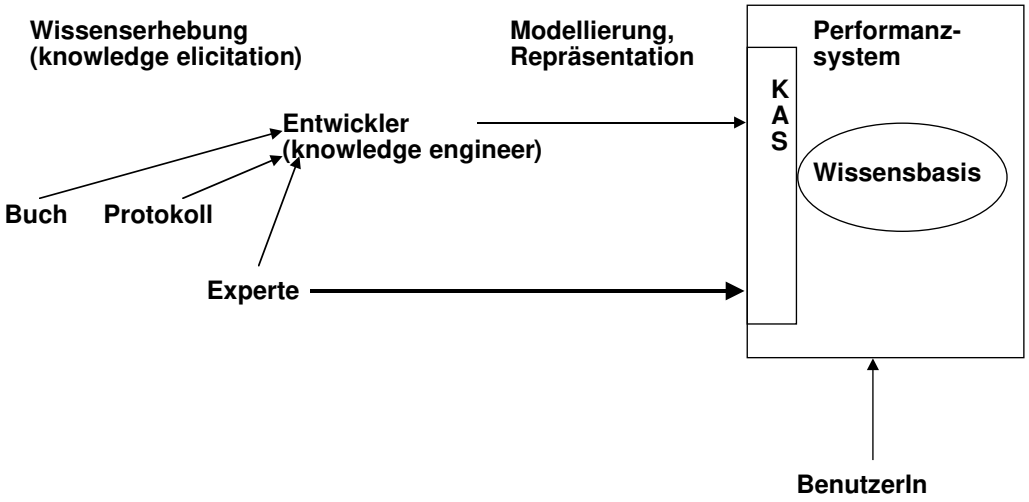
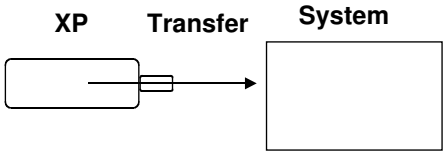
Frühe Wissenserwerbssysteme

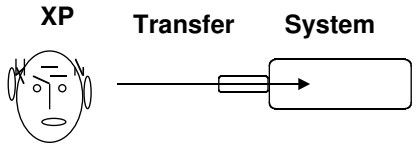


Flaschenhals

Knowledge acquisition is the bottleneck of expert system technology.



Wer ist die Flasche?
Wo liegt das Problem?



Wissen vs. Können

Der Experte als Flasche:

* Das Wissen ist <u>im</u> Experten, aber nicht bewußt, so daß der Experte unvollständige oder falsche Aussagen macht (Cooke, McDonald 1988)

Paradox: je besser der Experte desto schlechter seine Aussagen! (Johnson 1983)

Ablauf des menschlichen Erwerbs einiger Fertigkeiten:
bewußtes Wissen, Übung, stillschweigende Fertigkeit
Buchwissen, Anwendung des Wissens, Können

Amnesiepatienten lernten genauso schnell das Problem der Türme von Hanoi zu lösen wie andere Menschen. Aber keiner wußte (konnte sagen), daß er dies Problem kennt und lösen kann!
(Studie von Cohen 1987)

Sprechen, Komponieren, Dichten vs. Linguistik, Musikwissenschaft, Kritik

Introspektion

Nisbett, Wilson (1977) Telling more than we can know: Verbal reports on mental processes, Psychological Review Vol. 84, 231 - 259

Versuchspersonen bemerkten nicht,
welche Stimuli ihre Reaktion auslösten,
daß es überhaupt einen Stimulus gabe,
daß sie überhaupt reagierten.

Stattdessen erzählten sie ihre Vorstellungen kognitiver Abläufe gemäß ihrer naiven Psychologie.

Herkner (1980) Attribution - Theorie der Kausalität, Hans Huber Vg.
Erklärungen des eigenen Verhaltens werden genauso <u>konstruiert</u> wie Erklärungen des Verhaltens anderer. Es gibt keinen direkten Zugang zu eigenen kognitiven Prozessen!

Soziologie

Phänomenologie
Alfred Schütz Alltagswissen als soziale Realität

Ethnomethodologie
Wie Menschen Realität konstruieren und sich entsprechend verhalten

Interaktionismus
Die Rolle der Interaktion bei der Realitätskonstruktion

Alltagswissen vs. Wissenschaftliches Wissen:
Wissenschaftliche Aussagen werden systematisch Zweifeln ausgesetzt.

Literatur

Arbeitsgruppe Bielefelder Soziologen (Hg.)
Alltagswissen, Interaktion und gesellschaftliche Wirklichkeit
Rowohlt Verlag, 1973

Elmar Weingarten, Fritz Sack, Jim Schenkein (Hg.)
Ethnomethodologie -
Beiträge zu einer Soziologie des Alltagshandelns
Suhrkamp Taschenbuch Wissenschaft, 1979

Transfer

Übertragung des Expertenwissens in ein System

Davis (1979)"Interactive transfer of expertise: Acquisition of new inference rules", Artificial Intelligence J., Vol 12, 121 - 158
Boose (1988) "Expertise transfer and complex problems", Boose, Gaines (Hg.) Expertise Transfer System

Verschiedene (Fach-)Sprachen von Experte und Systementwicklerin
Mißverständnisse erkennen
unklare Punkte finden
unvollständige Angaben bemerken

Wissensrepräsentation

*** System als Flasche:**
Adäquatheit des Wissensrepräsentationsformalismus' für bestimmte Aufgaben:
wie leicht ist ein Sachverhalt des Anwendungsbereichs ausdrückbar im Formalismus?
***Representational mismatch* (Buchanan et al. 1983):**
auf welcher Ebene muß ein Sachverhalt im Formalismus beschrieben werden?
entsprechen Konzepte des Sachbereichs Konzepten des Formalismus?
in Hinblick auf Problemlösestrategie?
in Hinblick auf Sachbereichsterminologie?

ONCOCIN - OPAL

Mark Musen, Lawrence Fagan, David Combs, Edward Shortliffe
(Stanford Univ.)
Use of a domain model to drive an interactive knowledge-editing tool,
Int. J. Man-Machine Studies, 1987, Vol. 26, 105 - 121

Kommunikationsprobleme zwischen Experte und Entwickler:
- Entwicklern fehlt Bereichswissen, um Fragen zu stellen
- Experten fehlt die Begrifflichkeit von Problemlösung, Formalisierung
- Experten können ihr eigenes Verhalten schwer explizieren

Wenn nun Experten ihr Wissen selbst eingeben könnten?
- Vorstrukturierung anhand der Problemlösemethode
  der Repräsentationsstruktur
  des Sachbereichs

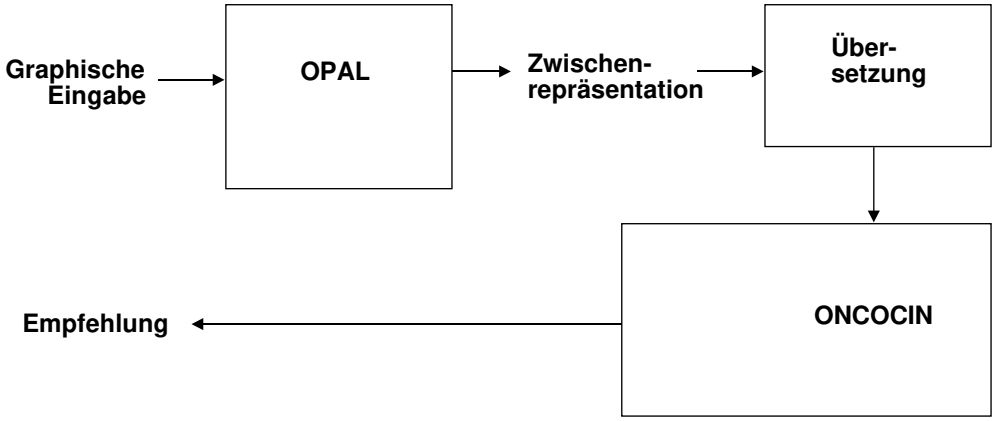
ONCOCIN

Krebstherapie, Protokolle von Behandlungen:
Chemotherapie, Laboruntersuchungen, Bestrahlung
in Abhängigkeit vom Wachstum des Tumors, Nebenwirkungen

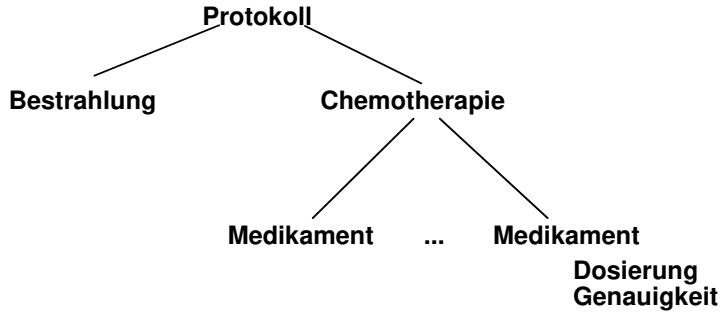
ONCOCIN erstellt eine Therapieempfehlung aufgrund von Patientendaten anhand des Sachbereichswissens.

Sachbereichswissen:
- Frames für Bestrahlung, Chemotherapie, die verschiedenen Medikamente und ihre Darreichungsformen (Objekthierarchie)
- Produktionsregeln für die Bestimmung bestimmter Parameter (Laborparameter, klinische Parameter, Dosierungen)
- Zustandstabellen für den Ablauf einer Therapie (Generatoren)

Transfer von OPAL nach ONCOCIN



Objekthierarchie



Für jedes neue Protokoll werden Instanzen erzeugt (z.B.: P1, C1, M1) und die Werte der Attribute (z.B.Dosis: 30 mg, 1 x pro Zyklus) erhoben.

Graphische Eingabe

Chemotherapie: VAM Tage in Therapie: 21

Medikamente:

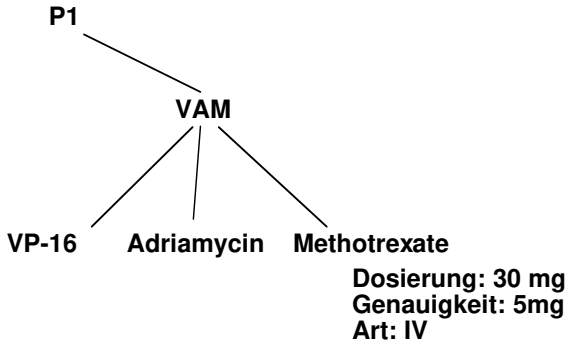
VP-16 Adriamycin Methotrexate _____

Dosis Dosis Dosis Dosis

Chemotherapie: VAM
Medikament: Methotrexate
Modus: normal

Dosis: 30mg Art: intravenoes Genauigkeit: 5 mg wann: 1

Zwischenrepräsentation



Handlungen

Bedingungen für die Änderung des Therapieplans:
Aufschieben der Behandlung
Ausschluß eines Medikaments

Chemotherapie: VAM
Medikament: Methotrexate
WBC (x 1000): ≥ 3.5
Platelets (x 1000): 100 - 150

delay

Bilirubin
Serum Creatinine

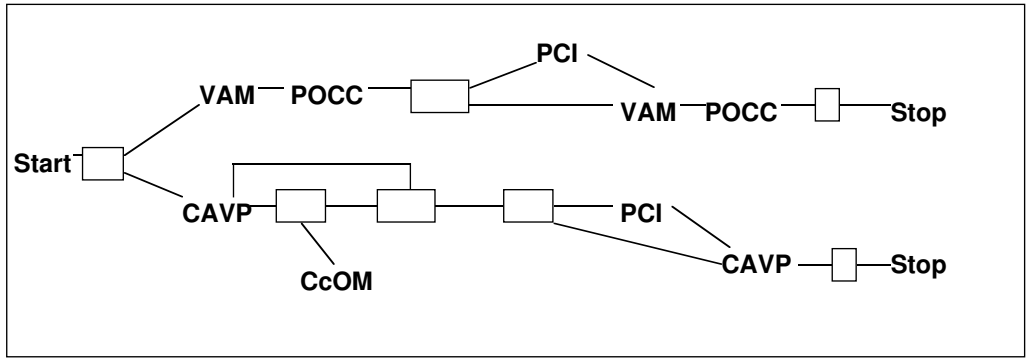
Chemotherapie: VAM
Medikament: Methotrexate
Wert: > 1.5
Handlung: zurückhalten

DELAY in der Zwischenrepräsentation

Handlungsteile von Produktionsregeln mit VISIT.type, VISIT.next.days, DRUGS.to.give werden für ONCOCIN erzeugt.

Bedingungsteile von Produktionsregeln werden für ONCOCIN erzeugt:
IF VAM, creatinine > 1.5
THEN omit methotrexate

Ablauf der Therapie



Graphische Spezifikation des Benutzers wird in Zustandstabelle übersetzt.

Zustand	Bedingung	Zielzustand
Start		VAM
Start		CAVP
VAM		POCC

Aufwand des Wissenserwerbs

ONCOCIN:
Lymphknotenkrebs-Protokolle: 2 Jahre mit 800 Stunden eines Experten
einige zusätzliche Brustkrebs-Protokolle: einige Monate

OPAL:
ein neues Protokoll: einige Tage + einige Tage testen und verbessern
"some 3 dozen protocols": 1 Jahr

Graphische Benutzeroberfläche

strukturiert durch ein gegebenes Sachbereichsmodell

Fenster entsprechen genau den Formblättern der Klinik.
Mediziner tragen das Wissen selbst ein.
Fenster entsprechen jeweils einem Sachbereichsobjekt.
Die Felder zum Eintragen entsprechen genau einem *slot* des Objekt-Frames.
Vollständigkeit der Einträge kann leicht geprüft werden.
Vererbung bestimmter Werte durch die vorgegebene Objekthierarchie.
Die Handlungsmenüs für Fenster entsprechen Handlungs- oder Bedingungs-teilen von Produktionsregeln. Sie verfeinern gegebene Regeln.
Benutzer muß Produktionsregeln nicht kennen, bleibt im eigenen Vokabular.

Vergleich

EMYCIN:
Modellierung auf der symbolischen Ebene;
Objekt: Regeln
große Allgemeinheit - keine Benutzerführung

Clancey:
Modellierung über Problemlösemethoden;
Objekt: Strategien
gültig für alle Probleme derselben Lösungsstrategien (heuristische Klassifikation) - Benutzerführung durch Lösungsstrategie

OPAL:
Modellierung ausschließlich durch Systementwickler;
Objekt: Sachbereichsobjekt
gültig für einen Sachbereich - starke Benutzerführung

... wie ging es mit OPAL weiter?

Protege

Mark Musen (1989)
Automated Support for Building and Extending Expert Models
***Machine Learning Journal,* Vol.4, No. 3/4, 347 - 376**

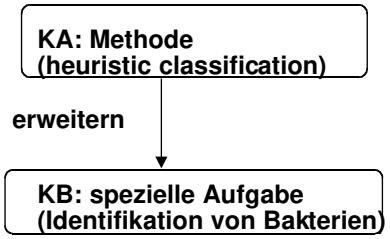
Building a knowledge-base is like building a scientific theory (of how a class of professionals approaches an application task).
In the course of building the expert system, both the knowl. engineers and the experts continually revise their respective mental models.
Knowledge bases are not embodiments of some human problem-solving expertise, but represent only of expert behavior.
KA tools help
system developers to craft a model of the application task or assume that a model of the task area already exists.

Aufbau eines Modells

Analyse auf der Wissensebene vor der Implementierung:
1. Ziele für das System aufstellen
2. Handlungen des Systems bestimmen
3. Wissen ermitteln, mit dem das System Handlungen auswählen kann, um ein Ziel zu erreichen

Sachbereichsunabhängiges Problemlösungswissen kann für verschiedene Modellierungen zur Verfügung gestellt werden:
KADS: interpretation model
Chandrasekaran: generic tasks
Clancey: problem-solving method

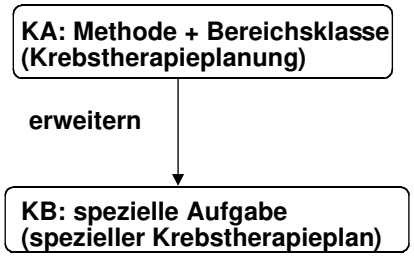
Dies ist der Aufbau des Modells.



Erweiterung eines Modells

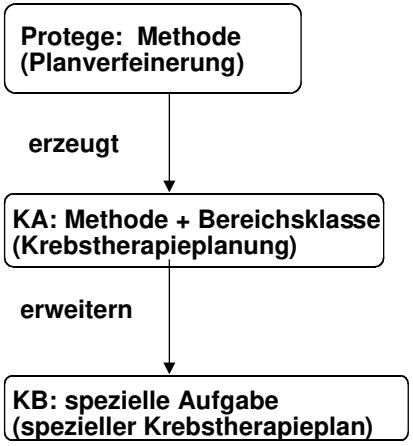
Jede spezielle Situation, in der ein Modell der Problemlösung angewandt werden kann, stellt eine Erweiterung dieses Modells dar.
Da hier nur noch Sachbereichswissen eingegeben werden muß, gibt es kein Paradox des Wissenserwerbs hier, Können muß nicht durch Wissen beschrieben werden.
Gegeben: Problemlösungsmethode + Klasse von Sachbereichen
gefordert: Wissen über einen speziellen Sachbereich
Beispiel eines Werkzeugs für die Erweiterung eines Modells: OPAL

Problem:
OPAL mußte neu programmiert werden, wenn das Modell der Krebstherapieplanung sich änderte!
Kein explizites Modell auf der Wissensebene!
Ausweg: Aufbau und Erweiterung in einem System zusammenbringen!



Protege

Protege unterstützt den Aufbau eines Modells für eine Klasse von Anwendungen;
Protege erzeugt mithilfe des Modells ein System, das die Erweiterung für eine spezielle Aufgabe unterstützt.
Protege hat p-OPAL erzeugt, das die Funktionalität von OPAL herstellt.
Ein neues Expertensystem, HTN, zur Behandlung von zu hohem Blutdruck wurde in wenigen Tagen mithilfe von Protege entwickelt.

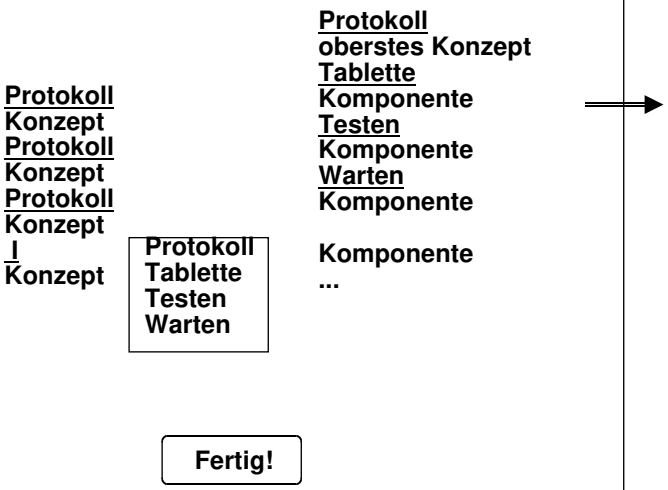


Wissensarten in Protege

Hierarchie von Planungskomponenten:
z.B. Chemotherapie, Medikament, Dosis, ...
mit Attributen
z.B. jetzige Dosis, Anfangsdosis, Dauer, Abbruchsbedingung, ...
Handlungen:
z.B. Zurückstellen einer Behandlung, Vermindern eines Dosis, ...
Eingabedaten:
z.B. Wachstum des Tumors

Graphische Eingabe wie bei OPAL: Ausfüllen von Formblättern, jetzt im Vokabular der Problemlösungsmethode für eine Sachbereichsklasse
Bereits eingegebenes Wissen wirkt sich sofort auf weitere Fenster aus.

Planungskomponenten



Fertig!

Attribute der Komponenten

Attribute von:	Tablette	ausgewähltes Attribut:	Anfangsdosis
Name			
Abbruchsbed.			
Dauer			
Anfangsdosis			
jetzige Dosis			
Intervall			
Darreichung			
...	...	...	...

Fertig!

Interne Datentypen

n-Tupel einer relationalen Datenbank
Datentypen der Attribute:
- konstant *v*
  - z.B. Darreichung bei Blutdruckmedikament immer *oral*
- zur Planungszeit zu bestimmen
  - z.B. jetzige Dosis
- abhängig von dem Planungskonzept
  - z.B. Anfangsdosis ist für ein Medikament vorgegeben, unabhängig vom Patienten

KE gibt Protege an, welchen Typ Werte für Attribute haben.

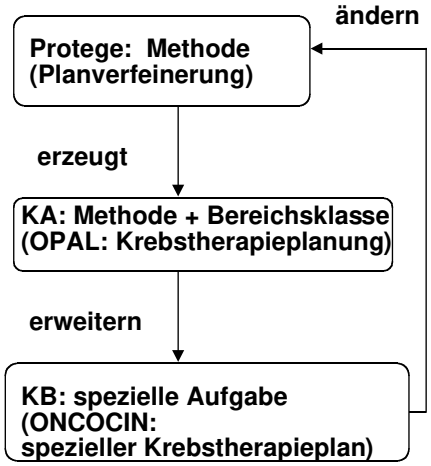
Protege erzeugt eine graphische Benutzeroberfläche (Wissenseditor), in der der Benutzer die tatsächlichen Werte eingeben kann.

Zyklus

Aufbau des Modells für eine Klasse von Anwendungen mit Protege
Protege erzeugt Wissenseditor
- z.B. p_OPAL, HTN

Erweiterung des Modells für eine spezielle Aufgabe mit dem Wissenseditor
Wissenseditor erzeugt Wissensbasis für XPS-Hülle (e-ONCOCIN)
- z.B. p_ONCOCIN, HTNXP

Eingabe eines speziellen Falls in das XPS
Änderungen:
- in der Wissensbasis mit dem Wissenseditor
- im Gesamtablauf mit Protege



... wie ging es mit den repertory grids weiter?

Wissenserhebungstechniken

Laddering (Verallgemeinerung der repertory grids):
- triadische Fragetechnik zur Erhebung von Attributen
- "What is the key difference between ITEM1 and ITEM2, ITEM3?"
- hierarchische Struktur des Interviews, top-down
- "Can you provide EXAMPLES of CLASS?" (abwärts)
- "What alternative examples of CLASS are there to ITEM1?"

Karten sortieren

Interview

Protokollanalyse

MOLE

Larry Eshelman, Damien Ehret, John McDermott, Ming Tan
Univ. Pittsburgh
MOLE: A tenacious knowledge acquisition tool
Int. J. Man-Machine Studies, Vol 26, 41 - 54

MOLE ist sowohl Performanz- wie auch Wissenserwerbssystem.
Wissenserwerb orientiert sich am Problemlösungsmodell.
Wissenserwerb umfaßt
- Aufbau der initialen Wissensbasis
- Verfeinerung der Wissensbasis

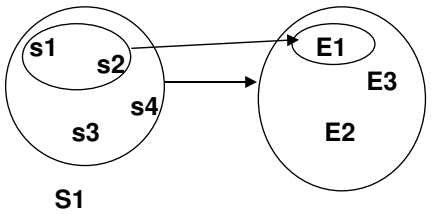
MOLE
das Performanzsystem

<u>cover and differentiate</u>
1) Symptome vom Benutzer erfragen
2) welche Hypothesen decken die Symptome ab (cover)
3) worin unterscheiden sich die Hypothesen (differentiate)
4) über diese Unterschiede (zusätzliche Symptome) den Benutzer befragen
5) ggf. mit neuen Symptomen zu 2)
6) Kombination notwendiger Hypothesen zur Erklärung der Symptome
7) falls einige Kombinationen nur unter bestimmten Umständen möglich sind - Benutzer nach diesen Umständen befragen und zu 2)
8) Ausgabe der Ergebnisse (Hypothesen)

Alle Symptome abdecken! So wenig Hypothesen wie möglich ausgeben!

Wissensarten

Erklärung:
E1 erklärt S1
Erwartung:
wenn nicht S1, dann auch nicht E1
Umstände:
häufiges (bzw. seltenes) gemeinsames Auftreten von E1 und S1
Verfeinerung:
unterscheidende Merkmale für Hypothesen
Modifikation:
Stärken oder Abschwächen einer Hypothese durch bestimmte Bedingungen



Kombination von Hypothesen

Aktive Hypothesen werden
- angenommen
- abgelehnt
- oder sind unentschieden

Gesucht ist die kleinste Menge von Hypothesen, die alle Symptome erklärt.
Es werden die besten Hypothesen angenommen und
alle Hypothesen, die Symptome erklären, die schon von angenommenen Hypothesen abgedeckt werden, abgelehnt.
Aus den unentschiedenen Hypothesen wird die kleinste notwendige Menge von Hypothesen herausgesucht, so daß alle Symptome abgedeckt sind.

Beispiel

**Kohlekraftwerk,
insbesondere Heizkessel**

Symptome:
s1) Verlust von Gas
s2) Asche ist schwarz
Hypothesen:
E11) hohe Gastemperatur
E12) *high excess air*
E21) *high excess air*
E22) *low excess air*
E23) große Brennstoffpartikel

1

1

Erwartungen:
high excess air --> hoher Aschenflug
low excess air --> niedrige Flamme

Alle Hypothesen für ein Symptom haben zusammeneinen Wert.
Wird eine Hypothese herabgesetzt, steigt der Wert der anderen automatisch.

Wenn kein hoher Aschenflug, dann sinkt *high excess air* etwa auf 0.1, aber die Summe der Hypothesenwerte ergibt weiterhin 1.
Wenn die Temperatur der Flamme nicht niedrig ist, bleibt nur noch E23 mit einer Bewertung > .5

E11) hohe Gastemperatur 0.9

E23) große Brennstoffpartikel 0.8

Diese Hypothesen werden als Symptome interpretiert, die erklärt werden.

große Brennstoffpartikel
1) Einstellung des Zerkleinerers verkehrt
2) Zerkleinerer kaputt

Umstände:
Der Mahlstein wurde gerade gewartet.

Also ist vermutlich die Einstellung des Zerkleinerers verkehrt.

MOLE
das Wissenserwerbssystem

Ziel: KE überflüssig machen
Probleme:
Unterbestimmtheit (welche Relationen zwischen Ereignissen?)
Unvollständigkeit
Anforderungen:
Wissenserwerbssystem soll Ambiguitäten selbst auflösen, Unvollständigkeiten und Widersprüche auffinden.
Validierung anhand eines Testfalls, dessen korrekte Lösung vom Experten gegeben wird.
Phasen:
initialer Aufbau
Verfeinerung, Validierung

initialer Aufbau

Ereignisse (Hypothesen, Symptome)
beobachtet, erschlossen
Relationen zwischen ihnen
covering, Erwartungswissen, Umstände, Verfeinerung, Modifikation
numerischer Wert für den Grad der Evidenz

Wie kann man dies erfragen?
Probleme:
Ereignis ist manchmal beobachtbar, manchmal nicht.
Ereignis ist mal ein Symptom, mal eine (Zwischen-)Hypothese.
Beispiel:
Verstopfung eines Rohrs kann als Erklärung für ungleichmäßige Wärmeabgabe dienen.
Verstopfung kann die Hypothese *ungleichmäßige Strahlung* als Erklärung für ungleichmäßige Wärmeabgabe unterstützen.

Probleme

Umstände werden selten vom Experten angegeben.
Beispiel:
l*ow excess air* soll erklärt werden.
1) Luftzufuhr verbraucht und 2) Leck im Rohr sind kausale Erklärungen.
Ventil kaputt unterstützt 1).
Installationsdatum des Rohrs wird nicht als relevanter Umstand für 2) angegeben.

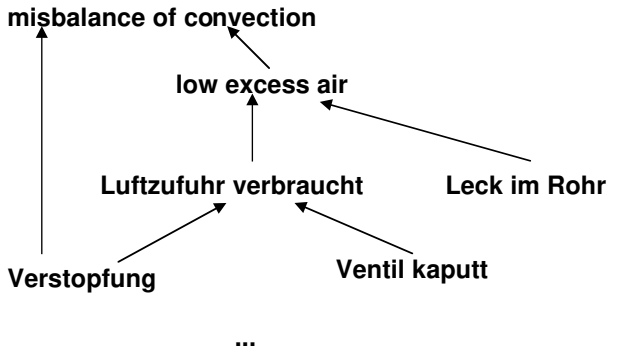
Richtung der Relation ist auch schwierig anzugeben.
Beispiel:
E1 erklärt E2 interpretiert als "E1 unterstützt Hypothese E2"
E1 bewirkt E2 interpretiert als " E2 bringt mich auf Hypothese E2"
E1 erklärt E2, wenn E1 vor E2 auftritt. (Ursache vor Wirkung)

Probleme

Erklärung: Hypothese erklärt Symptom
Erwartung: Symptom liefert Evidenz für Hypothese
XP: "E1 unterstützt E2", "E2 vor E1"
MOLE: E2 erklärt E1.
XP: "E1 unterstützt E2", "E1 vor E2"
MOLE: E1 erklärt E2 (cover),
E2 verstärkt E1 (Erwartung)
Außerdem:
Differenzierung zwischen Hypothesen liefert Erwartungswissen.

Experten können keine numerischen Werte für die Evidenz angeben!
Also verteilt MOLE die Werte gleichmäßig so auf die Hypothesen, daß die Summer 1 ergibt.

statische Analyse



Redundante Pfade finden und Experten befragen, ob kurze Regel Abkürzung für einen Pfad ist oder tatsächliche (kausale) Beziehung.

Verfeinerung

Lösung des Experten ist nicht die des Systems.
Fehlende Differenzierungen zwischen Hypothesen:
Gibt es inaktive Erklärungen, die als Erwartung interpretiert die richtige Lösung ergeben?
Gibt es einen besonderen Umstand in dem speziellen Fall?
Kann das Symptom verfeinert werden?
Gibt es Bedingungen, unter denen eine Relation (Erklärung, Erwartung, Umstand) stärker gewichtet werden muß?
Unter Rückverfolgung des vom System erstellten Lösungspfades und der möglichen Pfade zur Expertenlösung wird der Experte befragt und die Wissensbasis erweitert.

Würde eine Verschiebung der Evidenzstärke zwischen Hypothesen für dasselbe Symptom die richtige Lösung ergeben?
Wenn der Experte kein zusätzliches Wissen angeben kann, werden die Evidenzstärken so verschoben, daß die richtige Lösung herauskommt.

Verbesserung der Erklärungen

Hypothese war notwendig, um ein Symptom zu erklären, hätte aber abgelehnt werden sollen.
Experte wird nach alternativer Erklärung des ansonsten nicht abgedeckten Symptoms gefragt. Die ERklärung wird in die Wissensbasis eingetragen.

Hypothese wurde abgelehnt, da sie nicht notwendig war, um ein Symptom zu erklären, hätte aber angenommen werden sollen.
Experte wird nach einem Symptom gefragt, das von dieser Hypothese erklärt wird und im Testfall gegeben ist.
Beispiel:
Symptom: Gasverlust
abgelehnte Hypothese:hohe Gastemperatur
angenommene Hypothese: *high excess air* - erklärt obendrein hohen Aschenflug
Experte: *high gas stack temperature reading* - wird durch *high excess air* nicht erklärt, aber durch hohe Gastemperatur.
Neues Symptom mit Erklärung wird in die Wissensbasis eingetragen.

Kombination von Hypothesen

Die Strategie, so wenig Hypothesen wie möglich als Lösung auszugeben, ist nicht immer korrekt. In bestimmten Fälle müssen mehrere Hypothesen ausgegeben werden.
Beispiel:
Symptom: geringer Wärmeaustausch
Hypothesen: ungleichmäßige Strahlung, ungleichmäßige convection
Experte: bei *low excess air* müssen beide Hypothesen gewählt werden.

Conclusio

"MOLE plays the role of an experienced knowledge engineerwho is able to work in conjunction with a domain expert and build a diagnostic system, even though the knowledge engineer has little or no knowledge of the domain. "

Grundlage des Wissenserwerbs sind:
Wissensarten (Erklärung, Erwartung, Umstände, Modifikationen) und Erwartungen darüber, wie Experten sie ausdrücken.
Vergleich einer Systemlösung mit der richtigen des Experten und Annahmen über mögliche Ergänzungen, die die Abweichung verhindern würden.

AQUINAS

John Boose, Jeffrey Bradshaw (Boeing KS Lab.):
Expertise Transfer and Complex Problems Using AQUINAS as a Knowledge-Acquisition Workbench fo Knowledge-Based Systems,
in: Boose, Gaines (eds): Knowledge Acquisition Tools for Expert Systems, Academic Press, 1988, 39 - 64

ETS: mehrere Hundert Prototypen, in je 2 Stunden entwickelt

<table>
<tr><td colspan="6">dialog manager</td></tr>
<tr><td>Rep. grids tool (ETS)</td><td>hierarch. structure tool</td><td>uncertainty tool</td><td>inference engine</td><td>induct. tool</td><td>mult. experts</td></tr>
<tr><td colspan="6">object-oriented DBMS</td></tr>
</table>

Vorgehen

1. BEN gibt Objekte an.
2. SYS stellt alle möglichen Tripel von Objekten zusammen. BEN teilt Tripel jeweils in 2 Klassen durch Angabe einer Eigenschaft ein.
3. SYS stellt alle Kombinationen von Objekten und Eigenschaften vor. BEN ordnet jedem Tupel einen Wert zwischen 1 und 5 zu.
4. SYS schließt induktiv von Eigenschaftswerten auf Implikationen von Elgenschaften, so daß sich eine Hierarchie von Elgenschaften ergibt.
5. Durch cluster-Analyse erstellt SYS eine Hierarchie von Objekten.
6. SYS legt Fallbeispiel vor. BEN testet und erweitert.
7. SYS erfragt Begründungen für Eigenschaften von Objekten.
8. Repertory grids werden in Produktionsregeln transformiert.

Hierarchie von Eigenschaften

Die einzelnen repertory grids werden als Beispiele für iduktives Lernen genutzt.
attitude IMPLIES application area .66
4 von 6 Fällen haben denselben Wert bei *attitude* und *application area*
low attitude IMPLIES high availability .66
2 von 3 Fällen
low application area IMPLIES high availability .6
3 von 5 Fällen

BEN bewertet die Implikationen. Bei Ablehnung kann BEN zusätzliche Fälle angeben oder vorhandene Einträge ändern.

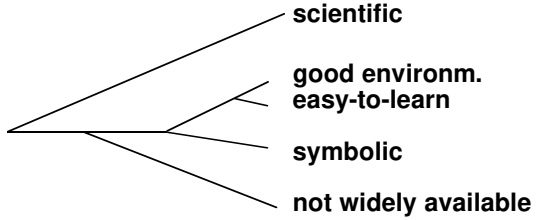
Cluster

Objekte mit denselben Eigenschaften werden zusammengruppiert.

Ideal
LISP
Prolog
ADA

Fortran
COBOL

Merkmale, die häufig zusammen auftreten, werden zusammengruppiert.



Allgemeinere Merkmale

Wat is a new trait that says why you think *good-environment* should be true of a *language for knowledge acquisition*?
: faster system development

Versuch, höheren Knoten der Hierarchien Bedeutung zuzuordnen.

How could a *language for knowledge acquisition* be characterized by *faster system development?*

Definition der höheren Eigenschaften.

Wie baut man ein wissensbasiertes System?

Ungefähr so wie andere Programme auch!

Wasserfallmodell

rapid prototyping

KADS

sloppy modeling

Wasserfallmodell

Plausibilitätsstudie

Konstruktion des Demonstrationsprototypen
Sachbereichsmodell für einenTeilbereich, Entwurf der Interaktion

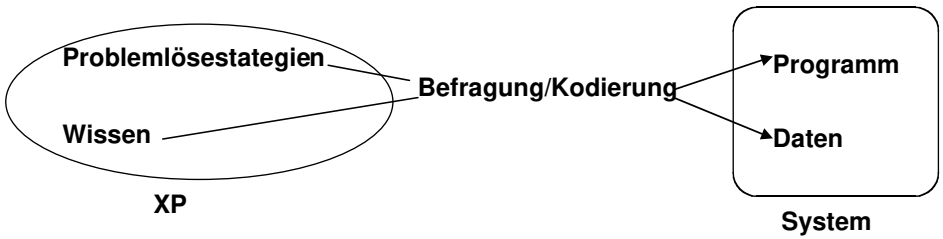
Konstruktion des vollständigen Prototypen
Wissenserhebung, Inferenzmaschine und Benutzerschnittstelle, Füllen der Wissensbasis, Testen

Implementation und Installation des Systems

Wartung und Erweiterung des Systems

Kritik

Hier wird vom Wissens<u>transfer</u> ausgegangen:



Die Rückkopplung zwischen Systementwicklung und Gedanken des XP ist nur rudimentär. Der anfänglich gesteckte Rahmen kann nicht verändert, sondern nur verfeindert werden.

rapid prototyping

Unter Verwendung einer Systemhülle wird aus den Angaben des Experten und einer tentativen Modellierung ein Prototyp erstellt, der dem Experten und den Benutzern vorgestellt wird.

Die Systemhülle enthält bereits Problemlösestrategie und Bausteine für Benutzerschnittstelle.

Eine Reihe verschiedener Modellierungen werden ausprobiert. Das Systemverhalten läßt sich vom Experten/Benutzer leichter evaluieren als Spezifikationen.

Wenn ein Prototyp positiv bewertet wurde, wird analog zum Wasserfallmodell verfahren.

Wissensebene

Allen Newell (1982) The Knowledge Level,
Artificial Intelligence Journal, 18, 87 - 127

Das Verhalten von Computer sollte nicht in Ausdrücken der Implementation und deren Methoden (*symbol level*) beschrieben werden, sondern auf der Wissensebene (*knowledge level*).

Prinzip der Rationalität: Wenn jemand weiß, daß eine der ihm möglichen Handlungen eines seiner Ziele erreichen wird, dann vollzieht er diese Handlung.

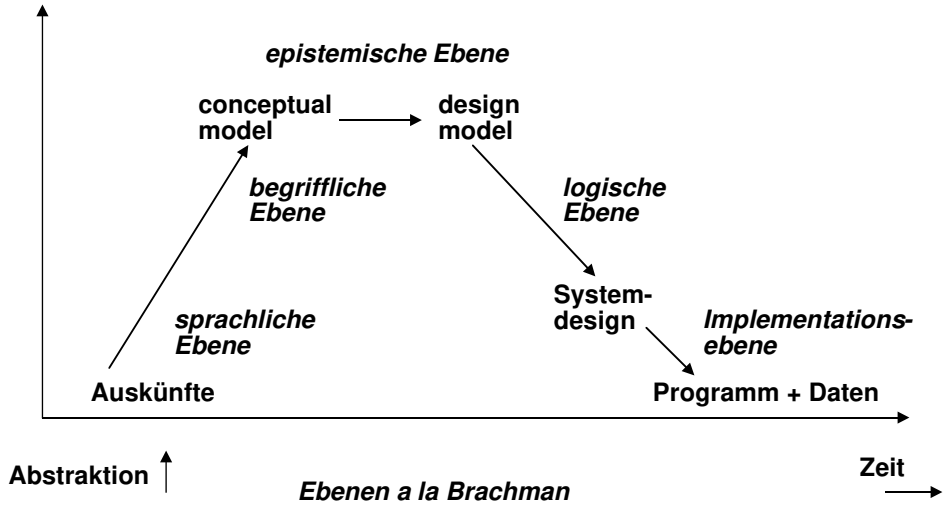
Wissen: Alles, was man jemandem zuschreiben muß, damit sein Verhalten aus dem Prinzip der Rationalität folgt.

Zwei Phasen Vier Ebenen = KADS

Das *conceptual model* ist eine Dokumentation dessen, was die Autfraggeber wollten. Es entsteht interaktiv und in beliebig vielen Iterationen. Es verwendet überwiegend Sachbereichsterminologie.
Es ist aber derart spezifiziert, daß der Transfer zum *design model* recht einfach sein sollte. Das *design model* spezifiziert ein System. Es verwendet Informatikterminologie.
Die Trennung von *conceptual* und *design model* (zwei Phasen) ermöglicht die Beschäftigung unabhängiger Teams:
Psychologen bei der Erstellung des *conceptual model* und Informatiker bei der Erstellung des *design model*.

Die Vorstrukturierung in (vier) Ebenen des Wissens soll das Vorgehen beim Wissenserwerb erleichtern.
Es gibt keine Einflüsse von der Design-Phase zurück in die Erwerbsphase.

Entwicklung eines WBS



Wissenserhebung

Knowledge elicitation

Protokollanalyse
XP protokolliert seinen Gedankengang während seiner Tätigkeit.

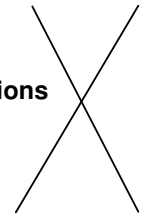
Beobachtung
Interaktion verschiedener, an der Lösung beteiligter Akteure:
teilnehmend: Beobachterin ist Teil der Interaktion
nicht teilnehmend: Beobachterin steht außerhalb

Befragung
Intensivbefragung - freie, vom Befragten gesteuerte Erhebung
strukturiertes Interview - Intensivbefragung mit Leitfaden
Fragebogen - vorstrukturierte, vom Interviewer gesteuerte Erhebung

allgemeine Strukturierung

LaFrance (1987) The Knowledge Acquisition Grid - A Method for Training Knowledge Engineers,
Int. J. Man-Machine Studies, 26, 245 - 256

Fragetypen:
grand tour
cataloging categories
ascertaining attributes
dterminate interconnections
seeking advice
cross-checking



Versionen des Wissens:
layout (grundlegender Plan)
stories (Fälle und Erfahrungen)
scripts (Ablauf der Problemlösung)
metaphors (Rückgriff auf versch. Bereiche)
rules of thumb (Zusammenfassung kausaler Beziehungen)

Fragetypen zur Erhebung von Begriffen

<u>grand tour:</u>
Could you describe the things that experts do? - A does B
Tell me an example! - <story>
<u>cataloging categories</u>:
What are the basic objects in Bgeneral? - B11, B12, B13, B21, B22, B31, B32
Are there different types of As? - yes, A1, A2, A3
<u>ascertaining the attributes</u>:
What distinguishes B11 from B12? - B11 constructs F1, B12 constructs F2.
<u>determining interconnections</u>:
Is B a sub-type of some other kind of Bgeneral? - yes, Bgeneral2
You said that B11 occurs before B12. Why is this the case? - Because B12 requires F1 which is constructed during B11.
<u>cross-checking</u>:
Let me play the devil's advocate. What if you were doing B12 while B11?

Übung

Übung: Interview über Programmierung, Studieren, Autofahren.

KADS

Knowledge Analysis and Documentation System
Knowledge Analysis and Design Support

1982 Annie Brooking (UK), Bob Wielinga (NL)
1983 - 1984 P12 KADS
1984 - 1985 P314 KADS mit Industriepartnern
1985 - 1990 P1098 KADS I
1990 - 1994 P5248 KADS II
Aim: de facto European standard for KBS development

KADS I Buch:
KADS A Principled Approach to Knowledge-Based System Development
Schreiber, Wielinga, Breuker (Hg.)
Academic Press 1993

Modelle in KADS

Organisationsmodell

In welcher Umgebung soll das System eingesetzt werden?
Wie wird das System die Arbeitsabläufe dort verändern?

Anwendungsmodell:

Funktionen und Aufgaben des Systems,
Einflüsse auf die Systemfunktionen

conceptual model

Beschreibung des Systemverhaltens auf vier verschiedenen Ebenen:
Sachbereichswissen, Inferenzstruktur, Aufgaben, Strategie

Die Beschreibung soll noch dicht an der Sachbereichsterminologie sein.

design model
Spezifikation des zu implementierenden Systems in Informatikterminologie.

Wissensarten

Deklarative Sachbereichstheorie (conceptualization of a domain)
Inferenzstruktur (Kontrollwissen)
Aufgabenstruktur (Kontrollwissen)
strategisches Wissen (Kontrollwissen)

Sachbereichstheorie

Conceptualization --> Genesereth, Nilsson 1987
epistemische Primitive --> Brachman, Schmolze 1985

Begriff
Merkmale/Werte
Relationen zwischen Begriffen
Relationen zwischen Merkmalswerten von Begriffen
Struktur zusammengesetztes Objekt

Man kann auch anderen Repräsentationsformalismus wählen.

ergibt die Signatur des Sachbereichsmodells

Sachbereichsschema

component — Begriff

IS-A component, component

SUB- COMPONENT-OF component, component

component: state-value

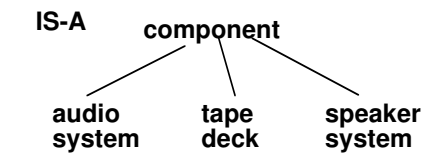
CAUSES component:state-value, component:state-value

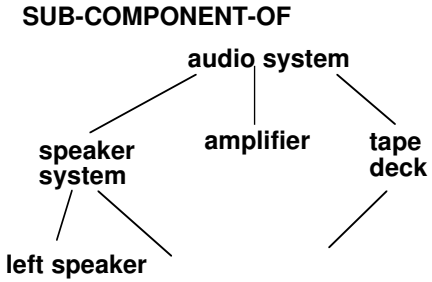
test

test:value — Begriff

INDICATES test:value, component:state-value

Sachbereichswissen





deck:function {stop, play, rew, ff, pause}
deck:power {on, off}
amplifier:input {deck, tuner, CD, ...}

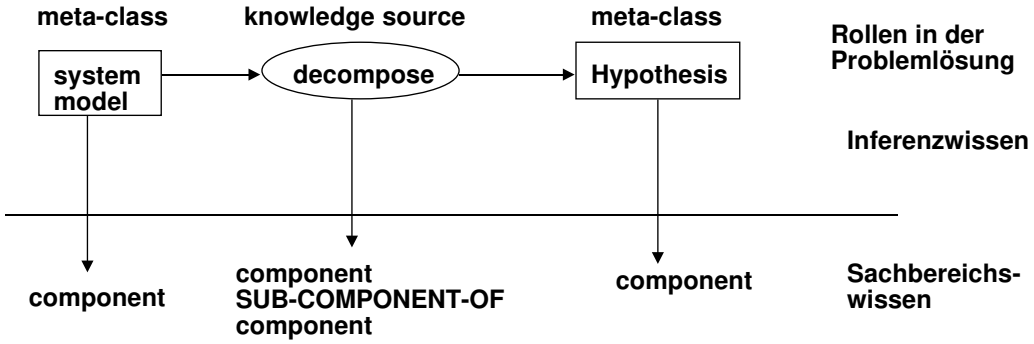
test:deck-p-switch {pressed, not-pressed}
test:input-selector {deck, tuner, CD, ...}

deck:power = on &
deck:function = play &
cable-connection:deck-amplifier = present
CAUSES
amplifier:input = deck

test:deck-p-switch= pressed
INDICATES
deck:power = on

Inferenzen

inference layer

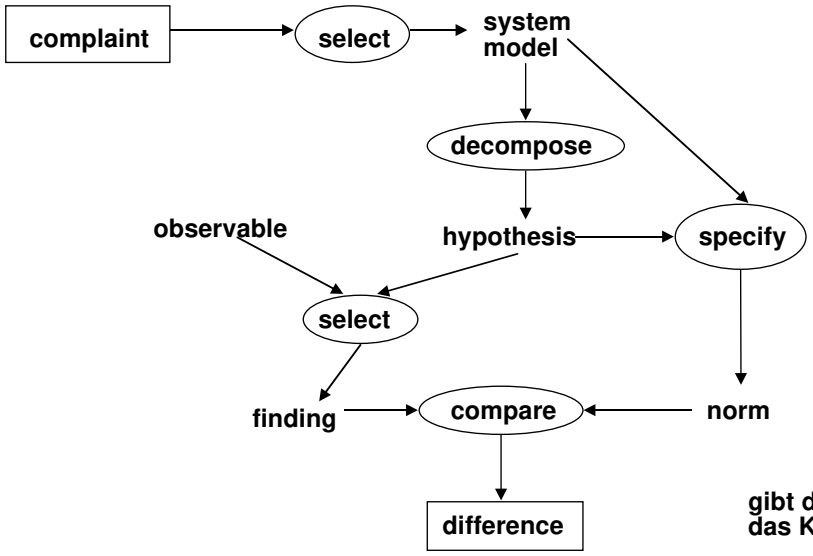


Rollen in der Problemlösung

Inferenzwissen

Sachbereichs-wissen

Inferenzstruktur



gibt das Vokabular für das Kontrollwissen an

Aufgaben

Aufgabe: Erreichen eines Ziels über mehrere Teilaufgaben
Kontrollausdrücke: Name für eine Menge von Elementen einer Metaklasse
Aufgabenstruktur:
einfache Aufgaben entsprechen Inferenzen
Aufgaben werden in Teilaufgaben zerlegt, möglicherweise rekursiv
Transfer-Aufgaben erfordern Interaktion mit der Außenwelt (Benutzer)
- **obtain: System fragt nach einer Information**
- **present: System gibt eine Information ab**
- **receive: System erhält eine Information (unaufgefordert)**
- **provide: System beantwortet eine Frage**

explizieren die Aufgaben a la Clancey

Aufgabe

task: systematic-diagnosis
goal: find smallest component with inconsistent behavior, if one.
input: complaint
output: inconsistent-sub-system: sub-part of the system with inconsistent behavior
control-terms: differential: set of currently active hypotheses
task-structure: systematic-diagnosis (complaint -> inconsistent-sub-system)=
- ***select* (complaint -> system-model) ; select is a knowledge source**
- **generate-hypotheses (system-model -> differential)**
- **REPEAT**
- **test-hypotheses (differential -> inconsistent-sub-system)**
- **generate-hypotheses (inconsistent-sub-system -> differential)**
- **UNTIL differential = 0**

test-hypotheses und generate-hypotheses sind Aufgaben, die entsprechend definiert werden.

Strategie

Was tun, wenn Widersprüche auftreten?
Was tun, wenn keine Lösung erstellt werden kann?
Kann die Aufgabe auch anders beschrieben werden?
Gibt es einen allgemeinen Plan für unvorhersehbare Fälle?

wurde in KADS I nicht gebraucht.

Interpretationsmodelle

... sind Modelle, deren Sachbereichswissen leer ist.
Sie bestehen also nur aus
- **Strategie**
- **Aufgaben und**
- **Inferenzstruktur.**

Sie sollen einem KE als Vorstrukturierung dienen.

Bezug zu rapid prototyping!

Übung

Skizzieren Sie
Anwendungs-, Organisationsmodell und *conceptual model* für
ein System, das einem Benutzer ein Hotelzimmer vermittelt.

Cover-and-Differentiate in KADS

Beschreibung von MOLE auf der Wissensebene
Verwendung der Spezifikationssprache ML2 (van Harmelen)

Sachbereichsschema:
Begriffe:
states Knoten in einem kausalen Netzwerk
qualifiers Beobachtungen, die keine Symptome sind

Relationen:
cause Erklärungsrelation zwischen *states*
manifestation Erwartungen und Modifikationen können eine Erklärung
mehr oder weniger plausibel machen

Sachbereichstheorie

```
theory causes
axioms
(coronary-artery-obliteration (70-percent) -->
mycardial-ischaemia (present) )
(mycardial-ischaemia (present) --> angina-pectoris (present) )
...
theory manifestations
axioms
blood-cpk (high) --> mycardial-necrosis(present)
physical-state (stress) --> (coronary-artery-obliteration (70-percent) -->
    myocardial-ischaemia (present))
```

Lifting definition

Sachbereichstheorie --> Metatheorie

```
lift-definition domain-schemata
from causes, manifestations to cover-theory, anticipate-theory, prefer-theory
functions:
    cover: state, state --> ...
    ...
mapping:
    lift (causes, P1 --> P2) |--> cover-relation( P1:state, P2:state);
    lift (causes, P1 --> P2) |--> anticipate-relation( P1:state, P2:state);
```

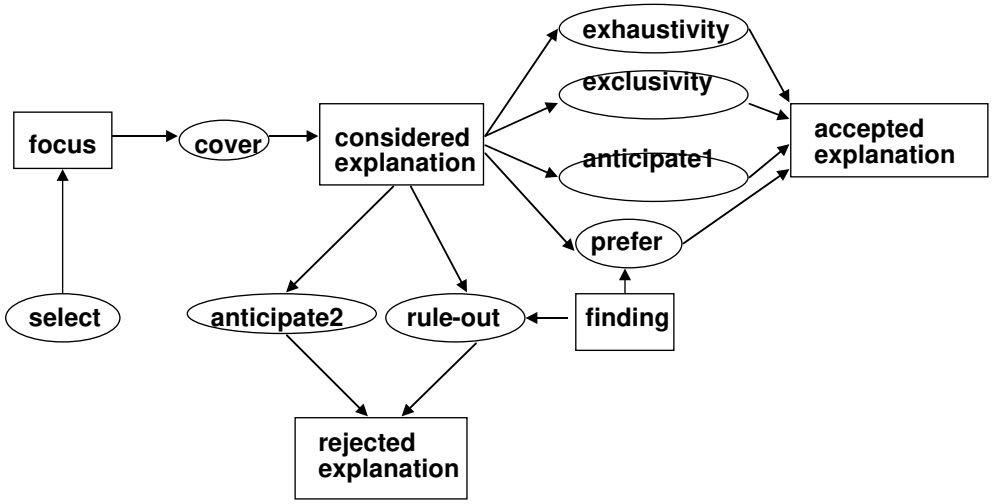
Inferenzen

Metaklassen:
Focus: nicht durch einen anderen Zustand erklärter Zustand
Finding: beobachteter Zustand oder Beobachtung
Explanation: angenommene, abgelehnte, betrachtete Erklärungen

theory cover
use domain-schemata
import role-defs (Deklaration der Metaklassen)
axioms
focus(S1) & ask (causes, cover-relation (S2, S1)) --> considered-explanation (S2, S1)

***ask* verwendet die *lift definition domain-schemata*, um herauszufinden, ob die *cover-relation* auf ein Axiom der Sachbereichstheorie *causes* anzuwenden ist.**
Alle Zustände, die in der Sachbereichstheorie und im Sachbereichswissen in einer *cause* Relation stehen, werden für den aktuellen Zustand betrachtet.

Inferenzstruktur



Vergleich cover-and-differentiate mit heuristic classification

vorgenommen von Guus Schreiber, Bob Wielinga und Hans Akkermans im KADS Buch, Kapitel 18

1. **Bei HC wird von den Symptomen in Symptomklassen abstrahiert. Etwas entsprechendes fehlt bei c&d.**
   ??? Vielleicht nur die Zwischenknoten falsch beschrieben?
2. **Bei c&d werden ausgehend von einer Beobachtung nur solche Hypothesen angestoßen, die direkt mit der Beobachtung verbunden sind. Bei HC wird eine Hypothese von mehrern Symptomen/Klassen angestoßen, aus irgendeiner Hierarchieebene heraus.**
3. **Das Ablehnen von Hypothesen bezieht sich bei c&d auf einzelne Hypothesen, bei HC auf die Konkurrenz zwischen Hypothesen.**
   ??? Das numerische Gewichten zwischen Hypothesen bei c&d fehlt in der KADS Beschreibung!

FRAUDWATCH

Expertensystem der Barclays Bank zur Überwachung von Scheckkarten, um Betrug aufzudecken -
von Touche Ross Management Consultants, London, mithilfe der KADS Methodik entwickeltes und gewartetes System

Weltweit hat der Scheckkartenbetrug ein Volumen von 1 Billion Dollars pro Jahr.
In GB 1991 waren es 150 Mio. Pfund.
Ziel: frühzeitig Erkennen, daß die Scheckkarte in falschen Händen ist und sperren.
70% des Mißbrauchs geschehen vor der Sperrung der Karte.
Statistische Verfahren konnten 300 falsche Transaktionen richtig erkennen, aber sie klassifizierten auch 100 000 richtige als falsch. Ausrichtung der Methode auf die korrekte Klassifikation fast aller falscher Transaktionen mußte genau umgekehrt werden: möglichst korrekt richtige Transaktionen erkennen!
Wissensbasis soll aus den Fällen eine handhabbare Menge von verdächtigen Transaktionen heraussuchen, die meisten Transaktionen belassen.

Nutzen des Systems

Wissensbasis wird über Nacht auf alle Transaktionen des Tages (ca. 2 Mio.) angewandt und liefert Liste verdächtiger Transaktionen (≤ 500) in 3 Stunden.
25% des Mißbrauchs (in $) konnte aufgedeckt werden:
von durchschnittlich 300 verdächtigen Konten sind 15 (20:1) tatsächlich betrügerisch benutzt worden
mit einem Betrag von durchschnittlich 600 Pfund.
Ersparter Betrag: 1,5 Mio. Pfund im Jahr.

Aufwand der Systementwicklung

4 Arbeitsjahre:
5 Arbeitsmonate Programmierung
9 Arbeitsmonate KADS Systementwurf
6 Arbeitsmonate Entwurf des Umgebungssystems
24 Arbeitsmonate Wissenserhebung, Analyse, Verbesserung der Wissensbasis
4 Arbeitsmonate Arbeitstreffen zur Integration des Systems

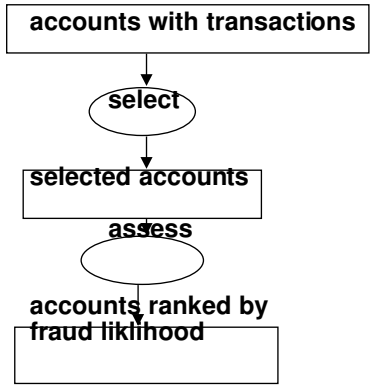
Vorgehen

3 Interviews, jedes 40 Minuten und auf Video- und Tonkassette aufgenommen
Analyse der Protokolle in einer Gruppensitzung, um Inferenzstruktur festzulegen:
Kombination datengesteuerten und modellgesteuerten Vorgehens

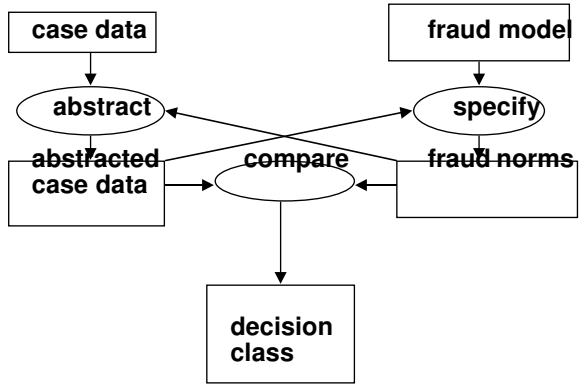
Kriterien für die Auswahl verdächtiger Konten:
UND Transaktion hat stattgefunden
ODER Anzahl der Transaktionen in den letzten 4 Tagen überschreitet Schwellwert
Bargeld- oder Scheckauszahlung überschreitet Schwellwert
Wert der Transaktionen des Monates überschreitet Schwellwert
Karte wurde erst kürzlich gültig

95% der Konten werden so aussortiert.
COBOL-Programm für die Auswahl.

Inference structure



overall procedure



assess

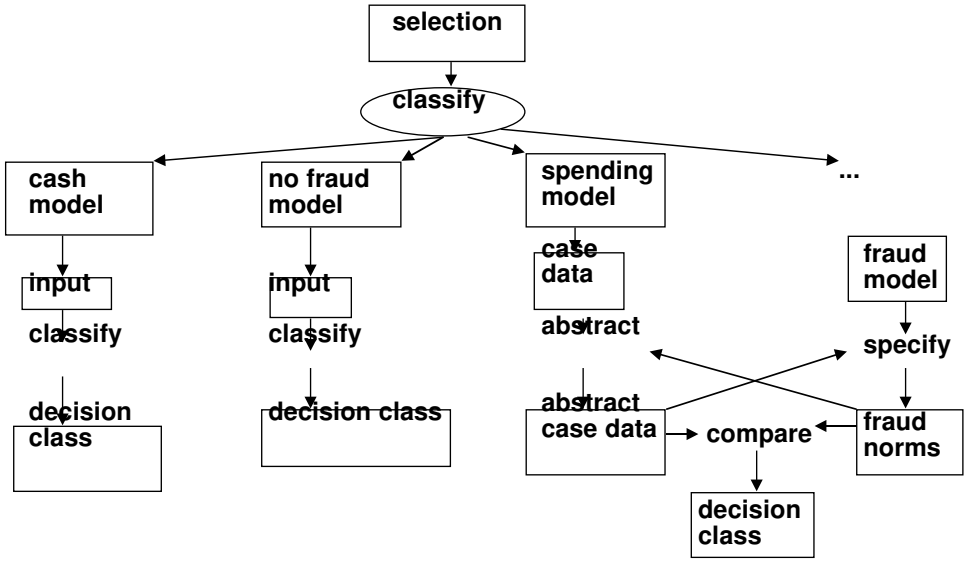
Bewertung

Daten einer Transaktion mit Verdächtigungsregeln abgleichen:
wenn ≥ 0.7 Verdächtigungspunkte vergeben wurden, wird das betreffende Konto als verdächtig ausgegeben.

a) jedes Konto gegen alle Regeln abgleichen,
positive Punkte - negative Punkte = Ergebnis dauerte zu lang

b) Betrugtypen mit positiven und negativen Punkten:
cash model, spending model, ...
Anzahl von Abhebungen in den letzten n Tagen,
typische hohe Überweisung, die keinen Betrug darstellt (Reisebüro)
Personentyp (z.B. jeden Tag viele kleine Transaktionen oder eine hohe Abhebung am Monatsanfang)

Inferenzstruktur



Tatsächlich?

Ist dies tatsächlich KADS?
Ist es eine Inferenzstruktur mit Metaklassen?
Oder ein Flußdiagramm (process breakdown)?
Wie wurden denn vorgegebene Inferenzstrukutren (C&D, HC,...) tatsächlich genutzt?
Wurde nur das Wissen über die Problemlösungsstrategien genutzt?
Muß dieses Wissen KADS zugeschrieben werden?

Analyse und Wartung

350 vom System als verdächtig klassifizierte Fälle und
die im selben Zeitraum betrügersich genutzten Konten, die nicht entdeckt wurden
wurden zur Verbesserung der Modelle *cash, spending, mixed* genutzt.
In einem Modell gibt es Begriffe wie
large cash, cash through ATM, first cash
Außerdem gibt es Kontotypen wie
schlafend, CVR (card valid recently)
Feststellen, wo der meiste Betrug vorkommt und folglich die Regeln verbessert werden müssen:

Kontotyp	F/G				
CVR					
dormant					
new account					
type change	first	high risk	ATM cash	no fraud	high volume

Betrugstyp

Conclusio von Killan im KADS-Buch S. 309

It is important to realize, of course, that this (KADS) help is on a purely conceptual level - the primacy of the CVR norm over any other norm is a fiction for the system and for the programming language used to implement these models. Being able to conduct "Gedanken" experiments, whether in the mind or on the computer, that are given structure by a KADS inference structure, allows for novel knowledge items to be synthesized from current data, which is frequently what FRAUDWATCH knowledge maintenance is about.