Zeitschrift: Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

Herausgeber: Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

Band: 83 (1992)

Heft: 17

Artikel: Aus alten Fehlern lernen : Case-Based-Reasoning : eine neue Software-Technologie für die technische Diagnostik

Autor: Lebsanft, Ernst / Prim, André / Braun, François

DOI: https://doi.org/10.5169/seals-902859

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften auf E-Periodica. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. Das Veröffentlichen von Bildern in Print- und Online-Publikationen sowie auf Social Media-Kanälen oder Webseiten ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Mehr erfahren

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. La reproduction d'images dans des publications imprimées ou en ligne ainsi que sur des canaux de médias sociaux ou des sites web n'est autorisée qu'avec l'accord préalable des détenteurs des droits. En savoir plus

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. Publishing images in print and online publications, as well as on social media channels or websites, is only permitted with the prior consent of the rights holders. Find out more

Download PDF: 09.07.2025

ETH-Bibliothek Zürich, E-Periodica, https://www.e-periodica.ch

Aus alten Fehlern lernen

Case-Based Reasoning – eine neue Software-Technologie für die technische Diagnostik

Ernst Lebsanft, André Prim und François Braun

Case-Based Reasoning (CBR) ist ein relativ neuer Ansatz im Bereich wissensbasierter Systeme. CBR eignet sich für Aufgaben der Diagnose, Planung und Konfiguration insbesondere dort, wo modellbasierte Ansätze zu aufwendig sind. In diesem Beitrag werden die Prinzipien des CBR vorgestellt und anhand eines Netzwerkdiagnosesystems illustriert.

Case-Based Reasoning (CBR) est une approche relativement nouvelle dans le domaine des systèmes basés sur la connaissance. CBR est prévu pour résoudre des tâches de diagnostic, de planification et de configuration, surtout là où les approches basées sur des modèles sont trop dispendieuses. Cet article présente les principes du CBR et l'illustre à l'aide d'un système de diagnostic de réseau.

Adresse der Autoren
Dr. *Ernst Lebsanft* und *François Braun*, Synlogic AG, 4102 Binningen und *André Prim*, GD PTT, 3029 Bern.

Die Technologie wissensbasierter Systeme (WBS) wird schon seit längerer Zeit und mit Erfolg in der Elektrotechnik und Elektronik angewendet [1] und ist im Bulletin SEV/VSE[1] in zahlreichen Artikeln ausführlich vorgestellt und gewürdigt worden. Beispielhaft seien etwa folgende Anwendungsgebiete genannt, in denen heute zahlreiche Systeme mit hohem Nutzen im operationellen Einsatz sind:

- Störungsdiagnose und -behebung in elektrischen Netzwerken
- Design und Planung elektrischer Netzwerke
- Design elektronischer Schaltkreise
- Fehlerdiagnose in elektronischen Schaltkreisen
- Präventive Wartung in Kraftwerken und Netzwerken
- Störungsdiagnose und -behebung an Maschinen

Der vorliegende Artikel befasst sich mit einer neueren wissensbasierten Softwaretechnik für die Diagnostik, Case-Based Reasoning, wobei die Anwendung in der Störungsdiagnose und -behebung sowie in der Überwachung und präventiven Wartung von elektrischen und Kommunikationsanlagen im Vordergrund steht. Dem Gebiet der wissensbasierten Diagnostik wird nicht zuletzt wegen des hohen Rationalisierungspotentials sehr gute Entwicklungsmöglichkeiten vorausgesagt. «... the fastest growing functional area for expert system development today», urteilt zum Beispiel Popolizio [2].

Eine Betrachtung im deutschsprachigen Raum bekannt gewordener wissensbasierten Diagnosesysteme [3...6] zeigt, dass – vereinfacht gesagt – das diagnostische Wissen entweder in Form oberflächlicher Symptom-Ursachen-Beziehungen (flaches Wissen, Shallow Reasoning) oder modellbasiert (Deep Reasoning, Expertensysteme der 2. Generation) repräsentiert wird [7]. Diese Methoden und Techniken haben sich vielfach bewährt und können heute als etabliert gelten. Es gibt aber diagnostische Problemstellungen, für die eine modellbasierte Systementwicklung Nachteile aufweist, für die es aber vielversprechende neuere Lösungsansätze wie eben zum Beispiel das Case-Based Reasoning gibt.

WBS im allgemeinen und damit auch wissensbasierte Diagnosesysteme werden heute meistens nach dem folgenden – vereinfacht dargestellten – Phasenschema realisiert, wobei das Vorgehen nicht streng sequentiell, sondern zyklisch ist:

1. Erhebung des Wissens aus Dokumenten durch Interviews usw., meistens auf Basis von Fällen (z.B. Störungen und deren Behebung), an die sich der Experte erinnert und die oft genug sogar dokumentiert (Störungsrapporte o.ä.) sind.
2. Erarbeitung eines analytischen Modells, das heisst Formulierung allgemeiner Regeln, kausaler Zusammenhänge u.a.
3. Erarbeitung eines Design-Modells (Systemdesign) unter Verwendung von Konzepten der WBS-Technologie.
4. Implementierung des Systems mittels geeigneter Programmiersprachen oder spezieller Software-Werkzeuge («Shells»).

Ausführliche Darstellungen konventioneller wissensbasierter Diagnosesysteme sind zahlreich in der Literatur [3; 7...10] zu finden, weshalb hier nicht im Detail auf sie eingegangen wird. Den zitierten modellorientierten

[1] Insbesondere in den Heften 15/1988 und 3/1989

Verfahrensweisen ist gemein, dass auf vergangener Erfahrung beruhendes Wissen mit entsprechendem Aufwand erhoben und in eine dem Lösungsansatz entsprechende Modellform transponiert werden muss [11]. Das hat zur Folge, dass

1. neu auftretende Probleme, die aufgrund der Erfahrung nicht direkt gelöst werden können, auch mit einem so modellierten System nicht gelöst werden können. (Die Systemergebnisse sind dann nicht vorhersagbar; es fehlt den Systemen an der sogenannten Graceful Degradation), und dass
2. Systeme in Bereichen, in denen sich Erfahrung und Wissen schnell ändern, immer hinter dem aktuellen Wissensstand hinterher hinken und somit gegebenenfalls sogar absolut unbrauchbar sind.

Ausserdem ist es aus Komplexitätsgründen oft schwierig, überhaupt ein Modell zu erstellen, umfangreichere Systeme zu validieren oder sie zu pflegen. Mit dem Ansatz des sog. Case-Based Reasoning (CBR) lässt sich mindestens ein Teil der erwähnten Schwierigkeiten, die man mit konventionellen wissensbasierten Systemen hat, lösen oder umgehen. Im folgenden werden die Prinzipien des CBR dargestellt, und es wird über erste Erfahrungen aus einem Projekt im Bereich der Störungsdiagnose in lokalen Netzen berichtet.

Case-Based Reasoning

CBR ist eine Technologie, die auf Theorien über menschliche Erkenntnis und über das Gedächtnis beruht [12; 13] und die neue Probleme aufgrund vergangener Erfahrungen (gespeicherte Fälle) zu lösen erlaubt. Unter Fall ist im Bereich der Diagnostik die Beschreibung eines Diagnoseproblems (zum Beispiel in Form einer Liste von Eigenschaften oder Beobachtungen) und dessen Lösung (d.h. die assoziierte Diagnose und gegebenenfalls die Therapie) zu verstehen. Im Unterschied zu konventionellen WBS ist die Kenntnis kausaler Zusammenhänge nicht notwendig. Seit den ersten Forschungsarbeiten auf diesem Gebiet, Mitte der 70er Jahre [14...17], sind etliche CBR-Systeme in den Bereichen Diagnose, Klassifikation, Vorhersage, Planung, Design, Prozesskontrolle, Überwachung und Konfiguration entwickelt und in Produktion genommen worden [12; 18]. Kommerzielle Produkte, Software-Werkzeuge also, die erlauben, CBR-Systeme in einem industriellen Umfeld zu realisieren, gibt es allerdings erst seit 1991 [19]. Bild 1 zeigt schematisch die unterschiedlichen Funktionsweisen eines CBR-Systems und klassischer (d.h. regelbasierter, modellbasierter, ...) wissensbasierter Systeme.

Funktionsweise von CBR-Systemen

Die Funktionsweise eines CBR-Systems wird anhand eines vereinfacht dargestellten Vorgehens- oder Lebenszyklusmodells erklärt. Die Schritte 1 bis 3 betreffen vor allem die Entwicklung eines CBR-Systems, Schritt 4 dessen Benutzung und Schritt 5 die Pflege und Wartung.

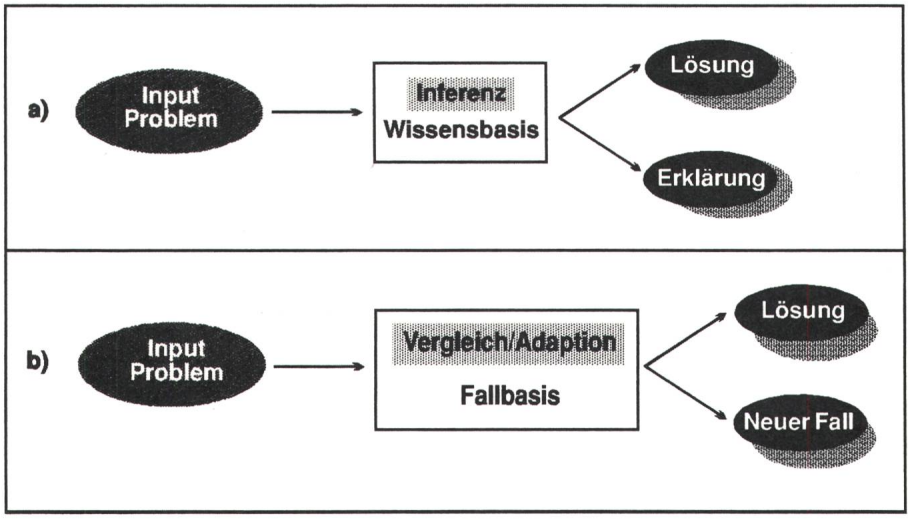

Bild 1 Funktionsweise wissensbasierter Systeme
a klassische Systeme
b fallbasierte Systeme

1. Schritt: Analyse und Beschreibung von Problemfällen

Der 1. Schritt besteht in der Sammlung und Darstellung geeigneter Fälle. Es ist zu analysieren, wie man Fälle konzeptuell und allgemein beschreibt, was die Fälle im einzelnen voneinander unterscheidet, ob und gegebenenfalls was für Strukturen vorhanden sind, wie der Problembereich durch die Fälle abgedeckt wird usw. Fälle können nach dem heutigen Stand der Technik beschrieben werden

- als eine Liste von Merkmalen (einfachste Form),
- als Beschreibung in freiem Format (in natürlicher Sprache) und
- als eine verbundene Menge von Teilfällen (komplexeste Form),

wobei die Techniken auch miteinander kombiniert werden. Diese Analyse erfordert Kenntnisse und Erfahrung im sogenannten Knowledge Engineering [11...13]. Das Ergebnis dieses Schrittes ist eine strukturierte und formalisierte Fallsammlung.

2. Schritt: Implementation einer Fallbasis auf dem Rechner

Die Strukturen der Fallsammlung sind nun in geeigneter Form auf dem Rechner abzubilden und die Fälle selbst sind zu erfassen. Im allgemeinen wird eine objektorientierte Darstellung gewählt, das heisst jeder Fall ist eine Instanz der Klasse «Fall» mit bestimmten Attributen und Werten. Diese Arbeit wird von heute verfügbaren CBR-Software-Werkzeugen in unterschiedlicher Weise unterstützt.

Die Art und Weise der Darstellung und Strukturierung bestimmt in signifikanter Weise Effektivität und Effizienz der Indexierung und des Retrievals (siehe unten) und erfordert darum entsprechende Kenntnis und Erfahrung. Sind die Grundstrukturen und eine Pflegemethodik jedoch einmal implementiert, kann die Fallbasis auch von Nichtinformatikern erweitert und gepflegt werden.

In manchen Problembereichen, wie zum Beispiel bei der Diagnose von Störungen in Rechnern, Kommunikationsnetzwerken oder elektrischen Netzen, können die ersten zwei Schritte gegebenenfalls dadurch erheblich vereinfacht werden, dass Störungen und damit verbundene Meldungen gelegentlich automatisch protokolliert und gespeichert werden und somit für eine Fallbasis bereits in elektronischer Form zur Verfügung stehen.

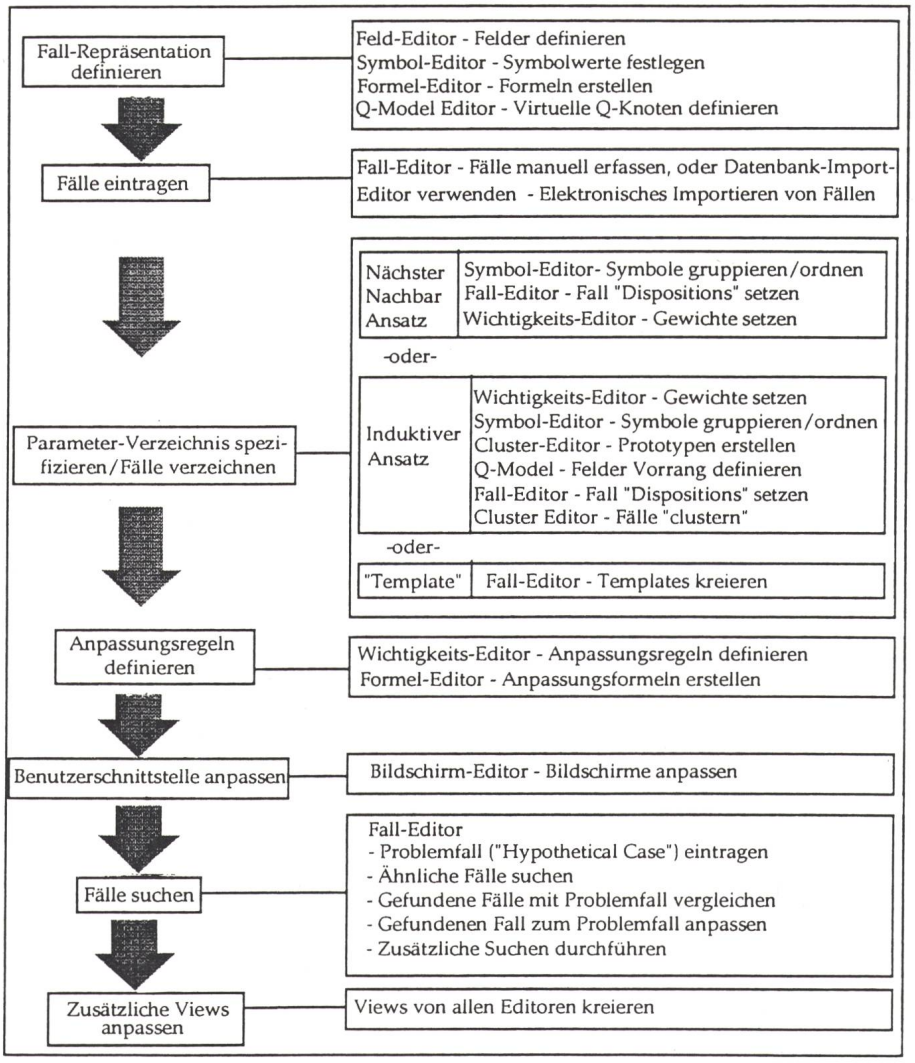

Bild 2 Vorgehensmodell für die Entwicklung eines CBR-Systems mit Remind

3. Schritt: Indexierung der Fälle

Die Mächtigkeit eines CBR-Systems beruht auf seiner Fähigkeit, relevante Fälle schnell und akkurat in der Fallbasis zu finden. Für diesen Prozess, das heisst die Bestimmung eines Ähnlichkeitsmasses zwischen gespeicherten Fällen und dem zu lösenden Fall sowie die Einordnung gespeicherter Fälle (Indexierung), sind folgende Techniken bekannt:

Der Nächste-Nachbar-Ansatz (am besten bekannt und meistgebraucht): Der zu lösende Fall wird mittels Gewichtung von Fallmerkmalen analysiert und mit auf gleiche Weise strukturierten vorhandenen Fällen auf maximale Ähnlichkeit (Berechnung von Gewichtssummen) verglichen.

Die induktive Indexierung: Die Falleigenschaften der zur Fallbasis gehörenden Fälle werden in einem Baum und mittels Algorithmen wie ID3, AQ11 oder CART [20], die auch von einfachen induktiven Werkzeugen wie 1st Class oder Expert Ease[2] benutzt werden, angeordnet. Das Auffinden des dem aktuellen Fall ähnlichsten Falles geschieht durch Abarbeiten des Baumes.

Die Erstellung einer Baumhierarchie prototypischer Fälle: Dabei werden abstrakte Klassen zwischen der Wurzel (der Klasse «Fall») und den spezifischen Fallinstanzen eingeführt.

Der wissensbasierte Ansatz: Dabei werden zum Beispiel Produktionsregeln zum Auffinden von Fällen aufgestellt und schliesslich: *Die Kombination der aufgeführten Techniken.*

Die Wahl und gegebenenfalls Kombination verschiedener adäquater Techniken sowie deren Implementation in einer konkreten Aufgabenstellung erfordert entsprechende Kenntnis und Erfahrung, trotz der Unterstützung heutiger Werkzeuge.

[2] Dies sind keine CBR-Werkzeuge!

4. Schritt: Fall-Retrieval

Das Auffinden von Fällen in der Fallbasis entspricht natürlich den für die Indexierung gewählten Techniken:

- das assoziative Suchen von Fällen bei der Nächsten-Nachbar-Indexierung,
- das hierarchische Suchen von Fällen bei induktiver oder hierarchisch-prototypischer Indexierung und
- das wissensbasierte Suchen in der Fallbank.

Für den Benutzer kommt es darauf an, zu seinem eingegebenen Fall möglichst schnell in für ihn natürlicher Weise die richtige Lösung zu finden. Das Retrieval muss so gestaltet sein, dass er – im Unterschied etwa zur Benutzung vieler Datenbanken – kein Wissen darüber benötigt, wie das Retrieval stattfindet. Ein gutes Design einer entsprechend komfortablen Benutzeroberfläche ist darum unerlässlich, wobei sich sogenannte Mixed Initiative User Interfaces bewährt haben. Ferner hängen die Suchzeiten in einer Fallbank stark von der gewählten Indexierungstechnik ab [11; 12; 18], was wegen der Benutzerakzeptanz in Schritt 3 bereits zu berücksichtigen ist.

5. Schritt: Adaption und Speicherung neuer Fälle

Bei der Benutzung eines CBR-Systems sind im Prinzip zwei Situationen denkbar:

1. Der Benutzer findet einen genau passenden Fall und hat damit die Lösung.
2. Der Benutzer findet keinen Fall in der Fallbank, der sein Problem genau löst.

Tritt die zweite Situation auf, so ist es oft sinnvoll und möglich, Funktionen zur Verfügung zu stellen, mit denen vorhandene Fälle so verändert werden können, dass sie eine Lösung darstellen. Diese – neuen – Fälle müssen natürlich dann auch in der Fallbank gespeichert und indexiert werden können. Die Bereitstellung solcher Funktionen, die ein CBR-System lernfähig macht, ist keine triviale Aufgabe, wenn Konsistenz, Effizienz und Effektivität des Systems sichergestellt sein sollen. Für die Falladaption gibt es in heutigen Werkzeugen wenig Unterstützung; entsprechende Funktio-

nen sind darum gegebenenfalls spezifisch zu implementieren. Erweiterung und Veränderung der Fallbank wird in der Praxis nur entsprechend autorisierten Personen gestattet sein.

Bild 2 zeigt schematisch ein solches Vorgehen bei Verwendung des Werkzeugs Remind (siehe unten).

Erfahrungen aus der Anwendung der CBR-Technologie

Das Projekt

Zwischen Januar 1990 und Mitte 1991 wurde gemeinsam von der Generaldirektion der PTT und der Synlogic AG das *modellbasierte* System *Mail-Diagnostic* entwickelt. Mail-Diagnostic ist ein funktionales Modul für das Fault-Management in einem Netzwerk. Es ermöglicht die Ermittlung und Behebung von Fehlern von VMS-Mail Version 5.4 und Decnet Phase IV für lokale Netze [21]. Die Entwicklung führte zum Ergebnis, dass modellbasierte Systeme im Bereich des Fault Managements in Netzwerken nur mit erheblichem Aufwand zu realisieren sind.

In einem Folgeprojekt wurde darum Mail-Diagnostic mittels CBR reimplementiert. Zielsetzung war es, die Tauglichkeit der CBR-Technologie im allgemeinen und des verwendeten Werkzeugs im besonderen für Diagnoseaufgaben im Bereich des Netzwerkmanagements im Vergleich zum vorgängig erfolgten Ansatz zu evaluieren. Bei diesem Projekt dürfte es sich um eine der ersten Anwendungen der CBR-Technologie im Telekommunikationsbereich handeln.

Aus den verschiedenen, auf dem Markt verfügbaren CBR-Software-Werkzeugen (siehe unten) wurde CBR-Express der Inference Corp. ausgewählt [11; 22...24]. Die Bilder 3a bis 3c zeigen Ausschnitte aus einer Diagnosesitzung mit dem System CBR-Mail-Diagnostic. Im Descriptions-Fenster ist die eingegebene Beschreibung des Problemfalles zu sehen, während das Questions-Fenster aufgrund der Suche nach vergleichbaren Fällen generierte Fragen zeigt. Im Recommended Actions-Fenster sind die Lösungen (d.h. die entsprechenden Fälle in der Fallbank) mit den zugehörigen Gewichten zu sehen, die das System als in Frage kommend erachtet. Bei Vergleich von Bild 3a und Bild 3b erkennt man, wie zum einen aufgrund der Beantwortung einer Frage neue mögliche Lösungen (mit weiteren assoziierten Fragen) gefunden werden und wie sich zum anderen aufgrund der durch die Anworten des Benutzers vermittelten zusätzlichen Informationen die Gewichte der möglichen Lösungen verändern und so schliesslich die zum Problemfall genau passende Lösung (hier: set line status at on on sender host; Gewicht 100, Veränderung der Farbe bzw. des Grautons im Recommended Actions-Fenster) gefunden wird. Der Benutzer nähert sich also durch einen Frage-Antwort-Dialog der gewünschten Problemlösung, ausgehend von seinem initial vorhandenen Informationsstand, den er zu Beginn einer Sitzung dem System mitteilt (bei CBR-Mail-Diagnostic in natürlicher Sprache). Es wird also im Prinzip ein Dialog des Benutzers mit einem Experten nachgebildet. Das Bild 3c zeigt schliesslich, wie der Benutzer die Aktion «set line status at on on sender host» genau durchzuführen hat. CBR-Mail-Diagnostic bietet also nicht nur Hilfe zur Diagnose einer Störung, sondern auch zu deren Behebung.

Ergebnisse und Schlussfolgerungen aus dem Projekt

Die Realisierung von Mail-Diagnostic mittels CBR-Express führte u.a. zu folgenden konkreten Ergebnissen:

– CBR erscheint für die Realisierung eines solchen Diagnosesystems als geeignet, da die Fault Management-Experten sich bei der Lösung von Problemen an früher aufgetretenen und gelösten Fällen orientieren, während die Bildung eines (fehlerorientierten) Modells – wie zuvor geschehen – respektive die Ableitung allgemeiner Diagnoseregeln doch etwas artifiziell ist. Der CBR-Ansatz kommt eben dem Denken der Sachgebietsexperten wesentlich näher als andere Ansätze und sollte darum, wann immer möglich, ernsthaft in Erwägung gezogen werden. Menschen leiten Wissen aus Erfahrung ab. Erfahrung aber gründet sich auf Fälle, an die man sich erinnert und die man per Analogie auf eine neue Situation wieder anwendet [12]. Sehr oft dürfte es auch gar nicht möglich sein, die (episodische) Erfahrung so zu generalisieren, dass Modelle konstruiert werden können, zum Beispiel dann, wenn es zuviele Ausnahmen oder Variationen von einer Regel gibt.

– Der CBR-Ansatz erforderte weniger Aufwand als der modellbasierte, da sich die Wissensakquisition auf die Sammlung und Aufbereitung von Fällen beschränkt, und man kein Modell zu erstellen und zu validieren braucht.

– Der Aufbau einer Fallsammlung für ein Fault Management ist relativ leicht, zumal Netzwerksoftware dies elektronisch unterstützt.

– Wegen der erforderlichen Effizienz und Robustheit der Fallbankstruktur und der Indexierung erfordert der Aufbau des CBR-Systems einen qualifizierten und in der Technologie wissensbasierter Systeme erfahrenen Informatiker oder Analytiker. Eine entsprechende Implementierung vorausgesetzt, ist dann auch der fachliche Experte in der Lage, das System zu pflegen und auszubauen, das heisst die Fallbank zu erweitern.

– CBR erlaubt in sehr natürlicher Weise, gleichzeitig mehrere Lösungswege zu verfolgen, indem zu einem Problem mehrere ähnlich aussehende Fälle aus der Fallbasis zur Entscheidung herangezogen werden (Bild 2 und 3a).

– Die Erzeugung dynamischer und kontextueller Erklärungen – ein Vorteil klassischer wissensbasierter Systeme – ist mit dem CBR-Ansatz schwer zu realisieren.

– Die Anwendung einer Entwicklungsmethodik ist zu empfehlen, wobei zum Beispiel bei Einsatz von Kads [21] diese Methodik entsprechend adaptiert werden muss [25].

Software-Werkzeuge zur Entwicklung von CBR-Systemen

Für die Implementierung von CBR-Systemen sind heute verschiedene Werkzeuge zu Preisen zwischen 1500 und 15000 Franken kommerziell verfügbar. CBR-Express (Inference Corp.) ist eine Tool-Book-Applikation auf Basis des sehr verbreiteten und mächtigen WBS-Werkzeugs ART-IM und ist auf PCs unter MS-Windows ablauffähig. CBR-Express erlaubt vor allem die Nächste-Nachbar-Indexierung und assoziatives Retrieval von Fällen. Durch die Kopplung mit ART-IM sind wissensbasiertes Retrieval und hybride fall- und modellbasierte Implementierungen möglich. Tool-Book erlaubt ferner die Realisierung sehr komfortabler Benutzeroberflächen, die zum Beispiel auch die Verwendung natürlicher Sprache bei Benutzeranfragen an das System zulassen. Mit CBR-Express/Tool-Book/ART-IM entwickelte Ap-

plikationen können in andere Applikationen und Datenbanken integriert werden.

Remind (Cognitive Systems) verfügt über nahezu alle bekannten CBR-Techniken (siehe oben) und ist in dieser Hinsicht mächtiger als CBR-Express. Remind ist auf PCs (MS-Windows, OS/2) und Macintosh II-Computern ablauffähig und ist in C++ geschrieben. Die Implementierung spezifischer Benutzeroberflächen ist im Gegensatz zu CBR-Express eher schwierig, wie uns erste Erfahrungen auch mit diesem Werkzeug zeigen.

Ein weiteres Werkzeug ist Esteem von Esteem Software Inc., das vor allem hierarchische Indexierung und hierarchisches Retrieval von Fällen erlaubt. Es wurde mit Intellicorps WBS-Werkzeug Kappa-PC realisiert (zu dem der Entwickler aber aus Esteem heraus keinen Zugriff hat) und ist unter MS-Windows ablauffähig. Im Unterschied zu den vorgängig genannten Werkzeugen zielt Esteem mehr auf den Endbenutzer.

Von der Aion Corp. wurde jüngst als Ergänzung zum sehr bekannten WBS-Werkzeug ADS das Tool ADS/CBR angekündigt. ADS/CBR ist eine in ADS integrierte Version von Remind.

Ein weiteres, eher einfaches CBR-Werkzeug ist Induce-It von Inductive Solutions Inc., das auf Excel aufsetzt.

Ein ausführliche Würdigung der oben genannten Werkzeuge findet sich in [19; 26]. Dass seit Mitte 1991 fünf CBR-Werkzeuge auf dem Markt erschienen sind, zeigt, dass die führenden Hersteller von WBS-Werkzeugen in der Anwendung der CBR-Technologie einen vielversprechenden Markt sehen.

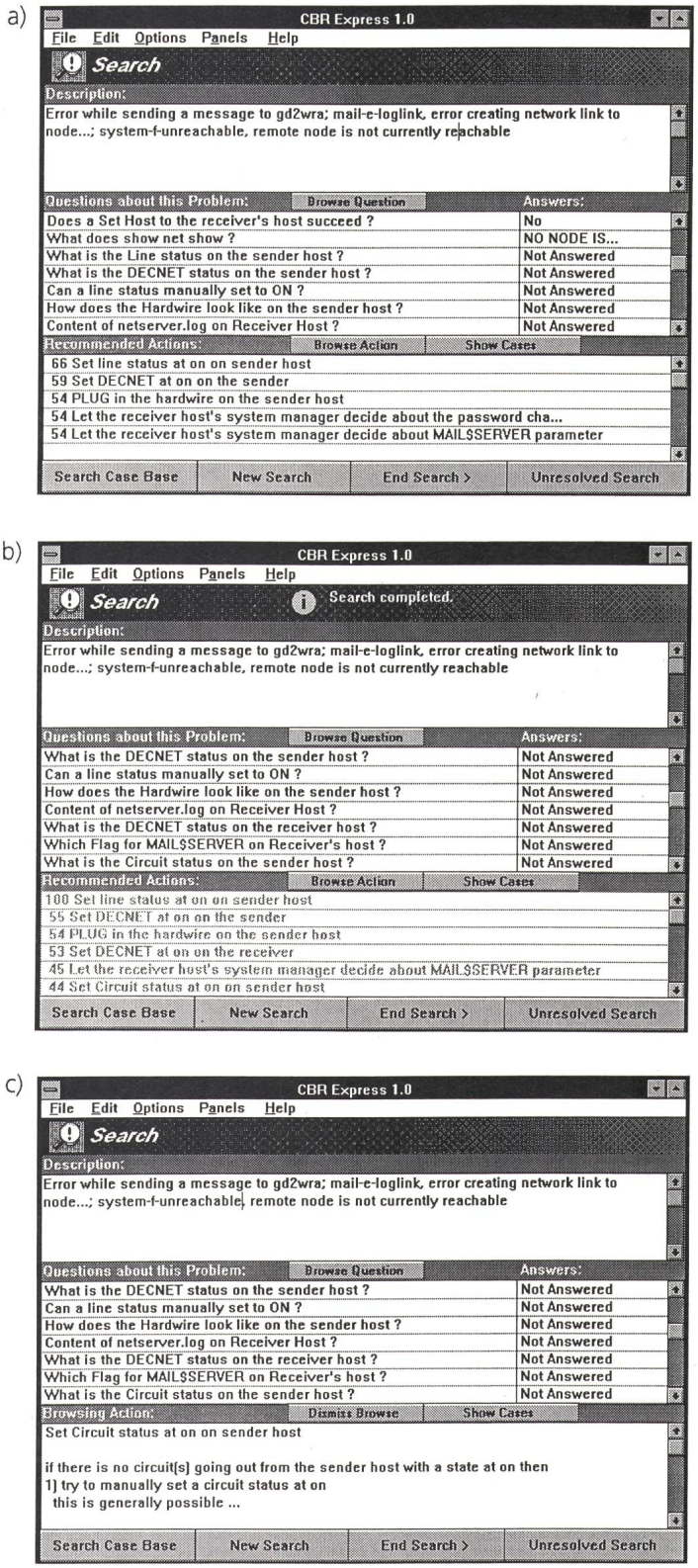

Bild 3
Ausschnitt aus einer Diagnosesitzung mit CBR-Mail-Diagnostic
a Phase 1
b Phase 2
c Phase 3

Anwendungspotential der CBR-Technologie

Die CBR-Technologie dürfte für sogenannte Help Desk-Diagnosesysteme – zum Beispiel für Hot Line-Dienste – besonders interessant sein [27]. Bekannte Anwender sind etwa American Airlines, Apple, Compaq, GTE, Lockheed, NCR und Nippon Steel [28]. CBR-Systeme stellen in solchen Bereichen eine natürliche Erweiterung des Problemrapportwesens dar und tragen dazu bei, dass gemachte Erfahrungen intensiver genutzt werden können. In Bereichen, in denen Störungen elektronisch erfasst und gespeichert werden, bietet sich eine Realisierung von CBR-Diagnosesystemen an, so zum Beispiel in den verschiedenen Bereichen der Telekommunikation oder der Erzeugung und Verteilung elektrischer Energie.

Schwerpunkt unserer Betrachtung war die Anwendung von CBR im Bereich der Diagnostik. CBR hat sich aber auch für andere Aufgabenstellungen als geeignet erwiesen, so zum Beispiel für Design- und Formulierungsaufgaben, Planung (Planning and Scheduling), Interpretation und Klassifizierung [13; 28]. So gibt es bereits CBR-Anwendungen zur Formulierung von Rezepten, zur automatischen Verarbeitung und Klassifizie-

rung von Telexen, für die militärische Planung oder für das Routing in Telefonnetzen [13; 18; 29].

Resumé

Mit Case-Based Reasoning steht, wie gezeigt wurde, für den in der Industrie wichtigen Bereich der technischen Diagnostik eine neue, höchst interessante Technologie zur Verfügung. Diese weist unter anderem die folgende Vorteile auf [20]:

- Wiederverwendbarkeit gefundener Lösungen
- Vermeiden vergangener Fehler
- Handhabbarkeit nur schwer formalisierbarer Aufgaben
- Lernen aus vergangener Erfahrung
- Komplementarität zu bekannten wissensbasierten Techniken
- Erleichterung des Knowledge Engineering

In der Schweiz haben Unternehmen verschiedener Branchen (Telekommunikation, Papierindustrie, Elektrizitätswirtschaft, Fahrzeugbau) bereits Interesse an dieser vielversprechenden Technologie bekundet.

Literatur

[1] *EC2 (Editor):* Onzièmes journées internationales des systèmes experts et leurs applications, Avignon, France, 27–31 mai 1991, conférence sectorielle: Intelligence artificielle et genie electrique, 7(1991), Nanterre.

[2] *John J. Popolizio:* Waging the Diagnostic Wars in the Early 1990, Artificial Intelligence Research, Part One and Part Two, February 20, 1989, pp. 947–956; March 20, 1989, pp. 996–1000.

[3] *Hans Voss, Frank Puppe (Hrsg.):* Abstracts über Diagnostikexpertensysteme, Arbeitspapiere der GMD Nr. 514, St.Augustin, Februar 1991.

[4] *Helmut Wollmersdorfer:* Artificial Intelligence in Österreich, ADV-Arbeitsgemeinschaft für Datenverarbeitung, Arbeitskreis Artificial Intelligence, Wien, Sommer 1988.

[5] *E. Marti:* Ausgewähltes Datenmaterial einer Untersuchung über Expertensysteme in der Schweiz, Arbeitsbericht Nr. 14 des Instituts für Wirtschaftsinformatik der Universität Bern, Juli 1987.

[6] *Peter Mertens:* Expertensysteme in der Produktion, Oldenbourg, München, 1990.

[7] *Peter Struss:* Model-Based Diagnosis, Tutorial T2, Second Annual Meeting of the Swiss Group for Artificial Intelligence and Cognitive Science, Geneva, October 2–3, 1990, Siemens AG, München, 1990.

[8] *F. Puppe:* Diagnostik-Expertensysteme, Informatik-Spektrum (1987)10, pp. 293–308.

[9] *F. Puppe:* Diagnostisches Problemlösen mit Expertensystemen, Informatik-Fachbericht, Band 148 der Gesellschaft für Informatik, Springer, Berlin, 1987.

[10] *Roman R. Laczkovich:* Expertensysteme zur technischen Fehlerdiagnose, Schmidt, Berlin, 1990.

[11] *Ernst Lebsanft:* Neue Software-Technologien für die technische Diagnostik, in: Michael Curth, Ernst Lebsanft (Hrsg.): Wirtschaftsinformatik in Forschung und Praxis, Hanser Verlag, München, 1992 (in Vorbereitung).

[12] *Stephen Slade:* Case-Based Reasoning: A Research Paradigm, AI Mag. 12(1991)1, pp. 42–55.

[13] *Janet L. Kolodner:* An Introduction to Case-Based Reasoning, Artificial Intelligence Review 6(1992)1, pp. 3–34.

[14] *Roger Schank:* Dynamic Memory: A theory of Learning in Computers and People, Cambridge University Press, Cambridge, 1982.

[15] *R. Schank, R. Abelson:* Scripts, Plans and Knowledge, in: Proc. of the Fourth International Joint Conference on Artificial Intelligence, Menlo Park, 1975, pp. 151–157.

[16] *M. Minsky:* A Framework for Representing Knowledge, in: P. Winston (Ed.): The Psychology of Computer Vision, McGraw-Hill, New York, 1975, pp. 211–277.

[17] *G. Sussman:* A Computer Model of Skill Acquisition, Artificial Intelligence Series, Volume 1, Elsevier, New York, 1975.

[18] *Ralph Barletta:* An Introduction to Case-Based Reasoning, AI Expert, August 1991, pp. 43–49

[19] *Paul Harmon:* Case-Based Reasoning III, ISS Vol. 8(1992)1, pp. 1–12, January 1992.

[20] *Kevin D. Ashley, Katia P. Sycara:* Case-Based Reasoning, Tutorial Notes, 12th International Joint Conference on Artificial Intelligence, Sydney, Australia, August 24–30, 1991.

[21] *Beat Liver, André Prim:* Wissensbasierte Systeme im Netzwerkmanagement, Technische Mitteilungen der PTT 2/1992, S. 38–44, Bern, Februar 1992.

[22] O. V.: CBR-Express TM 1.0 User's Guide, Inference Corp., El Segundo, 1991.

[23] *Dan Shafer:* CBR-Express Getting Down to Cases, PC AI July/August 1991, pp. 42–45.

[24] *Thomas Helton:* The Need for Case-Based Reasoning CBR-Express, AI Expert October 1991, pp. 55–57.

[25] *Brigitte Bartsch-Spörl:* A Simple Interpretation Model for Case-Based Reasoning, in: Christian Bauer, Werner Karbach (Eds.): Proceedings 2nd KADS User Meeting, Siemens AG, Munich, 17–18 February, 1992, Munich 1992 (Chapter 2.3).

[26] *Erick Brethenoux, A. Anthony Gee:* Who's Making the Best Case?, Applied Intelligent Systems, pp. 287–295, February 17, 1992, New Science Associates, Southport, USA.

[27] *Paul Harmon:* Case-Based Reasoning, ISS Vol. 7(1991)11, pp. 1–4.

[28] *Michelle Curtis:* Emerging Applications of Case-Based Reasoning, Applied Intelligent Systems, pp. 281–286, February 17, 1992, New Science Associates, Southport, USA.

[29] *John J. Popolizio:* Beyond Rules: Case-Based Reasoning, Artificial Intelligence Research, Part One and Part Two, May 16, 1988, pp. 678–680; June 6, 1988, pp. 685–694, New Science Associates, Southport, USA.