



Technische Universität München

Fakultät für Informatik

Forschungs- und Lehrereinheit Informatik VI

## **Überblick kognitiver Architekturen**

Seminar Kognitive Robotik (SS12)

**Sebastian Hofstetter**

**Iris Rothkirch**

**Betreuer:** Florian Röhrbein

**Leitung:** Prof. Alois Knoll

**Abgabetermin:** 21. Juli 2012

Ich versichere hiermit, dass die vorliegende Seminararbeit selbstständig verfasst und keine weiteren als die angegebenen Hilfsmittel benutzt, sowie die Stellen der Arbeit, die in anderen Werken dem Wortlaut oder dem Sinn nach entnommen sind, durch Angaben der Quellen sichtbar gemacht wurden.

München, 18. Juli 2012

# Inhalt

1. Ein kurzer Streif durch die Geschichte der Kognitiven Architekturen .....	4
2. Begriffsklärung: Kognition, Informationsverarbeitung und Intelligenz .....	5
3. Kognitive Architekturen .....	6
3.1 Grundlagen .....	6
3.2 Unterscheidungsmerkmale und Klassifizierungen .....	6
3.3 Strukturelle Analyse.....	8
3.4 Funktionale Analyse .....	11
3.4.1 Kognitives Spektrum .....	11
3.4.2 Verhaltensspektrum.....	12
4. Beispielarchitekturen .....	13
4.1 Kurzüberblick: SOAR .....	13
4.2 ICARUS'.....	14
4.2.1 Zielerreichung .....	14
4.2.2 Arbeitszyklus .....	14
4.2.3 Auszug aus der Dokumentation.....	14
4.2.4 Organisation.....	15
5. Bilanz und Ausblick.....	16
6. Literaturverzeichnis.....	17
7. Abbildungsverzeichnis.....	18
8. Index.....	18

# 1. Ein kurzer Streif durch die Geschichte der Kognitiven Architekturen

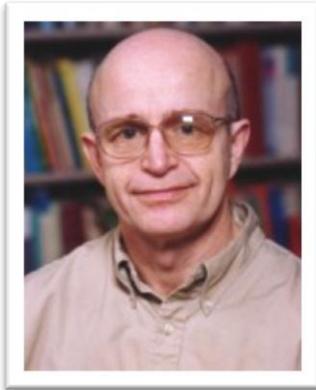


Abb.1: J. R. Anderson

Die Ursprünge des Konzepts der kognitiven Architekturen liegen wohl in dem von *John R. Anderson* (\*1947) im Jahre 1973 entwickelten Modell *ACT-R*, das sich im Wesentlichen auf die Derivate eigener Theorien stütze. Inspiriert wurde er dabei von *Allen Newell* (1927-1992), selbst Pionier im Bereich der Forschung um künstliche Intelligenz. Neu war, das nun versucht wurde, einen viel allgemeineren und umfassenderen Ansatz zur Modellierung der künstlich en Intelligenz zu suchen, der sich jedoch immer noch konkret auf Problemstellungen adaptieren ließ. Dieses Framework- oder auch Blueprint-Prinzip wurde schließlich inhärentes Charakteristikum kognitiver Architekturen.

Doch auch Newell selbst erarbeitete ab 1983 seine eigene Architektur *SOAR* (→4.1), in die seine Forschungen kumulierten und an deren Weiterentwicklung er sich fortan machte. Unterstützt wurde er dabei von *Herbert A. Simon* (1916-2001), mit dem er einen Lehrstuhl für künstliche Intelligenz an der *Carnegie Mellon University* in *Michigan* einrichtete. Guten Gewissens kann man hier die Wiege für die weiteren Entwicklungen im Bereich der kognitiven Architekturen suchen<sup>1</sup>, die nun in den folgenden Jahren, mit Ausnahme der *Psi-Theorie* von *Dörner* (Deutschland), fast ausschließlich in den USA stattfanden<sup>2</sup>.

Nach der Etablierung der Methodik waren nun Theoretiker in der Pflicht wieder Ordnung in den kaum noch überschaubaren „Zoo“ der Architekturen zu bringen. Grundlage für die Unterscheidung waren zum einen die verschiedenen Ansätze der Datenverarbeitung und die Struktur der Agenten<sup>3</sup>, die die Architekturen implementieren können (→3.2). Ob die Agenten dabei physisch in Form von Robotern existieren, oder nur rein virtuell, spielt für die Architektur nur eine untergeordnete Rolle.

Da die Ursprünge im Bereich der Psychologie liegen, findet man heutige Anwendungsgebiete von kognitiven Architekturen konzentriert in der Nachbildung der menschlichen Kognition. Ihr Einsatzgebiet beschränkt sich zwar noch weitgehend auf die Forschung – Das Konzept bietet aber Potenzial überall dort Einzug zu finden, wo effizient auf Bewährtes zurückgegriffen werden muss.

---

<sup>1</sup> (Vernon, et al., 2007) p. 162

<sup>2</sup> Zwei weitere europäische Architekturen: [DUAL](#) (Bulgarien) und [COGAFF](#) (England)

<sup>3</sup> "Agents are autonomous or semi-autonomous hardware or software systems that perform tasks in complex, dynamically changing environments." (Müller 1996, S. 1)

## 2. Begriffsklärung: Kognition, Informationsverarbeitung und Intelligenz<sup>4</sup>

**Kognition** charakterisiert die Gesamtheit aller Vorgänge, die den drei grundlegenden Transaktionen auf Information dienen, nämlich der *Aufnahme*, *Verarbeitung* und *Speicherung*. Angereichert wird diese Definition um eine transzendenten Dimension, getreu dem Leitsatz von Descartes: „*cogito ergo sum*“ („*ich denke, also bin ich*“).

Der Verarbeitung kommt dabei im Kontext kognitiver Architekturen eine besondere Bedeutung zu. Das **Informationsverarbeitungsgesetz** geht von der Annahme aus, dass sich menschliches Verhalten analog zu Computern in einem Modell symbolischer Manipulation beschreiben lässt:



Abb.2: Informationsverarbeitung bei Menschen und Computern

Der Grad, den man dieser Analogie zukommen lässt, ist eines von mehreren Unterscheidungsmerkmalen kognitiver Architekturen (→3.2).

Newell und Simon sahen eine Äquivalenz vorliegen und postulierten darauf aufbauend die *Physical Symbol System Hypothesis (PSS)*. Jedes symbolverarbeitende System könne demnach *Intelligenz* entwickeln und menschliche Intelligenz beruhe gerade auf der Manipulation von Symbolen.

Newell war davon überzeugt, dass sich die gesamte Bandbreite der Kognition in einer universellen Theorie vereinen ließe und implementierte SOAR (→4.1) als eigenen Kandidaten. Davon inspiriert bildete sich eine Ansammlung weiterer Theorien, die unter den *Unified Theories of Cognition (UTC)* zusammengefasst werden. Ihnen allen ist die PSS gemein.

Der Begriff der **Intelligenz** wurde vielfach versucht zu definieren. Der Kern spiegelt sich aber recht gut in einer Formulierung von *Phares*:

„**Intelligenz** umfasst die Fähigkeit

- zur adäquaten und schnellen Anpassung an neue Situationen und sich verändernde Anforderungen
- zum Lernen oder zur optimalen Nutzung von Erfahrung und Übung
- und zum abstrakten Denken und Gebrauch von Symbolen und Begriffen.“

*Phares, 1984*

---

<sup>4</sup>Siehe auch (Rothkirch, 2012)

## 3. Kognitive Architekturen

### 3.1 Grundlagen<sup>5</sup>

Im Kontext der Informationsverarbeitung versteht man unter einer **Architektur** die Menge der Mechanismen, die zum Erlangen kognitiver Fähigkeiten dienen. Diese sind, wie bereits erwähnt, im Rahmen einer Spezifikation erfasst und geben den daraus abgeleiteten **kognitiven Systemen** eine feste gemeinsame Struktur.

Die kognitiven Systeme unterscheiden sich ferner wesentlich in den eingepflegten und selbst erworbenen **Informationen**. Diese werden in ihrer *atomaren* Form als **Symbole** bezeichnet. Während die PSS davon ausgeht, dass die gespeicherten Symbole nur internen Repräsentationswert besitzen, existiert auch die *Computational Theory of Mind (CTM)*, welche darüber hinaus einen direkten Bezug zu externen Objekten postuliert. Bei letzterer ist unter Anderem problematisch, dass eine objektive Wahrnehmung der Umwelt gegebenenfalls gar nicht möglich ist und somit die Bewertung solch eines Bezugs schwer fällt.

Kognitive Architekturen sind in der Regel **berechnungsvollständig**, können also theoretisch einander emulieren. Wesentliches Unterscheidungsmerkmal ist allerdings der Aufwand, den man bei der konkreten Implementierung zusätzlich leisten muss, da sich die Architekturen auf verschiedene Aufgabengebiete spezialisiert haben.

### 3.2 Unterscheidungsmerkmale und Klassifizierungen

#### a) Interne Modellierungen<sup>6</sup>

Kognitive Architekturen lassen sich anhand der internen Vernetzung der Teilkomponenten aufteilen.

1. **Integrierend**: Komponenten sind eher unselbstständig, erfüllen einen bestimmten Zweck und werden in die Gesamtarchitektur integriert. Hierunter fallen auch z. B. hierarchische oder Kaskadenarchitekturen.
2. **Geschichtet**: Bestehen aus mehreren eigenständigen Schichten, ohne Rücksicht auf deren Anordnung (seriell, parallel oder hierarchisch).
3. **Modular**: Ähnlich aufgebaut wie integrierende Architekturen, aber die Komponenten erfüllen spezifischere Zwecke und sind prinzipiell eigenständig.
4. **Konnektionistisch**: Orientieren sich an den biologischen neuronalen Systemen, also an der Vernetzung vieler gleichartiger, kleiner, unselbstständiger Komponenten. Tritt oft in Zusammenhang mit emergenten Systemen auf (→3.2b).

---

<sup>5</sup> (Sessler, 1996) S. 12ff

<sup>6</sup> (Sessler, 1996) S. 15-16

## b) Kognitivistische und emergente Ansätze<sup>7</sup>

Zwei grundlegend philosophisch verschiedene Herangehensweisen werden durch *kognitivistische* (*cognitivist*) und *emergente* (*emergent*) Architekturen verkörpert. Während Ersteres dem Prinzip der Informationsverarbeitung (→2.) folgt, gehen emergente Architekturen davon aus, dass Intelligenz spontan aus dem Zusammenschluss der (primitiven) Komponenten *auftauchen* (*emergieren*) kann. Darüber hinaus gibt es natürlich auch hybride Architekturen, die beide Ansätze zu vereinen suchen.

<b>The Cognitivist vs. Emergent Paradigms of Cognition</b>		
<b>Characteristic</b>	<b>Cognitivist</b>	<b>Emergent</b>
Computational Operation	Syntactic manipulation of symbols	Concurrent self-organization of a network
Representational Framework	Patterns of symbol tokens	Global system states
Semantic Grounding	Percept-symbol association	Skill construction
Temporal Constraints	Not entrained	Synchronous real-time entrainment
Inter-agent epistemology	Agent-independent	Agent-dependent
Embodiment	Not implied	Cognition implies embodiment
Perception	Abstract symbolic representations	Response to perturbation
Action	Causal consequence of symbol manipulation	Perturbation of the environment by the system
Anticipation	Procedural or probabilistic reasoning typically using <i>a priori</i> models	Self-effected traverse of perception-action state space
Adaptation	Learn new knowledge	Develop new dynamics
Motivation	Resolve impasse	Increase space of interaction
Relevance of Autonomy	Not necessarily implied	Cognition implies autonomy

Abb.3: Characteristics of cognitivist and emergent Systems<sup>8</sup>

In der Abbildung sind ausführlich weitere Merkmale aufgeführt, die sich aus den verschiedenen Ansätzen ergeben. Auffällig ist, dass viele der emergenten Eigenschaften auch auf Neuronale Netze zutreffen und in der Tat sind gerade diese das klassische Beispiel einer emergenten Architektur.

<sup>7</sup> (Vernon, et al., 2007) pp 151-164 behandelt ausführlich die beiden Strömungen

<sup>8</sup> Detaillierte Beschreibung bei (Vernon, et al., 2007) p. 153

### 3.3 Strukturelle Analyse<sup>9</sup>

Eine kognitive Architektur besteht aus Teilsystemen, deren konkrete Struktur, Eigenschaften und Vernetzung charakteristisch sind.

#### a) Informationsarten

Das relevante Wissen befindet sich in Langzeitspeichern der Architektur und wird grundsätzlich vereinfacht in **deklaratives** und **nicht deklaratives** Wissen aufgeteilt:

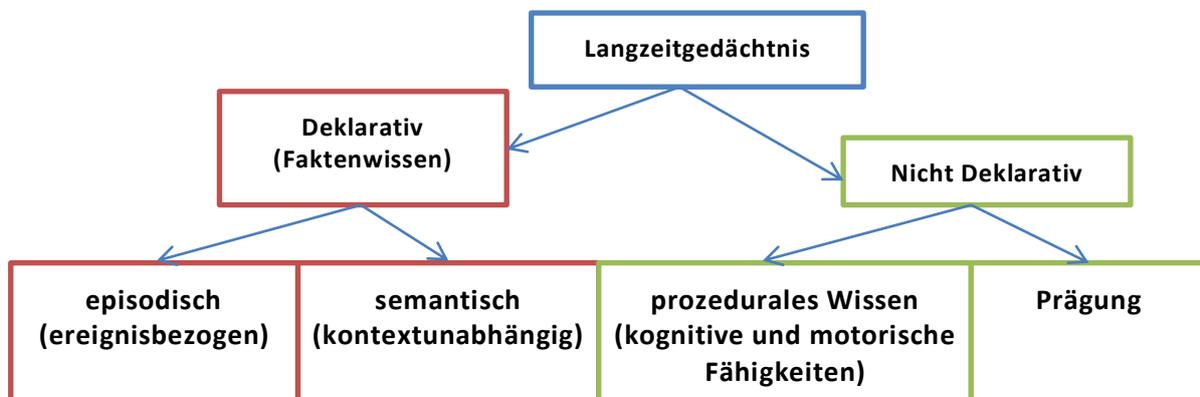


Abb.4: Hierarchie der Informationsarten

#### b) Informationsspeicher

Beim Menschen wird plakativ zwischen einem Kurz- und Langzeitgedächtnis als **Speichermodule** unterschieden. Eine räumliche Trennung oder klare Hierarchie wurde jedoch nach aktuellem Stand falsifiziert. Vielmehr existieren heute Modelle, die beispielsweise von einem Arbeitsgedächtnis ausgehen<sup>10</sup> oder sogar das gesamte Gedächtnis als einzelnes Netzwerk auffassen, das sich nur in der Konfiguration der Aktivierungen unterscheidet<sup>11</sup>.

Kognitive Architekturen bedienen sich trotzdem häufig des plakativen Ansatzes, setzen aber nicht notwendigerweise eine räumliche Trennung um. Langzeit-Information kann sich zusammen mit Kurzzeit-Informationen in Caches befinden.

<sup>9</sup> Siehe auch (Rothkirch, 2012), Abbildungen entsprechend auch hier entnommen

<sup>10</sup> Baddley 1974

<sup>11</sup> Cowan-Oberauer

Beim **Zugriff** spielt die **Speicherorganisation** eine wesentliche Rolle. Neben der hierarchisch kategorialen Klassifizierung von Informationen bietet sich vor allem eine assoziative Verknüpfung, wie in der folgenden Abbildung veranschaulicht, an. Solche Mindmaps, im wahrsten Sinne des Wortes, bieten starke Vorteile beim Zugriff auf Problemwissen, allerdings ist die Suche nach nicht assoziiertem Wissen sehr aufwendig.

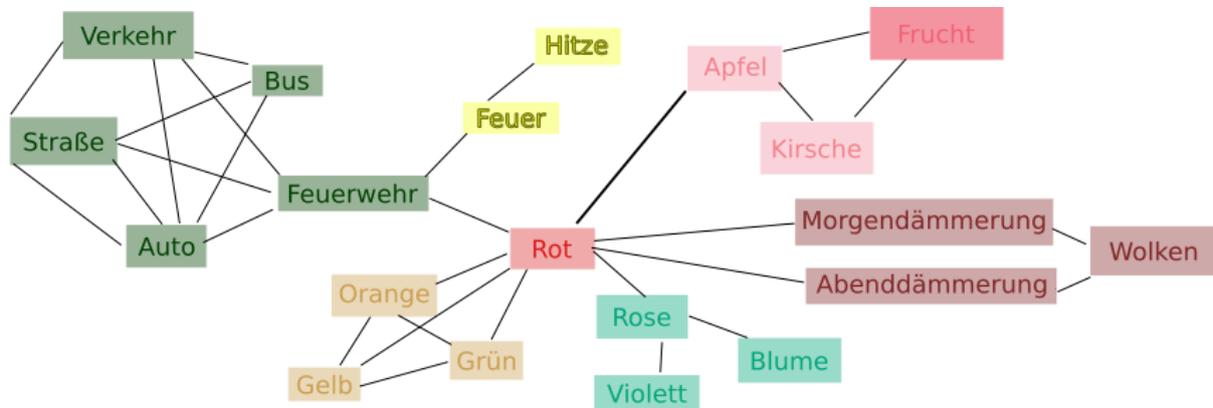


Abb.5: Aktivierungsausbreitung mit Assoziationsmechanismen

### c) Informationsverarbeitungsprozess

Beim Menschen gliedert sich die Informationsverarbeitung gemäß der Abbildung in einen Kreislauf aus Wahrnehmen, Denken und Handeln. Dabei stellt das *Denken* die kontrollierte Informationsverarbeitung dar, mit *Erinnern* und *Lernen* als Informationsabruf aus dem *Gedächtnis*.

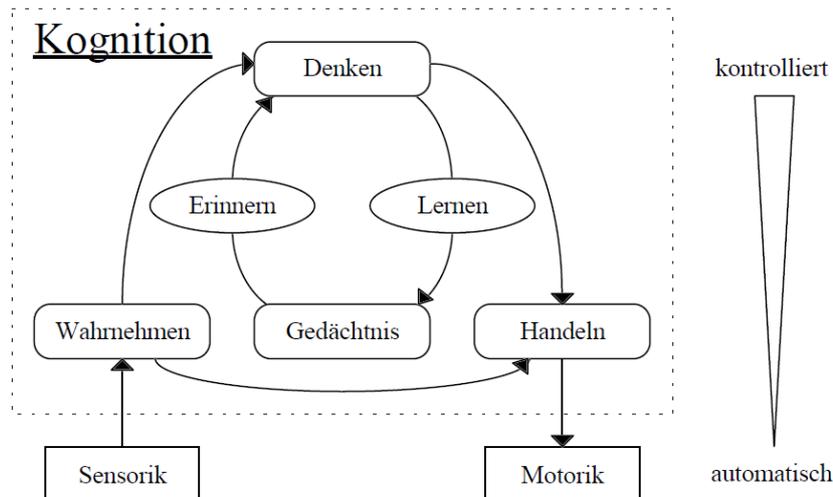


Abb.6: Informationsverarbeitungsprozess

Auch kognitive Architekturen implementieren diesen Prozess und stellen eine Auswahl der folgenden Fähigkeiten zur Verfügung.<sup>12</sup>

#### 1. Wahrnehmen und Handeln

Schnittstelle zum Kontext der Umwelt; unterscheiden sich häufig in den Implementierungen der gleichen Architektur.

#### 2. Erinnern und Lernen

Entspricht dem De- und Encodieren der Informationen auf Basis von Problemstellungen.

#### 3. Problemlösen und Planen

Architekturen müssen Problemstellungen und Ihre Lösungen als eine geordnete Folge von Aktionen formulieren können. Planen kann reaktiv, situativ oder deliberativ sein (→3.4.1).

#### 4. Logisches Denken und Entscheiden

Aus Wissen können deduktive und induktive Schlussfolgerungen gezogen werden, die die Entscheidung zur Problemlösung beeinflussen.

#### 5. Interaktion und Kommunikation

Häufig nur optional wird eine Kommunikation mit anderen Agenten angeboten (→3.4.1e)).

<sup>12</sup> Siehe auch (Rothkirch, 2012) für einen detaillierteren Einblick

## 3.4 Funktionale Analyse<sup>13 14</sup>

Die funktionalen Anforderungen an kognitive Architekturen umfassen einen vielfältigen Katalog und lassen sich im Wesentlichen in das kognitive und das Verhaltensspektrum gliedern. Umgekehrt kann man die Auflistung als Gütekriterien für eine gegebene Architektur betrachten. Diese lassen sich grundsätzlich in das *kognitive* und das *Verhaltensspektrum* unterteilen:

### 3.4.1 Kognitives Spektrum

#### a) Reaktivität

Dynamische Umgebungen erfordern dass ein Agent jederzeit und möglichst in Echtzeit zu reagieren in der Lage ist. Dazu ist insbesondere eine Unterbrechbarkeit und Priorisierung der Abläufe notwendig.<sup>15</sup>

#### b) Situativität

In der Realität kann ein Plan häufig nicht wie angedacht zu Ende geführt werden, sondern es muss auf unvorhergesehene und unplanmäßige Situationen adäquate Lösungsmöglichkeiten vorgesehen werden. Dies wird unter anderem durch ein hohes Abstraktionsvermögen vereinfacht.

#### c) Deliberativität

Das willkürliche Verhalten gemäß einer internen Zielsetzung wird häufig noch einmal separat betrachtet. Es findet zunächst unabhängig von äußeren Einflüssen statt.

#### d) Adaptivität

Die Anpassungsfähigkeit kann als Kombination aus Reaktivität und Situativität betrachtet werden. Da sie aber als zusätzliche Komponente eine gewisse *Permanenz* und damit Lernen fordert, erhält die Adaptivität eine eigene Dimension (→3.3b).

#### e) Sozialität

Unter Umständen lassen sich durch Informationsaustausch mit anderen Agenten erhebliche Vorteile erzielen (*Schwarmverhalten*). Probleme bereiten dabei allerdings die Synchronisierung und Verlässlichkeit der Daten. Trotzdem kommt der Sozialität in jüngster Zeit verstärkt Aufmerksamkeit zu.

---

<sup>13</sup> (Langley, et al., 2008) pp 13-15

<sup>14</sup> (Sessler, 1996) S. 111-122

<sup>15</sup> Vergleichbar mit den Anforderungen an Scheduler in Betriebssystemen

## 3.4.2 Verhaltensspektrum

### a) Rationalität

Rationalität lässt sich anhand des deliberativen Verhaltens (→3.4.1c) bewerten, betrifft aber die ganze Architektur. Teilkriterien wären **Effizienz, Kohärenz von Wissen und Handlung** und **Kohärenz des Gesamtverhaltens**. Fehlerquellen sind falsche Implementierungen der Schlussfolgerungen (Reasoning), aber auch Fehler und Probleme bei der Wissensaneignung<sup>16</sup> können zu augenscheinlich unlogischen Schlussfolgerungen führen. Häufig wird bei Widersprüchen einfach die ältere Wissenseinheit ersetzt. Die Kohärenz des Gesamtverhaltens ist umso stärker gefährdet, je dezentraler sie angesetzt ist. Insbesondere kann dann die Reaktivität einzelner Komponenten im Widerspruch zu dem Gesamtziel des Systems stehen.

### b) Robustheit

Robustheit spielt besonders bei physischen Agenten eine Rolle, wo analoge Sensordaten zu Unstimmigkeiten führen können, oder sogar gezielte Manipulation einwirken kann. Dabei ist eine Architektur umso robuster, je besser sie mit Widrigkeiten wie unvollständigem, unsicherem oder sogar falschen Wissen umgehen kann.

### c) Flexibilität und Vielseitigkeit

Ein weiteres Zeichen von Intelligenz ist *Flexibilität* im Umgang mit Problemen, was in der Regel auch mit einem größeren Einsatzbereich einhergeht. Diese korreliert auch grundsätzlich mit dem kompletten kognitiven Spektrum. Allgemein ist es anzustreben eine maximal *vielseitige* Architektur zu entwerfen, die sich flexibel auf einzelne Problemfelder spezialisieren kann. Immerhin wurde die prinzipielle Machbarkeit schon durch das menschlichen Gehirns belegt.

---

<sup>16</sup> Siehe dazu auch das sog. „Symbol Grounding Problem“

## 4. Beispiellarchitekturen

### 4.1 Kurzüberblick: SOAR<sup>17</sup>

SOAR (*State, Operator And Result*) wurde 1983 von A. Newell, J. Laird und P. Rosenbloom entwickelt und liegt heute in der Version 9.3.2 vor<sup>18</sup>. Alle Aufgaben werden als Problemräume repräsentiert, wobei die Suche darin im zentralen Arbeitsspeicher (*working memory*) stattfindet. Das Zusammenspiel mit dem Produktionsspeicher, wird in folgender Grafik veranschaulicht.

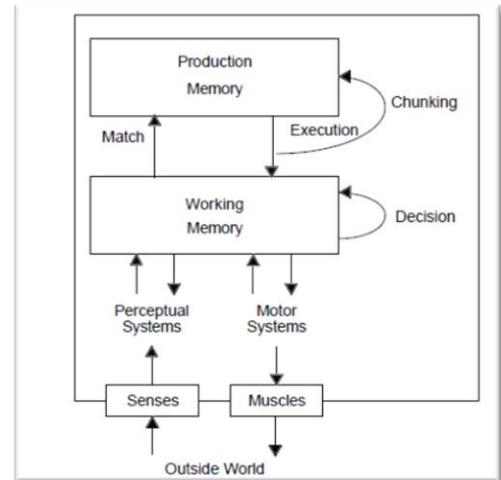


Abb.7: Architektur von SOAR

#### a) Grundprinzipien von SOAR

1. Problemlösen wird als Suche in Problemräumen repräsentiert.
2. *Dauerhaftes* Wissen wird durch Produktionsregeln im Langzeitspeicher repräsentiert.
3. *Temporäres* Wissen wird durch Objekte im Arbeitsspeicher repräsentiert.
4. Neue Ziele werden nur generiert, wenn *Sackgassen (Impasses)* auftreten.
5. Lernmechanismus: *Chunking* und ab Version 9.0.0 auch *Reinforcement Learning*

#### b) Elaborationszyklus

Ausgehend von einem Anfangszustand, wird dieser durch sukzessive Veränderung mittels Operatoren in einen Zielzustand überführt. Dabei orientiert man sich insofern am Menschen, dass eine Produktionsregel zwar einen Operator vorschlägt, aber situationsabhängig ein bestimmter ausgewählt werden muss. Eine Sackgasse tritt dann auf, wenn nicht genau ein Operator ausgewählt werden kann. Input und Output sind dabei optional.

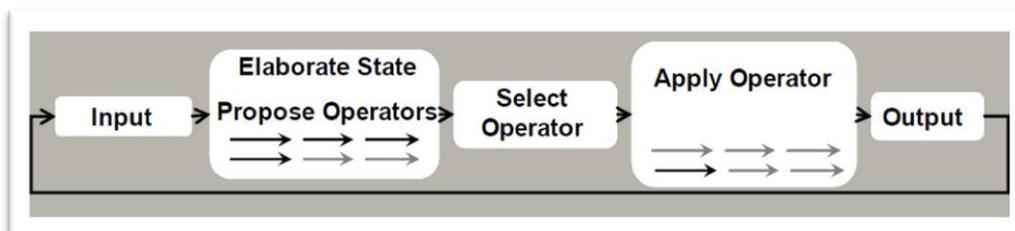


Abb.8: Elaborationszyklus bei SOAR

<sup>17</sup> Ausführlich bei (Rothkirch, 2012), Abbildungen auch entsprechend hier entnommen

<sup>18</sup> Kostenlos verfügbar auf <http://sitemaker.umich.edu/soar/home>

## 4.2 ICARUS<sup>19,20</sup>

ICARUS ist eine der neueren kognitiven Architekturen und wurde 2004 von *Langley, Cummings und Shapiro* entwickelt. Wissen wird hier entweder als *Konzept (concept)* oder *Fähigkeit (skill)* eingeordnet. Erstere repräsentieren Umgebungssituationen in Abhängigkeit von anderen Konzepten bzw. *Wahrnehmungen (percepts)*, Fähigkeiten beschreiben eine Anleitung zum lösen eines *Problems (goal)*, indem diese in Teilprobleme zerlegt werden.

### 4.2.1 Zielerreichung

Probleme sind direkt lösbar, wenn eine entsprechende Fähigkeit bereits erworben wurde. Andernfalls setzt ein Analyseprozess in gang, der sich rückwärts an ähnlichen Zielen und zugehörigen Fähigkeiten orientiert. Eine andere Möglichkeit ist die Umgebung so zu verändern, dass in diesem Konzept eine Fähigkeit anwendbar wird.

### 4.2.2 Arbeitszyklus

Jeder Zyklus folgt dem *recognize-act* Prinzip: Zunächst werden per Mustererkennung gefundene Wahrnehmungen in einen Buffer geschrieben und mit bekannten atomaren Konzepten verglichen. Übereinstimmungen werden als Vermutungen (*beliefs*) in einem Kurzzeitspeicher abgelegt. Nachdem iterativ auch abstraktere Konzepte erkannt wurde, wird nun hierarchisch absteigend im Fähigkeitenbaum nach anwendbaren Fähigkeiten gesucht, die noch unerfüllte Teilprobleme lösen können. Auf diese Weise wird Schritt für Schritt das Hauptproblem gelöst.

### 4.2.3 Auszug aus der Dokumentation<sup>21</sup>

In der Abbildung ist ein Auszug aus dem Wahrnehmungsbuffer zu sehen. Hier wird beispielhaft ein Sensorinput mit Koordinaten und Abmessungen veranschaulicht.

```
(a)
(block A xpos 10 ypos 2 width 2 height 2)
(block B xpos 10 ypos 4 width 2 height 2)
(block C xpos 10 ypos 6 width 2 height 2)
(table T1 xpos 20 ypos 0 width 20 height 2)
(hand H1 status empty)
```

Abb.9: Beispielinhalt des Wahrnehmungsbuffers

Die Programmierung ähnelt funktionalen Sprachen, wie man an den zwei folgenden Befehlen erkennen kann:

- *(run N)*: führt ein Programm in einer Schleife N mal aus.
- *(grun N)*: so wie run N, bricht aber auch schon bei Lösung aller Hauptprobleme ab.

---

<sup>19</sup> (Langley, et al., 2008) p. 4

<sup>20</sup> (Vernon, et al., 2007) p. 165

<sup>21</sup> Details entnimmt der geneigte Leser der aktuellen und ausführlichen Dokumentation (Langley, et al., 2011)

#### 4.2.4 Organisation<sup>22</sup>

Die vier Hauptmodule sind treffend mit *Argus*, *Daedalus*, *Labyrinth* und *Maender* benannt.

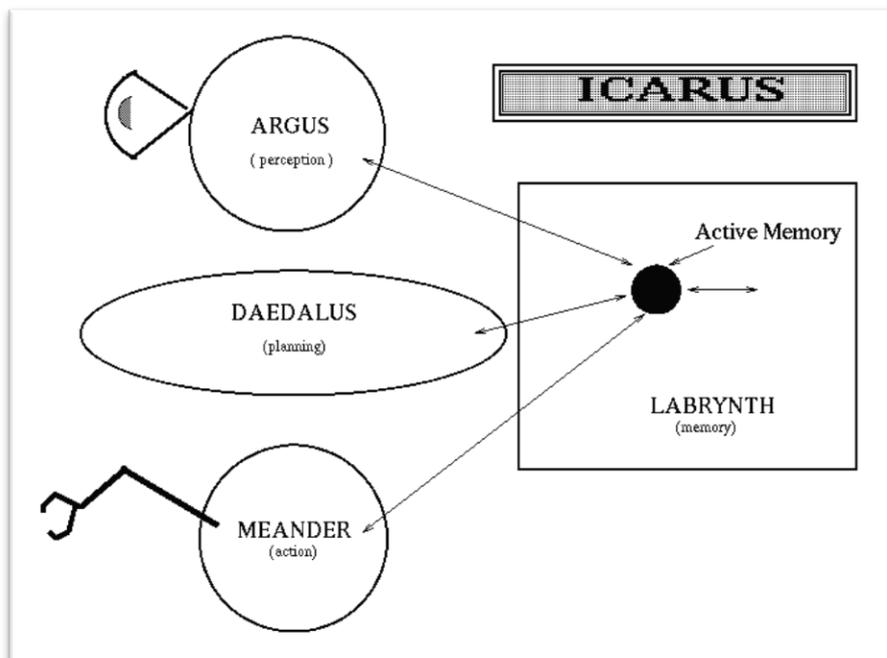


Abb.10: Hauptmodule von ICARUS

##### a) Argus (Wahrnehmung)

Argus generiert die Objekte aus der Umgebung. Dabei orientiert es sich am aktuellen Plan von Daedalus und setzt seine Ressourcen entsprechend ein. Außerdem kann ein spezielles Event an das Labyrinth gesendet werden, wenn die Umgebung außergewöhnliche Veränderung zeigt.

##### b) Daedalus (Planung)

Daedalus führt die Planung für die Probleme durch, die im Labyrinth abgelegt wurden (→4.2.1,4.2.2).

##### c) Labyrinth (Speicher)

Der Speicher unterscheidet sich von anderen Architekturen wesentlich dadurch, dass er auf eine Hierarchie probabilistischer Konzepte aufbaut, seinen Inhalten also Wahrscheinlichkeiten zuordnet. Damit bereitet ICARUS der Umgang mit widersprüchlichen Informationen wenig Probleme.

##### d) Maender (Output)

Maender führt die Pläne von Daedalus aus. Idealerweise befindet sich das System danach im angestrebten Zustand, allerdings kann Argus bei Abweichungen jederzeit eine Neuplanung anstoßen.

<sup>22</sup> Weitere auszeichnende Fähigkeiten: <http://cogarch.org/index.php/Icarus/Architecture>

## 5. Bilanz und Ausblick

Gezeigt wurde, was eine kognitive Architektur ausmacht und dies anhand von zwei konkreten Beispielen veranschaulicht. Zwar liegt ihr Haupteinsatzgebiet nach wie vor in der Forschung, allerdings sind die Erkenntnisse die man hieraus gezogen hat beträchtlich. Nicht nur ist man heute dazu in der Lage Autos autonom durch den Straßenverkehr fahren zu lassen<sup>23</sup>, mittlerweile lassen die USA Dronen sogar semiautonom im Krieg mitwirken und der Schachweltmeister wurde schon seit einiger Zeit bezwungen. Im Jahre 2050 soll planmäßig das erste Fußball-WM Spiel Mensch gegen Roboter stattfinden<sup>24</sup> und überhaupt wurden die wichtigsten Bereiche unseres Lebens von Anwendungen der künstlichen Intelligenz nicht nur erreicht, sondern regelrecht durchdrungen.

Allerdings gibt es durchaus noch Domänen die enorme Schwierigkeiten bereiten. Bewusstsein oder bewusstes erleben von Emotionen sind praktisch noch unerreicht. Natürlich liegt das auch daran, dass wir das menschliche Gehirn noch lange nicht vollständig verstanden und entschlüsselt haben. Erst wenn dies gelungen ist, kann man von künstlicher Intelligenz im Sinne von Hollywood reden.

Auch kognitive Architekturen selbst haben ihr Potenzial noch bei Weitem nicht ausgeschöpft. Zwar implementieren sie einige, jedoch selten alle der in Kapitel 3 aufgeführten Anforderungen und müssen noch häufig Kompromisse machen. Allein die Hardware oktroyiert schließlich immer noch ihre Beschränkungen auf die Systeme.

In jedem Fall wird die Entwicklung aber rapide voranschreiten und man darf gespannt sein, wohin sie führt. Dabei darf aber nicht - ganz besonders im Bereich der militärischen Anwendungen - der Mensch außer Acht gelassen werden. Mit Technologie geht immer, ganz ohne Wertung, eine große Verantwortung einher.

---

<sup>23</sup> <http://www.lasvegassun.com/news/2012/may/07/nevada-issues-google-first-license-self-driving-ca/>

<sup>24</sup> <http://www.heise.de/newsticker/meldung/10-RoboCup-German-Open-eroeffnet-1219435.html>

## 6. Literaturverzeichnis

**Funke Joachim** Kognitive Modellierung. - Uni Heidelberg : [s.n.], 2002.

**Gerke Peter R.** Wie denkt der Mensch? Informationstechnik im Gehirn [Buch]. - München : J. F. Bergmann Verlag, 1987.

**Horneck Max** Kognitive Dynamik [Buch]. - [s.l.] : Hempen, 1999.

**Kaczmarczyk Peter Paul** SOAR - Eine Kognitive Architektur [Bericht]. - Bexbach : Dr. Michael Kipp, Dr. Alassane Ndiaye, Dr. Dominik Heckmann, Michael Feld, 2006.

**Langley Pat und Choi Dongkyu** ICARUS Users's Manual [Artikel]. - Palo Alto : Institute for the Study of Learning and Expertise, 2011.

**Langley Pat, Laird John E. and Rogers Seth** Cognitive architectures: Research issues and challenges [Article] // Cognitive Systems Research. - 2008. - pp. 1-20.

**Michalk Christian** Eine Einführung in Kognitive Architekturen [Bericht]. - Losheim : Dr. Michael Kipp, Dr. Alassane Ndiaye, Dr. Dominik Heckmann, Michael Feld, 2006.

**Posperschill Markus** Konnektionismus und Kognition [Buch]. - [s.l.] : W. Kohlhammer, 2004.

**Rothkirch Iris** Überblick kognitiver Architekturen [Bericht]. - Technische Universität München : Florian Röhrbein, 2012.

**Sessler Ralf** Das Agentenmodell INTERAP im Vergleich mit kognitiven Architekturen [Bericht]. - [s.l.] : Professor J. Siekmann, 1996.

**Vernon David, Metta Giorgio and Sandini Giulio** A Survey of Artificial Cognitive Systems: Implications for the Autonomous Development of Mental Capabilities in Computational Agents [Article] // IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. - April 2007. - 11/2. - pp. 151-180.

## 7. Abbildungsverzeichnis

Abb.1: J. R. Anderson .....	4
	<a href="http://act-r.psy.cmu.edu/people/ja/">http://act-r.psy.cmu.edu/people/ja/</a>
Abb.2: Informationsverarbeitung bei Menschen und Computern .....	5
Abb.3: Characteristics of cognitivist and emergent Systems.....	7
	(Vernon, et al., 2007) p. 153
Abb.4: Hierarchie der Informationsarten.....	8
Abb.5: Aktivierungsausbreitung mit Assoziationsmechanismen.....	9
Abb.6: Informationsverarbeitungsprozess.....	10
Abb.7: Architektur von SOAR .....	13
Abb.8: Elaborationszyklus bei SOAR .....	13
Abb.9: Beispielinhalt des Wahrnehmungsbuffers .....	14
	(Langley, et al., 2011)
Abb.10: Hauptmodule von ICARUS .....	15
	<a href="http://cogarch.org/index.php/Icarus/Architecture">http://cogarch.org/index.php/Icarus/Architecture</a>

## 8. Index

ACT-R	3	Integrierende Architekturen	5
Adaptivität	10	Intelligenz	4
Agent	3	John R. Anderson	3
Allen Newell	3	Kaskadenarchitekturen	5
Arbeitszyklus	13	Kognition	4
Argus	14	Kognitive Architektur	5
Baddley	7	kognitiven Systemen	5
beliefs	13	Kognitives Spektrum	10
Chunking	12	Kognitivistisch	6
COGAFF	3	Kohärenz von Wissen und Handlung	11
Computational Theory of Mind	5	Konnektionistische Architekturen	5
Cowan-Oberauer	7	Labyrinth	14
Daedalus	14	Langley	13
deklaratives Wissen	7	Maender	14
Deliberativität	10	Modulare Architekturen	5
DUAL	3	percepts	13
Emergenz	6	Physical Symbol System Hypothesis	4
Flexibilität und Vielseitigkeit	11	Reaktivität	10
Funktionale Analyse	10	Robustheit	11
Geschichtete Architekturen	5	Situativität	10
Herbert A. Simon	3	Soar	3
hierarchische Architekturen	5	Sozialität	10
ICARUS	13	Unified Theories of Cognition	4
Informationsspeicher	7	Verhaltensspektrum	11
Informationsverarbeitungsgesetz	4		