

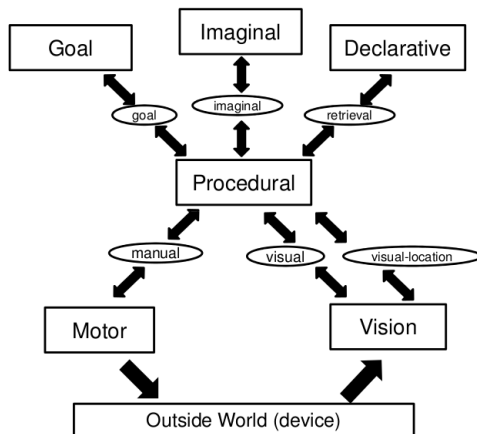
Cognitive Modeling

ACT-R

Michael Siebers

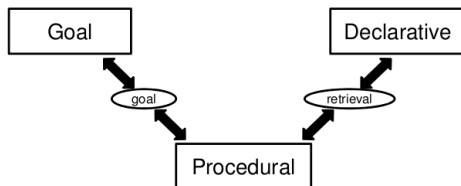
Kognitive Systeme, Angewandte Informatik, Universität Bamberg

25. November 2010



- ACT-R ist eine kognitive Architektur
- realisiert als Produktionssystem
- implementiert in Lisp

Genereller Aufbau



- Gliederung in einzelne *Module*
- Goal, Declarative, Procedural
- *buffer* als Bindeglieder zwischen Modulen
- *buffer* haben höchstens ein Element

Wissen in ACT-R

Prozedurales Wissen

- im Modul *Procedural*
- repräsentiert als Wenn-Dann-Regeln
- Bedingungen sind auf Buffern definiert
- Aktionen auch

Deklaratives Wissen

- organisiert als Chunks
- Chunks haben bestimmte Chunk-typen

Chunks

Chunk-typen erzeugen

- (**chunk-type** name slot-name-1 slot-name-2 ... slot-name-n)
- (chunk-type bird species color size)
- Name ist *kein* Teil des Chunks

Chunks definieren

- (name ISA chunk-type [slot-name slot-value]*)
- (tweety ISA bird species canary color yellow size small)
- (hedwig ISA bird species owl)

Produktions-Regeln definieren

```
(p Name \optional documentation string"  
  buffer tests  
==>  
  buffer changes and requests  
)
```

- Buffer-Test: =buffer-name> Bedingungen
- Buffer-Changes: =buffer-name> Änderungen

Beispiel Produktionsregel

(P counting-example

=goal>

isa count

state incrementing

number =num1

=retrieval>

isa count-order

first =num1

second =num2

==>

=goal>

number =num2

+retrieval>

isa count-order

first =num2

)

Produktionsbedingungen

Typanfragen

ISA	count
-----	-------

Bedingungen

state	incrementing
state	nil

Variablenzuweisung

number	=num1
--------	-------

Achtung: Matcht nicht auf nil

Aktionen auf Buffern

Modifikationen

=goal>

number =num2

Anfragen

+retrieval>

ISA count-order

first =num2

Löschen

-retrieval>

ACT-R starten

Download

<http://act-r.psy.cmu.edu/actr6/>

Windows Standalone version

MAC Standalone version

Linux Etwas komplizierter, siehe QuickStart.txt