
# Korpuslinguistik

# Annis$_3$-Korpussuchtool

## Suchen in tief annotierten Korpora

Anke Lüdeling, Marc Reznicek, Amir Zeldes,
Hagen Hirschmann

... und anderen Mitarbeitern der HU-Korpuslinguistik

# Ziele

- Wie/Was kann man in ANNIS suchen?
  - Wortformen?
  - Linguistische Muster?
  - Token-Annotationen (Lemmata/Wortarten)?
  - Spannenannotationen
  - Syntaktische Annotationen?
    - Konstituenten
    - Dependenzen
- Wie sucht man nach mehreren/beliebigen Annotationen gleichzeitig?
- Wie filtert man nach Metadaten?

# Ziele

- Wie/Was kann man in ANNIS suchen?
  - Wortformen?
  - Linguistische Muster?
  - Token-Annotationen (Lemmata/Wortarten)?
  - Spannenannotationen
  - Syntaktische Annotationen?
    - Konstituenten
    - Dependenzen
- Wie sucht man nach mehreren/beliebigen Annotationen gleichzeitig?
- Wie filtert man nach Metadaten?

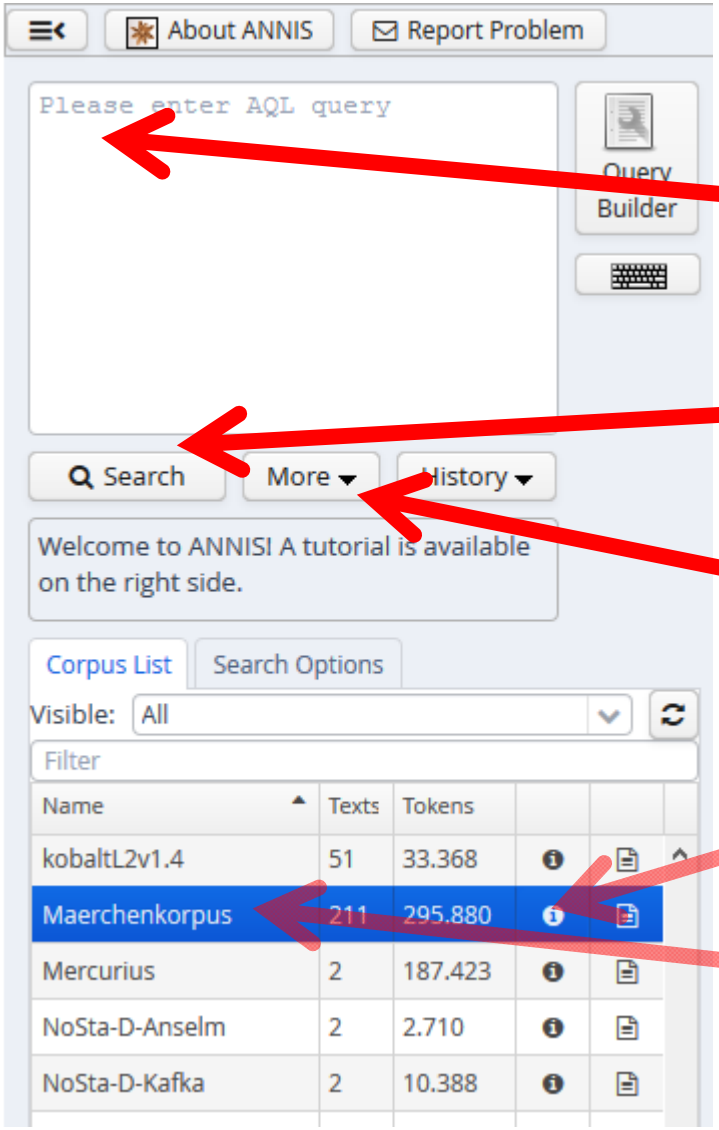
# Annis 3

- **Annis-Portal (öffentlich)**
  korpling.org/annis3/

- **Annis-Portal speziell für Lernerkorpora (Falko) (öffentlich)**
  korpling.org/falko-suche/

# Das Web-Interface: Abfrage

http://korpling.german.hu-berlin.de/falko-suche/search.html




Suche-Fenster (hier Eingabe)

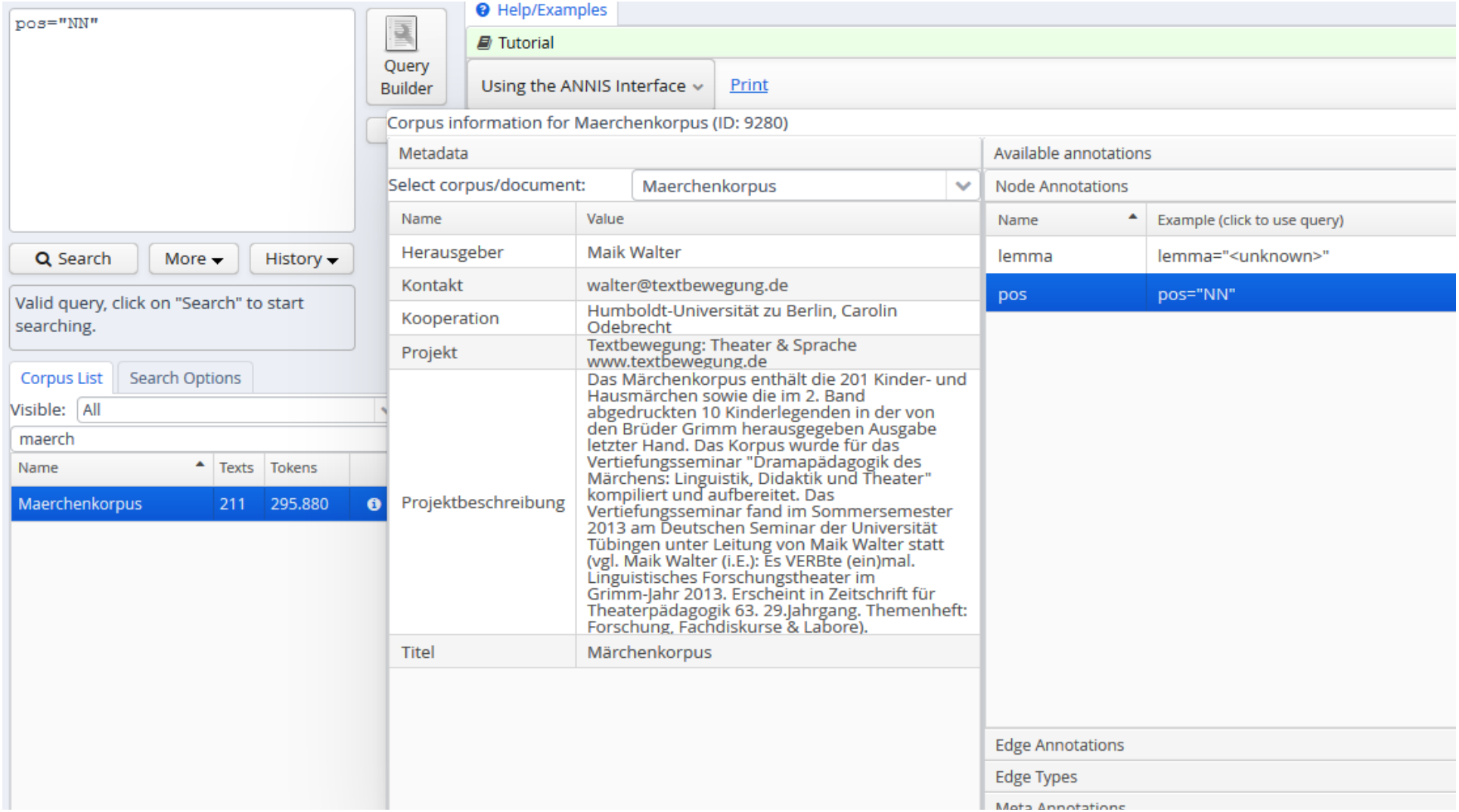
Abfrage starten

Trefferexport und Statistik

Pro Korpus Auflistung aller Annotationen und Metadaten

Auswahl der Korpora für die Suche (STRG+Klick für Mehrfachauswahl)

# Überführung exemplarischer Anfragen ins Suchfenster



# Prinzip I: Variablen-Wert-Paare

- Jedes Textkorpus enthält eine Ebene mit fortlaufendem, tokenisiertem Text
- Meistens: "tok"
- tok = "Frau"
  → Findet alle Vorkommen von "Frau" im Text

| **tok** | Sofern | die | Frau | herrscht |
|---|---|---|---|---|

# Prinzip I: Variablen-Wert-Paare

- tok = "Frau"
  → "Finde mir alle Vorkommen von 'Frau' auf der Ebene (Variable) 'tok' "
- tok
  → "Finde mir alle Vorkommen auf der Ebene 'tok' "
  → Korpus explorieren

# Prinzip I: Variablen-Wert-Paare

- Jedes Korpus in ANNIS enthält Annotationen (linguistische Informationen über die Sprache im Korpus)
- Unterschiedliche Annotationen werden auf Annotationsebenen gespeichert
- Meistens:
  - pos → Wortart
  - lemma → Lemma (Grundform)

# Prinzip I: Variablen-Wert-Paare

- pos = "NN"
  → "Finde mir alle Vorkommen von 'NN' auf Ebene (Variable) 'pos' "
  ('NN' steht für 'normales Nomen)
- pos
  → "Finde mir alle Vorkommen auf der Ebene 'pos' "
  → "Finde alle Wortarten"

# Prinzip I: Variablen-Wert-Paare

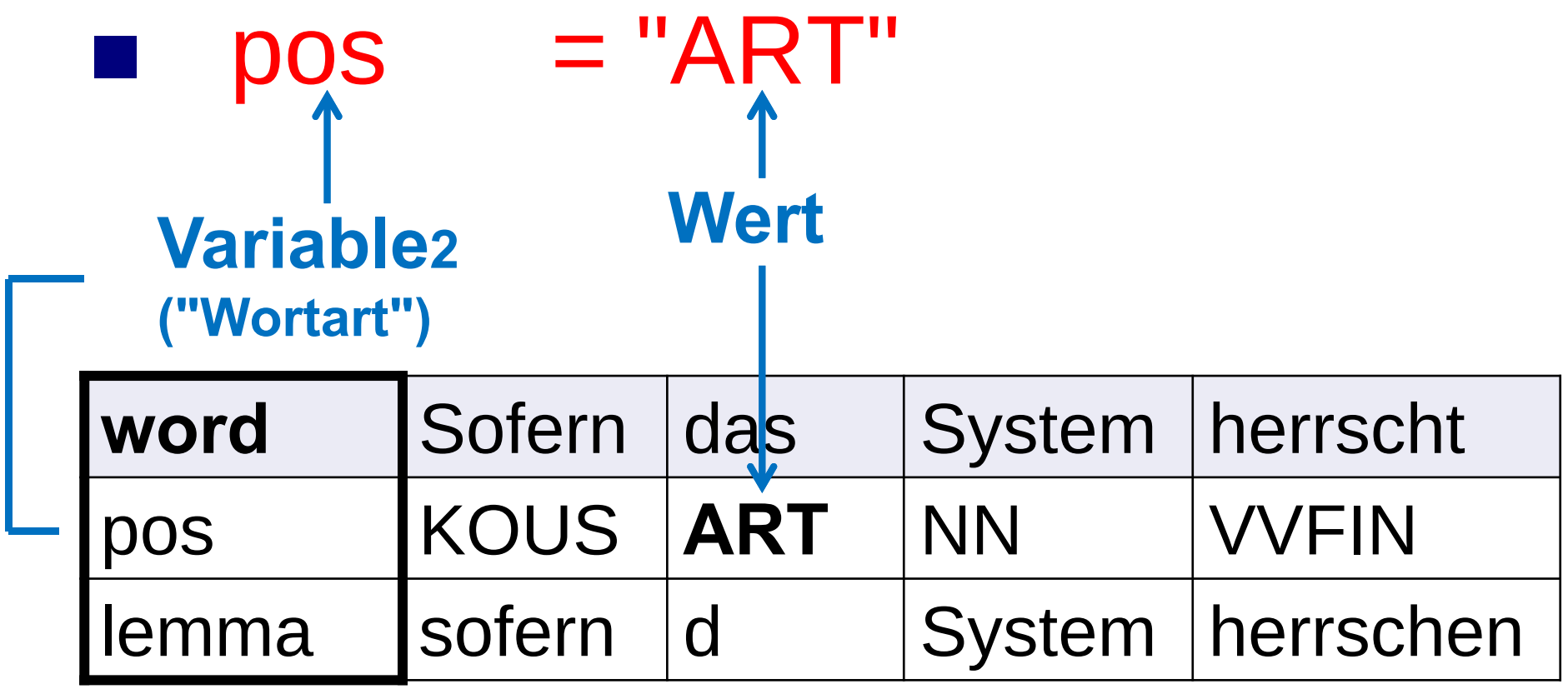
- tok = "das"

**Variable1 ("Wortform")** **Wert**

| **tok** | Sofern | **das** | System | herrscht |
|---|---|---|---|---|
| pos | KOUS | ART | NN | VVFIN |
| lemma | sofern | d | System | herrschen |

# Prinzip I: Variablen-Wert-Paare

- pos = "ART"




| **word** | Sofern | das | System | herrscht |
|---|---|---|---|---|
| pos | KOUS | **ART** | NN | VVFIN |
| lemma | sofern | d | System | herrschen |

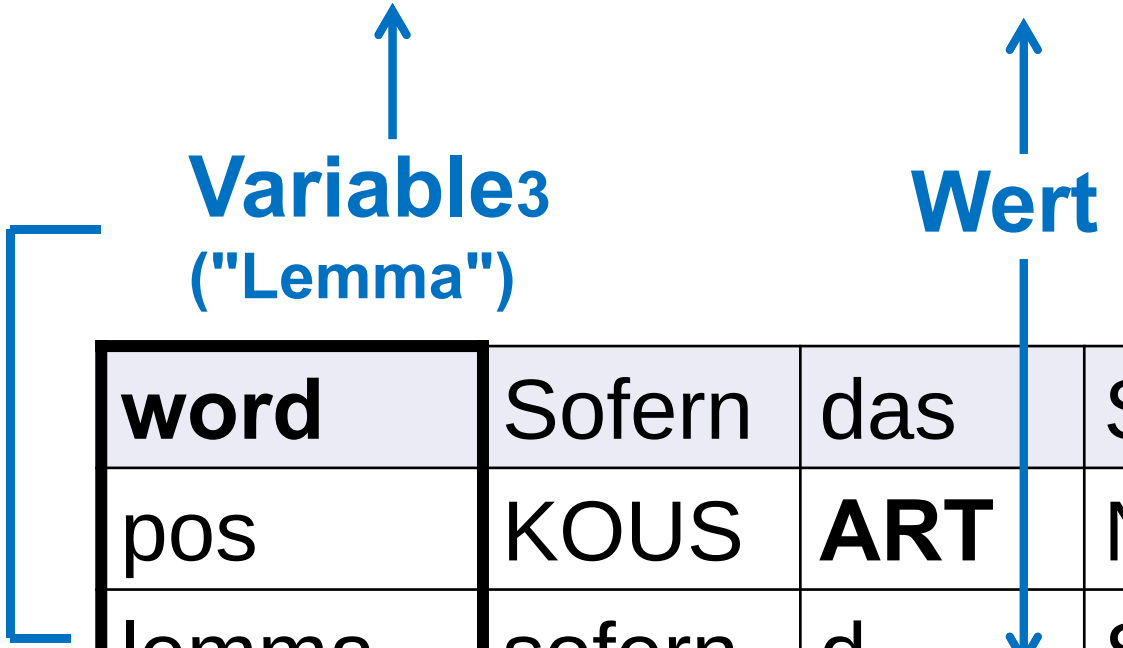
# Prinzip I: Variablen-Wert-Paare

- pos = "NN"

…findet *Riesen, Frauen, Student, …*

# Prinzip I: Variablen-Wert-Paare

- lemma = "d"




| **word** | Sofern | das | System | herrscht |
|---|---|---|---|---|
| pos | KOUS | **ART** | NN | VVFIN |
| lemma | sofern | d | System | herrschen |

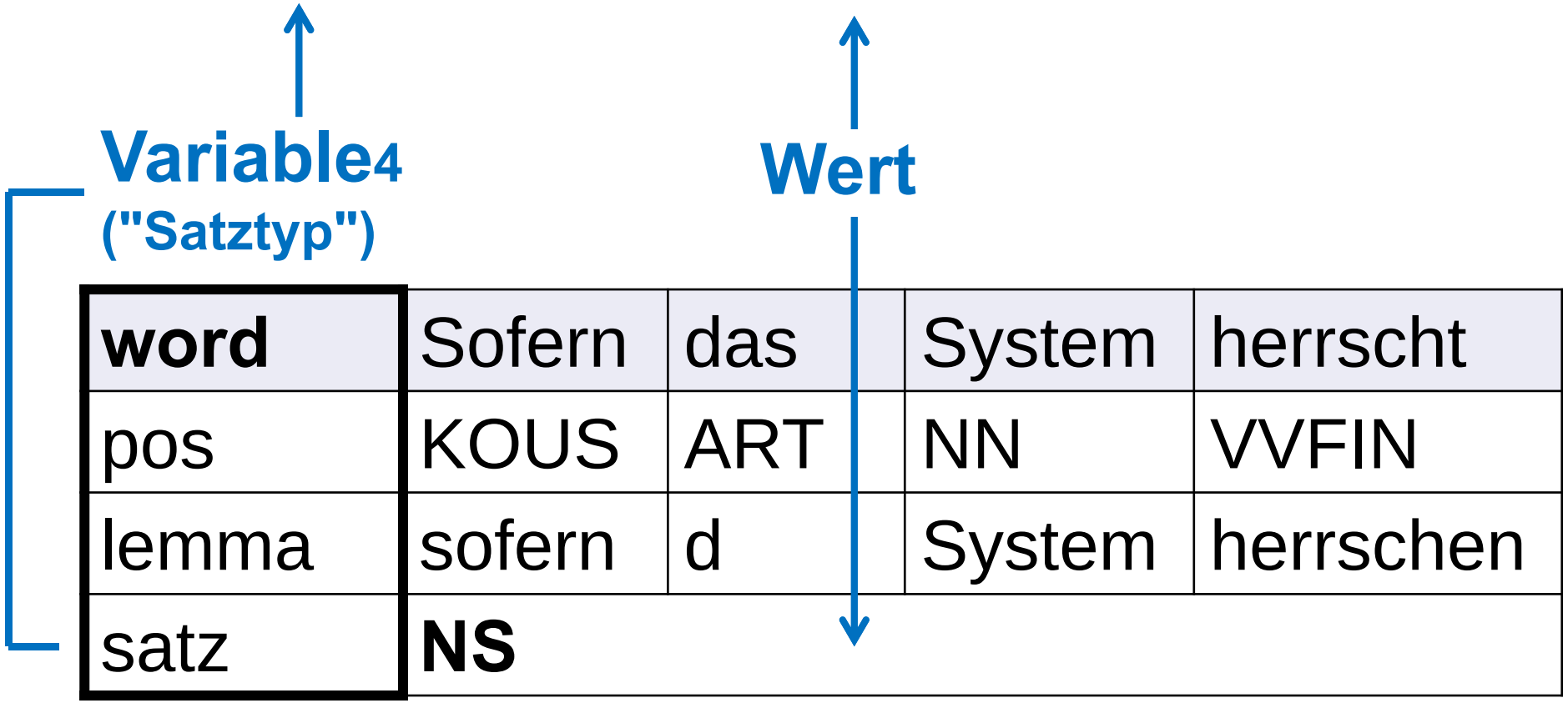
# Prinzip I: Variablen-Wert-Paare

- lemma = "d"

…findet *die, dem, den, …*

# beliebig erweiterbar…

- satz = "NS"




| **word** | Sofern | das | System | herrscht |
|---|---|---|---|---|
| pos | KOUS | ART | NN | VVFIN |
| lemma | sofern | d | System | herrschen |
| satz | **NS** | | | |

# beliebig erweiterbar…

- satz = "NS"

…findet alle Nebensätze wie
*Sofern das System herrscht*

(sofern die Daten wie gezeigt annotiert sind)

# Suche nach Strings

- **Suchen Sie nach allen Vorkommen der Wortform "meinen" in FalkoEssayL2V2.4**

```
tok = "meinen"
```

  - Was wird gefunden?
  - Ist das interessant?
  - Was wird nicht gefunden, was interessant sein könnte?

# Lemmata

- "Basisformen" von Wörtern
- **Suchen Sie nach allen Vorkommen der Formen des Verbs *meinen*:**

```
lemma = "meinen"
```

- → Problem: Lemmatisierung ist willkürlich; man muss wissen, wie lemmatisiert wurde.
- Beispiel: Lemma von *sich*

# Lemmata

- "Basisformen" von Wörtern
- **Suchen Sie nach allen Vorkommen der Formen des Possessivartikels:**

```
lemma = "mein"
```

# Mustersuche (reguläre Ausdrücke)

- Annis erlaubt Mustersuchen auf allen Annotationsebenen
- Mustersuchen werden statt in " " in / / eingefügt
- Z. B. kann man damit nach allen Wörtern suchen, die *…mein…* enthalten.

```
tok = /.*mein.*/
```

# Mustersuche: Joker .

- ein beliebiges Zeichen — al. → *als*, *alt*, ...
- zwei beliebige Zeichen — al.. → *alle*, *alte*, *also*
- drei beliebige Zeichen — al... → *alles*, *altes*, *alias*, ...

# Aufgabe

- **Welche Wortformen bekommen Sie mit?**

tok = /g.b./

# Mustersuche: **?** und ***** **+**


das vorherige Zeichen ist optional

→ϕ, s → *da, das*


das vorh. Zeichen kommt 0- bis ∞mal vor

→ ϕ, s, ss,… → *da, das, dass, dasssssssss*


das vorh. Zeichen kommt 1- bis ∞mal vor

→ s, ss,… → *das, dass, dasssssssssss*

# Aufgabe

- **Was passiert, wenn Sie die Operatoren kombinieren?**

```
tok = /Mann.?/
```

```
tok = / Mann.*/
```

```
tok = / Mann.+/
```

# Aufgabe

- **Versuchen Sie alle Wörter (Grundformen) zu finden, die auf *-lang* enden.**

# Aufgabe

- **Versuchen Sie alle Wörter (Grundformen) zu finden, die auf *-lang* enden.**

```
lemma = /.+lang/
```

**Treffer z.B.:**

*bislang*
*lebenslang*
*jahrelang*

# Aufgabe

- **Versuchen Sie alle Wörter (Grundformen) zu finden, die mit *lang-* beginnen.**

```
lemma = /lang.+/
```

**Treffer z.B.:**

*lange*
*langsam*
*langweilig*

# Gruppieren mit ()

- mit () kann man Ausdrücke als zusammengehörige Gruppen behandeln

```
tok = /(ja)+/
```

**findet**

*ja*
*jaja*
*jajaja*
...

# Alternativen: a oder b = (a|b)

- Mit Klammern und | ("oder") kann man gleichzeitig nach verschiedenen Wörtern suchen:

```
tok = /(Mann|Frau|Kind)/
```

- Nach verschiedenen Formen:

```
tok = /(Mann|Mannes)/
```

- Oder Zeichenketten:

```
tok=/bes(ser|t).?/
```

# Aufgabe

- **Finden Sie alle Formen des Verbs *meinen* im Präsens, aber keine anderen Formen.**

| | |
|---|---|
| mein | e |
| mein | st |
| mein | t |
| mein | en |
| mein | t |
| mein | en |

# Lösung

```
tok =/mein(s?t|en?)/
```

oder

```
tok =/mein(e|st|t|en)/
```

→ **Häufig gibt es alternative Suchanfragen für dieselben Treffermengen.**

# Lösung

t 2t|en?)/

oder

tok =/mein(e|st|t|en?)

- Problem:
  - Infinitiv
  - Formen des Possessivpronomens

# Suche nach Wortart

- Es gibt unterschiedliche Wortartensysteme (→Tagsets) für Korpora
- allgemein in der Linguistik unterschiedliche Wortartensysteme
- Die meisten deutschen Korpora benutzen das Tagset STTS
  - ADJA attributives Adjektiv
  - ADV Adverb
  - ART Artikel
  - NN normales Nomen
  - VVFIN finites Verb

  …

http://www.ims.uni-stuttgart.de/projekte/corplex/TagSets/stts-table.html

# Stuttgart-Tübingen-Tagset (STTS)

| ADJektiv | Nomen | Pronomen | Verb | ParTiKel | KOnjunktion |
|---|---|---|---|---|---|
| ADJA | NN | PDS | VVFIN | PTKZU | KOUI |
| ADJD | NE | PDAT | VVIMP | PTKNEG | KOUS |
| | | PIS | VVINF | PTKVZ | KON |
| | | PIAT | VVIZU | PTKANT | |
| | | PIDAT | VVPP | PTKA | |
| | | PPER | VAFIN | | |
| | | PPOSS | VAIMP | | |
| | | PPOSAT | VAINF | | |
| | | PRELS | VAPP | | |
| | | PRELAT | VMFIN | | |
| | | PRF | VMINF | | |
| | | PWS | VMPP | | |
| | | PWAT | | | |
| | | PWAV | | | |

# Stuttgart-Tübingen-Tagset (STTS)

| **VERB** | **Vollverb** | Auxiliar | Modalverb |
|---|---|---|---|
| finit | **V**VFIN | **V**AFIN | **V**MFIN |
| Imperativ | **V**VIMP | **V**AIMP | |
| infinit | **V**VINF | **V**AINF | **V**MINF |
| Infinitiv mit *zu* | **V**VIZU | | |
| Partizip 2 | **V**V**PP** | **V**A**PP** | **V**M**PP** |

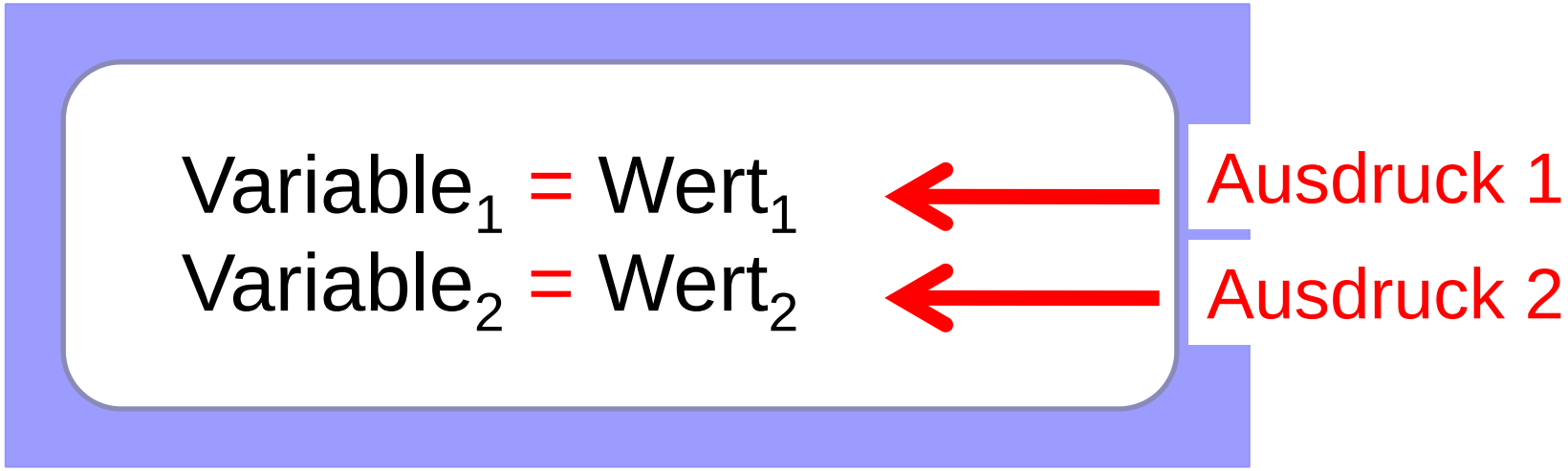
# Aufgabe

- **Suchen Sie in DDB.FNHD nach Possessiva**

pos =/PPOS(S|AT)/

# Prinzip II: Mehrere Suchbedingungen eingeben und Verknüpfen

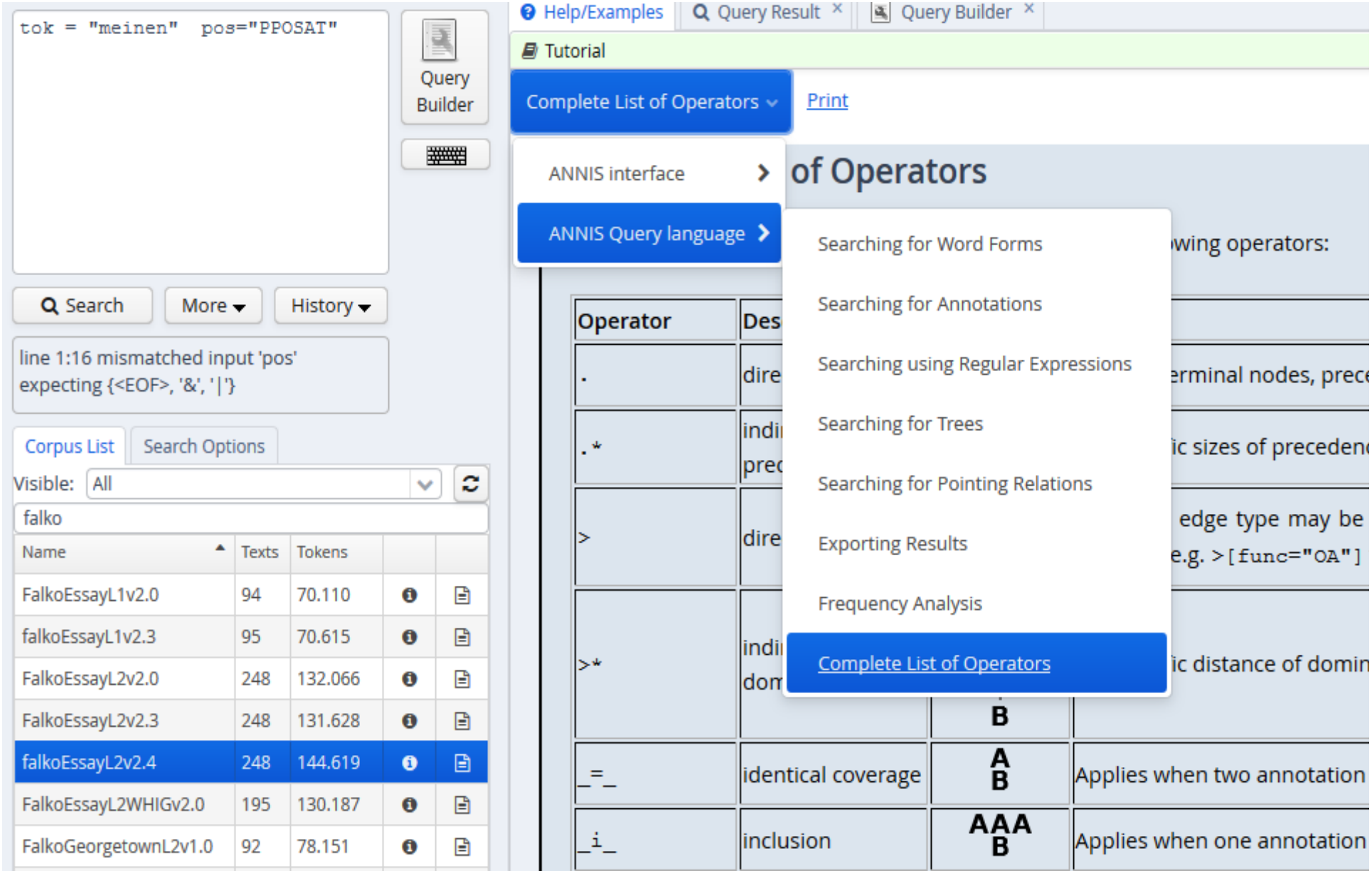
- Wir wollen mehrere Bedingungen an die Suche stellen, z. B. nach dem Lemma "auf" suchen, und dies soll gleichzeitig die Wortart "Verbpartikel" haben, weil wir keine Präpositionen finden wollen
- Einzelne **V**ariable-**W**ert-Paare werden durch einen passenden Befehl verbunden.



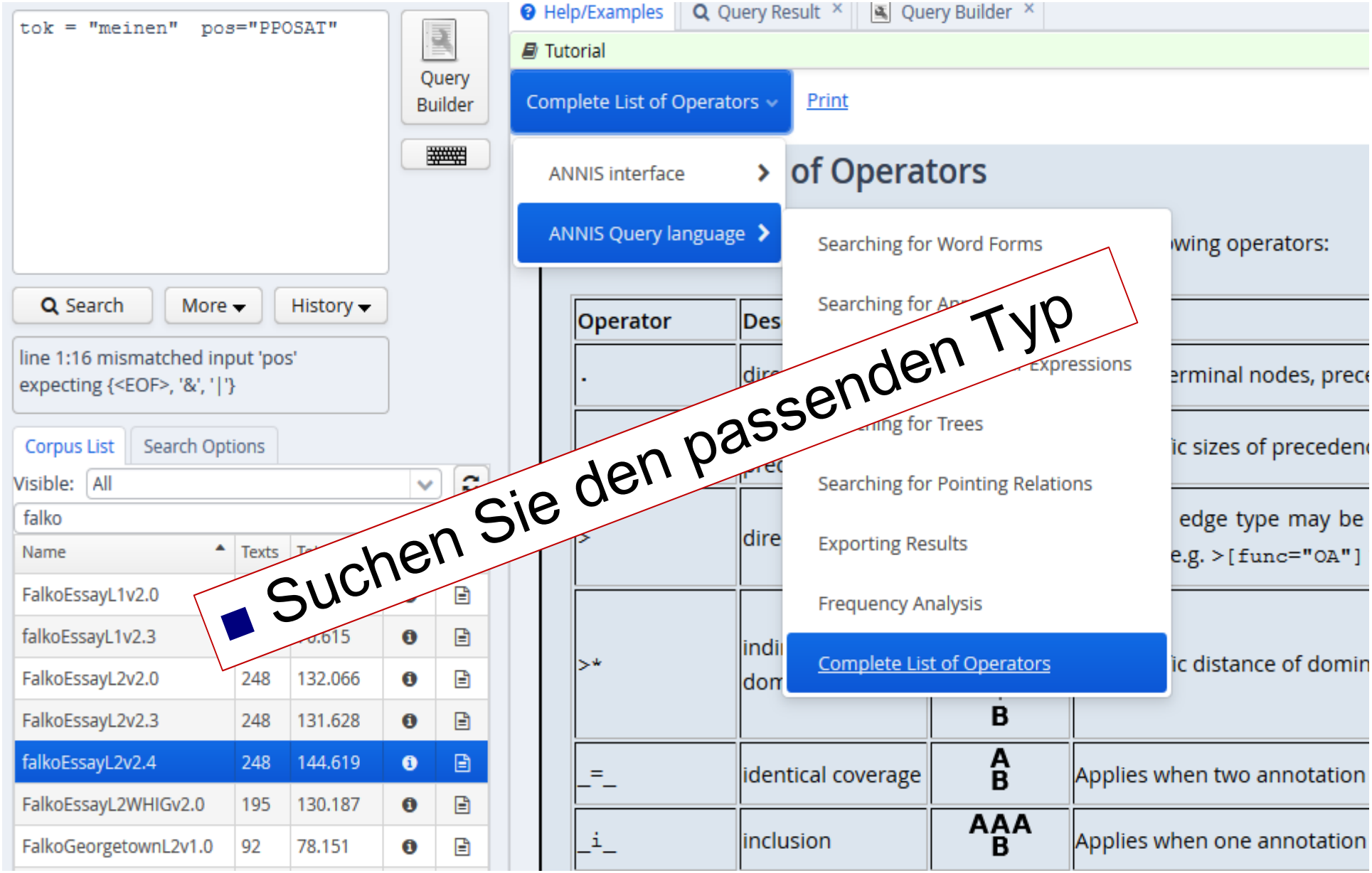
# Prinzip II: Verknüpfungstypen

- Wir können bereits die zwei Suchanforderungen für "Such mir alle Vorkommen von "auf", die Verbpartikeln sind" formulieren:
  lemma = "auf"
  pos="PTKVZ"
- Wir benötigen nun die passende Verknüpfung dieser Suchanforderungen

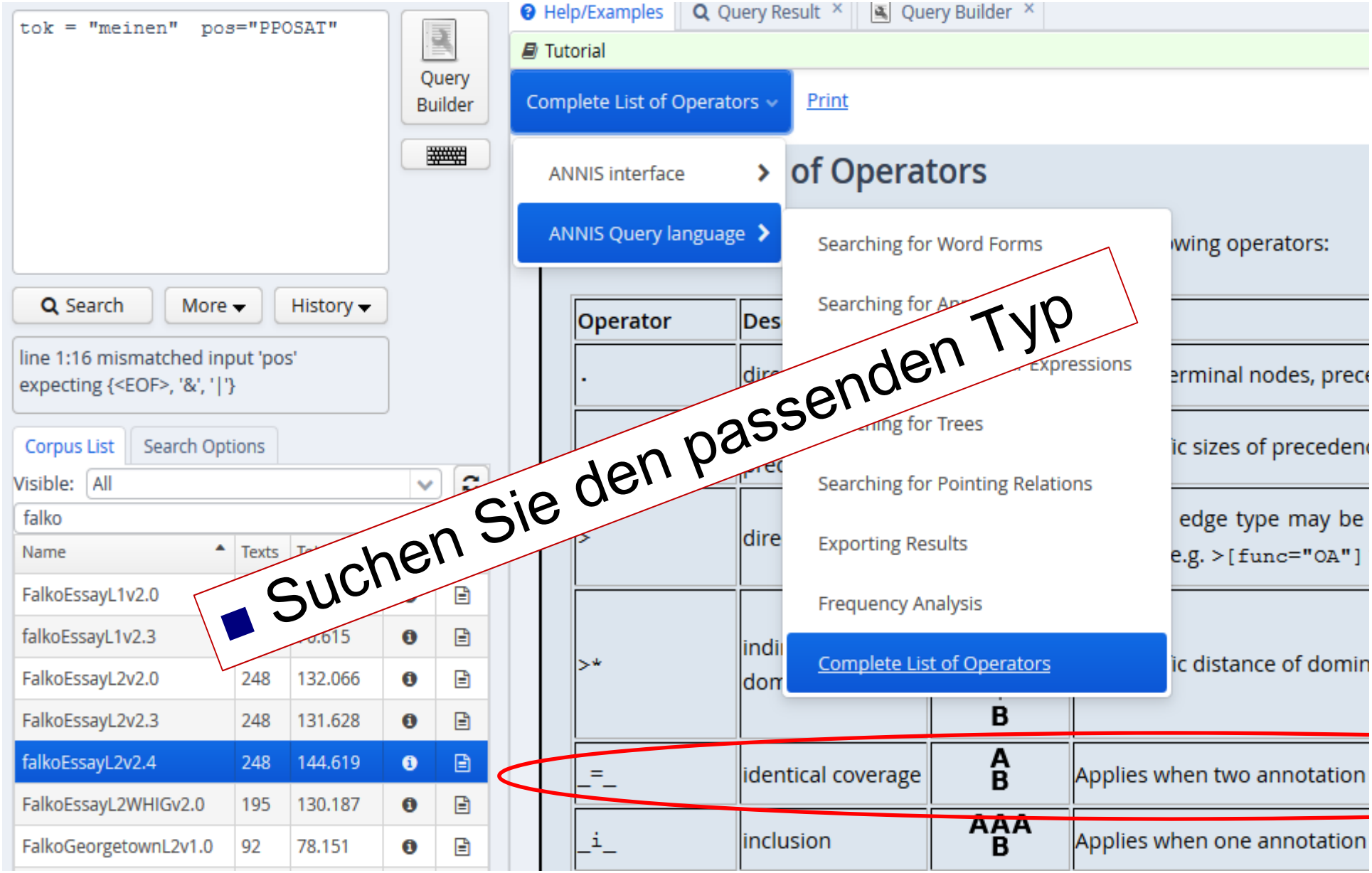
# Sämtliche Verknüpfungstypen in ANNIS



# Sämtliche Verknüpfungstypen in ANNIS



# Sämtliche Verknüpfungstypen in ANNIS



# Lösung

- **Suchanfrage für *auf* in der Funktion einer Verbpartikel**

**lemma =** "auf"
**_=_**
**pos= "PTKVZ"**

# Aufgabe

- **Finden Sie nun im Märchenkorpus Vorkommen von** lemma = "gehen" , die ausschließlich Imperative sind.

**lemma =** "gehen"
**_=_**
**pos= "VVIMP"**

# Negation !=

- ! bedeutet Negation
  - Der Operator wird vor dem "="-Zeichen eingefügt.
  - **Finden Sie im Märchenkorpus alle Vorkommen von** lemma ="denn" **, die keine Konjunktionen sind. Welche Wortart ist das? Welches Tag wird vergeben?**

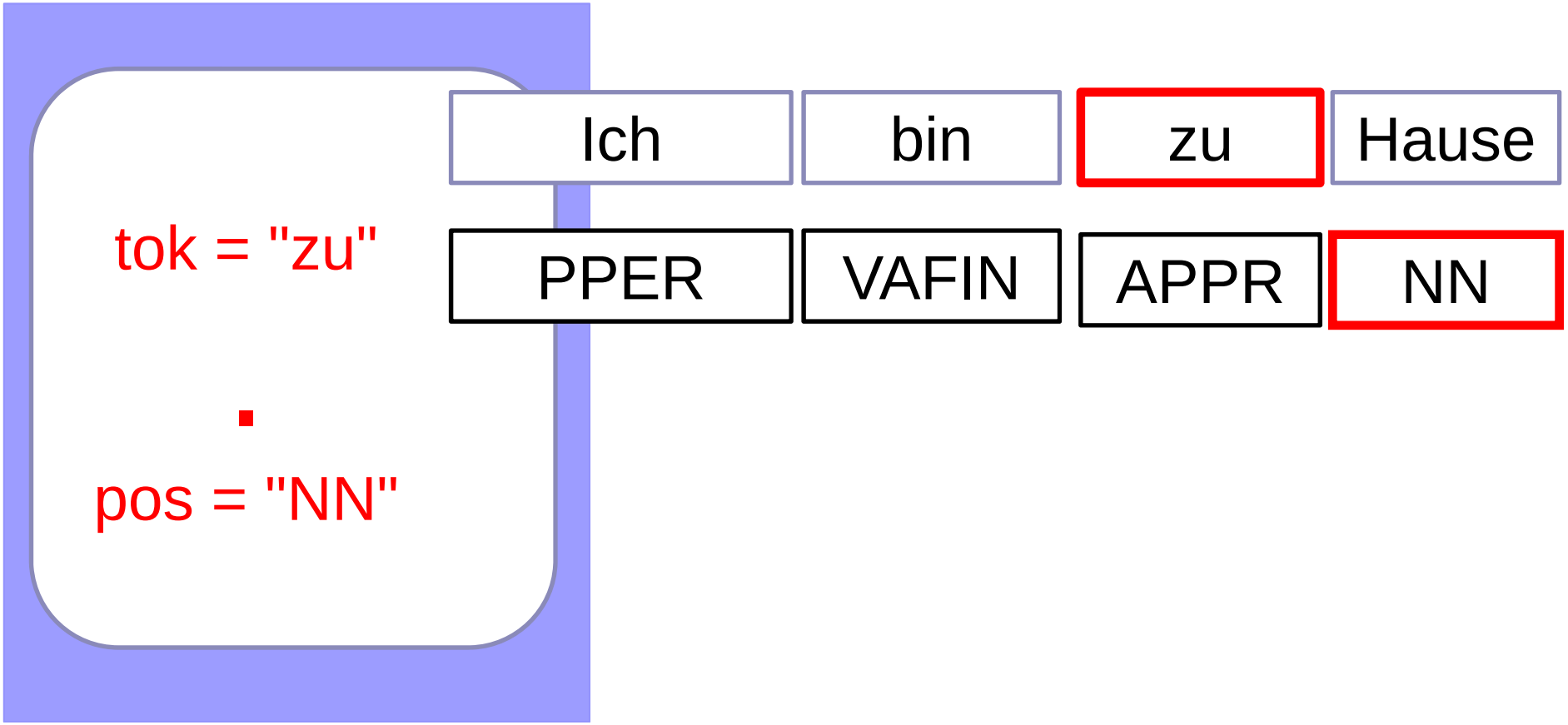
lemma ="denn" _=_ pos!="KON"

# Aufgabe

- **Finden Sie im Märchenkorpus alle <u>Abfolgen</u> von Präpositionen und Nomina (ohne Artikel oder Adjektiv dazwischen)**
- **(→Wieder zwei Bedingungen, aber jetzt ist die Verknüpfung der Bedingungen eine andere)**

pos ="APPR" . pos="NN"

# Suche nach Abfolgen:
# z.B. Nomen folgt auf "zu"



# Aufgabe

- **Finden Sie in DDB.MHD alle Satzbeendungszeichen, die drei Stellen hinter einem finiten Modalverb stehen**

```
pos="VMFIN" .3  pos =/\$\./
```

# Ambige Zeichen: Operatoren und Wort-/Satzzeichen

- Erinnerung: Der Operator "**.**" bedeutet "**ein beliebiges Zeichen**", wenn ich die Suche in / / formuliere
- Wie finde ich aber Wörter mit Abkürzungspunkten, wenn ich in der Suchanfrage reguläre Ausdrücke verwende?
- Der Operator "****" bedeutet

"**das folgende Zeichen ist wörtlich gemeint**".

# Ambige Zeichen: Operatoren und Wort-/Satzzeichen

- tok="usw."
  →Findet alle Vorkommen von ***usw.***
- **tok=/(B|b)./**
  **→ Findet in falkoEssayL2v2.4**
  ***B.*** **und** ***b.*** **(Kontext: *z. B.*),**
  **aber auch *BA* und *BH***
- tok=/**(b|B)\./**
  **→ Findet nur *B.* und *b.***

# Aufgabe - Statistik

- **Finden Sie in DDB.AHD alle Präpositionen**
- **Lassen Sie sich mithilfe der Funktion**
  **More>Frequency Analysis**
  **die Lemmata anzeigen**
- **Sie müssen hierzu die Suchanfrage**
  **pos="APPR"**
  **formulieren und in das Feld**
  **Select annotation of node**
  ***lemma***
  **eingeben (dann unten Mitte abschicken)**

# Zusammenfassung: reguläre Ausdrücke

| | |
|---|---|
| **.** | Ein beliebiges Zeichen |
| ***** | Beliebig viel (0 bis unendlich vom vorherigen Element) |
| **+** | Mindestens einmal (vorheriges Element) |
| **?** | Optional (vorheriges Element) |
| **** | wörtlich (folgendes Zeichen) |
| **!** | nicht |
| **[**abc**]** | Menge (oder **[^**abc**]** = alles *außer* die Menge) |
| **(**a**\|**b**)** | a oder b |
| a**{**2**,**3**}** | a 2 bis 3 mal |

# Zusammenfassung: Operatoren zur Verknüpfung von Suchanforderungen

**X . Y** → Y folgt direkt auf X

**X .2 Y** →
Y folgt nach 2 Stellen auf X

**X .1,3 Y** → Y folgt im Abstand von 1-3

**X .* Y** → Y folgt irgendwann auf X

**X _=_ Y** → Y deckt sich ab mit X
(bzw.: X und Y treffen auf dasselbe Element zu)

# Zusammenfassung

- Mit ANNIS kann man:
  - in unterschiedlichen Korpora (auch gleichzeitig) suchen
  - die Ergebnisse quantifizieren
  - die Ergebnisse exportieren
- Man kann auch nach Metadaten filtern (Tutorial)

Herzlichen Dank!