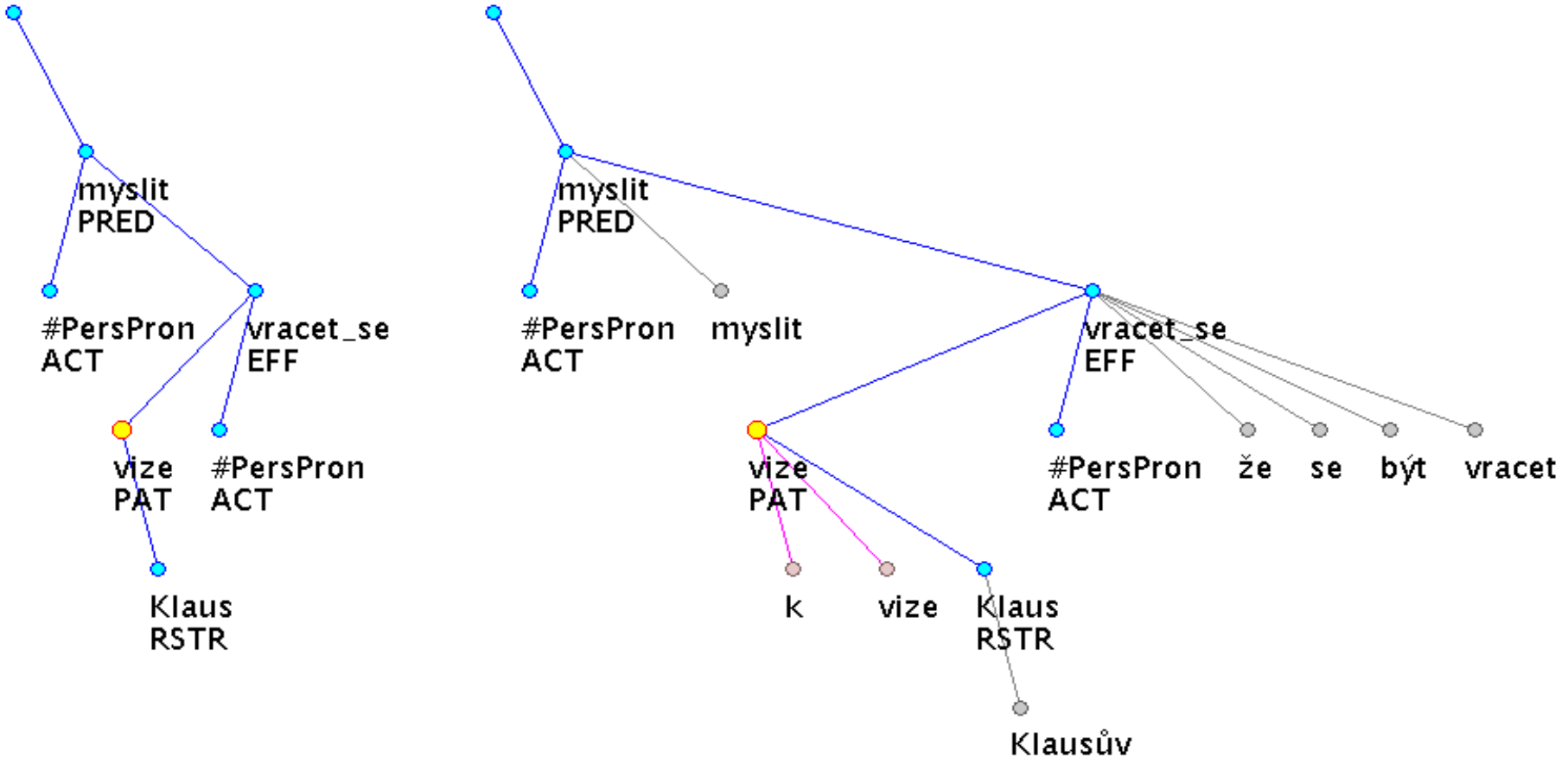
Introduction to the Query Language
Hidden Nodes – A Query



- PATient expressed with preposition **k** and a Noun in 3. case on the morphological layer

Introduction to the Query Language

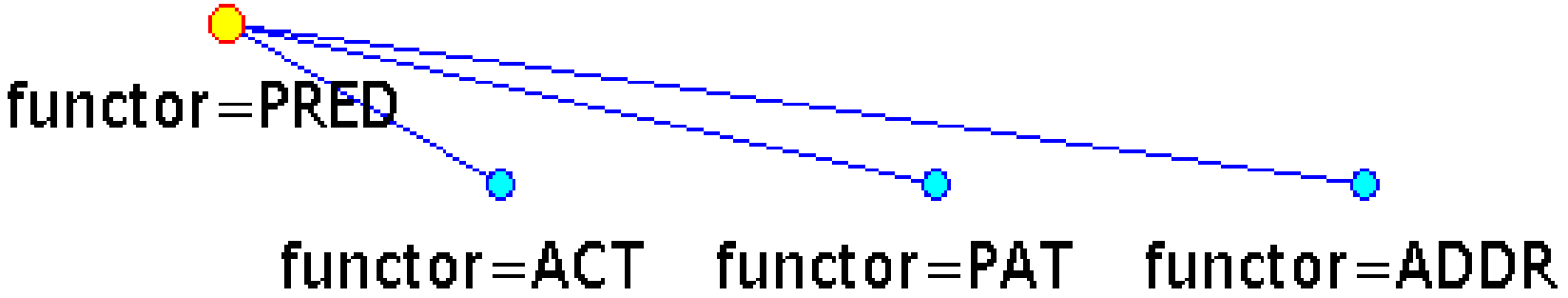
Hidden Nodes – A Result Tree



Abstract

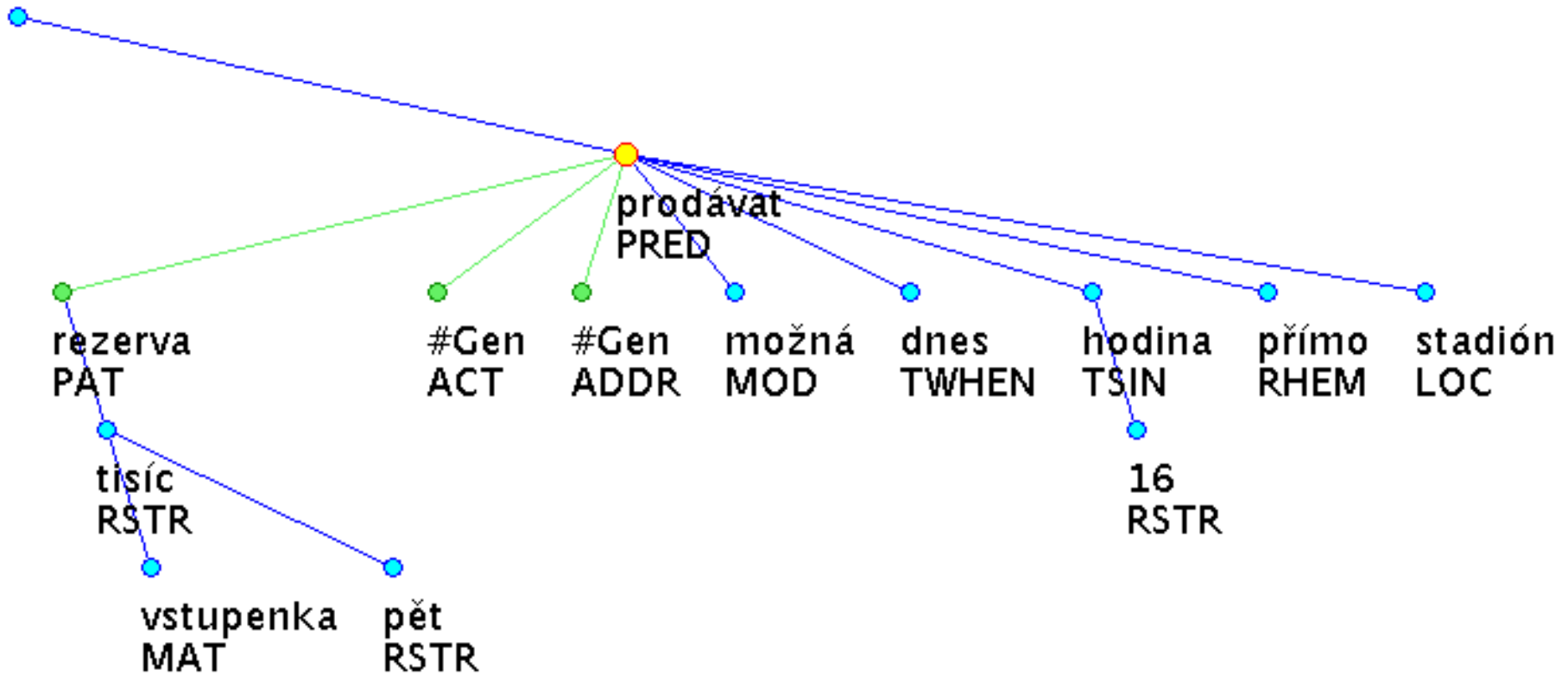
- Linguistic requirements on the query language
- Introduction to the query language
- Using the query language (examples in Netgraph)

Using the Query Language
A Simple Query



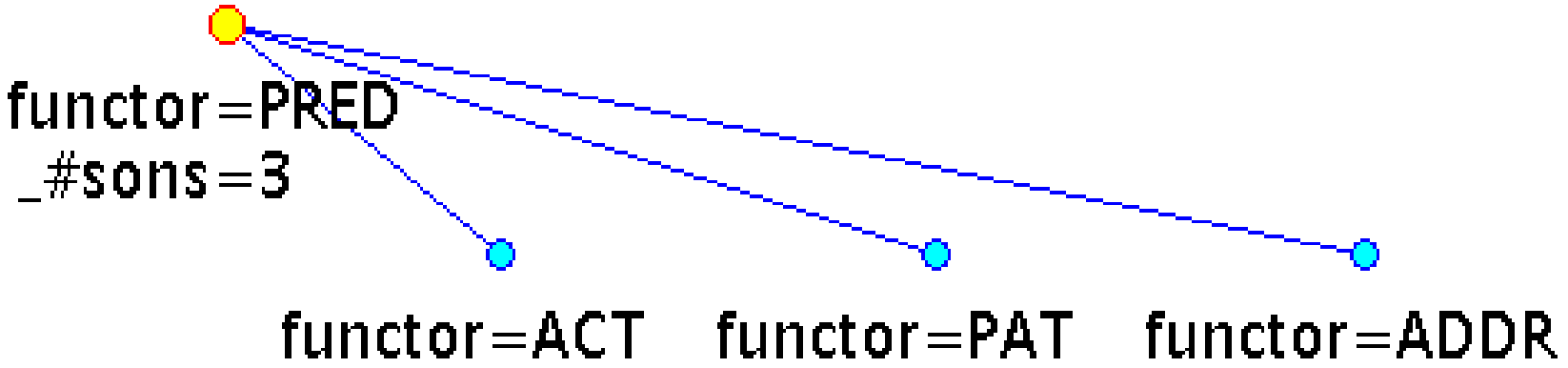
- A PREDicate governing an ACTor, a PATient, and an ADDRessee
- No condition on the order of the sons or their number

Using the Query Language
A Simple Query (Result)



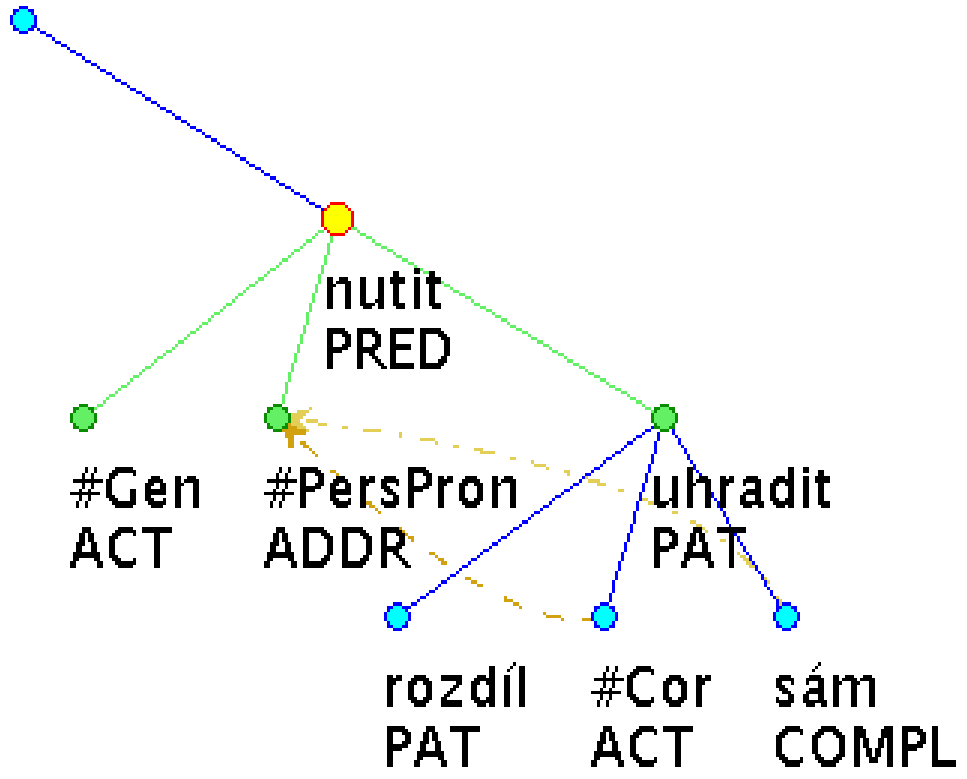
- Additional sons and different order of the sons

Using the Query Language
Restricting Number of Sons



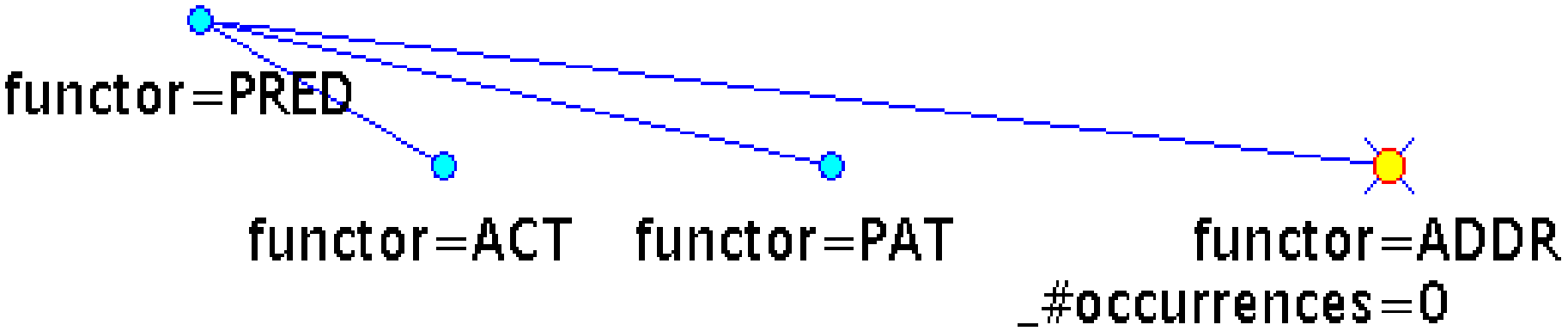
- A PREDicate governing an ACTor, a PATient, and an ADDRessee
- No other sons allowed

Using the Query Language
Restricting Number of Sons (Result)



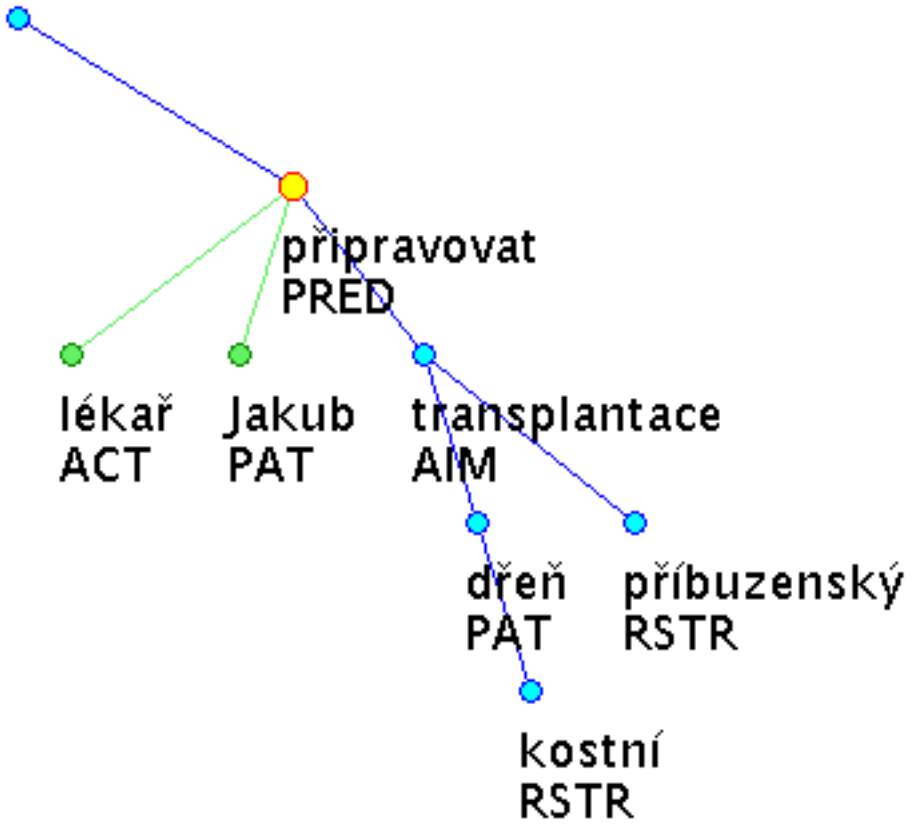
- No additional sons (though a different order)

Using the Query Language
Restricting the Type of Sons



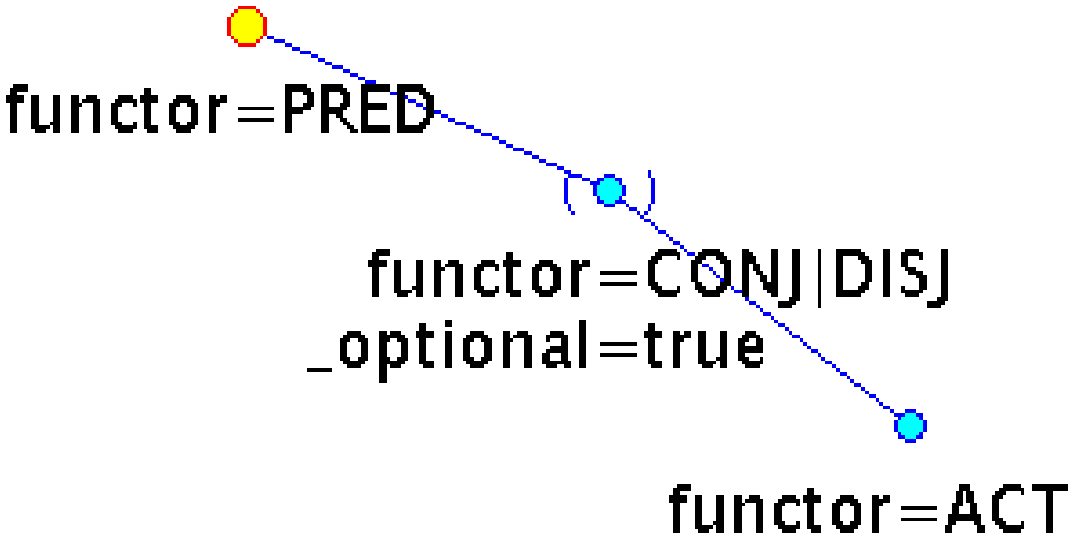
- A PREDicate governing an ACTor and a PATient, but not an ADDRessee

Using the Query Language
Restricting the Type of Sons (Result)



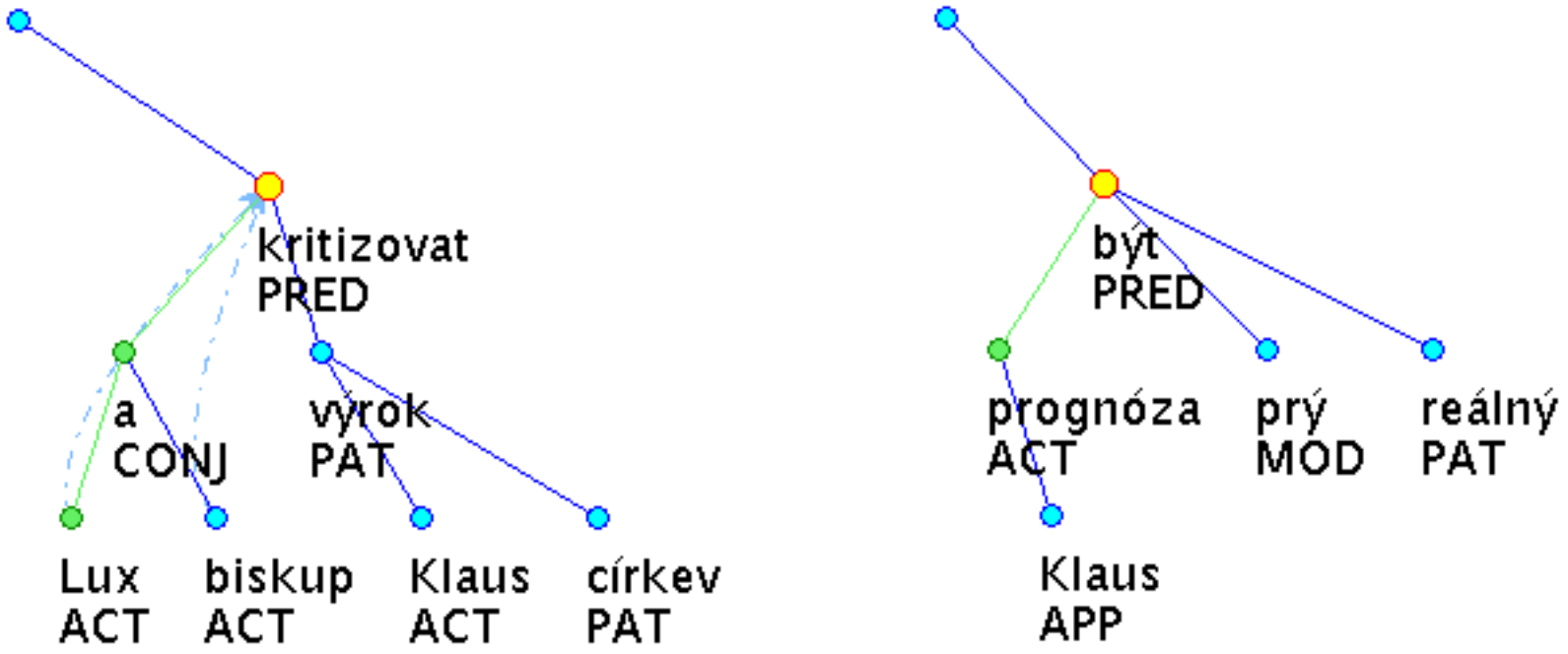
- No ADDRessee dependent on the PREDicate

Using the Query Language
An Optional Node



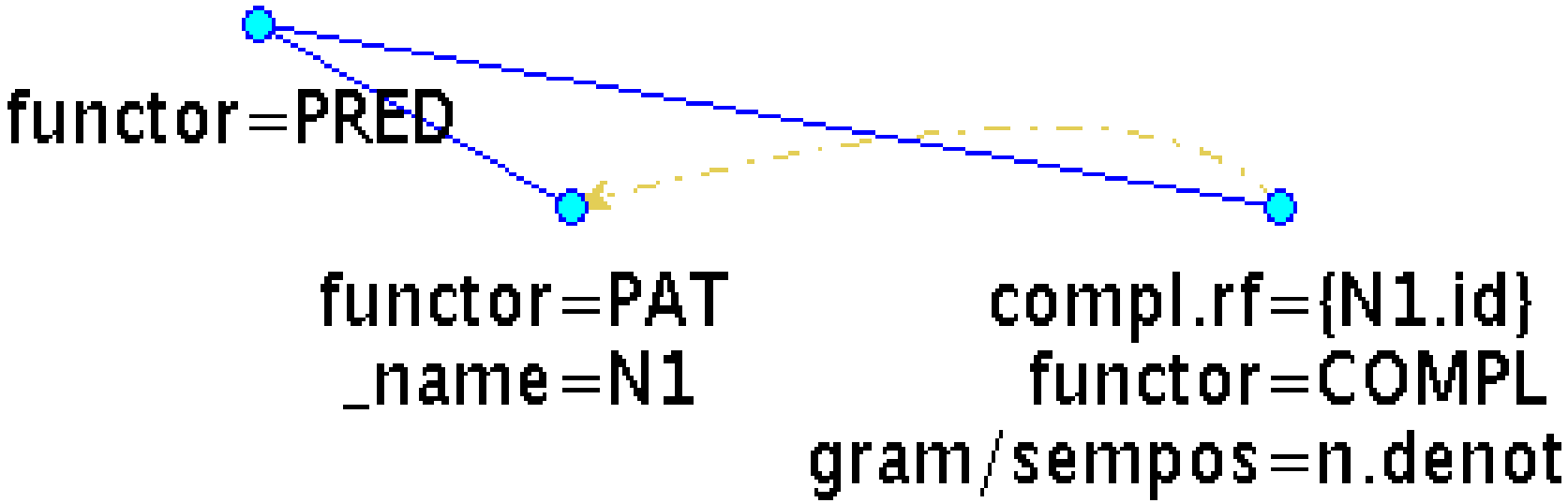
- A PREDicate governing an ACTor with an optional CONJunction or DISJunction inbetween (the ACTor may be coordinated)

Using the Query Language
An Optional Node (Result)



- Two possible types of results: with and without the optional coordination

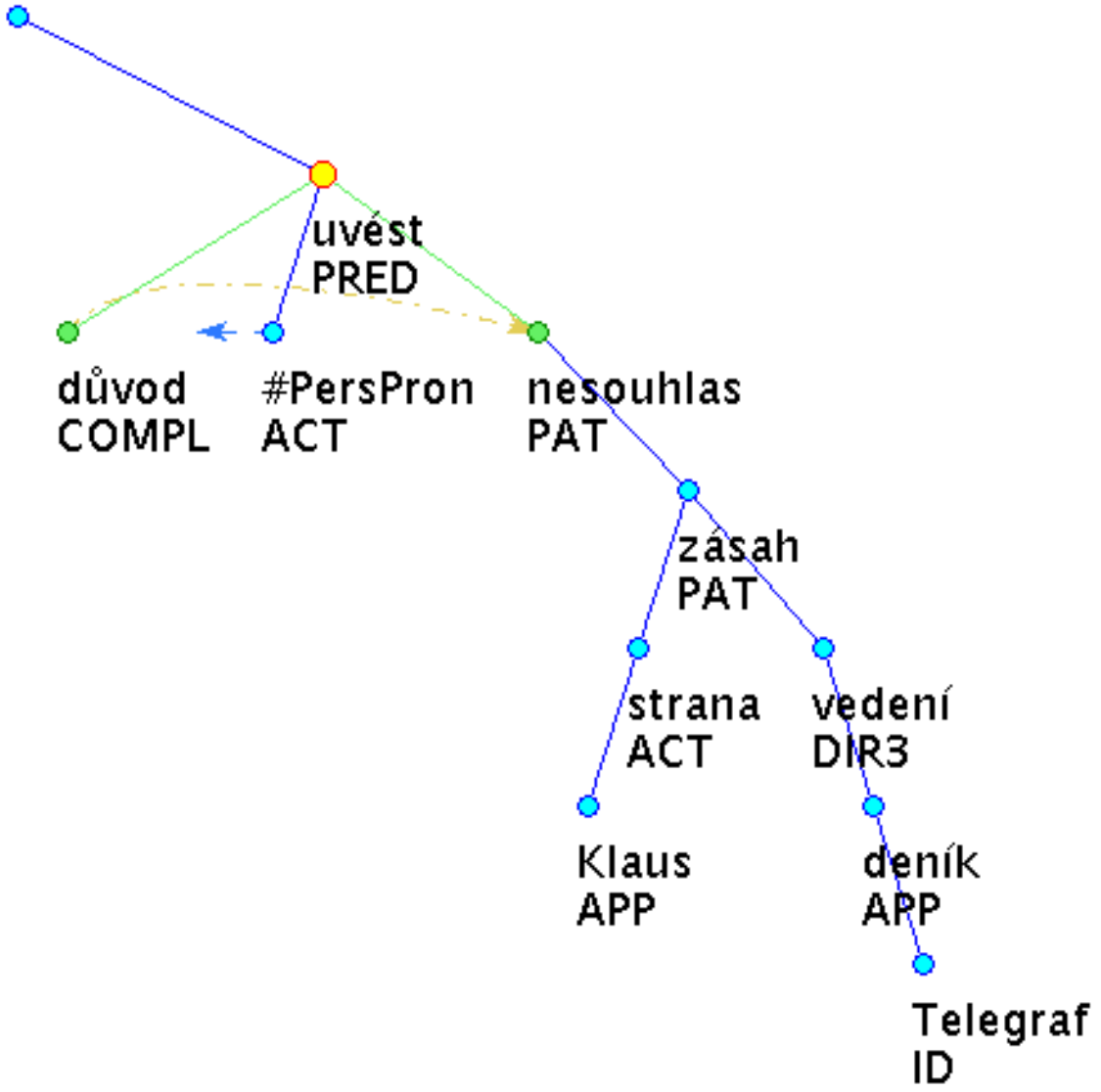
Using the Query Language
Predicative Complement



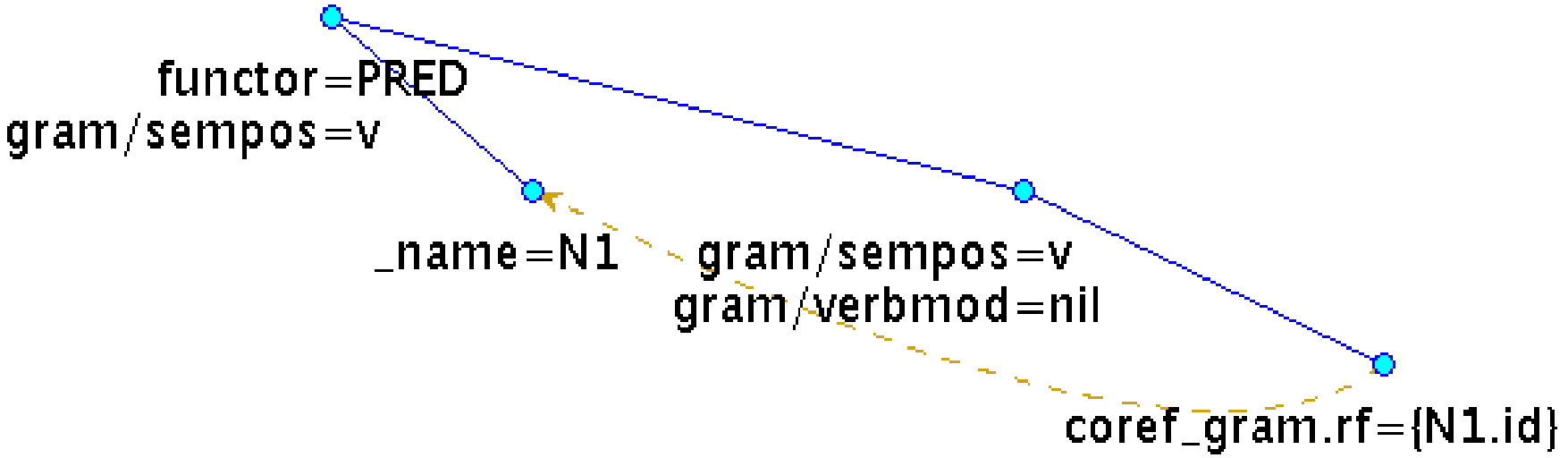
- A nominal predicative COMPLement with second dependency on a PATient

Using the Query Language

Predicative Complement (Result)



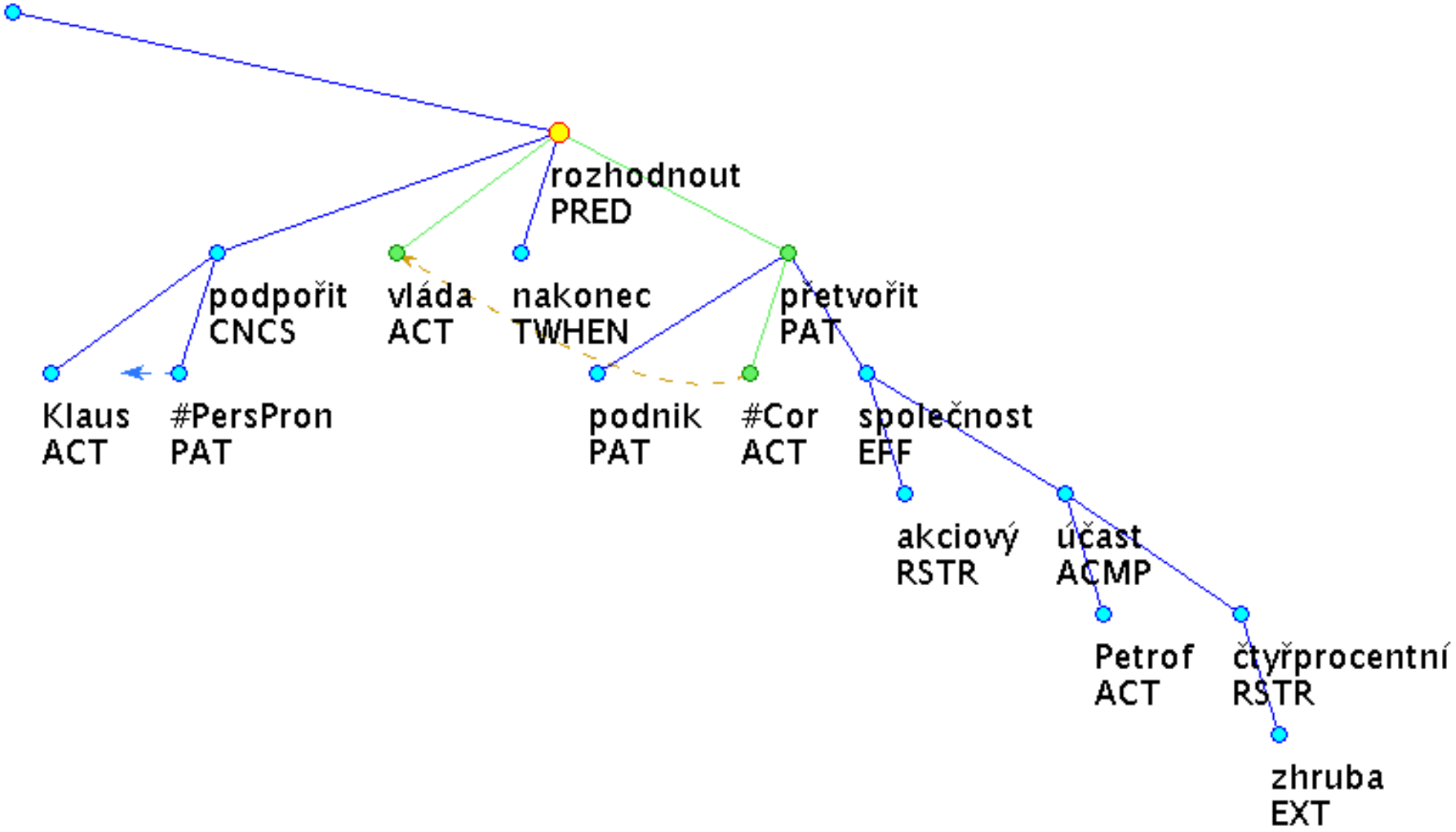
Using the Query Language
Type-1 Control Construction



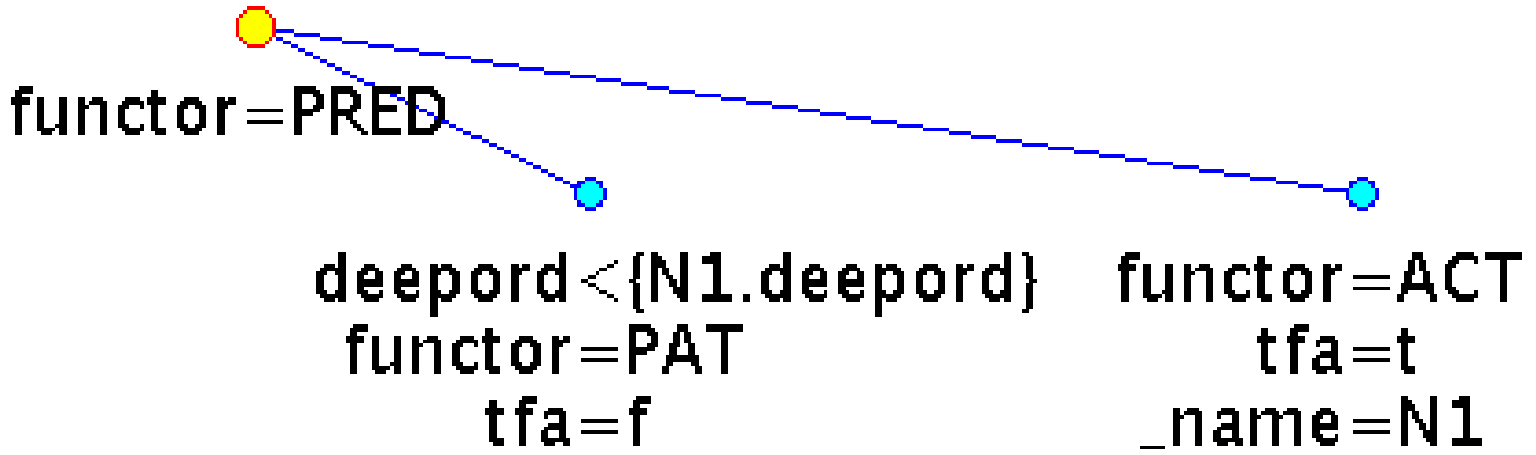
- Type-1 control construction (an infinitive depends on a verbal control PREDicate

Using the Query Language

Type-1 Control Construction (Result)

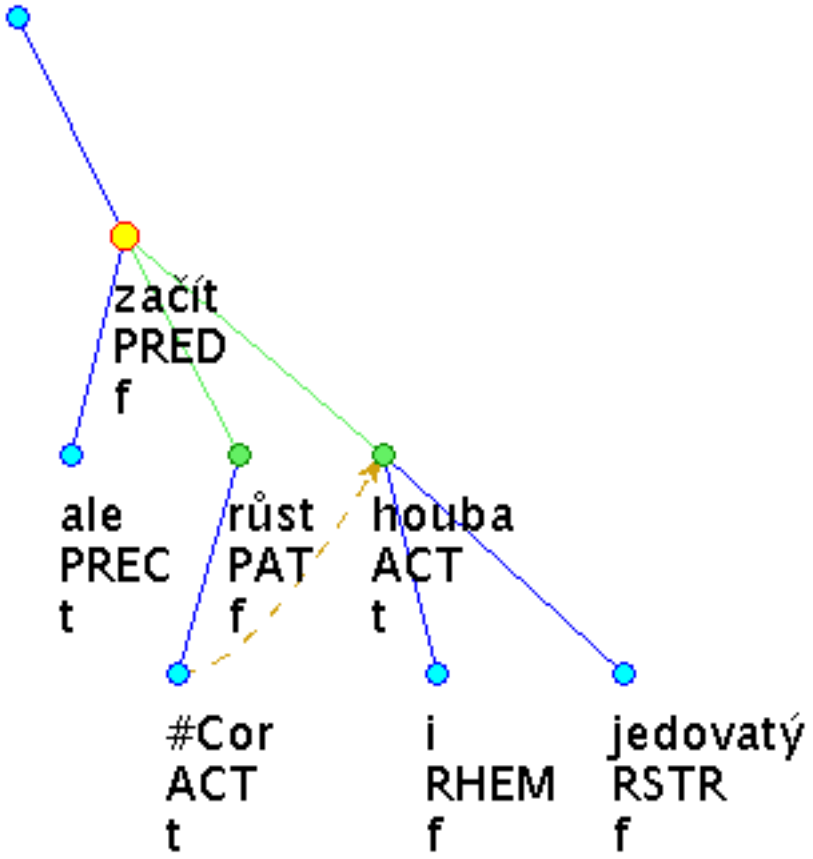


Using the Query Language
Topic-Focus (Deep Word Order)



- A PATient in focus on the left side (less dynamic) from an ACTor in topic

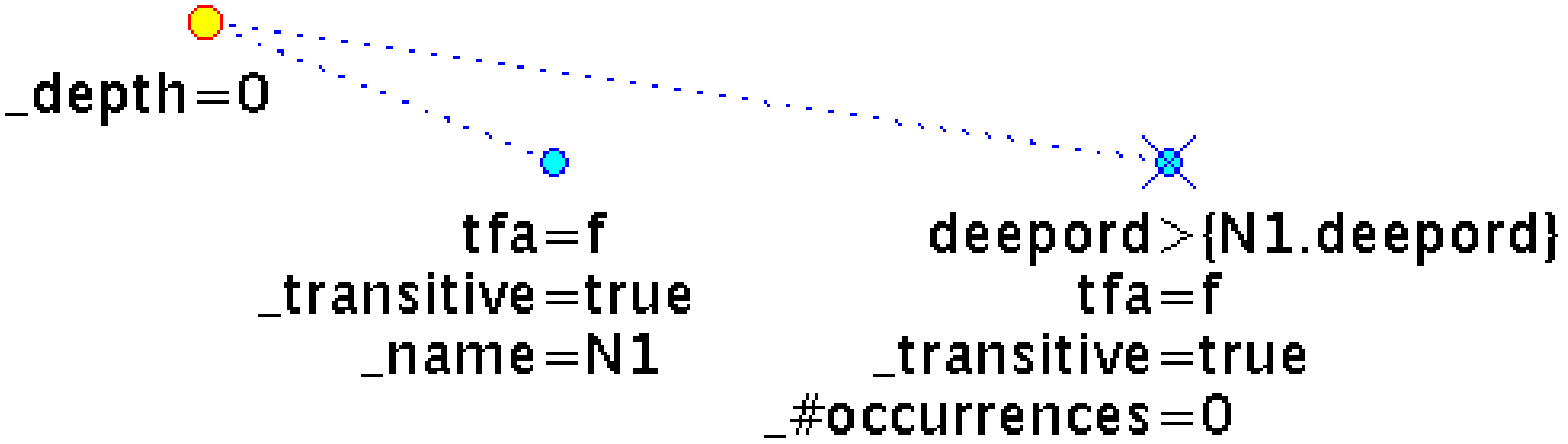
Using the Query Language
Topic-Focus (Deep Word Order) (Result)



Czech: Začaly ale růst i houby jedovaté.

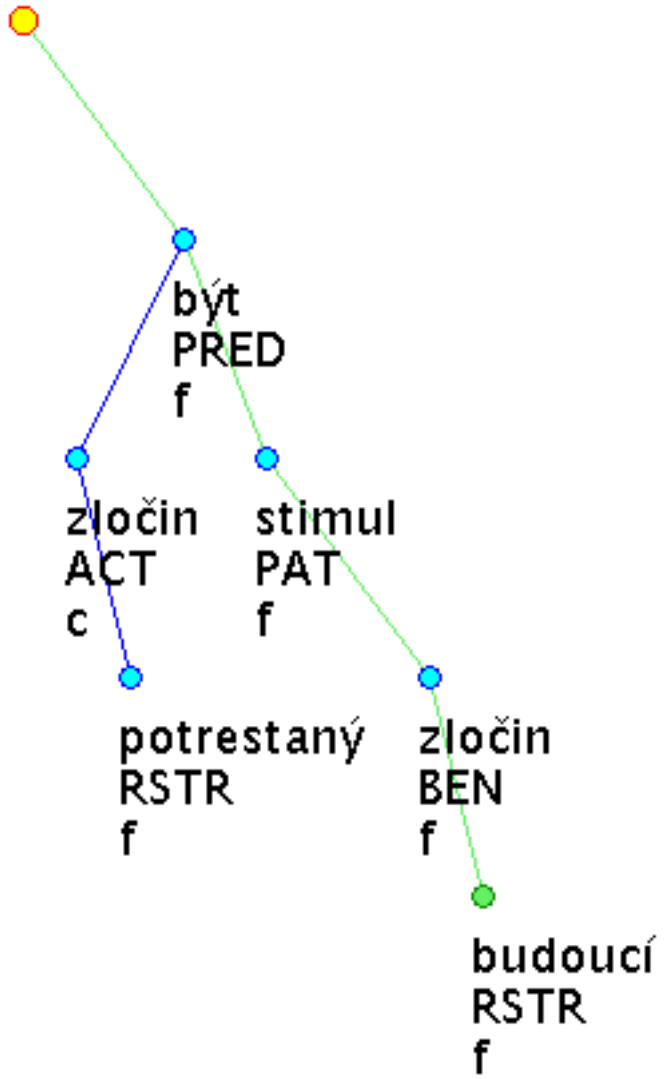
English (lit.): But also poisonous mushrooms started to grow.

Using the Query Language
Focus Proper



- There is not a more dynamic node in focus anywhere in the tree

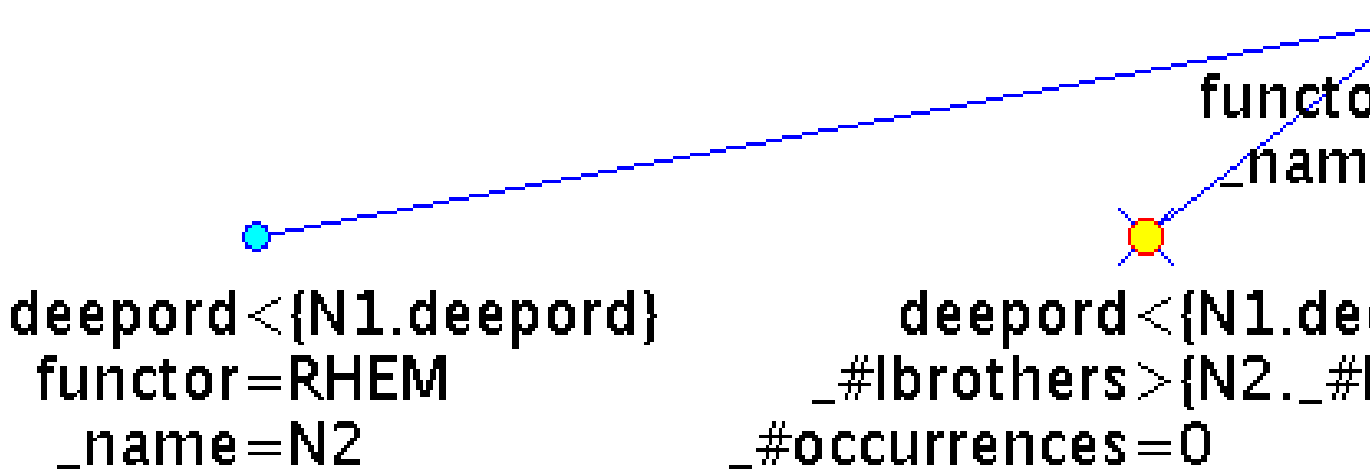
Using the Query Language
Focus Proper (Result)



Czech: Nepotrestaný zločin je stimulem pro zločiny budoucí.

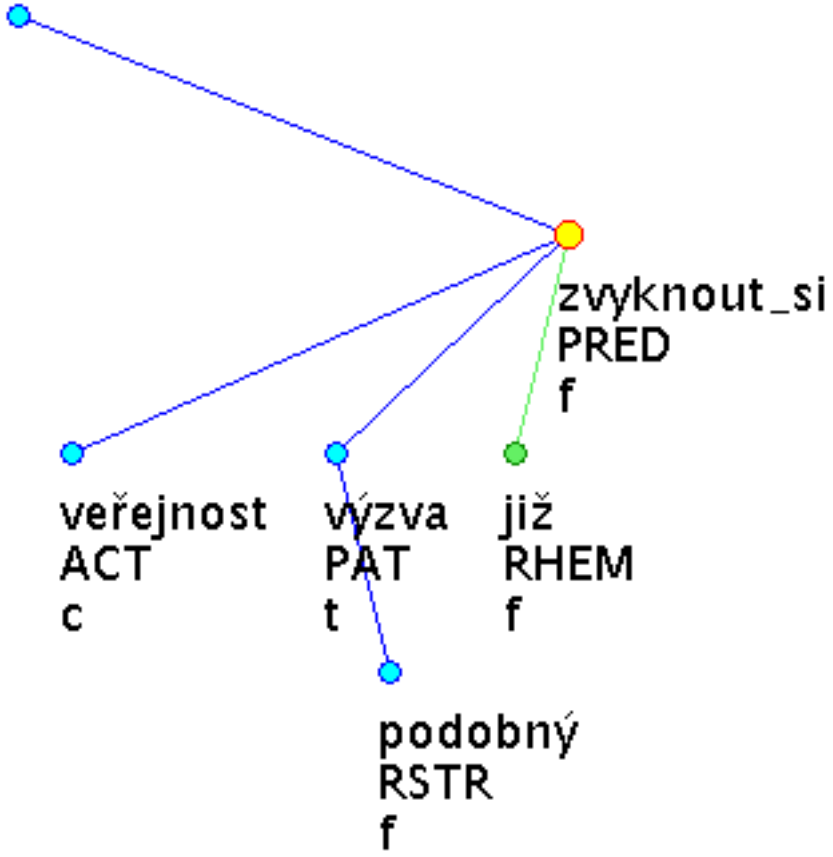
English (lit.): An unpunished crime is a stimulant for future crimes.

Using the Query Language
Rhematizer with Predicate in its Scope



- A RHEMatizer that is the leftmost son of a PREDicate (no left son of the PREDicate is on the right side from the RHEMatizer)

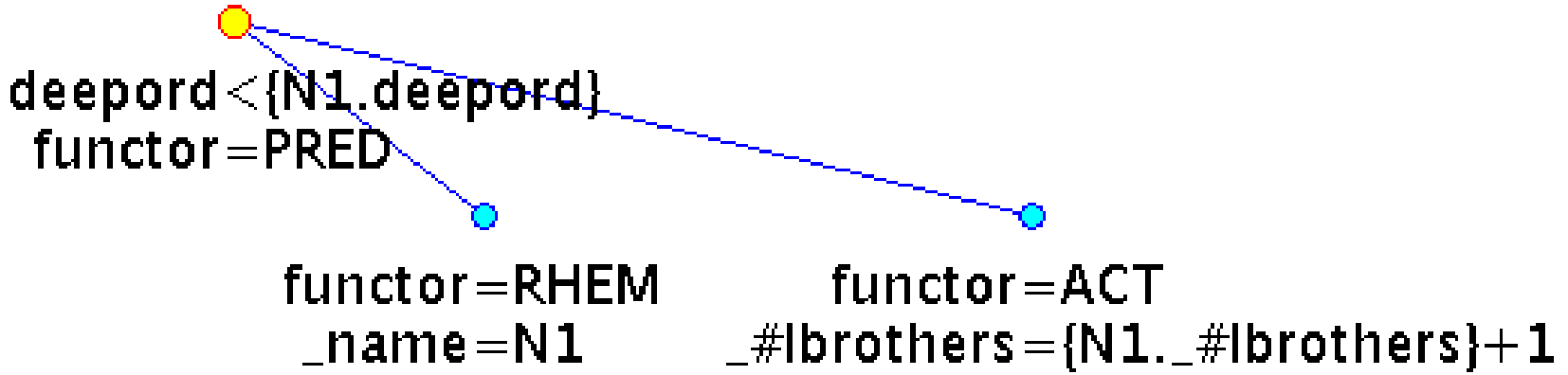
Using the Query Language
Rhematizer with Predicate in its Scope (Result)



Czech: Veřejnost si na podobné výzvy již zvykla.

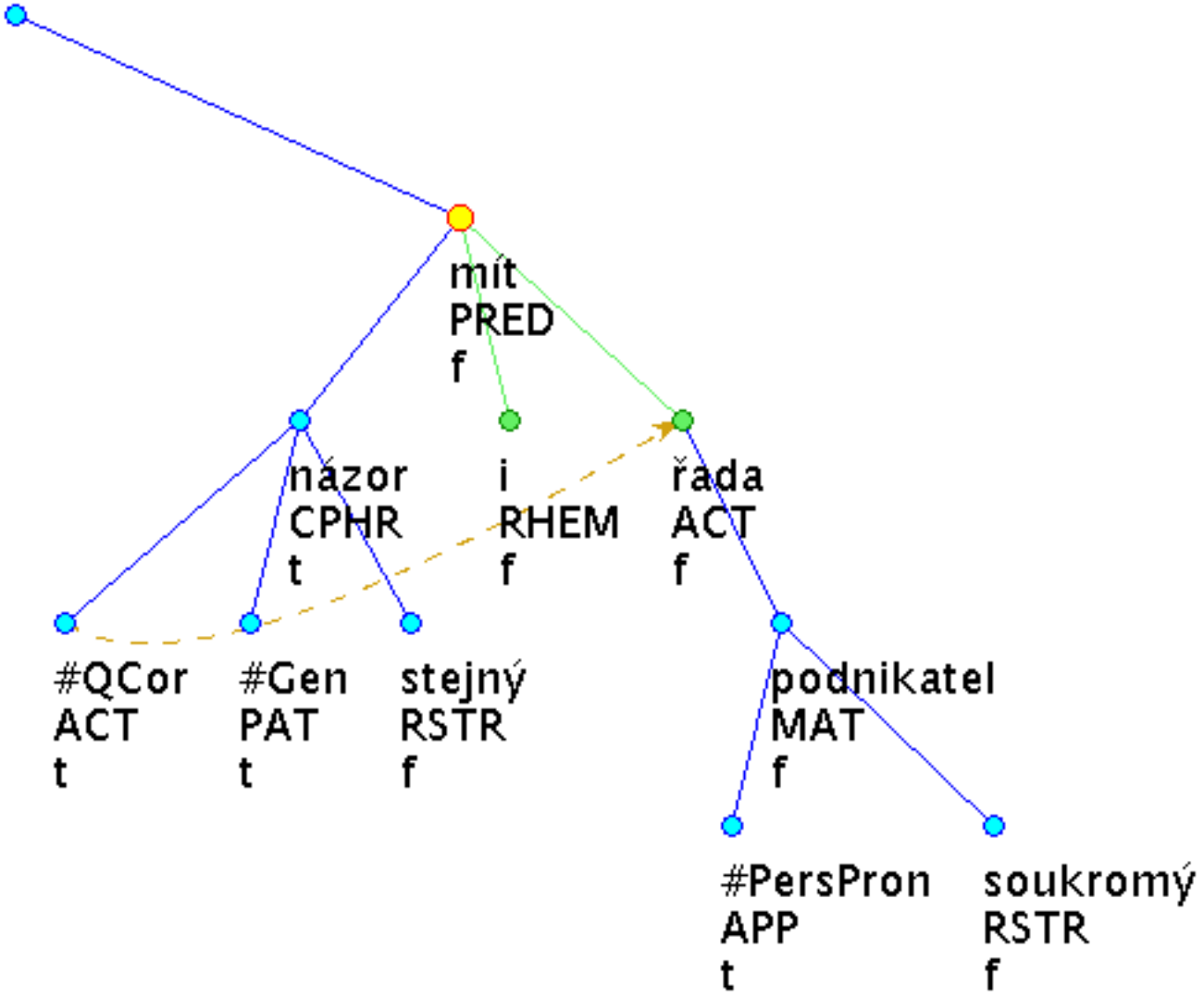
English (lit.): The public has already got accustomed to such calls.

Using the Query Language
Rhematizer without Predicate in its Scope



- A rhematized ACTor with a PREDicate outside the scope of the RHEMatizer

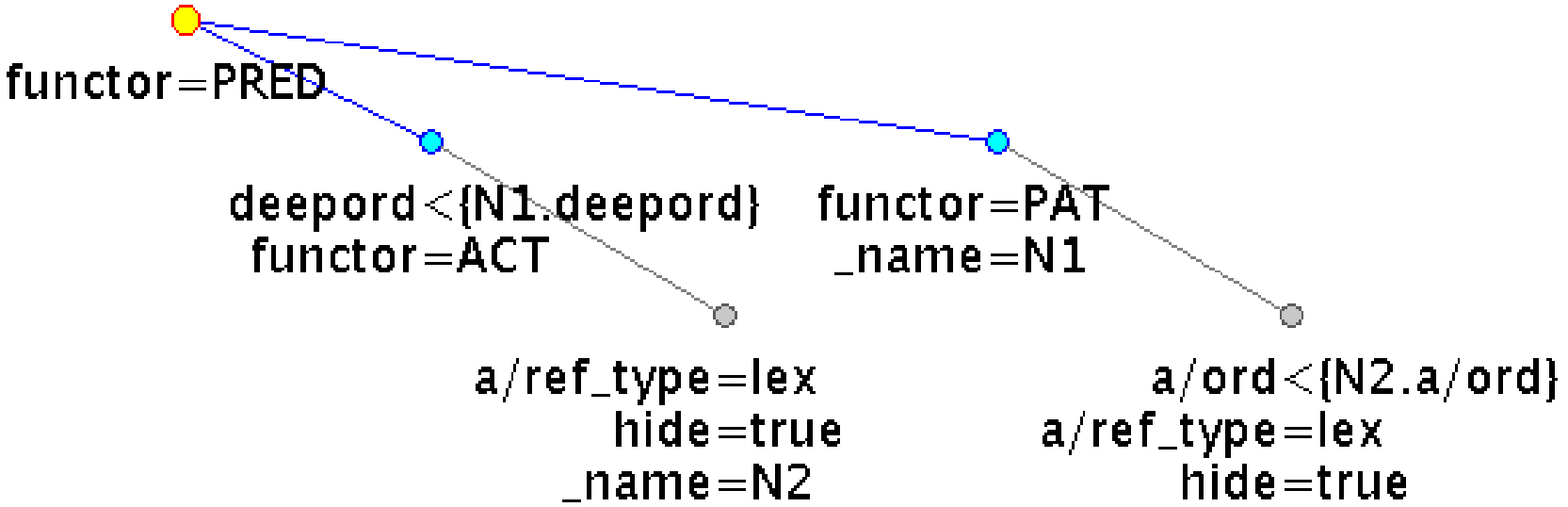
Using the Query Language
Rhematizer without Predicate in its Scope (Result)



Czech: Stejný názor má i řada našich soukromých podnikatelů.

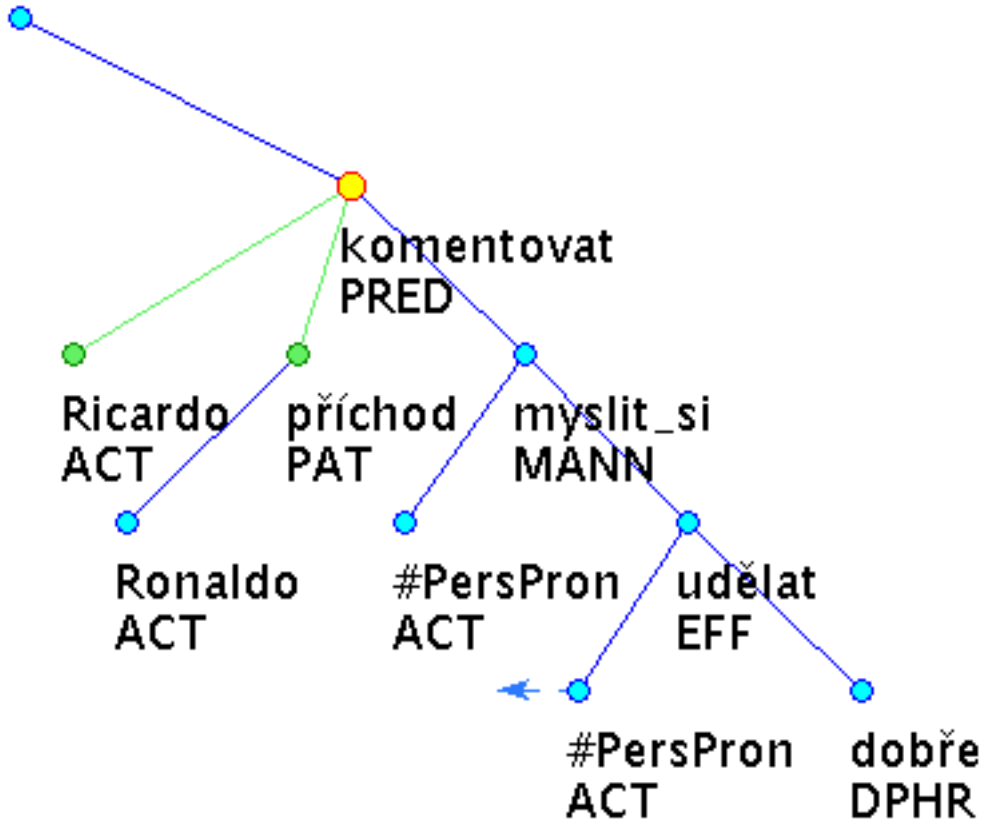
English (lit.): Also a number of our private investors has the same opinion.

Using the Query Language
Accessing Lower Layers



- An ACTor less dynamic than a PATient, but on the right side from it on the surface
- Lower layers accessible via hidden nodes

Using the Query Language
Accessing Lower Layers (Result)



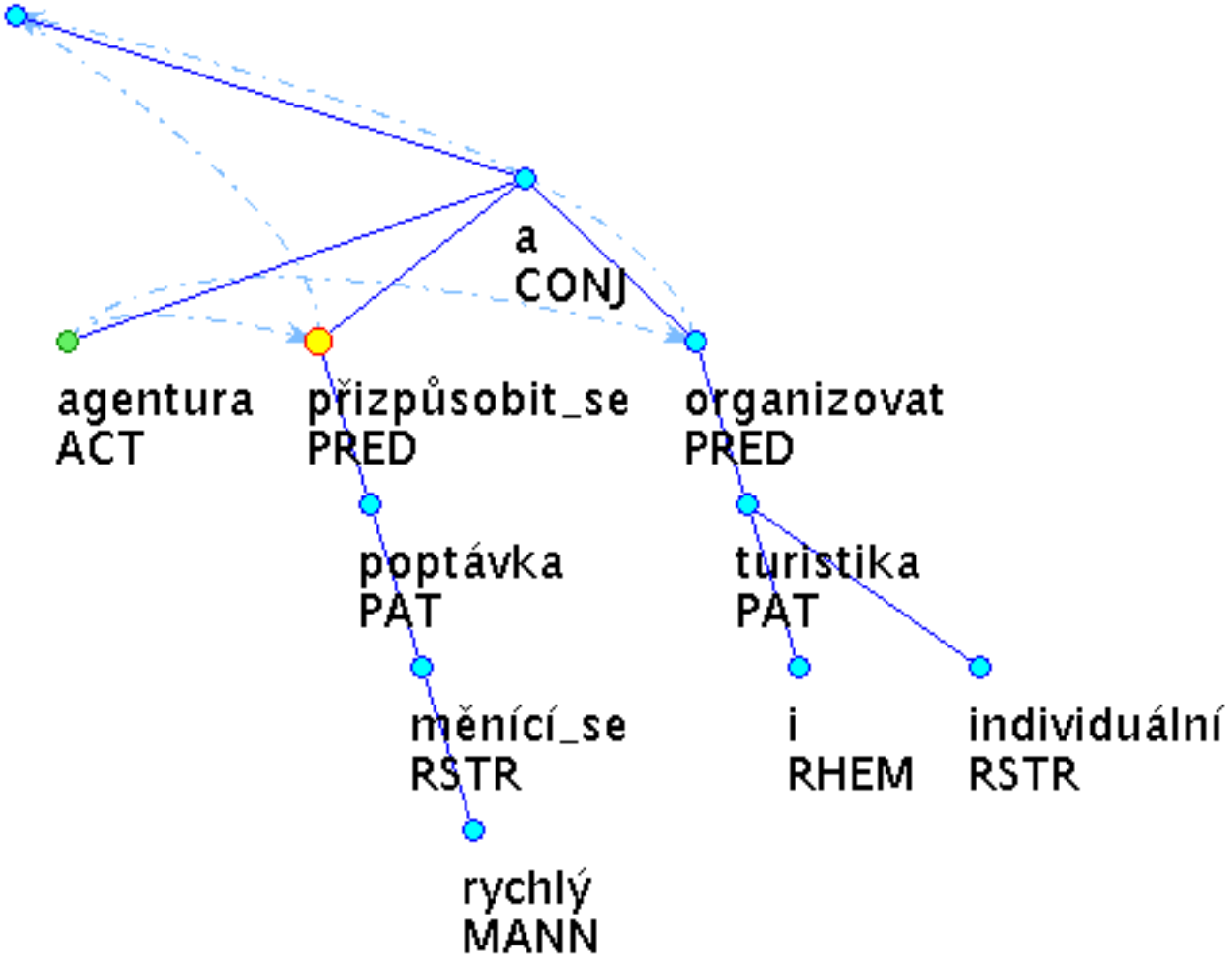
Czech: Myslím si, že udělal dobře, komentuje příchod Ronalda Ricardo.

English (lit.): I think that he did well, Ricardo says about Ronald's coming.

Using the Query Language
Effective Parentage

- A PREDicate effectively governing an ACTor (regardless of any possible combination of coordination)

Using the Query Language
Effective Parentage (Result)



Czech: Agentura se přizpůsobila rychle se měnící poptávce a organizuje i turistiku individuální.

English (lit.): The agency has adapted to a fast changing demand and organizes also an individual tourism.

References

Prague Dependency Treebank

- http://ufal.mff.cuni.cz/pdt

Netgraph home page

- http://quest.ms.mff.cuni.cz/netgraph