

Soar

kognitive Architektur

Peter Paul Kaczmarczyk

01. Februar 2007

Themen

- Produktionsregeln
- States
- Operatoren
- Wissen
- Lernmechanismen
- Soar Architektur
- Unified Theory of Cognition

Was ist Soar

- Soar steht für
 - State, Operator And Result
 - Problemlösung in Soar wird als Suche in einem Suchraum angesehen, in dem man Operatoren an Zustände anwendet um Ergebnisse zu erzielen

Was ist Soar

- Soar ist eine
 - allgemeine kognitive Architektur
 - für Entwicklungsumgebungen,
 - die intelligentes Verhalten erzeugen

Wer benutzt Soar

- Kognitionswissenschaft:
 - Kognition ist ein Informationsverarbeitungsprozess, indem Neues gelernt und Wissen verarbeitet wird
 - Kognitions-Psychologie
 - Neurowissenschaft
 - Linguistik
 - Philosophie
 - Informatik (speziell: KI Entwickler)

Die Soar Architektur

- Production Rules
- States
- Operators
- Memory
- Learning - mechanism
- I/O-Interface
- Entscheidungszyklus

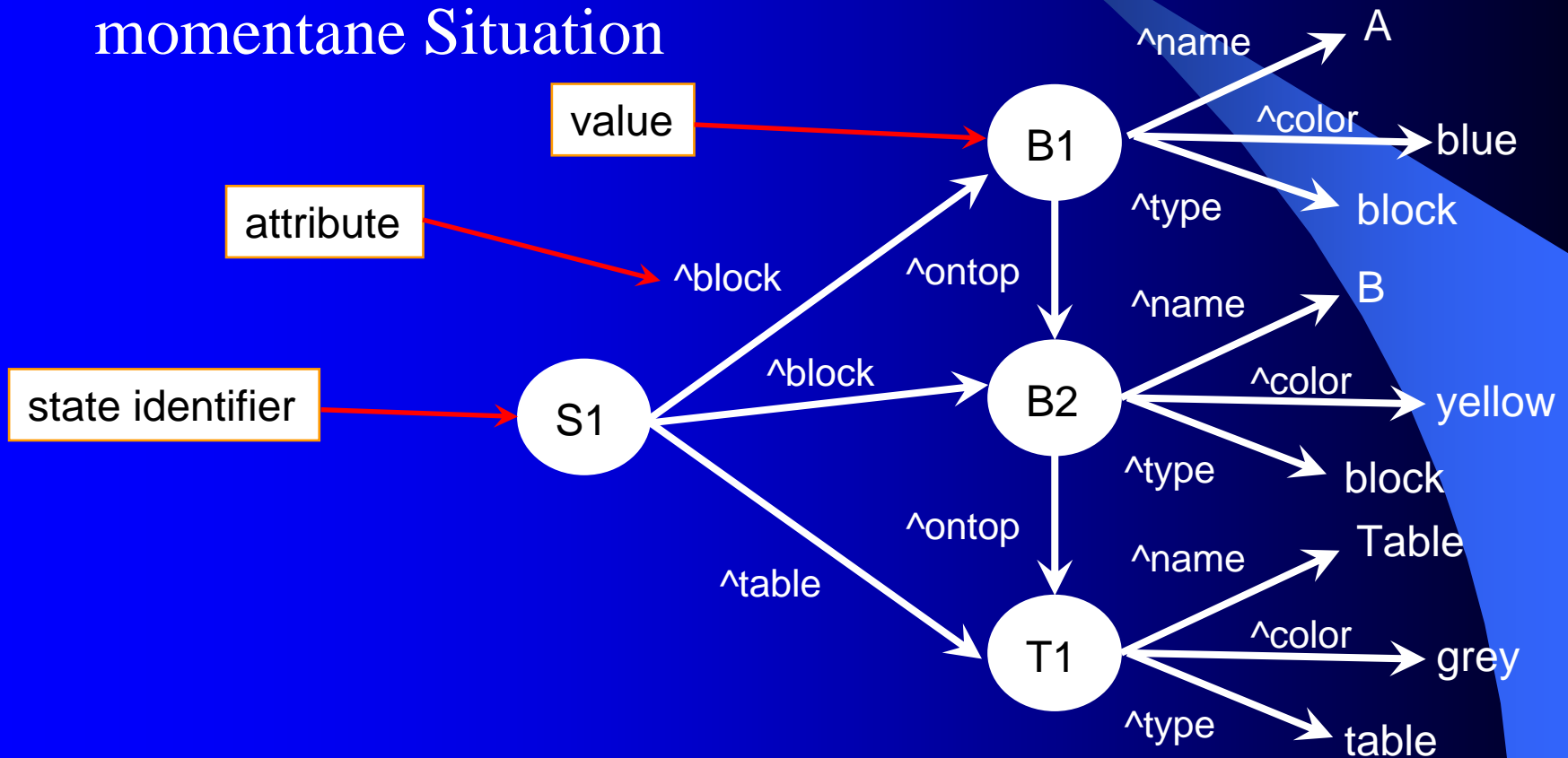
Produktionsregeln

```
sp {rule*name  
  (condition)  
  (condition)  
  ...  
-->  
  (action)  
  (action)  
  ...}
```

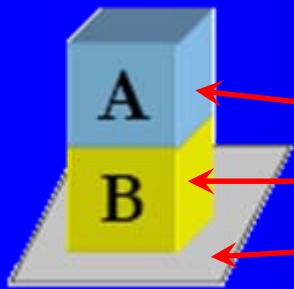
- Regel beginnt mit “sp”: “Soar production”
- Körper der Regel ist umschlossen von “{“ ... “}”
- Körper
 - Regelnamen
 - mind. eine Bedingung
 - dem Pfeilsymbol “-->“
 - mind. eine Aktion

States

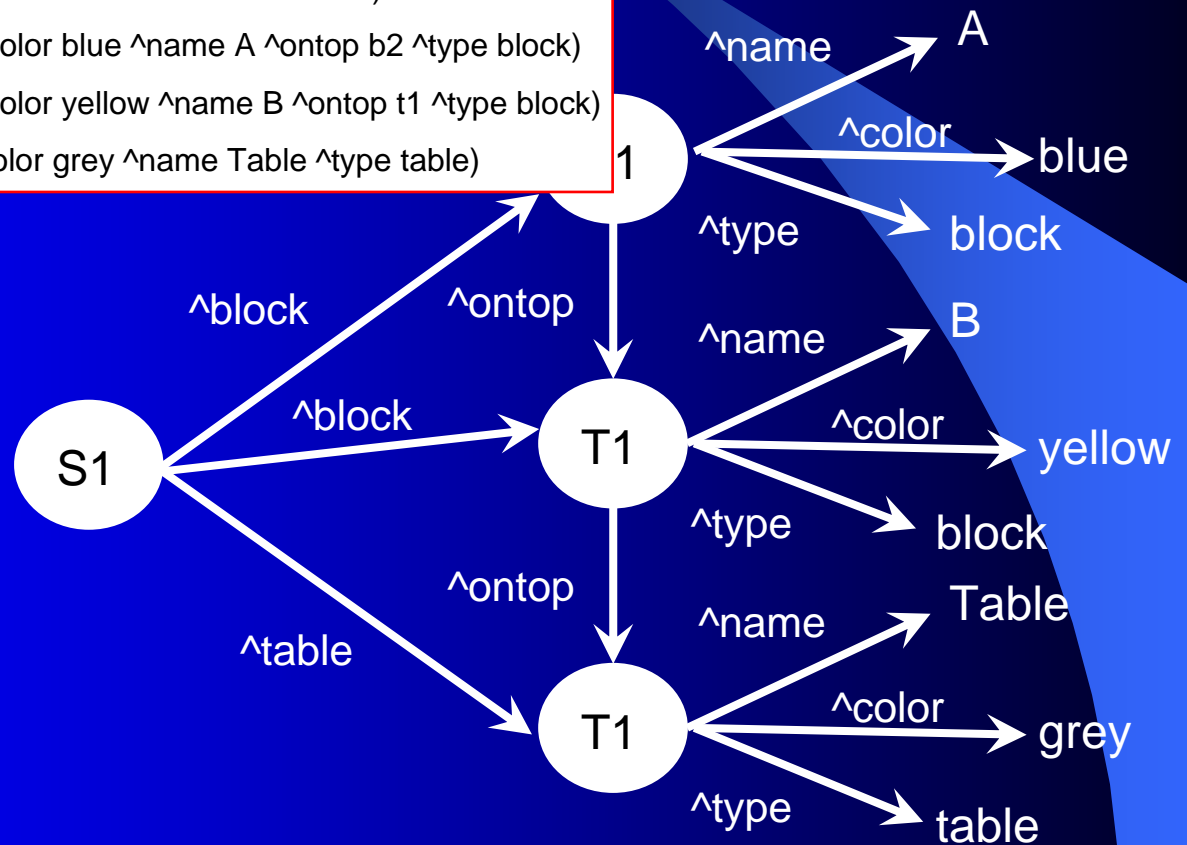
- Die States enthalten alle Informationen über die momentane Situation



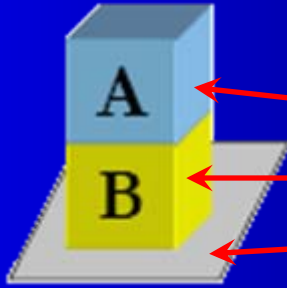
States



(s1 ^block b1 ^block b2 ^table t1)
(b1 ^color blue ^name A ^ontop b2 ^type block)
(b2 ^color yellow ^name B ^ontop t1 ^type block)
(t1 ^color grey ^name Table ^type table)



Operators



```
(s1 ^block b1 ^block b2 ^table t1)
(b1 ^color blue ^name A ^ontop b2 ^type block)
(b2 ^color yellow ^name B ^ontop t1 ^type block)
(t1 ^color grey ^name Table ^type table)
```

```
sp {propose*pick
(state <s> ^type state
- ^holding
^block <b>)
-->
(<s> ^operator <o> +)
(<o> ^name pick
^obj <b>)}
```

acceptable preference

```
sp {apply*pick
(state <s> ^operator <o>)
(<o> ^name pick
^obj <b>)
-->
(<b> ^ontop <x> -)
(<s> ^holding <b>)}
```

remove WM-element

Operators

- „operator application rules“
 - Bedingung enthält einen Operator
 - operator-support
 - „non-operator application rules“
 - instantiation-support
- SOAR Truth-Maintenance System (TMS)

Operators

- „non-operator application rules“

```
sp {elaborate*free
  (state <s> ^block <a>
    ^block <b>)
  (<a> ^ontop <b>)
  -->
  (<b> ^occupied)
  (<b> ^free -)
}
```

Operators

- „non-operator application rules“

```
sp {select*operator*pick
(state <s> ^operator <o> +)
(<o> ^obj <a>)
(<a> ^free)
-->
(<s> ^operator <o> > )
}
```

Working Memory (WM)

Es gibt also 2 Sorten von Speicher

- **o-supported WM**

- persistent
- erzeugt durch operator application rules

- **i-supported WM**

- abhängig von anderen Elementen im WM
- nach jeder Veränderung neu berechnet

Long-term Memory (LTM)

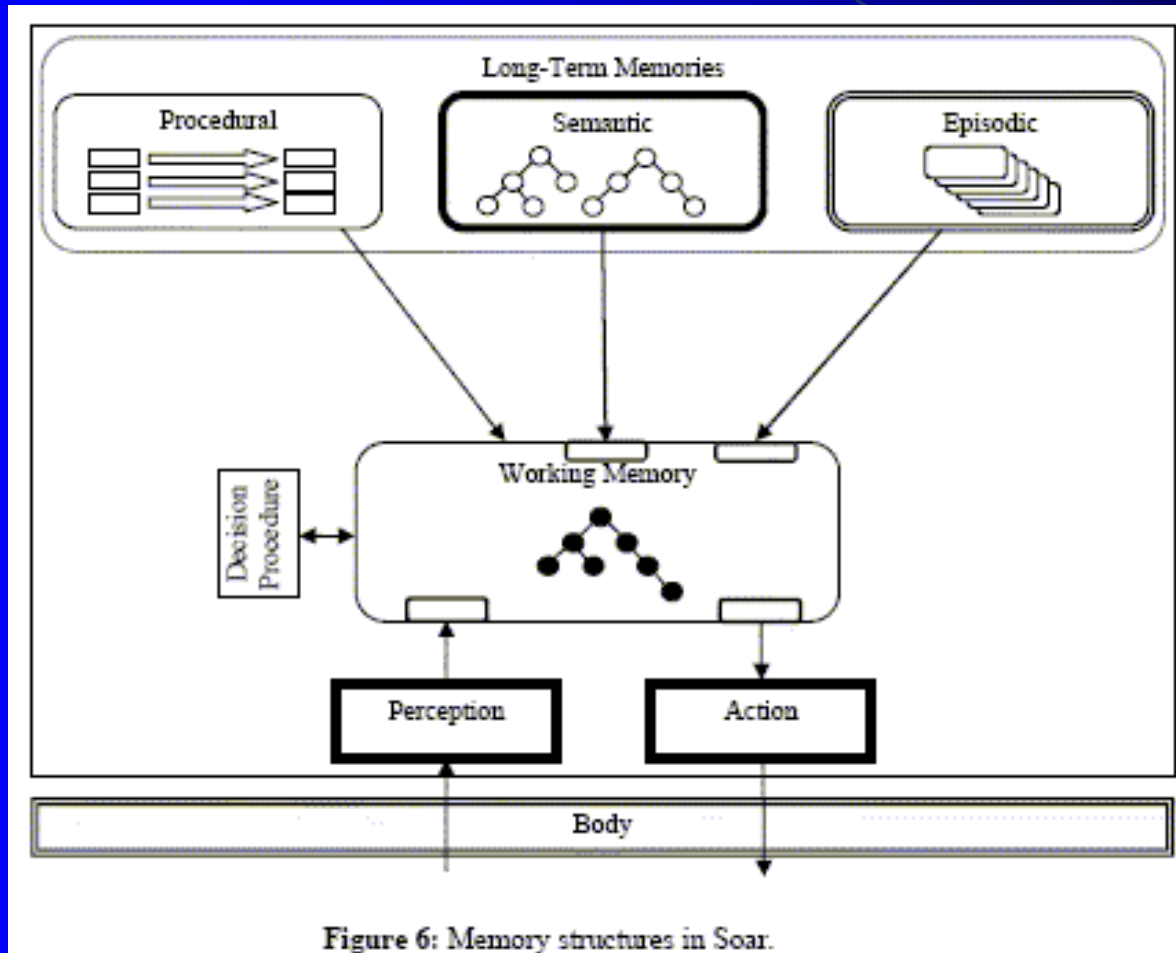


Figure 6: Memory structures in Soar.

Long-term Memory (LTM)

- Prozedurales Wissen
 - Regeln
- Semantisches Wissen
 - Deklarative Strukturen
- Episodisches Wissen
 - „Episodes“

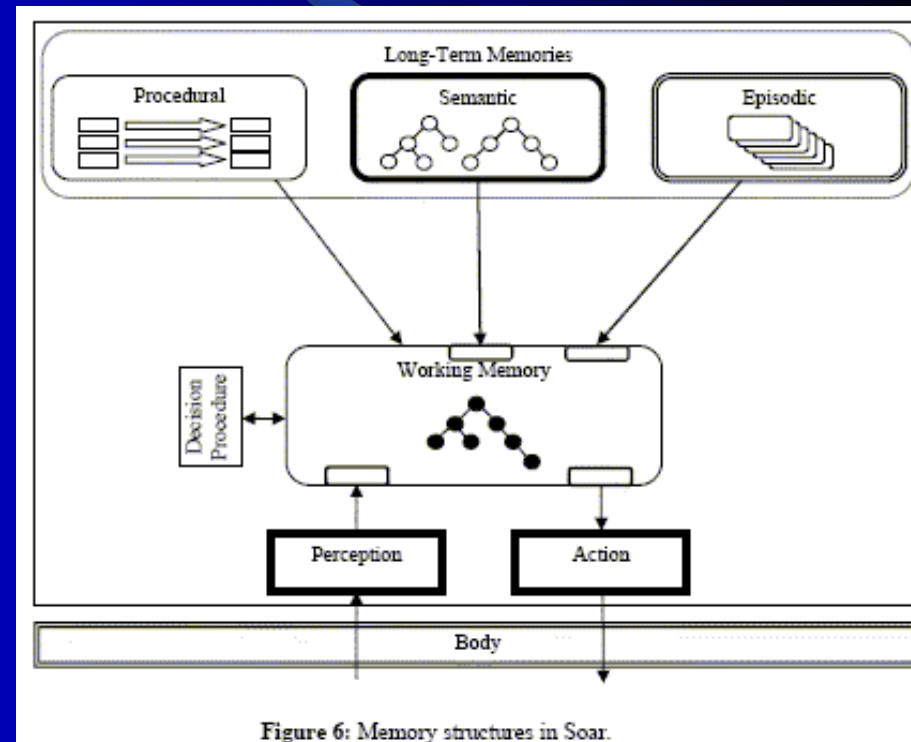
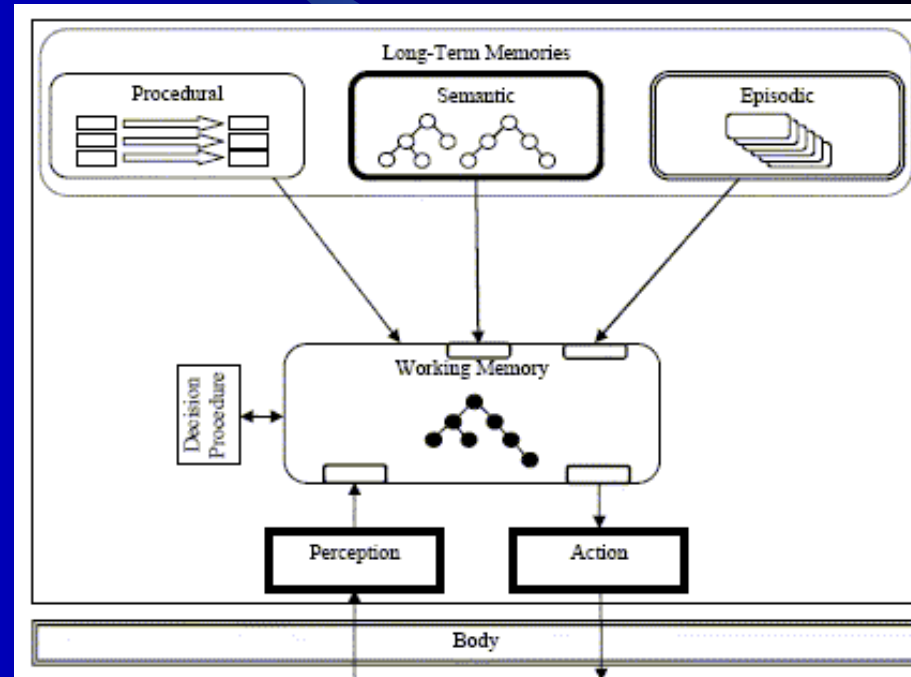


Figure 6: Memory structures in Soar.

Long-term Memory (LTM)

- **Prozedurales Wissen**
 - Regeln
- **Semantisches Wissen**
 - Deklarative Strukturen
- **Episodisches Wissen**
 - „Episodes“



SOAR kann sein LTM nicht direkt untersuchen
„memory retrievels“ verändern das WM

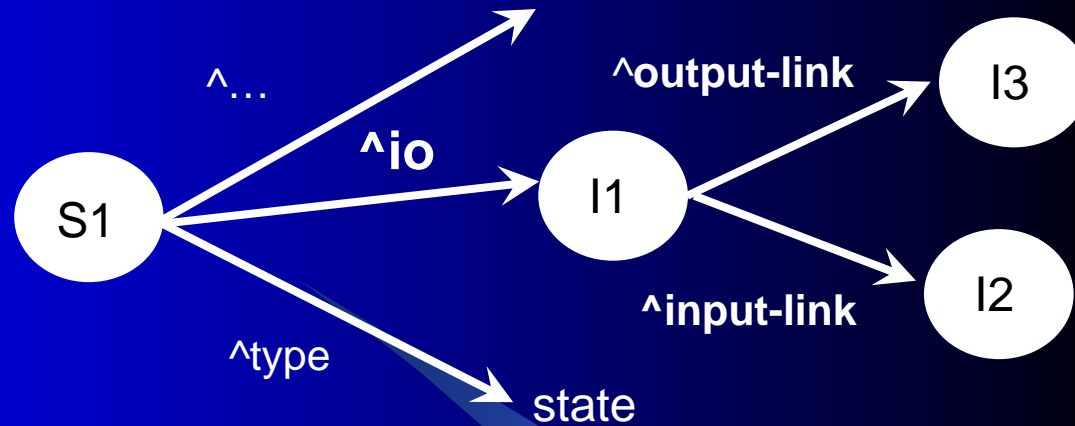
Sackgassen (impasses)

- Anzeichen einer Wissenslücke
 - Möglichkeit fürs Lernen
 - automatisch generiert durch Soar
- SOAR generiert automatisch einen neuen State, dieser hat das Ziel die Sackgasse zu lösen

Lernmechanismen

- **Chunking**
 - Erzeugt automatisch neue Regeln im LTM, wenn Ergebnisse durch einen impasse erzeugt wurden
- **Reinforcement learning**
 - Passt die Präferenzen zwischen Operatoren an
- **Episodic learning**
 - Speichert eine Historie von Erfahrungen
- **Semantic learning**
 - *Behandelt abstraktere deklarative Statements*

I/O-Interface

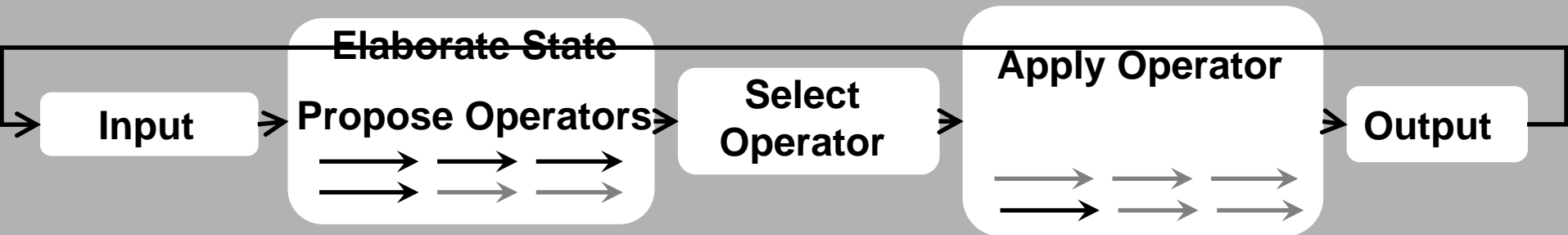


- ^io-Attribut an jedem State
- Mechanismus um ein Mapping zwischen Außenwelt und internen Repräsentation zu definieren
- Realisierung von
 - Wahrnehmungen
 - Aktionen

Problemstellung in SOAR

- **state representation**
- **initial state**
- **rules**
 - **proposal rules**
 - **application rules**
 - **elaboration rules**

Decision Cycle



- paralleles feuern von Regeln
- Auswahl eines Operator pro Zyklus
- Anwendungsregeln abhängig vom ausgewählten Operator

Demo

Unified Theories of Cognition

Menschen verfügen über zahlreiche kognitive Fähigkeiten:

- Gedächtnis
- Sprache
- Wahrnehmung
- Problemlösen
- geistiger Wille
- Aufmerksamkeit, ...

Unified Theories of Cognition

SOAR steht noch nicht als Architektur um alle kognitiven Prozesse zu simulieren.

Es wurden jedoch Projekte realisiert um Systeme miteinander zu vernetzen und auszubauen.

Wahrnehmung Lernen Problemlösen Sprache geistiger Wille

NTD-SOAR			X	X	
Instructo-SOAR		X	X	X	
IMPROV	X	X	X		
TacAir-SOAR, RWA-SOAR MOUTBOT	X		X	X	

NTD-SOAR

- Eine Berechnungstheorie des NASA Test Directors
- Er verwendete Materialien in seiner Umgebung und Kommunizierte mit anderen im Space Shuttle Start Team
- Kombiniert
 - NL-SOAR
 - NOVA (Model für Bildverarbeitung)
 - Wissen zur Entscheidungsfindung und Problemlösung, dass notwendig ist um ein Space Shuttle zu testen und auf den Start vorzubereiten

Instrukto-SOAR

- Berechnungstheorie wie Menschen durch interaktive Instruktionen lernen
 - Kombiniert
 - NL-SOAR
 - Wissen wie man aus Instruktionen während eines Problemlösungsprozesses lernen kann
- Instrukto-SOAR lernt neue Verfahren durch natürliche Sprache

IMPROV

- Berechnungstheorie um Wissen über die Konsequenzen von Aktionen zu korrigieren
- Kombiniert
 - SCA (Model für kategorisches Lernen)
 - Wissen wie man Konflikte zwischen dem internem WM und der Observation der Welt entdeckt

→ IMPROV kann automatisch sein Wissen über Interaktionen mit der Umgebung verbessern

Bsp.: <http://www.sunnyhome.org/pubs/mlw95.pdf>

TacAir-Soar und RWA-Soar

- Modelle von menschlichen Piloten
- Mit >8000 Regeln bei TacAir die größten SOAR Modelle
- Fähigkeiten:
 - Autonome Flugmissionen
 - Dynamische Missionswechsel bearbeiten
 - Einhaltung der Kommandostruktur riesigen Flugmanövern
 - Kommunikationsverbindung in „militär Englisch“

SOAR MOUTBOT

- Militär Operationen im urbanen Terrain
- SOAR kontrolliert mehrere individuelle Gegner, die Häuser verteidigen, gegen die das Militär stürmt
- Kombiniert
 - Integration von reaktiven Aktionen
 - Planung
 - Kommunikation mit anderen Individuen
 - Koordination
 - Räumliches Schließen

Zusammenfassung

- Produktionsregeln
- States
- Operatoren
- Wissen
 - o-supported
 - i-supported
- Lernmechanismen
- Unified Theory of Cognition

A surreal landscape with floating islands, a glowing blue figure, and musical notes. The scene is set against a blue sky with white clouds. In the foreground, a man lies on a grassy hill, listening to music from a boombox. A glowing blue figure, resembling a human form with a transparent body, stands on a floating island. A musical staff with notes is visible, and a large tree stands on the right. The text 'S.O.A.R.' is written in large, white, 3D letters at the bottom left.

Fragen ?

Ich bedanke mich für Ihr Interesse

S.O.A.R.