

---

## 2.4 Winograds SHRDLU-System

Literatur:

Terry Winograd: Understanding natural Language. New York und London: Academic Press, 1972.

Winograd nennt sein System eine 'understanding machine'

**Was bedeutet SHRDLU?**

Die Ordnund der engl. Buchstaben nach Häufigkeiten ist

ETAOINSHRDLU...

---

### 2.4.1 Definition des Verstehens (Winograd 1972, 23):

*"We can describe the process of understanding language as a conversion from a string of sounds or letters to an internal representation of 'meaning'. In order to do this, a language-understanding system must have some formal way to express its knowledge to a subject, and must be able to represent the 'meaning' of a sentence in this formalism. The formalism must be structured so the system can use its knowledge in conjunction with a problem-solving system to make deductions, accept new information, answer questions, and interpret commands."*

---

## Verstehen im **operationalen** Sinn

- Erkennen des Inhalts von Fragen,
- Lösung von natürlich-sprachlich formulierten Problemen
- inhaltliche Erschließung von Texten

Ein Computer 'versteht', wenn er in der Lage ist, **Eingabesätze**, die aus einer Untermenge der Sätze einer natürlichen Sprache entnommen sind, zu akzeptieren, und wenn er auf natürlichsprachlich formulierte Fragen **in der erwarteten Weise antworten kann.**

---

## Sprachverstehen des Systems ist

1. ein Prozess der Umwandlung einer Kette von Lauten oder Buchstaben in eine *internal representation*.
2. *Beweis* der internen Repräsentation, d.h. Abbildung auf den Zustand der betrachteten Welt,

Beweis heißt:

Entweder wird ein Befehl unmittelbar ausgeführt, indem etwa ein Gegenstand auf einem Tisch an einen anderen Platz gesetzt wird.

Oder: Zunächst werden andere Operationen durchgeführt, die der Befehl nicht enthält, die jedoch aufgrund des Zustandes der Modellwelt notwendig sind, damit der Befehl ausgeführt werden kann.

Beispiel: Einen Platz leerräumen, damit auf ihm etwas abgestellt werden kann.

---

## 2.4.2 Leistungen des Systems

Im einzelnen muss das System in der Lage sein,

- (a) Deduktionen durchzuführen,
- (b) neue Informationen aufzunehmen,
- (c) Fragen zu beantworten und
- (d) Befehle zu interpretieren (Winograd 1972, S. 23).

---

Die konstruierte Modellwelt, die Winograds Programm kennt, besteht aus folgenden Hauptkomponenten:

- den Teilnehmern am Dialog: Roboter **SHRDLU**, Person **FRIEND**
- **Hand** des Roboters
- verschiedene geomtr. **Körper: Würfel, Pyramiden, Schachtel** mit Namen B1, B2, B3
- einem **Tisch**, auf dem sich die Körper und die Schachtel befinden (vgl. Abb.)

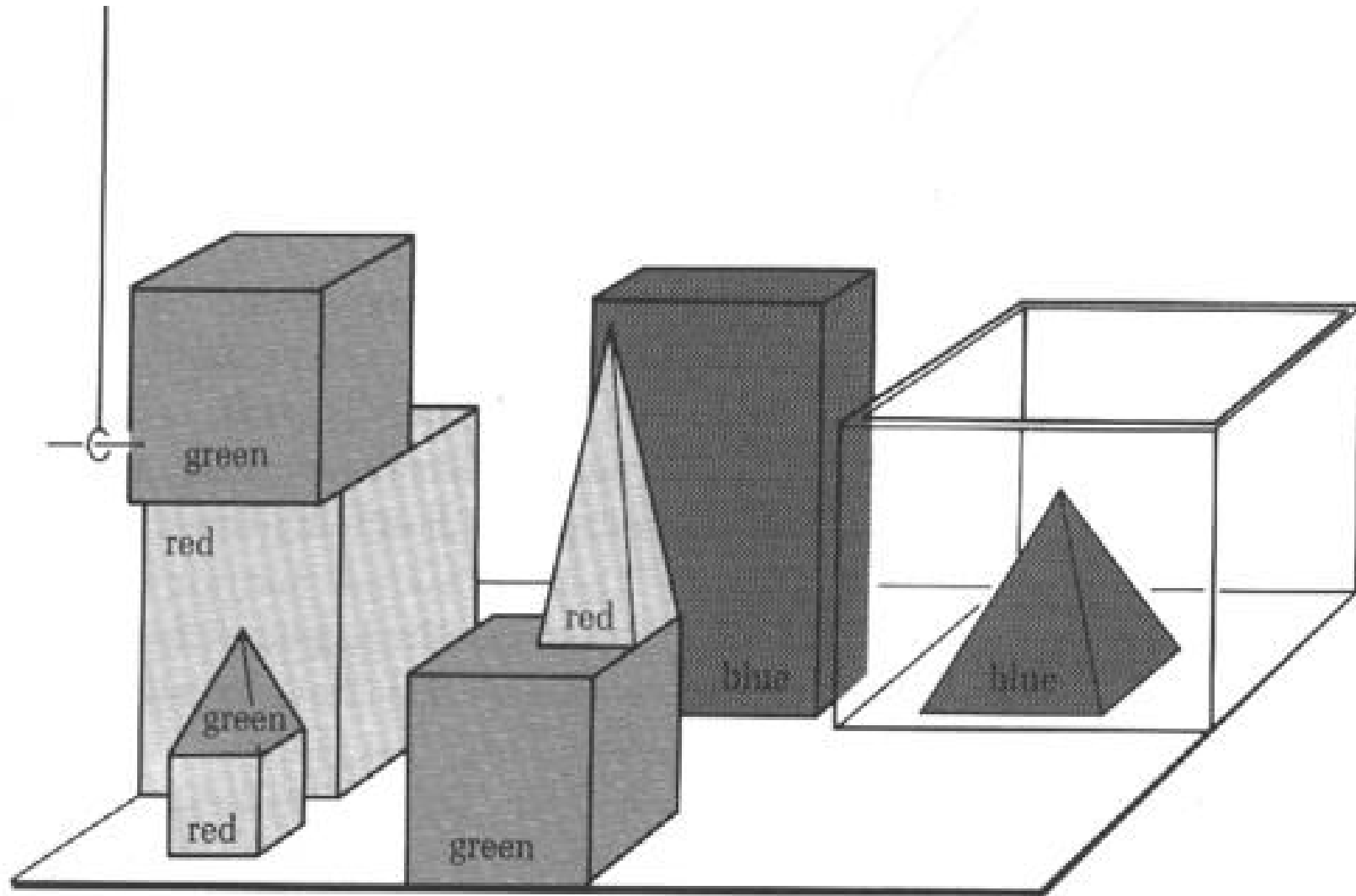


Figure 4.1 The Physical Environment of the Robot.

Winograd 1972: erster Versuch, alle Komponenten, die am Prozess des Sprachverstehens beteiligt sind, zu simulieren.

Erstmals klare Unterscheidung zwischen dem **sprachlichen** Wissens und dem **Weltwissen** eines Systems:

Das System muss in der Lage sein, jede Äußerung, die es empfängt, sprachlich zu analysieren und im Hinblick auf den gegenwärtigen Zustand seiner Welt zu interpretieren.

---

Wichtiger Unterschied in Winograds System gegenüber früheren:

Beschränkung auf eine Modellwelt:

Modellwelt: Begrenzung des Wirklichkeitsbereichs, auf den sich die Äußerungen beziehen: kleiner, genau abgegrenzten Wirklichkeitsbereich.

Dafür keine sprachliche Einschränkung, wie bei früheren Autoren.

---

Einschränkung auf Modellwelt ermöglicht es, alle Komponenten, die am Prozess des Sprachverstehens beteiligt sind, zu simulieren:

- Grammatik
- Semantik
- Inferenzkomponenten

Das System **beantwortet Fragen**, führt **Befehle** aus und empfängt **Informationen** in einem interaktiven englischen Dialog.

---

## Beispieldialog:

1. pick up a big red block.

OK

2. grasp the pyramid.

I DON'T UNDERSTAND WHICH PYRAMID YOU MEAN.

.

4. what does the box contain?

THE BLUE BYRAMID AND THE BLUE BLOCK.

5. what is the pyramid supported by?

THE BOX.

.

13. the blue pyramid is mine.

I UNDERSTAND.

.

15. do I own the box?

NO.

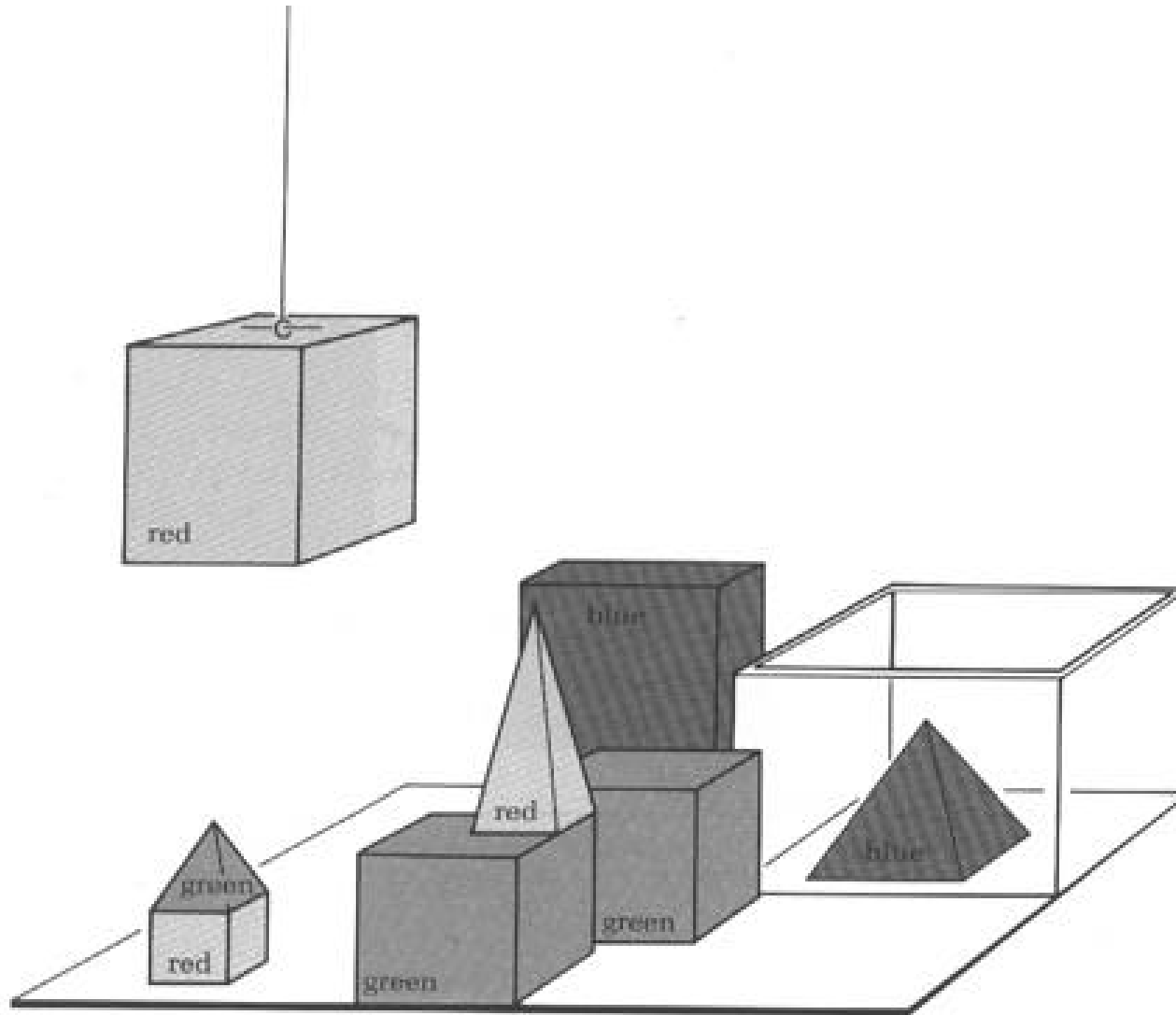


Figure 4.2 "Pick up a big red block."

---

## 2.4.3 Sprachverarbeitung in SHRDLU

### a) Grundprinzip:

Zeichenkette (z.B. Fragesatz

wird umgewandelt in

*internal representation of meaning*

---

**b) Darstellung von Objekten, Eigenschaften, Relationen (Repräsentationssprache)**

:HOUSEB74            (: ist Identifikator für Objekte)

(#CITY :BOISE)        # ist Identifikator für Eigenschaften und Relationen

(#FATHER-OF :NOAH :JAFETH)

Ein relationaler Ausdruck kann selbst wieder Relation sei:

(#FATHER\_OF :NOAH :JAFETH :REL76)

---

Damit sind Relationen der folgenden Art möglich:

(#KNOW :I :REL76)

Beispiel: "Harry slept on the porch after he gave Alice the jewels."

(#SLEEP :HARRY :REL1) (#LOCATION :REL1 :PORCH)

(#GIVE :HARRY :ALICE :JEWELS :REL2)

(#AFTER :REL1 :REL2)

---

## c) Darstellung des sprachlichen Wissens, des relationalen Wissens und des Weltwissens

- Sprachlichen Wissen

- lexikalisches Wissen
  - syntactic features
  - semantic definitions
- grammatisches Wissen
  - z.B. Wissen über die mögliche Struktur von Nominal- und Verbalgruppen

- **Relationales Wissen**

(Relationen, die das System kennt)

#IS	schreibt Objekten und Eigenschaften einen Namen zu
#SUPPORT (A,B)	A stützt B, B trägt A
#CLEARTOP (A)	A befindet sich oben, auf der Spitze
#CONTAIN (A,B)	A enthält B
#PART (A,B)	A ist Teil von B
#RIGHT (A,B)	A ist rechts von B
#BEHIND (A,B)	A ist hinter B
#ABOVE (A,B)	A ist oberhalb von B

- **Weltwissen**

über den Zustand der Modellwelt (sog. 'Assertions')

Der jeweilige Zustand der Modellwelt wird durch eine Liste sog. Assertions (Behauptungen) ausgedrückt. Jede Assertions definiert ein Objekt, dessen Eigenschaften und seine räumlichen Relationen, und zwar mithilfe der o.g. repräsentationssprachlichen Mittel:

#IS ordnet Objekten und Eigenschaften Namen zu,  
#AT ordnet einen Ort in der Modellwelt zu,  
#SUPPORT, #CLEARTOP, #CONTAIN und #PART ordnen räumliche Relationen zu.

Darstellung eines Weltzustandes nach Winograd

---

```
(#IS :B1 #BLOCK)
(#IS :B2 #Pyramid)
(#IS :B3 #BLOCK)
(#IS :B4 #Pyramid)
(#IS :B5 #PYRAMID)
(#IS :B6 #BLOCK)
(#IS :B6 #BLOCK)
(#IS :B10 #BLOCK)
(#IS #RED #COLOR)
(#IS #BLUE #COLOR)
(#IS #GREEN #COLOR)
(#IS #WHITE #COLOR)
(#IS #BLACK #COLOR)
(#IS #RECTANGULAR #SHAPE)
(#IS #SHRDLU #ROBOT)
(#IS FRIEND #PERSON)
(#AT :B1 (100 100 0))
(#AT :B2 (100 100 100))
(#SUPPORT :TABLE : B1)
(#SUPPORT :TABLE :BOX)

(#AT :BOX (600 600 0))
(IS :BOX #BOX)
(IS :TABLE #TABLE)
(#CONTAIN :BOX :B4)
(#SHAPE B1 #RECTANGULAR)
(#COLOR :B1 #RED)
#COLOR :B2 #GREEN)
#COLOR :B3 #GREEN)
(#SUPPORT :B1 ;B2)
(#CLEAR TOP :B2)
(#MANIP :B1)
(#MANIP :B2)
(#CALL :SHRDLU SHRDLU)
(#CALL :FRIEND YOU)
```

- Wissen über Prozeduren zum Problemlösen (Theoreme)

Theoreme stellen die möglichen Veränderungen in der Modellwelt dar.

Beispiel: Das Theorem **Evaluate** - in der speziellen Wissens-Repräsentationssprache formuliert.

Aufgabe des Theorems: Bewertung einer Examensarbeit ('thesis')

(DEFTHEOREM EVALUATE	EVALUATE is the name we are giving to the theorem.
THCONSE(X Y) (#ACCEPTABLE \$?X)	This indicates the type of theorem, names its variables and states that we are trying to prove something acceptable.
(THGOAL(#THESIS \$?X))	Show that X is a thesis. The "\$?" indicates a variable.
(THOR	THOR is like "or", trying things in the order given until one works.
(THGOAL(#LONG \$?X) (THUSE CONTENTS-CHECK COUNTPAGES))	THUSE says to try the theorem named COUNTPAGES.
(THAND	THAND is like "and".
(THGOAL(#CONTAINS \$?X \$?Y))	Find something Y which is contained in X.
(THGOAL(#ARGUMENT \$?Y))	Show that it is an argument.
(THGOAL(#PERSUASIVE \$?Y) (THTBF THTRUE))))))	Prove that it is persuasive, using any theorem which are applicable.

Das Theorem EVALUATE ist ein Computerprogramm.

Es wird aufgerufen, wenn in einer Äußerung verlangt wird, eine bestimmte Examensarbeit zu bewerten.

z.B. "Is Sam's thesis acceptable?".

PLANNER-Formulierung dieser Frage:

(THGOAL (#ACCEPTALBE :SAM-THESIS) (THUSE EVALUA-  
TE)

---

Das Theorem prüft,

- ob die 'thesis' einen Inhalt hat,
- wenn nicht, wieviele Seiten sie hat,
- ob sie Argumente enthält und
- ob die Argumente überzeugend sind.

Bei Ausführung des Theorems werden die Bedingungen, unter denen die Bewertung steht, geprüft. Diese Bedingungen müssen im Wissen gespeichert sein.

Nach Ausführung des Programms wird dem PLANNER-Ausdruck ein logischer Wert ('wahr' oder 'falsch') zugeordnet.

Ähnlich werden alle Befehle, Fragen und Aussagen bearbeitet.

Beispiel:

"Pick up a big red block and put it in the box"

zugehöriger PLANNAR-Ausdruck:

```
(THAND (THGOAL(#PICKUP :BLOCK23)) (THGOAL(#PUTIN  
:BLOCK23 :BOX7)))
```

PICKUP und PUTIN sind Namen für Theoreme.

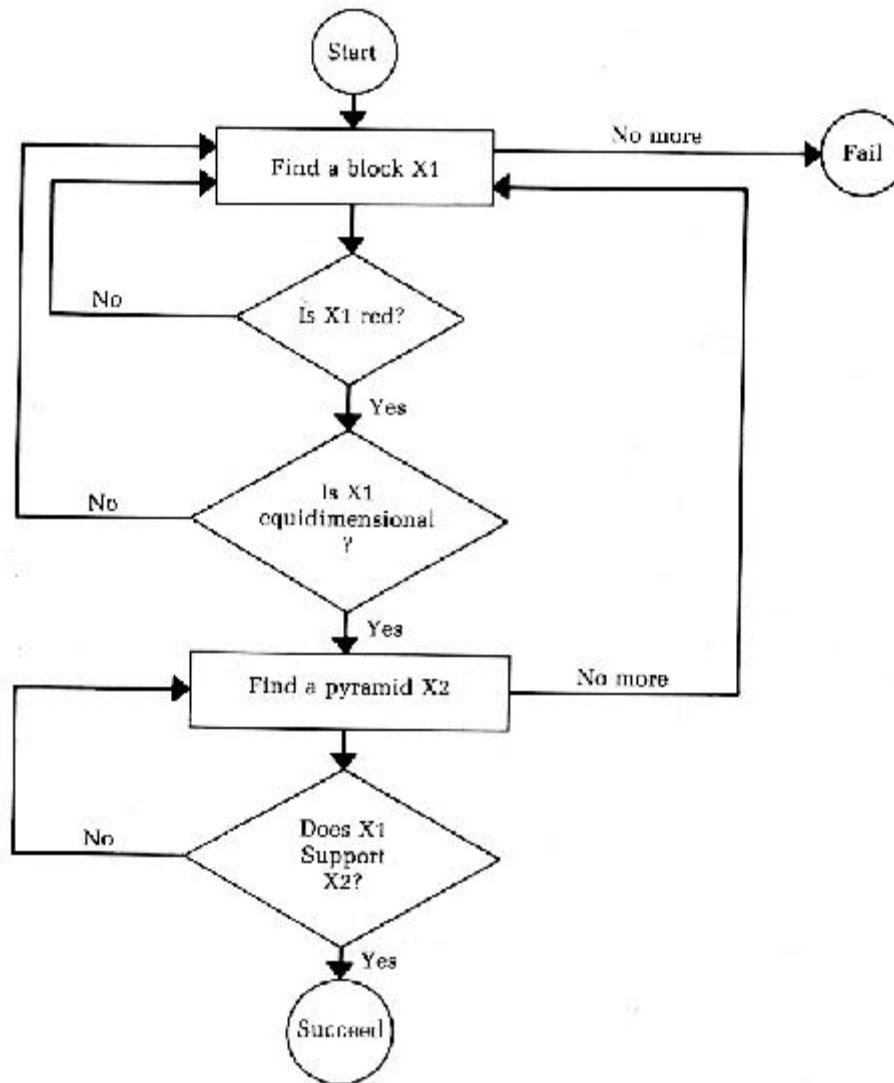


Figure 4.7 Procedural Representation of "a red cube which supports a pyramid."

Darstellung  
der prozeduralen Arbeitsweise  
durch ein Flussdiagramm

## d) Beispiel für die Abarbeitung eines Satzes: Pick up a big red block

Das System nimmt eine Transformation vor:

natürlich-sprachlicher Ausdruck ---> PLANNER-Ausdruck (=Ausdruck in einer Repräsentations-sprache)

Arbeitsschritte:

- 1) 'parsing', d.i. Erkennen der syntaktischen Struktur,
2. **semantische Interpretation**, d.i. Darstellung in PLANNER-Ausdrücken,
- 3) **Deduktion**, d.i. Ableiten von Antworten aus der Modellwelt, die zur Beantwortung der Fragen erforderlich sind.

**parsing** "pick up a big red block"

---

## Ergebnisstruktur:

(**CLAUSE** MAJOR IMPER PRT TRANS) pick up a big red block)  
(**VG** IMPER) (pick) (**VB** MVB INF TRANS VPRT) pick (**PRT**) up  
(**NG** OBJ OBJ1 DET INDEF NS) (a big red block)  
(**DET** INDEF NS) a  
(**ADJ**) big  
(**ADJ**) red  
(**NOUN NS**) block

---

Beim Parsing benutzt das System auf allen Ebenen detaillierte Beschreibungen der Konstituenten, die in Form von Netzwerken (*choice-systems*) vorliegen:

Konstituenten:

Clause (Ebene der Satztypen: Hauptsatztypen, Nebensatztypen)  
(Ebene der möglichen Ergänzungen: Transitivität etc.)

Noun Group (Nominalgruppen)

Preposition Groups (Präpositionalgruppen)

Adjektive Groups (Adjektivgruppen)

Verb Groups

## Wörter, Wortklassen

Typen von Determinern

Typen von Verben

Typen von Konjunktionen

Die Erkennung der Wortformen wird über eine Analyse der Wortendungen durchgeführt.

---

## Semantische Interpretation

Umwandlung der Einheiten, die im Verlaufe der syntaktischen Analyse ermittelt wurden, in PLANNER-Ausdrücke.

Für jede ermittelte syntaktische Einheit wird ein **spezielles semantisches Interpretationsprogramm** (*semantic specialist*) aufgerufen.

Durch die semantische Interpretation mittels der *semantic specialists* werden aus

Nominalgruppen	÷	<i>Object Semantic Structures: OSS</i>
Verbalgruppen	÷	<i>Relation Semantic Structure: RSS</i>

---

Die *semantic specialists* bedienen sich bei der Überführung in PLANNER-Ausdrücke der semantischen Informationen des Wörterbuchs:

Diese besteht aus semantischen Markern:

( Cube: #BLOCK, #MANIP; red: #PHYSOB)

und semantischen Zuordnungen:

(cube: ((#IS \*\*\* #BLOCK)(#EQIDIM \*\*\*)); red: (#COLOR \*\*\* #RED)).

---

Für die Nominalphase *a red cube* ermittelt das System folgende **semantische Interpretation** (*Object Semantic Structure*):

```
(( ((X1) 200      (THGOAL(#IS$?X1 #BLOCK))
      (THGOAL(#COLOR $?X1 #RED))
      (#EQDIM ?X1))
  (0 #BLOCK #MANIP #PHYSOB #THING)
  (#MANIP #PHYSOB #THING))
X1
(NS INDEF NIL)
NIL)
```

Wie ist dieser Ausdruck zu lesen?

---

Dieser Ausdruck stellt ein Theorem mit dem Namen THGOAL dar, das zu beweisen ist, d.h. eine Anweisung an das System, festzustellen, ob es in der Datenbasis ein Objekt X1 gibt, das den Namen BLOCK trägt, von roter Farbe und von gleicher Kantenlänge ist. Dem Theorem wird ein Wahrheitswert ("wahr" oder "falsch") zugeordnet.

"Object Semantic Structures" (OSS) =  
PLANNAR-Ausdrücke, die er den Nominalgruppen zugeordnet sind.

"Relation Semantic Structures" (RSS) =  
Plannar-Ausdrücke, die den übrigen Konstituentengruppen zugeordnet werden.

Beispiel einer *Relation Semantic structure*:

(NG1 (#SUPPORT NG1 NG2) NIL) (0)

Ganze Sätze sind RSS, weil in ihnen Verben die Beziehungen zwischen den OSS herstellen.

---

W. unterscheidet drei Satztypen (Typen von Sprechakten):

Fragesatz, Deklarativsatz, Imperativsatz.

Jeder Typ veranlasst bestimmte Aktionen.

Der Fragesatz wird an einem Fragewort identifiziert. Die Antwort wird auf folgende Weise erzeugt:

Abarbeitung einer Frage:

Der Fragesatz wird in eine RSS überführt. Der entstandene PLANNER-Ausdruck tritt dann als Argument in ein spezielles Theorem ein, das eine entsprechende RSS in der Datenbasis sucht. Wird eine solche nicht unmittelbar gefunden, so kann sie doch implizit enthalten sein. Sie muss durch Deduktion abgeleitet werden.

## Deduktion

---

Die Datenbasis enthalte die Aussage

(THASSERT (#HUMAN :TURING)) (=Turing ist ein Mensch)

und das Theorem

(DEFTHEOREM THEOREM1  
(THCONSE (X) (#FALLIBLE \$?X)  
(THGOAL (#HUMAN \$?X)))) (=alle Menschen sind fehlbar)

---

Wird die Frage gestellt:

(THGOAL (#FALLIBLE :TURING) (=ist Turing fehlbar?))

so ist eine RSS, die dieser Frage entspricht, nicht vorhanden, d.h. die Antwort muss abgeleitet werden.

Dazu werden alle vorhandenen Strukturen der Form

(\$?Z :TURING) (=ist Turing Z?)

überprüft. Gefunden wird

(#HUMAN :TURING) (=TURING ist ein Mensch)

Dieser Ausdruck fungiert als neues Ziel des Suchprozesses; er wird im Theorem1 gefunden, welches die Implikation "alle Menschen sind fehlbar" repräsentiert. Durch dieses Theorem kann dem Ausdruck ein Wahrheitswert zugeordnet werden.

Auf ähnliche Weise werden Imperativsätze und Deklarativsätze interpretiert und entsprechende Aktionen als Änderungen in der Datenbasis ausgeführt.