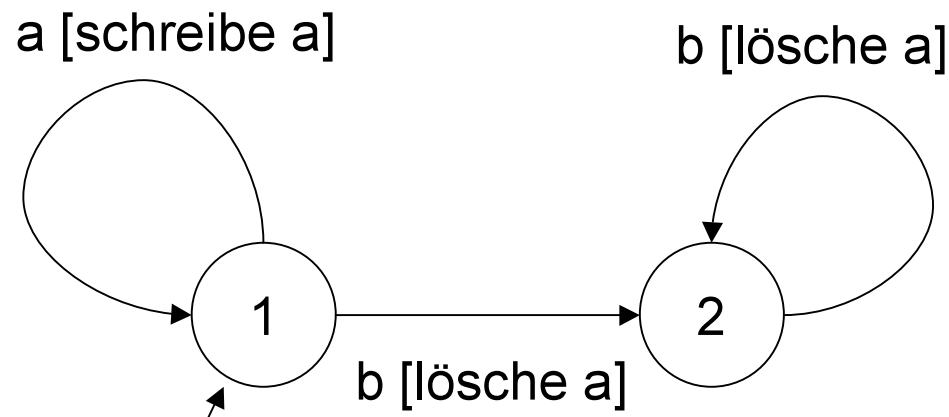


# Syntaktische Verarbeitung

- Kontextfreie Grammatiken erlauben die einfache und elegante Beschreibung syntaktischer Strukturen (ähnlich wie Zustandsdiagramme/deterministische endliche Automaten das für morphologische Strukturen tun).
- Wie sehen Analysesysteme für kontextfreie Sprachen aus (syntaktische Parser)?
- Die einfachste Version ist der "Kellerautomat": Wir nehmen einen Automaten an, der zusätzlich zu Zuständen und Übergängen über einen Speicher als Stapel (Stack, Keller) verfügt. Der Automat liest Symbole der Eingabekette und kann bei jedem Übergang Symbole in den Stack schreiben oder vom Stack nehmen. Eine Eingabe wird akzeptiert, wenn ein Endzustand erreicht, das Eingabewort abgearbeitet, und der Stack leer ist.

# Ein Beispiel

- Ein Kellerautomat, der  $a^n b^n$  akzeptiert:



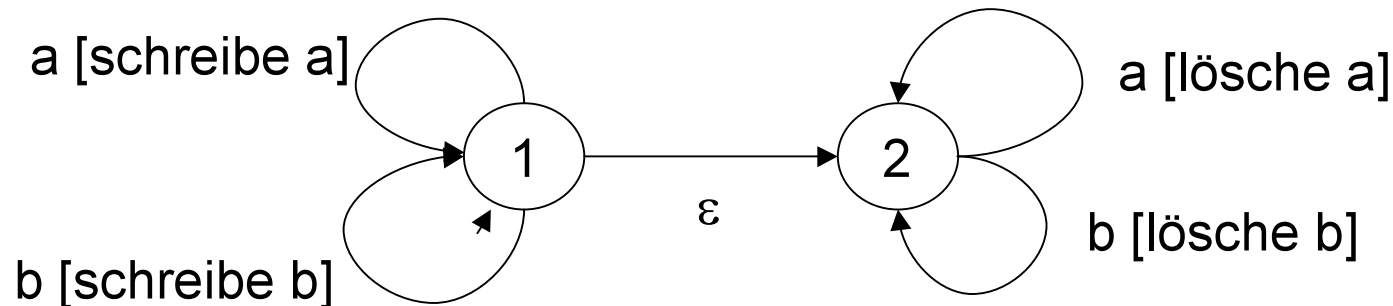
- Im Zustand 1 werden a's gelesen und in den Stack geschrieben. Beim ersten b wechselt der Automat in den Zustand 2 und löscht für jedes gelesene b ein a vom Stack. Eingabe und Stack sind beide leer, wenn die Eingabe gleichviele a's und b's enthielt. (Beide Zustände sind Endzustände.)

# Deterministische syntaktische Verarbeitung?

- Der Automat aus Beispiel 1 ist deterministisch: Er führt in jedem Zustand bei jedem Eingabesymbol eine bestimmte Aktion aus, benötigt deshalb keine aufwändigen Suchverfahren und analysiert jede Eingabe in linearer Zeit, ganz wie ein DEA,
- Gibt es eine generelle Möglichkeit, kontextfreie Grammatiken (analog zum NEA-DEA-Überführungsalgorithmus) in ein Format zu übersetzen, das deterministisches Parsing erlaubt?
- Die Antwort ist Nein. Viele kontextfreie Sprachen sind inhärent nicht-deterministisch (dazu auch das Beispiel auf der nächsten Folie).

## Ein einfaches Beispiel

- Die Sprache  $L = \{ww^R \mid w, w^R \in \{a,b\}^*\}$  kann nicht deterministisch geparkt werden ( $w^R$  ist die Spiegelung von  $w$ ).
- Auch dieser Automat hat zwei Zustände. Bis zur Mitte des Eingabewortes speichert er a's und b's im Stack, dann springt er in den zweiten Zustand, in dem er neu gelesene Symbole mit den gespeicherten Stack-Symbolen vergleicht.



- Problem: Woher weiß der Automat, dass er die Mitte des Wortes erreicht hat. Er muss raten, und wenn er falsch geraten hat, zurücksetzen und eine andere Option verfolgen.

# Verarbeitung von CFGs für natürliche Sprachen

- Programmiersprachen, Sprachen der Logik und Mathematik sind meist so konzipiert, dass sie deterministische Analyse zulassen; eine einfache Möglichkeit, Nicht-Determinismus zu vermeiden, besteht in der Verwendung von Klammern.
- Natürliche Sprachen lassen keine deterministische Analyse zu, sie erfordern nicht-lineare Suchverfahren /Parsingalgorithmen, die im besten Fall quadratische Zeit erfordern (d.h., der Zeitbedarf für eine Eingabe der Länge  $n$  beträgt  $c \cdot n^2$ ;  $c$  ist ein konstanter Faktor).

# Konstituentenstruktur

- Die Erzeugung eines Satzes mit einer kontextfreien Grammatik (und die Verarbeitung des Satzes mit einem Parser, der die Grammatik in ein Analysesystem umsetzt) liefert nicht nur die Information, ob der Satz zur Sprache gehört/"grammatisch ist". Als "Seiteneffekt" liefert sie den Ableitungsbaum/"Parse Tree", der die "Konstituentenstruktur" und damit einen wesentlichen teil der syntaktischen Struktur beschreibt.
- Eine Grammatik ist eine angemessene Beschreibung der syntaktischen Struktur einer Sprache, wenn sie
  - genau die grammatisch akzeptablen Sätze der Sprache erzeugt
  - für die grammatischen Sätze plausible Konstituentenstrukturen erzeugt.

# Grammatische Mehrdeutigkeit

- Natürlich-sprachliche Sätze sind oft grammatisch mehrdeutig. Die grammatische Mehrdeutigkeit ist relevant. Sie führt im allgemeinen zu semantischer Mehrdeutigkeit.
- Natürliche Sprachen benötigen deshalb kontextfreie Grammatiken, die Sätzen unterschiedliche syntaktische Strukturen zuweisen können (und deshalb übrigens per se nicht-deterministisch sind).

# Grammatische Mehrdeutigkeit, Beispiele 1

- Die folgende Grammatik erlaubt Präpositionalphrasen (PPs) als Satz- und als NP-Modifikatoren

$S \rightarrow NP VI$

$S \rightarrow NP VT NP$

$S \rightarrow S PP$

$NP \rightarrow ART NN$

$NP \rightarrow PN$

$NP \rightarrow NP PP$

$PP \rightarrow P NP$

- Der Satz: *Peter sah den Mann mit dem Teleskop* hat in dieser Grammatik zwei Analysen, die zwei Bedeutungsvarianten entsprechen:
  - $[_S \text{Peter} [_{VT} \text{sah} [_{NP} [_{NP} \text{den Mann} ] [_{PP} \text{mit dem Teleskop} ] ] ] ]$
  - $[_S [_S \text{Peter} [_{VT} \text{sah} [_{NP} \text{den Mann} ] ] ] [_{PP} \text{mit dem Teleskop} ] ]$
- Der Satz *Peter rief den Kollegen aus München mit dem Handy an* hat mindestens 4 Lesarten (welche?).



## Eigenschaften der syntaktischen Struktur [3]

- Sätze setzen sich aus Satzteilen (Konstituenten) zusammen, die wiederum aus einfachen oder ihrerseits komplexen Satzteilen bestehen können. Sätze können deshalb beliebig lang und beliebig tief geschachtelt sein.
- Die Syntax natürlicher Sprachen erlaubt variable Wortstellung: Wörter und Konstituenten mit der gleichen Funktion können an unterschiedlichen Stellen eines Satzes stehen. Unterschiedliche Sprachen erlauben sehr unterschiedliche Freiheitsgrade.
- Die grammatischen Eigenschaften unterschiedlicher Wörter und Konstituenten im Satz hängen voneinander ab – zum Teil auch in Fällen, in denen die Wörter und Konstituenten im Satz weit auseinander liegen.

# Interaktion von grammatischen Merkmalen

- Kongruenz: bestimmte grammatische Merkmale verschiedener Ausdrücke müssen übereinstimmen
- Rektion: ein Ausdruck (lexikalischer Kopf) verlangt das Vorliegen bestimmter Merkmale bei abhängigen Ausdrücken
- Beispiele für Kongruenz:
  - in Nominalausdrücken (Numerus, Genus, Kasus)  
*der [sg,nom,m]interessierte [sg,nom,m] Student [sg,nom,m]*  
*dem[sg,dat,m] interessierten[sg,dat,m] Studenten[sg,dat,m]*
  - Subjekt-Verb-Kongruenz (Person, Numerus)  
*Der Student[sg] arbeitet[sg] / Die Studenten[sg] arbeiten[sg]*
- Beispiele für Rektion (Kasus)
  - *die Übungen [akk] abgeben / der Dozentin [dat] zuhören*
  - *mit den Kollegen [dat] / ohne die Kollegen [akk]*
  - *dem Fach [dat] zugetan / des Faches [gen] überdrüssig*

## Ein komplexerer Fall: Relativpronomen

Das Relativpronomen bekommt seine grammatischen Merkmale durch zwei unterschiedliche Mechanismen zugewiesen:

- Numerus , Genus-Kongruenz mit dem Kopf der Nominalphrase
- Kasusreaktion durch das Hauptverb des Relativsatzes

*der Student [m,sg], den[m,sg,akk] die Computerlinguistik interessiert*  
*der Student [m,sg], dem [m,sg,dat] die Computerlinguistik gefällt*  
*die Studentin [f,sg], die [f,sg,akk] die Computerlinguistik interessiert*  
*die Studentin [f,sg], der [f,sg,dat] die Computerlinguistik gefällt*

# Die Beschreibung von Merkmalsstrukturen

- Die Interaktion von grammatischen Merkmalen ist neben der Konstituentenstruktur ein zweiter wichtiger Bereich für die grammatische Analyse. Das wichtigste Werkzeug zur Beschreibung von Merkmalsstrukturen ist die Operation der Unifikation.
- Unifikation testet die Verträglichkeit von zwei Merkmalsstrukturen (z.B. Kongruenzmerkmalen innerhalb einer NP oder zwischen NP und Verb). Wenn die Merkmalsstrukturen verträglich sind, liefert die Operation als Resultat eine neue Merkmalsstruktur, die die Vereinigung der Merkmalsinformationen in den Ausgangsstrukturen enthält.
- Die wichtigsten existierenden Grammatikformalismen (z.B. LFG, HPSG) verwenden in irgendeiner Form eine Kombination aus "kontextfreiem Gerüst"+ Unifikation für die Merkmalsbehandlung.
- Wir haben leider keine Zeit, Unifikation näher zu erläutern.

# Stand der Technik in der syntaktischen Verarbeitung

- Syntaktische Verarbeitung komplexer Sätze mit großen Grammatiken war bis vor kurzem ein großes Problem. Inzwischen sind hier große Fortschritte auf dem Weg zu effizienter Verarbeitung erzielt worden.
- Die Abdeckung von Grammatiken ist immer noch problematisch: Bisher existiert keine Grammatik für das Deutsche (oder auch das Englische, obwohl man hier ein bisschen weiter ist), die eine verlässliche und vollständige syntaktische Analyse von Zeitungstexten erlaubt.
- In vielen Fällen benötigt man aber keine kompletten oder 100% sicheren syntaktischen Analysen. Für sprachtechnologische Anwendungen z.B. im Bereich von Dialog oder Informationsextraktion hilft man sich mit Parsern, die sehr schnell sind, aber nur partielle Analysen (z.B. von einfachen NPs und PPs) liefern: "flache syntaktische Verfahren", statistische Analyse (s. Folien 8).