



8050

Wissenbasierte Systeme

Dirk Krechel

SS 2009



Überblick über das Modul

1. Einführung
2. Grundtechniken
 - ◆ Case Based Reasoning
 - ◆ Logik als Grundlage der Wissensrepräsentation
 - ◆ Klassische Wissensrepräsentation & -akquisition
3. Spezielle Aufgabenklassen & Systeme
 - ◆ Systeme für Planungsaufgaben
 - ◆ Systeme für Konfigurationsaufgaben
 - ◆ Systeme für Diagnoseaufgaben
 - ◆ Semantic Web



2.1 Case Based Reasoning

2.1.1 Einführung

2.1.2 Fallrepräsentation

2.1.3 Ähnlichkeitbestimmung

2.1.4 Retrieve: Effiziente Fallauswahl

2.1.5 Reuse: Lösungsanpassung

2.1.6 Retain: Lernen

2.1.1 Einführung

Überblick

- Was ist fallbasiertes Schließen ?
- Ein erstes Beispiel
- Geschichte des fallbasierten Schließens
- Verwandte Gebiete
- Motivationen für fallbasiertes Schließen
- Zyklus des fallbasierten Schließens
- Positionierung von fallbasierten Systemen

Literatur

David Leake (1996). *CBR in Context: The Present and Future*. In: David Leake (Ed.), *Case-Based Reasoning, Experiences, Lessons & Future Directions*, AAAI Press, 1996.

Ralph Bergmann, et al. *Developing Industrial Case-Based Reasoning Applications*, Kap. 1.

Agnar Aamodt & Enric Plaza (1994). *Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches*. *AI Communications*, Vol 7, Nr. 1.

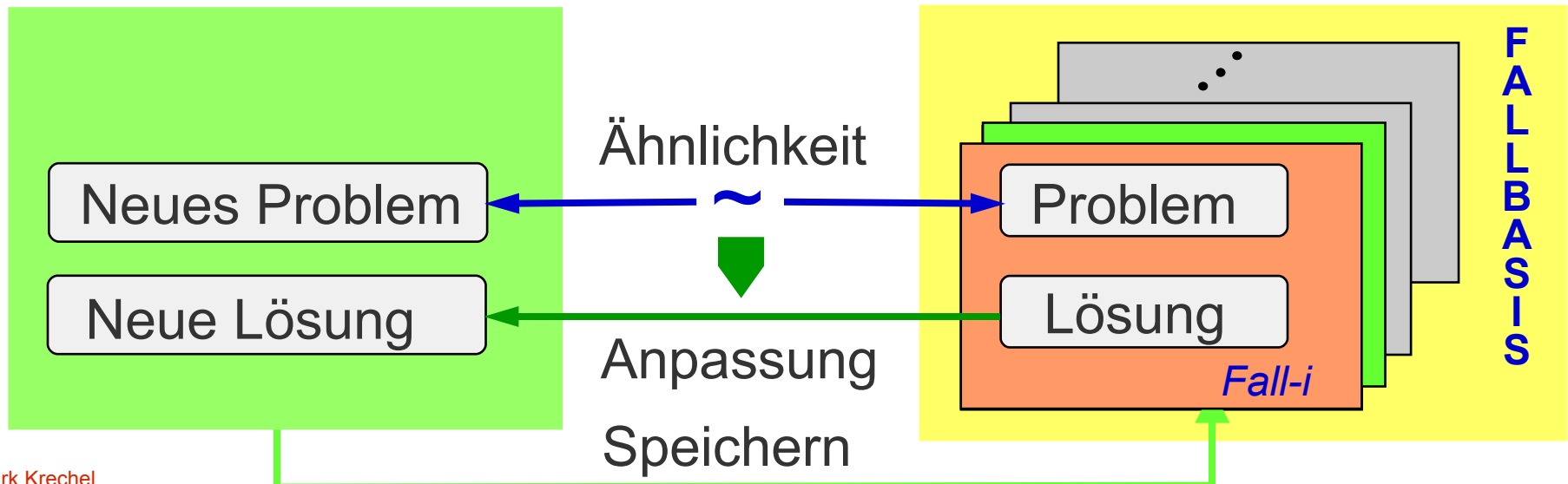
Fallbasiertes Schließen ist ...

- ein kognitiver Ansatz zur Modellierung des menschlichen Problemlösens
- ein Ansatz zur Entwicklung von intelligenten Systemen.
- Zentrale Grundidee:
 - ◆ gemachte Erfahrungen (Fälle) werden gespeichert
 - ◆ Zum Lösen einer neuen Aufgabe werden:
 - ähnliche Erfahrungen aus dem Speicher abgerufen
 - die Erfahrung im Kontext der neuen Situation (ganz, teilweise, oder modifiziert) wiederverwendet.
 - hierbei neu gewonnene Erfahrungen werden wieder gespeichert
- Grundannahme: Ähnliche Probleme haben ähnliche Lösungen



Sicht der Informatik: Uns interessiert primär die Entwicklung von intelligenten Systemen

- Erfahrungswissen wird gespeichert (Fallbasis)
- Zum Lösen einer neuen Aufgabe:
 - ◆ ähnliche (nützliche) Fälle aus dem Speicher abrufen (Retrieval)
 - ◆ Lösung auf die neue Situation anpassen (Anpassung, Reuse)
 - ◆ hierbei neu gewonnene Erfahrung speichern (Lernen, Retain)



Menschen setzen Fälle beim Problemlösen ein ...



- Ein Arzt erinnert sich an die *Krankengeschichte* eines anderen Patienten.
- Ein Jurist argumentiert mit einem ähnlichen *Präzedenzfall*.
- Ein Architekt studiert die *Konstruktionszeichnung* eines bereits existierenden Gebäudes.
- Ein Arbeitsplaner erinnert sich bei der Planung der Fertigung eines Werkstückes an einen *Fertigungsplan* eines vergleichbaren Werkstückes (Variantenplanung).
- Ein Mathematiker versucht einen bekannten *Beweis* auf ein neues Problem zu übertragen.
- Ein Verkäufer schildert den erfolgreichen Einsatz des *Produktes* bei einem anderen Kunden mit ähnlichen Problemen.
- Ein Servicetechniker erinnert sich an einen ähnlichen *Defekt* bei diesem Gerätetyp.

- Call Center
 - ◆ Mobilfunk, Internet Service Provider, Computerspiele-Hersteller etc.
- Knowledge Management
 - ◆ FAQ-Suche, Dokumentenmanagement etc.
- Medizin
 - ◆ Krankheitsbilder, Diagnosen etc.
- Produktsuche
 - ◆ Last-Minute-Reisen, Katalogsuche etc.
- Rechtsprechung

Ein Fall enthält ...

- mindestens :
 - ◆ Problem
 - ◆ Lösung

- zusätzlich :
 - ◆ Kontext (Rechtfertigungen usw.)
 - ◆ Lösungswege
 - ◆ Pointer auf andere Fälle
 - ◆ Güteinformation
 - ◆ ...

- Technische Diagnose (hier KFZ-Diagnose):
 - ◆ An einem technischen Gerät tritt ein Fehler auf
 - ◆ Wir beobachten Symptome (z.B. Motor startet nicht) und erheben Messwerte (z.B. Batteriespannung = 8.3V)
 - ◆ Ziel: Wir sind daran interessiert,
 - ◆ die Fehlerursache zu finden und
 - ◆ eine Reparaturhilfe zu bekommen

- Fallbasierte Diagnose:
 - ◆ Ein Fall beschreibt die Diagnose eines Fehlers und enthält:
 - ◆ Beschreibung der Symptomatik
 - ◆ Beschreibung des Fehlers
 - ◆ ggf. Beschreibung der Abhilfemaßnahme
 - ◆ Diagnosefälle werden gespeichert
 - ◆ Bei neuem Problem, finde Fall mit ähnlicher Symptomatik

Einfaches Beispiel (2)

Wie sieht ein Fall aus?



- Ein Fall beschreibt eine spezielle Diagnosesituation
- Ein Fall erfasst mehrere Attribute und ihre spezifischen Ausprägungen in der speziellen Situation

⇒ Ein Fall ist keine Regel !!

	Attribut	Wert
F A L L 1	Problem (Symptome) <ul style="list-style-type: none">• <i>Problem</i>: Fernlicht funktioniert nicht• <i>Typ</i>: VW Golf IV, 1.6 l• <i>Baujahr</i>: 1998• <i>Batteriespannung</i>: 13,6 V• <i>Zustand der Scheinwerfer</i>: OK• <i>Zustand des Schalters</i>: OK	
	Lösung <ul style="list-style-type: none">• <i>Diagnose</i>: Fernlichtsicherung defekt• <i>Reparatur</i>: Fernlichtsicherung ersetzen	

Einfaches Beispiel (3)

zwei Fälle in der Fallbasis



- Jeder Fall beschreibt eine spezielle Situation
- Alle Fälle sind unabhängig von einander

F A L L 1	Problem (Symptome) <ul style="list-style-type: none">• <i>Problem:</i> Fernlicht funktioniert nicht• <i>Typ:</i> VW Golf IV, 1.6 I• <i>Baujahr:</i> 1998• <i>Batteriespannung:</i> 13,6 V• <i>Zustand der Scheinwerfer:</i> OK• <i>Zustand des Schalters:</i> OK
	Lösung <ul style="list-style-type: none">• <i>Diagnose:</i> Fernlichtsicherung defekt• <i>Reparatur:</i> Fernlichtsicherung ersetzen

F A L L 2	Problem (Symptome) <ul style="list-style-type: none">• <i>Problem:</i> Fernlicht funktioniert nicht• <i>Typ:</i> Audi A4• <i>Baujahr:</i> 1997• <i>Batteriespannung:</i> 12,9 V• <i>Zustand der Scheinwerfer:</i> beschädigt• <i>Zustand des Schalters:</i> OK
	Lösung <ul style="list-style-type: none">• <i>Diagnose:</i> Glühbirne defekt• <i>Reparatur:</i> Glühbirne ersetzen

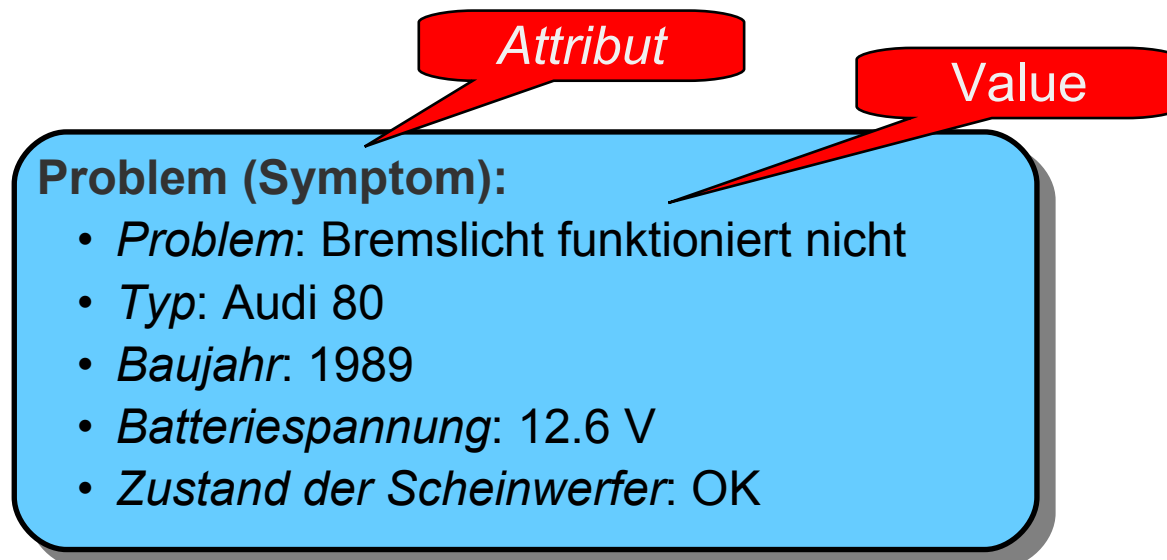
Einfaches Beispiel (4)

Lösung eines neuen Problems



- Ein neues Problem soll gelöst werden
- Wir machen einige Beobachtungen in der neuen Situation
- Beobachtungen definieren das neue Problem
- Nicht alle Merkmale müssen bekannt sein.

Bemerkung: Es ergibt sich ein neuer Fall ohne Lösungskomponente



Einfaches Beispiel (5)



- Wann sind zwei Fälle ähnlich
⇒ **Ähnlichkeit ist ein sehr wichtiges Konzept beim fallbasierten Schließen**
- Wir können die Ähnlichkeit auf der Grundlage der Ähnlichkeit der Attribute beurteilen.
- Ähnlichkeit der Attribute wird über Ähnlichkeit der Werte festgelegt
- Aber: Die Wichtigkeit der Attribute kann unterschiedlich sein.

Einfaches Beispiel (6)

Ähnlichkeitsermittlung

- Ermittlung von Attributähnlichkeiten.
- Grad der Ähnlichkeit wird durch eine reelle Zahl zwischen 0 and 1 ausgedrückt
- Examples:

nicht ähnlich

sehr ähnlich

- ◆ Attribut: *Problem*

Fernlicht funktioniert nicht	← 0.8 →	Bremslicht funktioniert nicht
Fernlicht funktioniert nicht	← 0.4 →	Motor springt nicht an

- ◆ Attribut: *Batteriespannung* (Ähnlichkeit basiert auf Differenz)

12.6 V	← 0.9 →	13.6 V
12.6 V	← 0.1 →	6.7 V

- Attribute haben unterschiedliche Wichtigkeit (Gewichtung)!
- ◆ Hohe Wichtigkeit: Problem, Batteriespannung, Zustand der Scheinwerfer, ...
- ◆ Geringe Wichtigkeit: Typ, Baujahr, ...

Einfaches Beispiel (7)

Ähnlichkeitsvergleich mit Fall 1



Problem (Symptom):

- *Problem:* Bremslicht funktioniert nicht
- *Typ:* Audi 80
- *Baujahr:* 1989
- *Batteriespannung:* 12.6 V
- *Zustand der Scheinwerfer:* OK

Problem (Symptome)

- *Problem:* Fernlicht funktioniert nicht
- *Typ:* VW Golf IV, 1.6 l
- *Baujahr:* 1998
- *Batteriespannung:* 13,6 V
- *Zustand der Scheinwerfer:* OK
- *Zustand des Schalters:* OK

0.8

0.4

0.6

0.9

1.0

Hohe Wichtigkeit: Gewicht = 6



Geringe Wichtigkeit: Gewicht = 1



Lösung

- *Diagnose:* Fernlichtsicherung defekt
- *Reparatur:* Fernlichtsicherung ersetzen

- Ähnlichkeit als gewichteter Mittelwert

$$\text{sim}(p_{\text{neu}}, \text{fall}_1) = 1/20 * [6*0.8 + 1*0.4 + 1*0.6 + 6*0.9 + 6* 1.0] = \mathbf{0.86}$$

Einfaches Beispiel (8)

Ähnlichkeitsvergleich mit Fall 2



Problem (Symptom):

- *Problem*: Bremslicht funktioniert nicht
- *Typ*: Audi 80
- *Baujahr*: 1989
- *Batteriespannung*: 12.6 V
- *Zustand der Scheinwerfer*: OK

Problem (Symptome)

- *Problem*: Fernlicht funktioniert nicht
- *Typ*: Audi A4
- *Baujahr*: 1997
- *Batteriespannung*: 12,9 V
- *Zustand der Scheinwerfer*: beschädigt
- *Zustand des Schalters*: OK

0.8

0.8

0.4

0.95

0

Hohe Wichtigkeit: Gewicht = 6



Geringe Wichtigkeit: Gewicht = 1



Lösung

- *Diagnose*: Glühbirne defekt
- *Reparatur*: Glühbirne ersetzen

- Ähnlichkeit als gewichteter Mittelwert

$$\text{sim}(p_{\text{neu}}, \text{fall}_2) = 1/20 * [6*0.8 + 1*0.8 + 1*0.4 + 6*0.95 + 6*0] = \mathbf{0.585}$$

Fall 1 ist ähnlicher wegen des Merkmals "Zustand der Scheinwerfer"

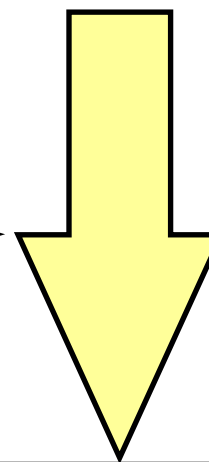
Einfaches Beispiel (9)

Anpassung von Fall 1

F A L L 1	Problem (Symptome): <ul style="list-style-type: none">• <i>Problem:</i> Fernlich funktioniert nicht• ...
	Lösung <ul style="list-style-type: none">• <i>Diagnose:</i> Fernlichtsicherung defekt• <i>Reparatur:</i> Fernlichtsicherung ersetzen

Problem (Symptom):

- *Problem:* **Bremslicht** funktioniert nicht
- *Typ:* Audi 80
- *Baujahr:* 1989
- *Batteriespannung:* 12.6 V
- *Zustand der Scheinwerfer:* OK



Lösungsanpassung:

Wie wirken sich die Änderungen auf die Lösung aus?

Neue Lösung:

- *Diagnose:* **Bremslichtsicherung** defekt
- *Reparatur:* **Bremslichtsicherung** ersetzen

Einfaches Beispiel (10)

Neue Erfahrung speichern



Wenn die Diagnose bestätigt wurde:
speichere neuen Fall in der Fallbasis

F A L L 3	Problem (Symptome) <ul style="list-style-type: none">• <i>Problem:</i> Bremslicht funktioniert nicht• <i>Typ:</i> VW Golf IV, 1.6 I• <i>Baujahr:</i> 1998• <i>Batteriespannung:</i> 13,6 V• <i>Zustand der Scheinwerfer:</i> OK• <i>Zustand des Schalters:</i> OK
	Lösung <ul style="list-style-type: none">• <i>Diagnose:</i> Bremslichtsicherung defekt• <i>Reparatur:</i> Bremslichtsicherung ersetzen

CBR R4-Zyklus



- Formen des Wissens:
 - ◆ Vokabular
 - ◆ Fallbasis
 - ◆ Ähnlichkeitsmaß
 - ◆ Anpassungen

- Retrieve:

Auswahl ähnlicher Fälle

- Reuse:

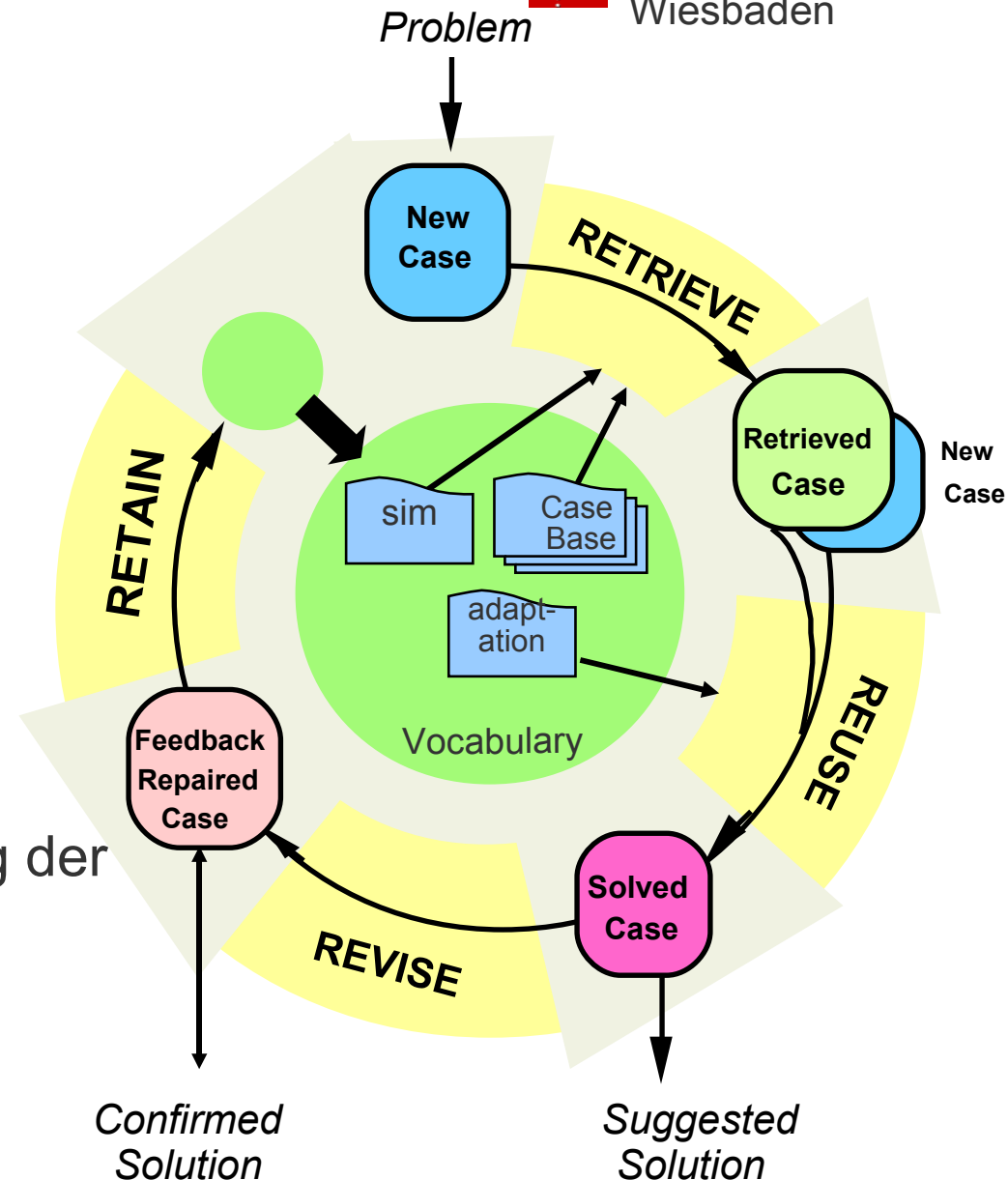
Wiederverwendung, d.h. Problemlösung unter Ausnutzung der Erfahrung

- Revise:

Überprüfung der Lösung

- Retain:

Lernen aus der Problemlösung



in Anlehnung an Aamodt & Plaza (1994)

Retrieve



- finde einen ähnlichen Fall bzw. finde eine Menge ähnlicher Fälle -

- Ähnlichkeit
 - ◆ Maße (z.B. Abstand nach Euklid, Tversky Kontrastmodell)
 - ◆ Erklärungs-basierte Verfahren
 - ◆ Abstraktionsebenen
- Organisation der Fallbasis
 - ◆ Lineare Liste
 - ◆ Indexstrukturen
 - ◆ Diskriminationsnetze
 - ◆ Dynamische Datenbankabfrage
- Speicherung der Fallbasis
 - ◆ Hauptspeicher
 - ◆ Dateien
 - ◆ Datenbanken

- Lösungsübertragung, Adaption -

- Keine Adaption, Kopieren der Lösung
- Manuelle/interaktive Lösungsanpassung durch den Benutzer
 - ◆ Bereitstellung geeigneter Editoren
- Vollautomatische Lösungsanpassung
 - ◆ Transformational Analogy: Transformation der Lösung
 - Regeln oder Operatoren passen die Lösung in Bezug auf die Differenz der Problemstellung an
 - Zusätzliches Wissen erforderlich über den Einfluss von Problemunterschieden
 - ◆ Derivational Analogy: Wiedergabe und Anpassung des Lösungsweges
 - Erfordert generativen Problemlöser, der grundsätzlich ohne Fälle auskommt
 - Hierfür Problemlösungswissen erforderlich
 - ◆ Kompositionelle Adaption: Konstruktion einer Lösung aus Teillösungen mehrerer unterschiedlicher Fälle.

- Überprüfung/Verbesserung der Lösung -

■ Umfang der Überprüfung

- ◆ keine Verbesserung
- ◆ Überprüfung der Lösung durch Simulation
- ◆ Überprüfung/Bewertung der Lösung in der realen Welt/Umgebung

■ Revisionskriterien

- ◆ Korrektheit der Lösung
- ◆ Qualität der Lösung
- ◆ Andere (z.B. Benutzerpräferenzen)

- **Bewahrung der gemachten Erfahrung** -

■ Was wird gelernt

- ◆ Neue Erfahrung (neuer Fall)
- ◆ Ähnlichkeitsbewertung, Bedeutung von einzelnen Merkmalen
- ◆ Organisation der Fallbasis (Effizienz)
- ◆ Lösungsanpassung
- ◆ Vergessen (Löschen) von Fällen, aus Effizienzgründen oder wegen mangelnder Aktualität

■ Methoden

- ◆ Auswendiglernen (speichern neuer Fälle)
- ◆ Kompetenzbewertung von Fällen
- ◆ Konnektionistische Verfahren (ähnlich Neuronaler Netze)
- ◆ Genetische Algorithmen
- ◆ Symbolische induktive Lernverfahren

- Wissen ist in verschiedenen Formen enthalten (*Wissenscontainer*):
 - ◆ Wissensrepräsentation (z.B. verwendete Merkmale)
 - ◆ Fallbasis
 - ◆ Ähnlichkeitsbeurteilung (Maß)
 - ◆ Adaption

- Vorteile von CBR
 - ◆ hohe Flexibilität: vorhandenes Wissen kann auf die Container aufgeteilt werden
 - ◆ Schwerpunkt liegt auf Fallwissen
 - ◆ Wissen in Fallbasis, Ähnlichkeitsbeurteilung und Adaptation kann gelernt werden

- U.S.A. -

Roger Schank, Yale University: Kognitionswissenschaftliche Arbeiten

1977: Scripts als Wissensrepräsentation (Schank, Abelson)

1983: Dynamic Memory Theory, Memory Organization
Packets CYRUS: Erstes implementiertes CBR-System
(Kolodner)

1983-1988: Weitere Systeme, z.B. : JUDGE, SWALE, CHEF

Bruce Porter, Austin Texas: Arbeiten zum Konzeptlernen

1986-89: System PROTOS (Konzeptrepräsentation durch
Fälle)

Edwina Rissland, U. of Massachusetts: Arbeiten zur Verwendung von Fällen in der Rechtssprechung (seit 1983)

1990-92: Systeme HYPO (Ashley) und CABARET (Skalak)

Jaime Carbonell & Manuela Veloso, Carnegie Mellon U.: Analogie

seit 1992 Prodigy/Analogy: Fallbasierte Planung durch
Analogie

Steigendes Interesse an CBR in USA (viele neue Forschungsgruppen)

seit 1988 regelmäßige DARPA und AAAI Workshops



Geschichte des fallbasierten Schließens

- Europa -

Michael Richter, U. Kaiserslautern: CBR im Kontext von Expertensystemen

1988-1991 Systeme Moltke und Patdex zur techn. Diagnose
seit 1991 Fallbasierte Planung: Systeme Caplan/CbC, PARIS
seit 1992 erste Europäische Projekte: INRECA,
seit 1999 erprobte Methodologie zur Systemwicklung: INRECA-II

Ramon Mantaras, Enric Plaza, IIIA Blanes, Spanien: CBR im Kontext von ML

1990 Fallbasiertes Lernen zur medizinischen Diagnose

Agnar Aamodt, U. Trondheim, Norwegen: CBR und Wissensakquisition

1991 System CREEK: Integration von Fällen und allgemeinem Wissen

Mark Keane, Trinity College, Dublin: Kognitionswissenschaften

seit 1988 Theorie des analogen Schließens

Seit 1991 Starkes Interesse in Europa (viele neue Forschungsgruppen)

1991 Erster Deutscher CBR Workshop (AKCBR, GWCBR)

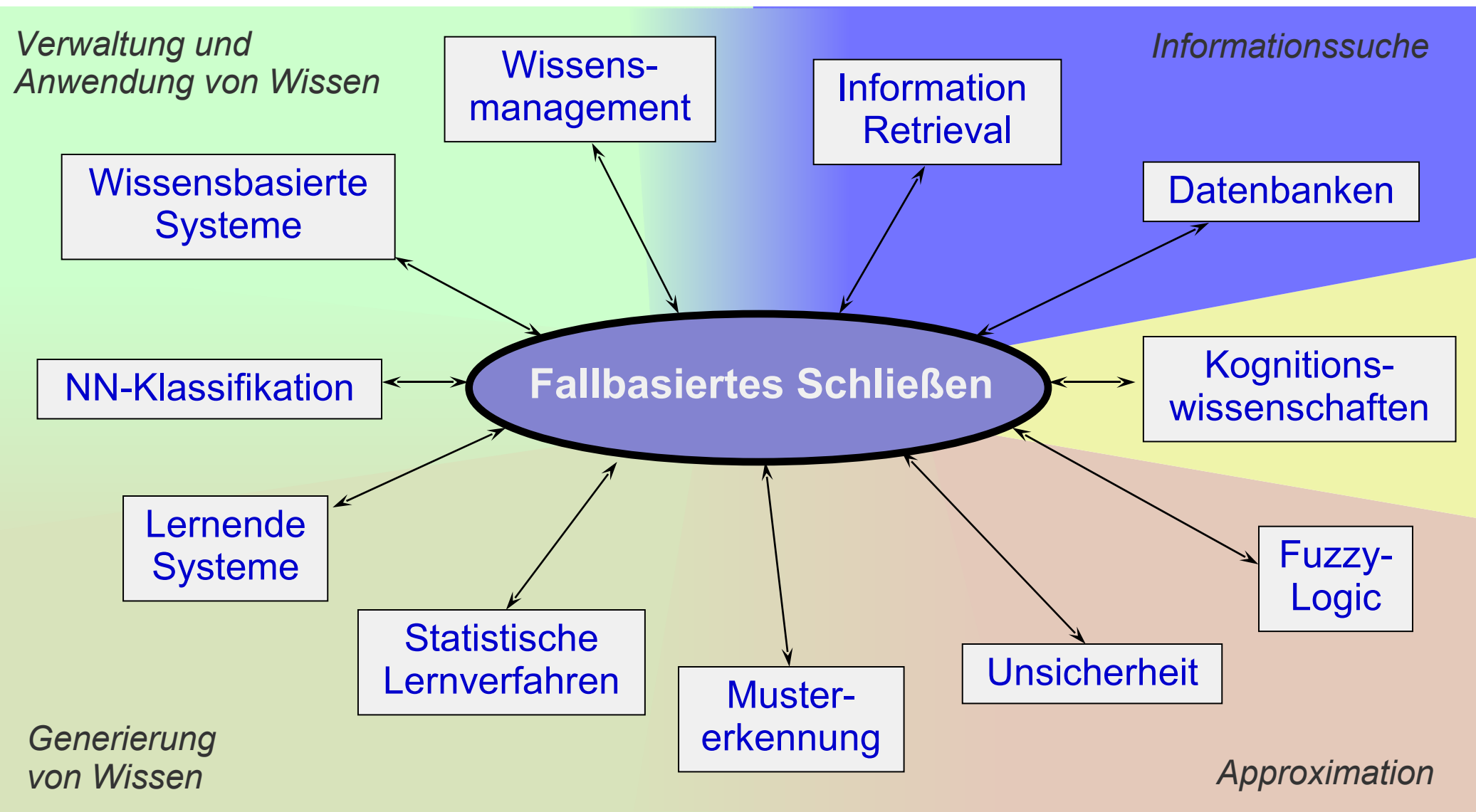
1993 Erster Europäischer CBR Workshop (EWCBR)

1995 Erste Internationale CBR Konferenz (ICCBR)

Fallbasiertes Schließen Heute

- Mehrere kommerzielle Werkzeuge zur Entwicklung von CBR-Anwendungen verfügbar
- erprobte Referenzmodelle zur Entwicklung von fallbasierten Anwendungssystemen vorhanden
- Vielzahl von Anwendungen im täglichen Einsatz
z.B. in den Bereichen
 - ◆ Kundenberatung im Electronic Commerce
 - ◆ Wissens- und Erfahrungsmanagement
 - ◆ Technische Diagnose
 - ◆ Help-Desk-Support
 - ◆ Designunterstützung in der Produktion (Prototypen)
- Regelmäßige wissenschaftliche Veranstaltungen: von nationalen Workshops bis zur internationalen Konferenz
- Aktuelle Information zu CBR im WWW
 - ◆ www.fgwm.de (Fachgruppe Wissensmanagement der GI)
 - ◆ www.ai-cbr.org (CBR Portal)
 - ◆ www.iccbr.org (Konferenzseite)

Verwandte Gebiete aus methodischer Sicht

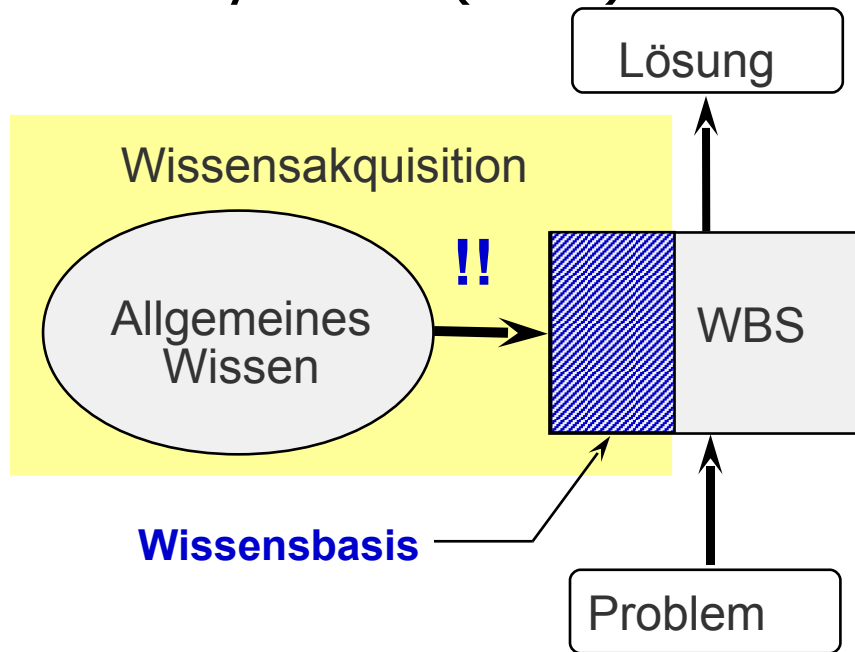


Motivationen für fallbasiertes Schließen

- Vermeidung eines hohen Wissensakquisitionsaufwandes bei der Realisierung wissensbasierter Systeme
- Einfache Wartung des Wissens des Systems
- Höhere Qualität der entstehenden Lösungen
- Höhere Effizienz beim Problemlösen
- Intelligente, wissensbasierte Suche in Datenbeständen
- Benutzerakzeptanz

Vermeidung eines hohen Wissensakquisitionsaufwandes

Klassische Wissensbasierte Systeme (WBS)



Akquisition von allgemeinem
regelhaftem Wissen ist sehr
schwierig !!

Fallbasierte Systeme

- Wenig allgemeines Wissen erforderlich
- Fallwissen einfach zu akquirieren (ggf. bereits vorhanden)

Was passiert wenn sich das Einsatzumfeld ändert ?

- Regelbasen sind schwer zu warten
 - ◆ Viele Abhängigkeiten zwischen Regeln
 - ◆ Regeln sind oft schwer zu verstehen (aufgrund der Repräsentation)
 - ◆ Auswirkung von Änderungen sind schwer vorherzusehen (erfordern oft einen KI-Experten)
- Fallbasen sind leichter zu warten:
 - ◆ Fälle sind in der Regel unabhängig voneinander
 - ◆ Fälle sind auch für Laien leicht zu verstehen
 - ◆ Wartung der Fallbasis kann durch Hinzunahme oder Löschen von Fällen geschehen.

Höhere Qualität der entstehenden Lösungen



- Oft sind Problembereiche nicht vollkommen verstanden.
- Daher: Selbst Experten können keine Regeln angeben um gute Lösungen zu finden.
- Fallbasierte Systeme speichern gute Lösungen in der Fallbasis
- Wenn lediglich geringe Anpassung erforderlich ist, wird dies keine große Auswirkung auf der Qualität der Lösung haben.

- Lösen von Problemen “nach ersten Prinzipien” ist oft sehr aufwendig.
 - ◆ Lösungen müssen von Grund auf komplett neu erzeugt werden
 - ◆ Beispiel: Handlungsplanung ist NP-vollständig.
- Fallbasiertes Schließen ermöglicht das Wiederverwenden von Erfahrungen:
 - ◆ Abändern einer Lösung ist oft einfacher, d.h. effizienter
 - ◆ Aber: Zur Zeit noch kein formaler Nachweis der Effizienzsteigerung bekannt.

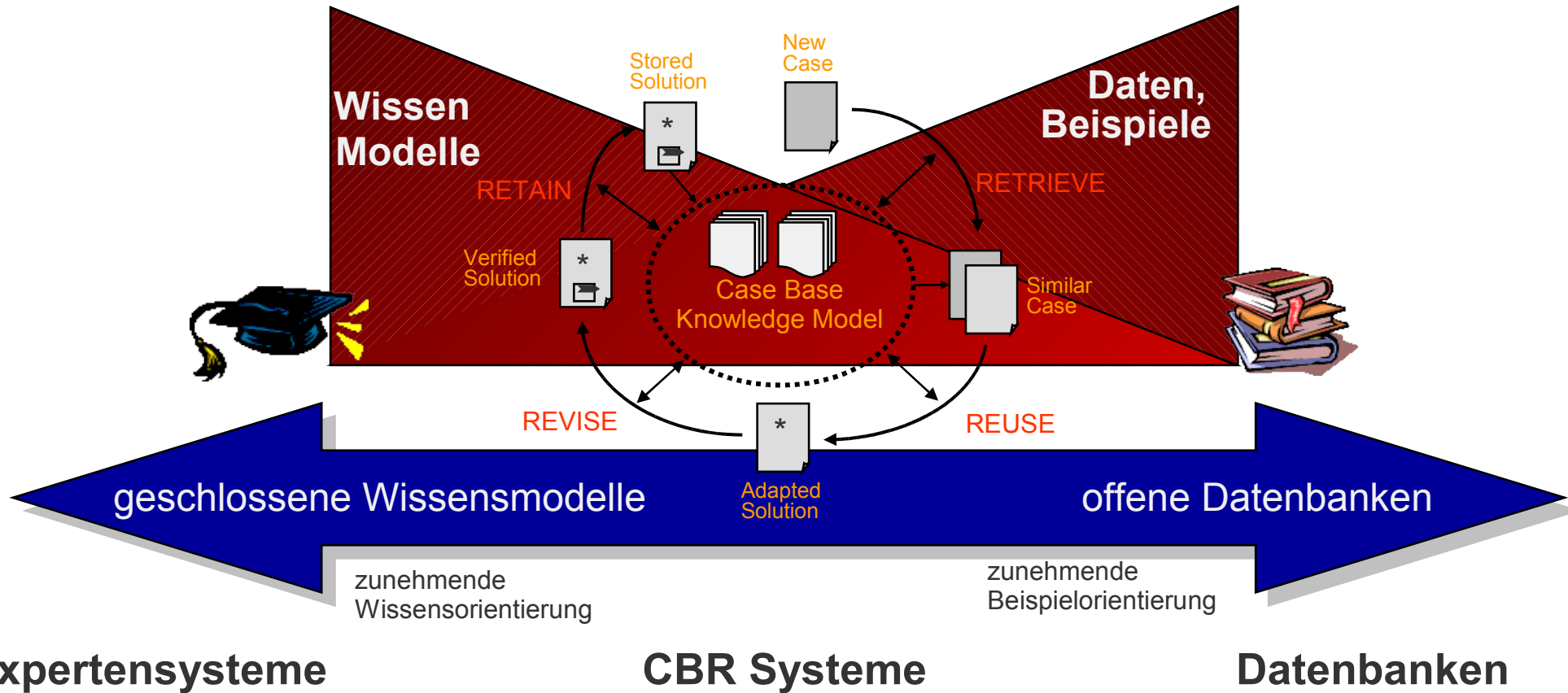
Intelligente, wissensbasierte Suche in Datenbeständen



- Oft sind Datenbestände bereits vorhanden
- Datenbanken ermöglichen nur eine begrenzte Ausnutzung der Datenbestände
- Fallbasiertes Schließen ermöglicht:
 - ◆ Intelligenteres Retrieval (Intelligenz in der Ähnlichkeitsbeurteilung)
 - ◆ Weiterverarbeitung (Anpassung) der gefundenen Daten

- Wie kann die Lösung die das System gefunden hat dem Benutzer erklärt werden ?
- Wie bekommt der Benutzer Vertrauen in das Ergebnis ?
- Problem bei regelbasierten Systemen oder Neuronalen Netzen:
 - ◆ Inferenzprozeß zur Ergebnisfindung oft sehr kompliziert
 - ◆ Ergebnis Neuronale Netze überhaupt nicht erklärbar
- Fallbasierte Systeme haben eine hohe Erklärungsfähigkeit:
 - ◆ Ausgewählter Fall und Lösungsanpassungen können dem Benutzer präsentiert werden

Positionierung von CBR Systemen



Expertensysteme vs. CBR

Expertensystem	CBR System
• Experten sollen ersetzt werden	• Unerfahrene Benutzer sollen unterstützt werden
• Sollen neues Wissen erzeugen	• Suche und Anpassung
• Implizit gespeichertes Wissen	• Explizites Wissen in Fällen im Vordergrund
• Wartung aufwendig	• Leichter zu warten
• Zielen auf 100%ige Abdeckung des Gegenstandsbereiches ab	• Abdeckung von relevanten Teilbereichen

DBMS	CBR System
<ul style="list-style-type: none">• Einfache Suche \Rightarrow "alles oder nichts"	<ul style="list-style-type: none">• Suche nach ähnlichen Daten, ggf. unter Verwendung von DBMS
<ul style="list-style-type: none">• Häufig zu viele oder keine Treffer	<ul style="list-style-type: none">• Anordnung der Ergebnisse; zeigen der ersten n Treffer
<ul style="list-style-type: none">• Kein domänenspezifisches Wissen bei der Suche verwendet	<ul style="list-style-type: none">• Domänenspezifisches Wissen kann in Ähnlichkeitsmaßen kodiert werden

Information Retrieval (Suchmaschinen) vs. CBR



Suchmaschinen (IR)	CBR System
<ul style="list-style-type: none">• Indexieren das Internet	<ul style="list-style-type: none">• spezifisch für eine Anwendung
<ul style="list-style-type: none">• Suche nach Webseiten	<ul style="list-style-type: none">• Suche nach Lösungen
<ul style="list-style-type: none">• Schlüsselwortorientiert	<ul style="list-style-type: none">• IdR. merkmalsorientiert
<ul style="list-style-type: none">• Kein Domänenwissen bei der Suche	<ul style="list-style-type: none">• Domänenspezifisches Wissen kann in Ähnlichkeitsmaßen kodiert werden

Wann ist CBR gut geeignet ?

- ... Benutzer sollen unterstützt werden
- ... Fälle sind vorhanden oder können leicht erfasst werden
- ... Wartung erforderlich
- ... keine 100%ige Abdeckung des Gegenstandsbereiches nötig
- ... Ähnlichkeitsbasierte Suche ist schnell genug

CBR kann als Fusion unterschiedlicher Konzept verstanden werden

