

Interdisziplinäre  
fachdidaktische Übung:  
Modelle für  
formale Sprachen

SS 2015: Grossmann, Jenko

# Modelle für formale Sprachen

Die Beschreibung orientiert sich am Begriffssystem der Beschreibung natürlicher Sprachen

Formale Sprache steht dabei stellvertretend für alle drei Begriffe

- Formale Sprache
- Programmiersprache
- Computersprache

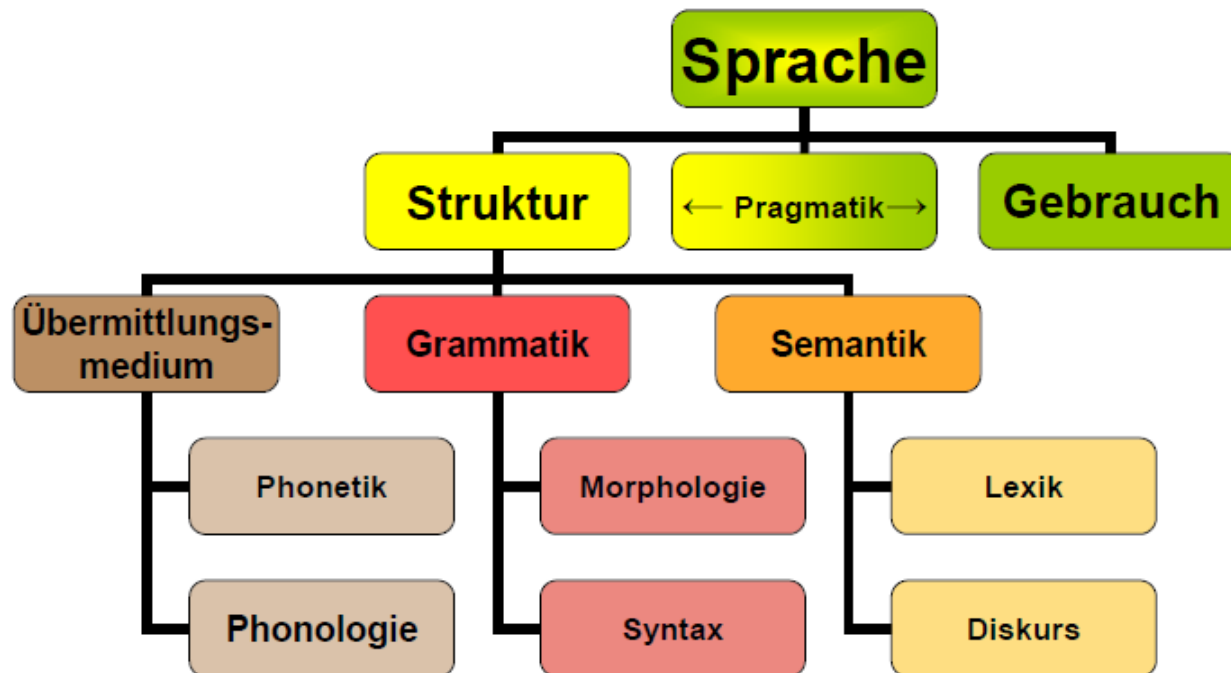
# Modelle für formale Sprachen

Da eine Maschine keine Sprachfähigkeit hat muss der Begriff Sprache von einem abstrakten Modell her entwickelt werden

Die Strukturbeschreibung formaler Sprachen wird wesentlich durch die Aufgabe „Kommunikation mit einer Maschine“ bestimmt

# Modell für formale Sprachen

Betrachtungen eine formalen Sprache  
(vgl. Modell natürliche Sprache



# Modell für formale Sprachen

Diese Konzepte werden

- unterschiedlich interpretiert
- haben bei den drei Begriffen Formale Sprachen – Programmiersprache – Computersprache unterschiedliches Gewicht

# Formale Sprache - Übermittlungsmedium

Das Übermittlungsmedium ist sehr vielfältig  
(Sprache, Zeichen, Bilder, Gesten,...)

## Multimedialität

Für die Kommunikation ist dies ein  
wesentlicher Aspekt für Computersprachen

- Ikonen, Graphische Eingabesprachen,  
Bildbeschreibungssprachen
- Human Computer Interfaces
- Spracheingabe

# Formale Sprache - Übermittlungsmedium

In der Kommunikation mit einer Maschine wird alles in eine Zeichenkette umgewandelt

Für die verschiedene Übermittlungsmedien gibt es eigene Programmiersprachen für die Umwandlung der Eingabe in eine formale Maschinensprache und der Rückübersetzung in das Ausgabemedium

# Formale Sprache - Übermittlungsmedium

Dabei wird eine Folge von Transformationen angewendet, die unterschiedliche Sprachen verwenden





# Formale Sprache - Übermittlungsmedium

Jedes Eingabemedium nutzt Wissen von unterschiedlichen Disziplinen

- Fragen der Phonetik und der Phonologie spielen bei der Spracheingabe eine Rolle
- Andere Disziplinen wie Graphik (statisch oder dynamisch), oder Haptik spielen eine zentrale Rolle beim Übermittlungsmedium

# Formale Sprache – Grammatik, Morphologie

## Struktur von Wörtern

Der Begriff „Wort“ wird in formalen Sprachen allgemeiner als eine Zeichenkette verstanden

Programmiersprachen haben üblicherweise eine Reihe von reservierten Wörtern

Es gibt auch Regeln zur Konstruktion von neuen Wörtern

# Formale Sprache – Grammatik, Morphologie

## Fragen:

- Beispiele von reservierten Wörtern in der Kommunikation mit dem Computer
- Beispiele von Regeln zur Konstruktion von neuen Wörtern in der Kommunikation mit einem Computer
- Beispiele von grammatischen Kategorien in Programmiersprachen

# Formale Sprache – Grammatik, Syntax

In formalen Sprachen spielt eine präskriptive Grammatik eine zentrale Rolle

- Regeln zur Bildung von korrekten Ausdrücken
- Regeln welche die Überprüfung der Korrektheit eines Ausdruckes im Sinne der Regeln erlauben

# Formale Sprache – Grammatik, Syntax

Einige Grammatische Kategorien formaler Sprachen:

- Zeichen (Alphabet, Symbole)
- Zeichenketten (Strings) „Wörter“
- Zahlen
  - Natürliche Zahlen,
  - Dezimalzahlen mit vorgegebenen Kommastellen

# Formale Sprache – Grammatik, Syntax

- Reservierte Worte für die Beschreibung von Abläufen
  - Bedingungen: **wenn** *bedingung* **dann** *aktivität*
  - Alternativen: **wenn** *bedingung* **dann** *aktivität1*  
**sonst** *aktivität 2*
  - Wiederholtes Ausführen:  
**wiederhole** *aktivität* **n-mal**
- Aus diesen Konstruktionen können durch Schachtelung komplexe Ausdrücke gebildet werden

# Formale Sprache – Grammatik, Syntax

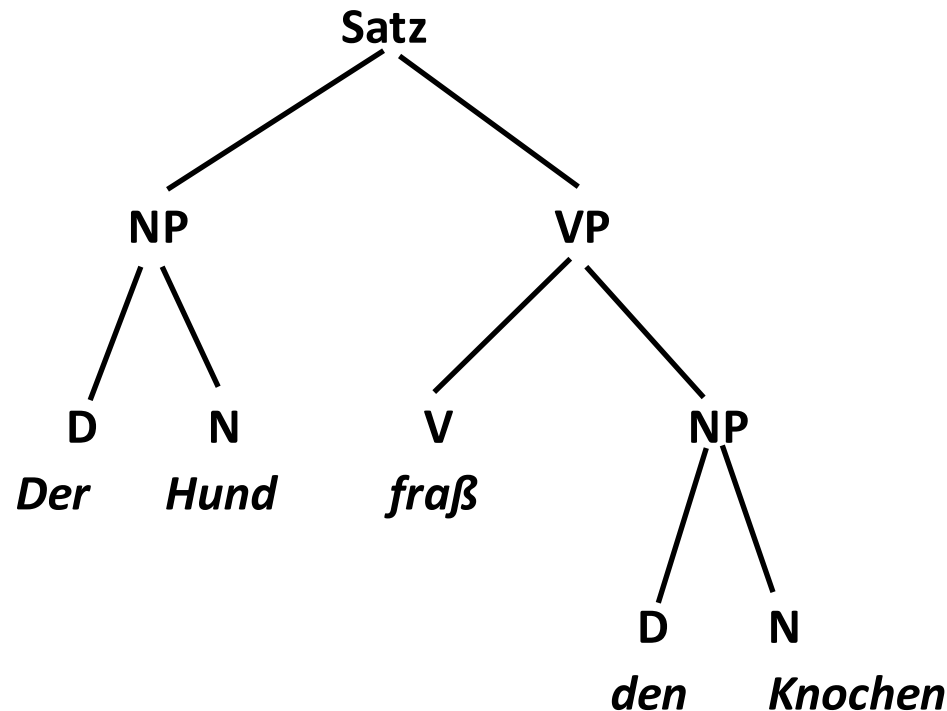
**Satzsyntax:** Wir konstruieren einen komplexen Ablauf oder eine Berechnung, ähnlich wie einen Satz in einer natürlichen Sprache in einer generativen Grammatik

Ein Satz wird aus einzelnen Phrasen aufgebaut

Dieser Aufbau kann durch einen Baum dargestellt werden (Ableitungsbaum)

# Formale Sprache – Grammatik, Syntax

Beispiel für einen Ableitungsbaum





# Formale Sprache – Grammatik, Syntax

Ähnlich definieren wir eine formale Sprache

- Eine endliche Menge von Symbolen, die man Alphabet bezeichnet. Sie bilden die Terminalen ***Symbole***
- Eine endliche Menge von Nicht-terminalen Symbolen, die man auch ***Variable*** nennt
- Ein ***Startsymbol***
- Produktionsregeln:

Ausdruck linke Seite  $\rightarrow$  Ausdruck rechte Seite

Ausdrücke werden als Terme bezeichnet

# Formale Sprache – Grammatik, Syntax

Im Prinzip gibt es nur drei Regeln um Terme zu bilden:

- Alle Elemente Alphabets sind Terme
- Verkettung von Termen:  $(x, y) \rightarrow xy$
- Alternative Auswahl von Termen:  $x \mid y$

Wesentlich für die Charakterisierung von formalen Sprachen sind Produktionsregeln  
Jede Produktion wird **Wort** genannt

# Formale Sprache – Grammatik, Syntax

Die Verwendung dieser Syntax ist zweifach

- Konstruktion von syntaktisch richtigen Sätzen durch Menschen oder eine Maschine, die ein ausführbares Programm definieren
- Analysieren von Sätzen, d.h. die Maschine kann nach diesen Regeln das eingegebene Programm „verstehen“ und die Handlungsanweisungen in eindeutiger Weise ausführen

# Formale Sprache – Grammatik, Syntax

Die Syntax von natürlichen Sprachen erlauben im allgemeinen nicht eine eindeutige Rekonstruktion eines Satzes

Beispiele:

*Ich sah den Mann mit dem Fernglas*

*Der Chef lachte über die Forderung der Angestellten nach einer Gehaltserhöhung im Mai*

**Aufgabe:**

Bilde mögliche Ableitungsbäume für diese Sätze

# Formale Sprachen – Grammatik, Synatx

Grammatiken werden meist durch eine der folgenden beiden Methoden beschrieben

- Angabe der Produktionsregeln(Backus – Naur Form)
- Angabe der Struktur regulärer Ausdrücke

# Formale Sprachen – Semantik

Semantik formaler Sprachen beschäftigt sich wie bei natürlichen Sprachen mit dem Aufbau und der Bedeutung der durch die Grammatik gebildeten Ausdrücke

Man unterscheidet statische und dynamische Semantik

Beschreibung der Semantik oft in natürlicher Sprache

# Formale Sprachen – Semantik

## Statische Semantik

Regeln, die in der Sprache erfüllt sein müssen und formal überprüft werden können

Regeln entsprechen unserer Vorstellung von „semantisch sinnvollen“ Ausdrücken

Beispiel: Addition nur für Zahlen sinnvoll

Mojca+Wilfried = ?

# Formale Sprachen – Semantik

Lösung dieses Problems durch  
Typisierung:

Addition von ganzen Zahlen ist anders  
zu verstehen als Addition von  
Dezimalzahlen (Rechengenauigkeit)



# Formale Sprachen – Semantik

## Dynamische Semantik

Semantik, die sich aus der Abarbeitung der Befehle ergibt

Meist in einem eingeschränkten Kontext behandelt: Ist Programm **korrekt**?

**Korrekt** bedeutet, dass das Programm seine Analyse in endlicher Zeit mit einer Antwort endet

# Formale Sprachen – Semantik

Beispiel für ein nicht korrektes Programm:

$x = 1$

**solange**  $x > 0$  **führe durch**

*addiere zu  $x$  den Wert 1*

Diese Anweisungen sind syntaktisch korrekt, aber im Sinne der dynamischen Semantik nicht korrekt

# Formale Sprachen – Semantik

Eine Präzisierung dieser Idee führt zur Frage der Berechenbarkeit und der Entscheidbarkeit

# Formale Sprachen – Semantik

Diese Semantik ist offensichtlich viel einfacher als die Semantik natürlicher Sprachen

Zur Erreichung einer komplexeren Semantik benötigen wir Programme, welche eine Problemstellung auf die Anwendung von Ersetzungsregeln für immer komplexer werdende Bausteinen zurückführen