INSTITUT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Diplomarbeit

Einsatz von Bayesschen Netzen für den Intelligent Assistant des LRZ

Bearbeiterin:	Birgitt Mayer
Aufgabensteller:	Prof. Dr. Heinz–Gerd Hegering
Betreuer:	Dr. Gabi Dreo Rodosek (LRZ), Christian Ensel (MNM–Team)

INSTITUT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Diplomarbeit

Einsatz von Bayesschen Netzen für den Intelligent Assistant des LRZ

Bearbeiterin: Birgitt Mayer
Aufgabensteller: Prof. Dr. Heinz–Gerd Hegering
Betreuer: Dr. Gabi Dreo Rodosek (LRZ),
Christian Ensel (MNM–Team)
Abgabetermin: 15. Mai 1999

Ich versichere, daß ich die vorliegende Diplomarbeit selbständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 15. Mai 1999

....

(Birgitt Mayer)

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Kapitel 1

Einführung

1.1 Einleitung und Motivation

Das Münchener Leibniz–Rechenzentrum (LRZ) der Bayerischen Akademie der Wissenschaften bietet als wissenschaftliches Rechenzentrum für die Hochschulen in München und landesweites Zentrum für technisch–wissenschaftliches Hochleistungsrechnen ein breitgefächertes Spektrum an Serviceleistungen an. Im Kompetenzzentrum für Datenkommunikationsnetze werden neueste Technologien in Pilotprojekten in Zusammenarbeit mit dem Verein Deutsches Forschungsnetz (DFN) erprobt. Zu den Aufgaben des LRZ gehören die Infrastrukturbereitstellung der Kommunikationsnetze, die Forschung auf dem Gebiet des Netz– und Systemmanagements und die Dienstleistungen, die die Infrastruktur betreffen. Über das LRZ ist der Münchener Hochschulraum einer der größten Internetnutzer in Deutschland.

Eine reibungslos funktionierende Kommunikationsinfrastruktur, wie sie im LRZ vorhanden ist, ist eine grundlegende Voraussetzung im Informationszeitalter. Es gehört zu den Leistungen eines Kommunikationsdienstleisters, eines sogenannten Dienstproviders, seinen Kunden (Dienstnutzer) ein gewisses Maß an Kommunikationsqualität zu garantieren. Diese Qualität wird in sogenannten Dienstgüteparametern festgelegt und mit den Kunden des Providers, zumindest im kommerziellen Bereich, in Dienstgütevereinbarungen (Service Level Agreements) vertraglich geregelt. Der Funktionsbereich des Netzmanagements eines Providers wird in mehrere Ebenen aufgeteilt. Im Leistungsmanagement eines Dienstanbieters werden Quality of Service (QoS)–Parameter definiert. Ein Beispiel für einen derartigen Parameter stellt die Zeit von der Fehlermeldung eines Dienstnutzers bis zur ersten Reaktion des Dienstanbieters dar. Zu einer der Aufgaben des Fehlermanagements zählt es, die Verfügbarkeit einer Kommunikationsinfrastruktur möglichst hoch zu halten. Dies kann durch ständige Überwachung und entsprechende Reaktion auf Fehler bzw. Fehlermeldungen erreicht werden.

Der *Help Desk* eines Kommunikationsdienstanbieters setzt sich überwiegend aus den Experten zusammen, die für die Fehlerlokalisierung und Fehlerbehebung zuständig sind. Die Experten erkennen Probleme und sind fähig, diese zu lösen. Dazu wird die Expertenriege am LRZ zur (telefonischen) Hilfestellung in einer sogenannten Hotline organisiert. Sie dient dem Benutzer als zentrale Anlaufstelle.

Die Experten benutzen zur Unterstützung der Überwachung des Netzes und der Lokalisierung der auftretenden Fehler Werkzeuge als Hilfsmittel.

Ein sogenanntes Trouble–Ticket–System (TTS) zählt dabei zu den passiven Werkzeugen in diesem Managementumfeld. Es bietet dem *Help Desk* die Fehlermeldungen in einem

Dokumentationssystem an. Von der ersten Erfassung eines Fehlers über die gestellte Diagnose bis hin zu den Einzelschritten der Fehlerbehebung kann der Experte sich während seiner Hilfestellungsphase informieren sowie seine eigenen Schritte dokumentieren.
Am LRZ wird ein dem Trouble–Ticket–System vorgeschaltetes Werkzeug eingesetzt. Die Aufgabe des Intelligent Assistant (IA) ist es, den Vorgang der Erstellung eines Trouble–Tickets (TT) von der Fehlermeldung bis zur Fehlerlokalisierung weitgehend zu automatisieren. Dieser Assistent kann in den sogenannten First Level Support eingeordnet werden. Das ist die Hilfe, die der Benutzer zuerst erfährt und in Anspruch nehmen kann. Benutzergeführt mit Fragen, die dem Benutzer gestellt werden, oder mittels Diagnosefunktionen, die durchgeführt werden können, wird ein Fehler in einem Rechnernetz lokalisiert. Die Bedienung erfolgt dabei auf zwei Arten. Einerseits nutzt der Experte am LRZ den Assistenten und nimmt die Fehlermeldungen der Kunden telefonisch entgegen, trägt sie ein und kann zugleich Rückfragen stellen. Andererseits in der direkten Bedienung per WWW–Browser kann der Dienstnutzer benutzergeführt den Fehler eingrenzen und ein Trouble–Ticket erstellen. Derzeit ist der Intelligent Assistant für die Problembereiche Verbindung, Durchsatz und Mail realisiert.
Die Automatisierung dieser „Ersthilfe“ bedeutet für die am LRZ zuständigen Experten eine Entlastung hinsichtlich zeitaufwendiger Routineanfragen. Des weiteren können in der telefonischen Beratung des LRZ auch weniger routinierte Experten eingesetzt werden. In jedem Fehlerfall werden durch den Intelligent Assistant Trouble–Tickets erstellt, qualifizierte Fehlermeldungen, auf die der Experte um so schneller und präziser reagieren und damit den Fehler entsprechend beheben kann.
Eingebettet in ein Expertensystem benutzt der Intelligent Assistant derzeit eine regelbasierte Technik zur Wissensrepräsentation. Auf diese Weise wird das Vorgehen in der Fehlerlokalisierung und Diagnose bestimmt. Dazu wurden die Fragestellungen der Experten an die Benutzer im Zusammenhang mit durchzuführenden Testroutinen dokumentiert und in Entscheidungsbäumen implementiert. Die schrittweise Abarbeitungsreihenfolge der Fragen und Tests wird je nach Testverlauf und Testergebnis durch diesen Baum vorgegeben. Die Experten haben die Kosten im Sinne von Nutzen und den Informationsgehalt einer Frage oder eines Tests in der Diagnose durch ihre Wissensvorgaben in die Entscheidungsbäume eingebaut und berücksichtigt. Durch diese Vorgaben ist die Effizienz der Diagnose, vor allem die Abarbeitungsreihenfolge betreffend, weitgehend eingeschränkt.
Daher wird in diesem Expertensystem nach weiteren Möglichkeiten der Wissensrepräsentation gesucht, um den statischen und wenig „intelligenten“ Ablauf in der Entscheidungsphase zu verbessern.
Die Bayesschen Netze bieten hier eine Möglichkeit, Wissen in anderer Form zu repräsentieren. Mit dem Einsatz von Wahrscheinlichkeiten kann der Grad der Überzeugung, wie es Entscheidungen unter Unsicherheit erfordern, dargestellt werden. Der Umgang mit unsicherer Information wird durch den Einsatz der Wahrscheinlichkeitstheorie ausgedrückt.
Die Abhängigkeiten einzelner Entscheidungen lassen sich, direkt oder indirekt, von Ursache und auftretenden Fehlern (Wirkung), graphisch darstellen. Der Leitgedanke ist, Schlußfolgerungen herzuleiten, ähnlich dem probabilistischen Schließen. Dieses Vorgehensmodell ist der menschlichen Entscheidungsfindung zur Lösung eines Problems sehr ähnlich.

1.2 Ziel der Arbeit

Die Bayesschen Netze werden aufgrund ihrer Komplexität und ihrer theoretischen Grundlage selten in technischen Gebieten eingesetzt. Das Netzmanagement, insbesondere die Situation am Leibniz–Rechenzentrum in München, bietet für derartige Expertensysteme und deren Implementierung ein sehr praxisnahes, technisches und vor allem dynamisches Einsatzfeld.
Das Spannungsfeld zwischen der Entscheidungstheorie in der Künstlichen Intelligenz (KI) und der Diagnoseanwendung im Rechnernetzmanagement des Leibniz–Rechenzentrums zu analysieren, ist Ziel dieser Arbeit. Aus den differenten Aspekten, die sich aus den beiden Bereichen ergeben, sollen schrittweise Lösungen für die Bildung und den Einsatz der Bayesschen Netze aufgezeigt werden. Die Anwendung des Intelligent Assistant und der Einsatz der Experten am Leibniz–Rechenzentrum soll dabei soweit wie möglich unterstützt werden.

1.3 Gliederung der Arbeit

Ausgehend von der Motivation und dem Ziel der Arbeit wird der Leser im folgenden Kapitel in die notwendigen *Grundlagen* zum Verständnis der Materie eingeführt. Das *Netzmanagement am Leibniz–Rechenzentrum* wird dargestellt, um den Einsatz des *Intelligent Assistant* zu erläutern. Über die Einführung in *Expertensysteme* wird mit dem Schritt zu den *Bayesschen Netzen* dieses Kapitel abgeschlossen.
In der *Analyse der bestehenden Situation* wird der *Intelligent Assistant* untersucht und seine Anwendung aufgezeigt. Zugleich ergeben sich daraus die *Anforderungen aus dem Netzmanagement* und die *Anforderungen aus dem Help Desk Szenario.*
Diese beiden Anforderungsgebiete werden zu *Anforderungen an die Bayesschen Netze* zusammengeführt und ergeben so ein Gesamtbild vom *Entwurf*, den *Eigenschaften* und der *Realisierung* der Bayesschen Netze.
Die *Modellierungsaspekte* zeigen auf, welche spezifischen Überlegungen in dieser Phase getroffen werden müssen. Eine Umsetzung der erarbeiteten Aspekte speziell für den Intelligent Assistant findet im darauffolgenden Kapitel statt.
Die *Lösungskonzepte* stellen eine mögliche Umsetzung der durch Anforderungen und Modellierungsaspekte ausgearbeiteten Grundlagen vor.
Diese Arbeit wird durch eine *Zusammenfassung und Ausblick* abgeschlossen.

Kapitel 2

Grundlagen

2.1 Netzmanagement am Leibniz–Rechenzentrum

Das Münchener Leibniz–Rechenzentrum betreibt hochleistungsfähige Rechensysteme zusammen mit einer sehr leistungsfähigen Infrastruktur. An das Netzmanagement werden dabei spezielle Anforderungen gestellt.

2.1.1 Leistungen

Als wissenschaftliches Rechenzentrum für die Hochschulen in München, als landesweites Zentrum für technisch–wissenschaftliches Hochleistungsrechnen und als Kompetenzzentrum für Datenkommunikationsnetze hat das LRZ ein breites Spektrum an Aufgaben. Dazu gehört die Forschung im Bereich des integrierten Netz– und Systemmanagements. Die Infrastrukur wird mit ca. 20000 Netzanschlüssen, ca. 250 Subnetzen und bis zu 250 Modem– oder 360 ISDN–Verbindungen bereitgestellt. Im Bereich Dienstleistungen bietet das LRZ von Druckschriften über Kurse und Praktika zu kommerziell erhältlichen Anwendungsprogrammen bis zur Spezialberatung von Unternehmen eine große Palette von Leistungen an.
Dazu zählen Internet–Dienste wie World Wide Web, Email, News, Proxy–Dienst und FTP–Server sowie Nameserver–Dienste. Zum Gesamtleistungsangebot des LRZ gehört auch die Organisationsform der Experten im *Help Desk* und die zugehörige Hotline.
Am LRZ ist der *Help Desk* in einer Telefon–Hotline organisiert. Während der Geschäftszeiten können die Mitarbeiter des LRZ direkt telefonisch erreicht werden, um bei Problemstellungen weiterzuhelfen. In einer Hilferufeliste ist jeder Mitarbeiter als Experte eines oder mehrerer Gebiete aufgelistet. Der Anspruch dabei ist, Probleme sofort telefonisch zu beheben oder bei der Lösung Hilfestellung zu leisten. In den Abendstunden übernehmen studentische Operateure diesen Dienst. Es ist intuitiv klar, daß die Qualität dieser Hilfestellung nicht der der Experten entsprechen kann. Dennoch soll dem Benutzer weitergeholfen werden und sei es nur durch eine Auskunft.
Die Fehler, die von den Benutzern auch tagsüber gemeldet werden, liegen häufig in einer Kategorie, die eine sofortige Problemlösung erlauben. Die Anhäufung trivialer und einfach zu lösender Fehler macht den Einsatz eines Werkzeugs wie dem Intelligent Assistant zur automatischen Hilfestellung und Fehlerdiagnose nicht nur möglich, sondern unabdingbar. Die Entlastung der Experten ist offensichtlich.

2.1.2 Fehlermanagement

Das integrierende Management von Rechnernetzen und vernetzten Systemen wird in mehrere Ebenen aufgeteilt. Das Netzmanagement befaßt sich schwerpunktmässig mit dem Management von Kommunikationsdiensten und eingesetzten Komponenten. In einer darüberliegenden Ebene ist das Systemmanagement einzuordnen, das sich vor allem mit den in Endsystemen und Systemverbunden vorhandenen Ressourcen beschäftigt.
Das Management derartiger Netze ist von vielfältigen Faktoren abhängig, so z.B. technische, betriebswirtschaftliche oder organisatorische Faktoren. Einer der technischen Aspekte in verteilten Systemen ist eine heterogene Systemumgebung, die nicht in jedem Einsatzbereich die gleichen Managementfunktionen fordert und verwendet. Die konkret vorliegende IT–Struktur bestimmt die Komplexität des Managements, das wiederum hinsichtlich verschiedener Funktionsbereiche aufgeteilt werden kann.
Die Aufteilung eines Rechnernetzmanagements erfolgt für die Funktionen in Konfigurations–, Fehler–, Leistungs–, Abrechnungs– und Sicherheitsmanagement.
Das Leistungsmanagement stellt die Funktionsfähigkeit des Gesamtsystems sicher. Es ist verantwortlich für die Einhaltung der mit einem Kunden vereinbarten QoS–Parameter.
Die Anbieter von Kommunikationsdiensten und –netzen vereinbaren und beschreiben ihre Leistungen gegenüber ihren Kunden in sogenannten Dienstleistungsvereinbarungen (engl.: Service Level Agreement, SLA).
Die Güte einer Leistung kann in quantifizierbaren Größen wie Mittelwert angegeben werden. Diese Quality of Service (QoS)–Parameter definieren die Schnittstellen im Gefüge der Kommunikationsinfrastruktur. Der *Help Desk*, wie er am Leibniz–Rechenzentrum besteht, zählt bei einem Kommunikationsdienstanbieter zum Bereich des Service, der das Fehlermanagement bedient.
Teilaufgaben des Fehlermanagements bestehen darin, die Dienstverfügbarkeit und die Verfügbarkeit des gesamten Systems im Kommunikationsnetz sicherzustellen. Dazu zählen die Entdeckung, Isolierung und Reparatur von Netzproblemen sowie die Verfolgung und Lokalisierung von Störungen.
Der *Help Desk* ist dabei ein wichtiges Instrument des Dienstanbieters. Da aus Gründen der Netzlast nicht jederzeit eine aktive Qualitätsprüfung des Systems und seiner Komponenten durchgeführt werden kann, werden Fehler auf verschiedene Art und Weise festgestellt. Komponenten, die defekt sind oder die einen Defekt selbsttätig melden, lösen Alarme im Überwachungssystem aus. Benutzer, die die Nichtfunktionalität von Komponenten oder Diensten feststellen, melden dies dem Betreiber des Netzes. Der erste Kontakt findet in diesen Fällen meistens zum *Help Desk* statt.

Reaktive und proaktive Maßnahmen

Für jeden Managementbereich ergeben sich zwei verschiedene Arten, Maßnahmen zu etablieren. Im Fehlermanagement werden ein dynamisches oder adaptives Routing als Maßnahmen verstanden, die im voraus erbracht und geleistet werden. Dabei erfolgt ein Neueintrag in die Routing–Tabellen, um Änderungen der Konfiguration im Netz oder in Geräten zu dokumentieren. Im Vorfeld werden Fehler verhindert oder minimiert und zählen damit zum proaktiven Management.
Im Gegensatz dazu steht das reaktive Management, das auf Fehler nur zu reagieren vermag. Dazu zählen unter anderem Expertensysteme, die aufgrund von Störungsmeldungen oder Feh-

lermeldungen entsprechende Maßnahmen zur Fehlerbehebung anstoßen können.

Fehlerquellen

Die auftretenden Fehler können im Hardwarebereich wie Datenübertragungssystemen, Netzkomponenten oder Endsystemen sowie in der auf den Systemen installierten Software auftreten. Ein nicht unerheblicher Faktor ist der durch Fehlbedienung ausgelöste Fehler.

Phasen

Die Diagnose und Lokalisierung eines Fehlers im Fehlermanagement läßt sich in vier Phasen aufteilen. In einem ersten Schritt wird ein Fehler erkannt. Im zweiten Schritt findet eine Fehlerkompensierung statt. Ein Operator sollte dabei eine passende Aktion für den Ausgleich dieses Fehlers auswählen. Dabei sollte er sich der Konsequenz seines Handelns bzw. seiner Entscheidung bewußt sein. In einem dritten Schritt findet eine genaue Fehlerdiagnose statt, die sich zusammensetzt aus der Isolierung und intensiven Beobachtung des Fehlers. In einem vierten und letzten Schritt wird der Fehler korrigiert. Hierzu stehen wiederum alternative Aktionen zur Verfügung. Damit wird auch der Einsatz von Expertensystemen im reaktiven Fehlermanagement eines Rechnernetzes deutlich.

2.1.3 Help Desk

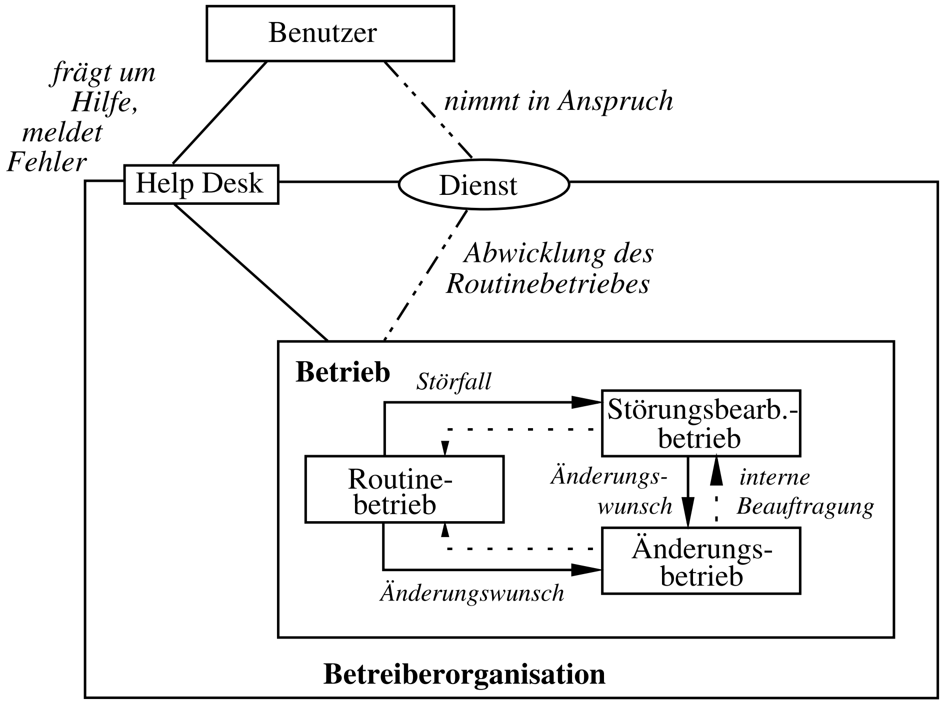

Abbildung 2.1: zentrale Betriebsprozesse und Schnittstellen

Der Netzbetreiber LRZ als Dienstanbieter besitzt mehrere Schnittstellen zu seinen Betriebsprozessen, wie *Abbildung 2.1* zeigt (vgl. *[HAN 99a, S. 425]*). Ein Benutzer nimmt einen Dienst

in Anspruch und bei Komplikationen wendet er sich an den *Help Desk*.
Der Betrieb eines vernetzten Systems wird in Routine–, Änderungs– und Störungsbearbeitungsbetrieb aufgeteilt. Der Übergang vom Routinebetrieb z.B. in den Störungsbearbeitungsbetrieb erfolgt durch einen im Routinebetrieb festgestellten Störfall. Durch interne Beauftragung können Rückkopplungen erfolgen. Der Betrieb besitzt zwei Schnittstellen. Über die Dienstschnittstelle wird der Routinebetrieb realisiert und abgewickelt. Zum Benutzer besteht eine weitere Schnittstelle in Form des *Help Desk*, die er in Anspruch nehmen kann, sobald Wünsche, Änderungen oder Fehlermeldungen außerhalb des Routinebetriebes auftreten. Der *Help Desk* ist eine einheitliche Schnittstelle zum Dienstnutzer und Kunden.
„Helpdesks fördern die Selbsthilfe der Kunden“ *[Wale 99]* — damit werden die Aufgaben dieser Institution deutlich. Im Fokus der Kommunikationsdienstanbieter steht eine weitgehend automatisierte Benutzerbetreuung, da sich damit eine hohe Servicequalität zu niedrigen Kosten erzielen läßt. Informationsübermittlung, Annahme von Beschwerden, vor allem Bearbeitung von Anfragen, Störungsmeldungen und Änderungswünschen. Diese zu automatisierenden Aufgaben können über verschiedene Medien wie Telefon, E–Mail oder eine Web–basierte Schnittstelle übermittelt und abgewickelt werden. Die Hilfe, die der Kunde dabei in Anspruch nehmen kann, teilt sich in mehrere Schichten auf. Erster Anlaufpunkt ist der First Level Support.

2.1.4 Netzmanagementwerkzeuge

Das Management einer Kommunikationsinfrastruktur bedient sich zur Erfüllung seiner Aufgaben einer Reihe von Werkzeugen. Diese kann man sich nicht nur als Hardware wie Prüfgeräte und Protokollanalysatoren in Kombination mit darauf laufender Software vorstellen, sondern auch als reine Softwareprodukte.

Internet–Bereich

Zum Standard eines TCP/IP–basierten Rechnernetzes zählen Mechanismen, die auf Standardprotokollen beruhen. Sie haben den Vorteil, daß keine managementspezifischen Anforderungen entstehen.
Z.B. *ping (packet internet groper)* und *traceroute*:

- *ping* testet die Verbindung zwischen zwei IP–Adressen. Dazu sendet das Programm in einem voreingestellten Zeitraum ein 64 Byte langes Datenpaket mit Echo–Request und stellt fest, wie lange es dauert, bis der angerufene Rechner antwortet. Dabei wird eine Schicht 3–Verbindung getestet, spezifiziert durch ihren Host–Namen oder IP–Adressen.

- *traceroute* schickt ein erstes Datenpaket mit $ttl = 1$ zum Zielrechner. Der erste Router verwirft dieses Paket und schickt es mit der Nachricht *ttl exceeded* wieder zur Quelle zurück. Die Testroutine *traceroute* extrahiert daraus die IP–Adresse und erhält damit den ersten Netzknoten, der erreichbar ist. Im zweiten Schritt sendet *traceroute* ein Datenpaket mit $ttl = 2$ und so weiter. So kann festgestellt werden, welchen Weg die Datenpakete zum Zielrechner nehmen und bis zu welchem Knoten die Strecke in Ordnung ist.

Trouble–Ticket–System

Dieses Hilfsmittel unterstützt eine koordinierte Zusammenarbeit der Spezialisten bei der Bearbeitung von Problemen. Dabei zählt es zu den passiven Maßnahmen im Netzmanagement.

Die Vorgabe einer Reaktionszeit von 4 Stunden auf Fehlermeldungen eines Benutzers von seiten des LRZ, bedingt den Einsatz eines Trouble–Ticket–Systems (TTS). Durch eine automatisch erzeugte E–Mail wird dem Fehlermeldenden die Bearbeitungsnummer seines Tickets gemeldet. So kann der Benutzer seinerseits die Bearbeitung seines Tickets nachvollziehen und verfolgen.
Das Trouble–Ticket–System wird zur Unterstützung eines *Help Desk* eingesetzt. Dieses *integrierende Werkzeug [HAN 99a, S. 309]* dokumentiert Fehlerfälle und sonstige Problembeschreibungen. Hierzu zählen Angaben zum Problem und zum Kunden ebenso, wie die Dokumentation der bereits durchgeführten Aktionen während der Bearbeitung und eventuell durchgeführte Lösungsschritte. Durch eine einheitliche Beschreibung, ähnlich einem Formular, wird eine qualitativ hochwertige Grundlage zur Erstellung eines solchen Tickets geschaffen. Die Reduktion der Informationswege zwischen den einzelnen Experten und somit eine bessere Nutzung des Expertenwissens ist ein weiterer Vorteil.
Ein TTS wird (siehe *Abbildung 2.2*) in mehrere Module aufgeteilt. Das Filtermodul sorgt hierbei für die beschränkte und nicht wiederholte Aufnahme eines protokollierten Fehlers in die Datenbasis. Durch Auswahl der Tickets kann eine Mindestqualität des Trouble–Ticket–Systems garantiert werden.
Das Diagnosemodul liefert die Grundlage für ein Expertensystem. Die Ansammlung von Tickets kann zu einer Wissensdatenbank ausgebaut werden, die für Fehlersuche und Fehlerdiagnose verwendet werden kann.
Die Trouble–Tickets bieten für eine Fehlerlokalisierung bzw. einen Fehler die entsprechend dazu aufgetretenen Symptome. Um den Rückschluß und den Zusammenhang von Trouble–Tickets auf Fehlerursachen bewerkstelligen zu können, muß entsprechend aufbereitetes Expertenwissen zur Verfügung stehen.
Die Informationspfade, auf welchen Daten und Informationen in das Trouble–Ticket–System gelangen, werden, neben manuellen Eingaben durch LRZ–Mitarbeiter, auch vom Intelligent Assistant verwendet.

Intelligent Assistant

Der Intelligent Assistant (IA), wie er am Leibniz–Rechenzentrum in München eingesetzt wird, ist ein Werkzeug im Fehlermanagement für den First Level Support, das dem Trouble–Ticket–System vorgeschaltet ist (siehe *Abbildung 2.3*). Seine Aufgabe ist es, den Vorgang der Erstellung eines Trouble–Tickets auf dem Weg von der Fehlermeldung bis zur Fehlerlokalisierung weitgehend zu automatisieren.
Diese Schnittstelle bietet ein Web–basiertes Frontend, das sowohl vom Kunden als auch vom jeweils zuständigen Experten bedient werden kann. Die Aufgaben dieses Assistenten sind eine benutzergeführte Fehlerlokalisierung und eine automatische Erfassung von relevanten Informationen.
In dieser Stufe des Supports gleichen sich die Anfragen und gemeldeten Fehler überwiegend. Auch von Expertenseite werden zu Beginn einer Fehlermeldung immer spezifische Informationen benötigt. Die durch den Assistenten bereitgestellte Routine stellt eine Automatisierung dieser Stufe in der Fehlerlokalisierung und Fehlerdiagnose dar.
Da der Prozeß der Fehlerverfolgung und Fehlerlokalisierung in verteilten Systemen zunehmend komplexer wird, ist diese Aufgabe schwierig und zeitaufwendig. Eine Automatisierung dieser „Ersten Hilfe“ stellt damit vor allem eine Entlastung für die zuständigen Experten dar. Sie werden von Routineanfragen und –aufgaben entlastet und haben mehr Zeit sich

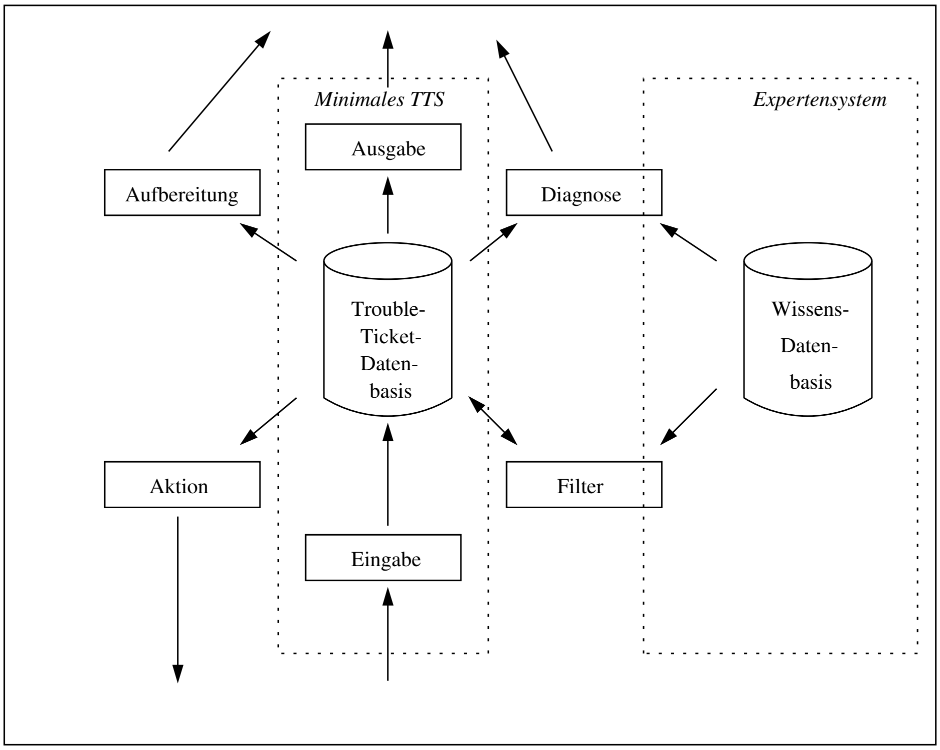

Abbildung 2.2: Aufbau eines Trouble–Ticket–Systems *[HAN 99a, S. 310]*

intensiver komplexen Problemen und Fehlern zu widmen. Somit wird auch für das Netz die Gesamtqualität erhöht, da in hohem Maße komplexe und komplizierte Fehler schneller behoben werden können.
In seiner benutzergeführten Fehlerlokalisierungmöglichkeit kann der Anwender selbst Diagnoseaktionen durchführen. Auf diese Weise wird eine Historie erstellt und formalisiert in Trouble–Tickets eingetragen. Diese gesammelten Informationen stehen dem Experten dann zur weiteren Problemlösung in qualitativ einheitlicher und hochwertiger Weise zur Verfügung und garantieren damit eine schnellere Bearbeitung und Fehlerbehebung.
Dem Intelligent Assistant liegt für die Durchführung und Anwendung das Expertenwissen in Form von Entscheidungsbäumen zu Grunde.
Das Netzmanagement profitiert vom Einsatz Künstlicher Intelligenz und ihrer Techniken. Dies soll in einer allgemeinen Einführung in die Expertensysteme erläutert werden.

2.2 Expertensysteme

Die Phasen des Fehlermanagements verdeutlichen die Korrelation zu Expertensystemen. Kurz gesagt „Fault Management is Decision Making“ *[Thij 87]*. In der zweiten Phase, der Fehlerkompensierung, kann die Auswahl einer Aktion durch den Operateur als Unsicherheit bezeichnet werden. Im Diagnoseschritt wird diese Unsicherheit durch die entsprechenden Maßnahmen weiter reduziert. In der Korrektur des Fehlers wird als Ergebnis deutlich, wie gut die Ent-

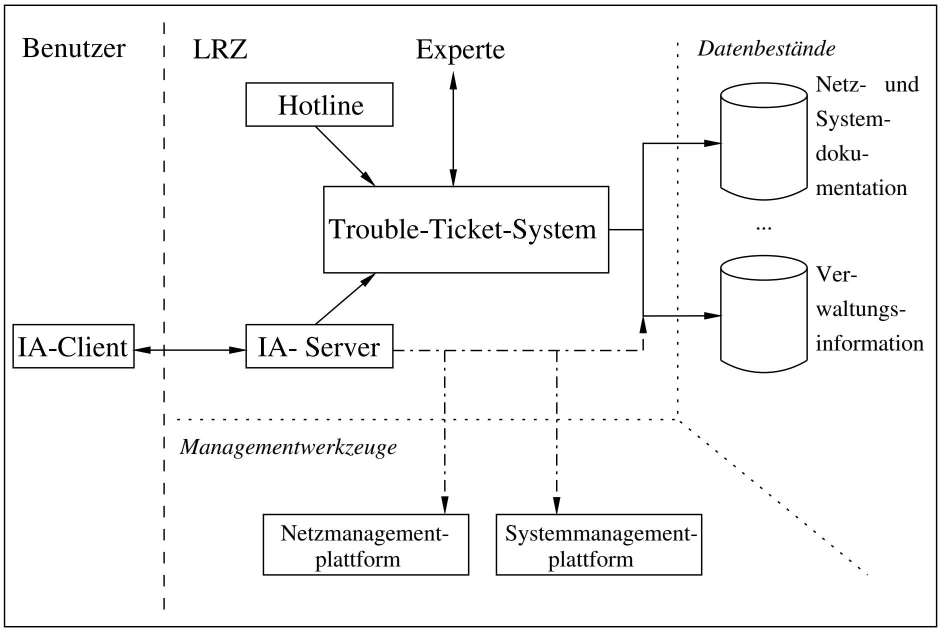

Abbildung 2.3: Integration des Intelligent Assistant in die Managementumgebung

scheidungsfindung war, indem der Fehler teilweise oder komplett behoben ist.
Die Annäherung eines Expertensystems an die Fehlerdiagnose ist damit intuitiv, da Symptome explizit mit den Ursachen in Zusammenhang gebracht werden können.
Das Einsatzgebiet der Expertensysteme liegt überwiegend im Gebiet der heuristischen Diagnostik und hier in der Wartung und Reparatur von Geräten und in der Prozeßüberwachung.

2.2.1 Charakterisierung

Das Gebiet der Wissenrepräsentation und Inferenz umfaßt einen zentralen Bereich des Gebietes der Künstlichen Intelligenz und der Kognitionswissenschaft. Es behandelt einerseits die Fragen nach einer formalen Beschreibung von Wissen jeglicher Art, besonders unter dem Aspekt einer maschinellen Verarbeitung in modernen Computern. Andererseits versucht es, das Alltagsschließen des Menschen so zu formalisieren, daß logische Schlüsse auch von Maschinen ausgeführt werden könnten.
In Expertensystemen wird das Spezialwissen und die Schlußfolgerungsfähigkeit qualifizierter Fachleute auf eng begrenzten Aufgabengebieten nachgebildet *[Pupp 91]*. Die Experten setzen dabei die Vorgehensweise in ihren Problemlösungsstrategien aus ihren bisher gewonnenen Erfahrungen und Einzelerkenntnissen zusammen. Diese werden selektiert und in passender Anordnung eingesetzt. Expertensysteme sind der Versuch, diese Strategien auf eine Menge von formalisierten und maschinenverarbeitbaren Operationen abzubilden.

2.2.2 Wissensrepräsentation

Die Wissensrepräsentation in Expertensystemen kann auf differente Weise erfolgen. Hier sollen die im Intelligent Assistant implementierte Repräsentationsform der Regeln und die in den

Bayesschen Netzen interpretierte Form des probabilistischen Schließens formal kurz vorgestellt werden.

Regeln

Regeln sind die am weitesten verbreitete Wissensrepräsentation in Expertensystemen. Sie bestehen aus einer Vorbedingung und einer Implikation bzw. einer Aktion. *Wenn A, dann gilt B*, oder *Wenn A, dann führe B aus.* Im Gegensatz zur konventionellen Programmierung legt der Experte mit einer Regel nur fest, was in einem beschriebenen Zusammenhang getan wird, während die Reihenfolge der Regelausführung von der Gültigkeit der Regel bzw. der Aktion bestimmt wird. Dies bleibt einem sogenannten Regelinterpreter überlassen. Dieser geht von einer vorhandenen Datenbasis aus und wählt sich aus den anwendbaren Regeln mittels einer Konfliktlösungsstrategie eine Regel aus, deren Aktionsteil ausgeführt wird. Dabei kann die Datenbasis verändert werden. Dieser Zyklus wird so lange wiederholt, bis ein Terminierungskriterium erfüllt oder keine Regel mehr anwendbar ist. Das Terminierungskriterium im Intelligent Assistant ist die Erstellung eines Trouble–Tickets.
Häufig verwendete Konfliktlösungsstrategien sind die Auswahl nach der *Reihenfolge* oder nach *Spezifität* (z.B. ist die Regel $A \;\&\; B \rightarrow C$ spezifischer als die Regel $A \rightarrow C$, da die Bedingung B zusätzlich gilt). Die Auswahl nach *Aktualität* geht von der Neuigkeit der Vorbedingungen einer Regel aus und aufgrund von *Zusatzwissen* kann ebenso eine Auswahl getroffen werden. Im einfachsten Fall besteht die Vorbedingung einer Regel aus nur einer Aussage. *[Görz 95]*

Probabilistisches Schließen

In vielen Anwendungsbereichen der Expertensysteme sind die Fakten bzw. das Wissen inhärent unsicher. Eine Aussage wird für wahrscheinlich gehalten oder man weiß nichts darüber. Um diesen Zwischenraum im Anwendungsbereich der Expertensysteme darzustellen, bedarf es einer Erweiterung der Wissensrepräsentation und der Problemlösungsstrategien.
Ein Ansatz dafür ist das probabilistische Schließen. Jede Aussage wird mit einer Wahrscheinlichkeit, die den Grad der Unsicherheit repräsentiert, bewertet. Dabei ist nach der Herkunft der Wahrscheinlichkeiten zu unterscheiden. Stammen die Angaben aus statistisch hergeleiteten Angaben, spricht man von probabilistischem Schließen. Werden die Wahrscheinlichkeiten von Experten geschätzt, wird dies in die Kategorie des unsicheren Schließens mit Evidenzen oder Sicherheitsfaktoren eingeordnet.[1] Unter Evidenz versteht man Annahmen und Aussagen, die sicher oder beobachtbar sind und damit über eine gewisse Sicherheit verfügen.
Voraussetzung für die Anwendbarkeit des probabilistischen Schließens und der dazugehörigen fundierten statistischen Verfahren ist die Unabhängigkeit der Symptome sowie deren Vollständigkeit und der wechselseitige Ausschluß der Diagnosen. Dies kann in den meisten Fällen nicht garantiert werden. Die Mechanismen für die Repräsentation unsicheren Wissens stammen deshalb überwiegend aus dem Bereich des unsicheren Schließens.
Das wichigste Einsatzgebiet in Expertensystemen für den Umgang mit Unsicherheiten stellt die Diagnostik dar. Die Bewertungen der Diagnosen basieren dabei auf einem einheitlichen Algorithmus:

1. Alle Diagnosen werden mit a priori (unbedingten) Wahrscheinlichkeiten belegt.

[1] in dieser Arbeit werden im Folgenden unsicheres und probabilistisches Schließen gleich bewertet, da im IA die Wahrscheinlichkeiten sowohl geschätzt werden als auch aus Statistiken stammen

2. Für jedes Symptom wird die Wahrscheinlichkeit aller Diagnosen entsprechend der bedingten Wahrscheinlichkeiten modifiziert.

3. Die wahrscheinlichste Diagnose wird selektiert.

Zur Anwendung benötigt man alle Berechnungen und Abschätzungen der bedingten Wahrscheinlichkeiten und sämtliche a priori Wahrscheinlichkeiten.

2.3 Verarbeitung unsicheren Wissens

2.3.1 Entscheidungen unter Unsicherheit

Aus den Expertensystemen ist bekannt, daß Wissen oftmals unvollständig ist. In der realen Welt werden in vielfältiger Weise Entscheidungen unter Unsicherheit getroffen. Die Vorbedingungen einzelner Aktionen sind nicht bekannt. Dennoch muß agiert und reagiert werden. Dazu stehen in der Simulation der Expertensysteme verschiedene Aktionen zur Auswahl. Im Intelligent Assistant sind dies die Fragen, die dem Benutzer gestellt werden und die Tests, die durchgeführt werden können.
Jede einzelne Aktion kann zu verschiedenen Ergebnissen führen und damit die Wahrscheinlichkeiten und im gleichen Schritt auch die Unsicherheit bzw. Sicherheit in Richtung einer Aktion verschieben. Vergleichbar im Intelligent Assistant, wenn der Test *traceroute* erfolgreich durchgeführt wurde, erübrigt sich ein *ping*.
Diese Aktionen sind auch mit Kosten behaftet, die durch die Experten oder den Entscheidungsträger wiederum subjektiv eingeschätzt werden können. Im oben bezeichneten Fall ist der Test *traceroute* mit höheren Kosten verbunden.
Die Ergebnisse wiederum haben verschiedenen Nutzen, der ebenfalls subjektiv eingeschätzt werden kann. *Traceroute* liefert im Gegensatz zu *ping* mehr Informationen und besitzt damit einen größeren Nutzen.
Die beste Entscheidung, die hier rational getroffen werden kann, ist eine Aktion zu wählen, die den größten zu erwartetenden Gesamtnutzen besitzt, aber in einem gewissen Kostenrahmen bleibt.
Die Entscheidungstheorie wird hier durch die Kopplung der Nutzentheorie und der Wahrscheinlichkeitstheorie ausgedrückt.

2.3.2 Theorem von Bayes

Die Wissensrepräsentation durch probabilistisches Schließen modelliert genau diese Entscheidungen unter Unsicherheit. Der Grad der Überzeugung, ob eine Aussage wahr oder falsch ist, wird durch den Einsatz von Wahrscheinlichkeiten ausgedrückt.
Dabei basiert diese Darstellungsart auf dem *Theorem von Bayes*, welches hier kurz erläutert wird.

Unbedingte Wahrscheinlichkeiten

$P(A)$ bezeichnet die *unbedingte* oder *a priori* Wahrscheinlichkeit, daß ein Ereignis A eintreten wird. Hierfür sind keine zusätzlichen Informationen notwendig.
Die Zufallsvariable eines Ereignisses kann nicht nur boolesche Werte wie *wahr* oder *falsch*

Verbundwahrscheinlichkeit			
	Husten	¬ Husten	$\sum$
Bronchitis	0.04	0.01	0.05
¬ Bronchitis	0.16	0.79	0.95
$\sum$	0.2	0.8	1.0

Tabelle 2.1: Verbundwahrscheinlichkeiten

annehmen, sondern beliebig skaliert werden. Im Intelligent Assistant kann z.B. im Durchsatzproblem die Skalierung über mehrere Wertebereiche des Durchsatzes erfolgen.

Bedingte Wahrscheinlichkeiten

Neue Information, die bei der a priori Wahrscheinlichkeit nicht notwendig war, kann die Wahrscheinlichkeit verändern.
$P(A \mid B)$ bezeichnet dabei die *bedingte* oder *a posterio* Wahrscheinlichkeit von A, unter Beobachtung von B. Das Ereignis oder die Beobachtung B wird auch als Evidenz bezeichnet.
Die bedingten Wahrscheinlichkeiten ergeben sich per Definition aus den unbedingten Wahrscheinlichkeiten (falls $P(B) > 0$):

$$P(A \mid B) = \frac{P(A \wedge B)}{P(B)}$$

- Produktregel: $P(A \wedge B) = P(A \mid B)P(B)$
- Äquivalent: $P(A \wedge B) = P(B \mid A)P(A)$
- $P(A \mid B) + P(\neg A \mid B) = 1$
- A und B heißen unabhängig voneinander, falls
  $P(A \mid B) = P(A)$ und $P(B \mid A) = P(B)$,
  dann gilt: $P(A \wedge B) = P(A)P(B)$

Verbundwahrscheinlichkeit

Ein atomares Ereignis ist eine Zuweisung von Werten an alle Zufallsvariablen $(X_1, \ldots, X_n)$ in einer Tabelle.
Diese Verbundwahrscheinlichkeitsverteilung $P(X_1, \ldots, X_n)$ weist jedem Ereignis eine Wahrscheinlichkeit zu. Dies soll an einem Beispiel aus der Medizin, dem häufigsten Einsatzgebiet von Expertensystemen mit Bayesschen Netzen, in *Tabelle 2.1* im Folgenden erläutert werden.

Alle atomaren Ereignisse sind disjunkt und damit ist die Summe über alle Ereignisse 1.
Damit lassen sich alle für eine Berechnung interessanten Ereignisse aus den Verbundwahrscheinlichkeiten errechnen.

$$P(Bronchitis \mid Husten) = \frac{P(Bronchitis) \wedge P(Husten)}{P(Husten)} = \frac{0.04}{0.04 + 0.16} = 0.2$$

Bayessche Regel

Aus der oben bezeichneten Produktregel und ihrem Äquivalent läßt sich durch Gleichsetzen berechnen:

$$P(A \mid B)P(B) = P(B \mid A)P(A)$$

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Für mehrwertige Variablen gilt:

$$P(Y \mid X) = \frac{P(X \mid Y)P(Y)}{P(X)}$$

Diese Gleichung wird von überwiegend allen probabilistischen Systemen verwendet.

Die allgemeine Bayessche Regel mit zusätzlichen Evidenzen:

$$P(Y \mid X, E) = \frac{P(X \mid Y, E)P(Y \mid E)}{P(X \mid E)}$$

Daraus lassen sich unbekannte Wahrscheinlichkeiten wie $P(Y \mid X)$ durch bekannte Wahrscheinlichkeiten berechnen.

Anwendung der Bayesschen Regel

Das Theorem von Bayes eignet sich dazu, aus den a priori Wahrscheinlichkeiten $P(A_i)$ einer Menge von Diagnosen und aus den bedingten Wahrscheinlichkeiten $P(B_j \mid A_i)$, der Häufigkeit des Auftretens eines Symptoms bei Vorhandensein einer Diagnose, die wahrscheinlichste Diagnose unter der Annahme der Symptome $B_1 \ldots B_m$ zu berechnen.
Dies soll an einem Beispiel aus der Medizin deutlich werden:
Die Statistiken zeigen folgende Wahrscheinlichkeiten:

$$\begin{aligned}
P(\text{Bronchitis}) &= 0.05 \\
P(\text{Husten}) &= 0.2 \\
P(Husten \mid Bronchitis) &= \tfrac{0.04}{0.05} = 0.8
\end{aligned}$$

Im Beispiel bedeutet $P(\text{Husten} \mid \text{Bronchitis}) = 0.8$, daß ein Bronchitispatient in 80% der Fälle auch Husten hat.
Dann gilt:

$$P(Bronchitis \mid Husten) = P(Bronchitis) * \frac{P(Husten \mid Bronchitis)}{P(Husten)} = 0.05 * \frac{0.8}{0.2} = 0.2$$

Die Wahrscheinlichkeit, daß ein Patient mit Husten Bronchitis hat, ist 20%.
In diesem Fall wird die Wahrscheinlichkeit für nur ein Symptom, Husten, berechnet. Wird die Anzahl der Symptome vergrößert, müßte man für jede relevante Symptomkonstellation Korrelationen zu Diagnosen bestimmen. Dies würde zu einer *kombinatorischen Explosion [Pupp 91]* führen. Beispielsweise müßten für 50 Symptome bereits 2^{50} Kombinationen berechnet werden.

Deshalb nimmt man in der Anwendung des Bayesschen Theorems an, daß die Symptome, wenn sie nicht direkt in derselben Diagnose auftreten, voneinander unabhängig sind.
Dafür sind einige Voraussetzungen erforderlich:

- Unabhängigkeit der Symptome untereinander. Dies wird umso kritischer, je mehr Symptome betrachtet werden.
- Wechselseitiger Ausschluß der Diagnosen.
- Fehlerfreie und vollständige Statistiken zur Herleitung der a priori Wahrscheinlichkeiten.
- Konstanz der Wahrscheinlichkeiten. Dies ist problematisch, da z.B. neue Diagnosemethoden oder eine andere Arbeitsteilung zu neuen Vorselektionen und damit zu anderen a priori Wahrscheinlichkeiten führen können.

Ein exakter Umgang mit Wahrscheinlichkeiten ist in den meisten Anwendungsbereichen nicht möglich. Durch Zusatzwissen können die Fehlerquellen im Theorem von Bayes verringert werden. Im Intelligent Assistant bieten sich die weiter unten aufgeführten Informationsquellen und Werkzeuge an. Mit jedem zusätzlichen Mechanismus wird andererseits die theoretische Absicherung schwieriger.

2.4 Bayessche Netze

In diesem Kapitel wird kurz in die Bayesschen Netze eingeführt. Zur intensiveren Literaturstudie sei auf *[Klin 98]* hingewiesen.
Bayessche Netze bilden eine Schnittstelle zwischen statistischen Verfahren und Expertensystemen. Sie modellieren die Welt im Bayesschen Sinne, indem Ereignisse mit Wahrscheinlichkeiten belegt werden. Dazu werden Merkmale in eine Abhängigkeitsstruktur eingebettet, die in der Form eines gerichteten Graphen repräsentiert wird. Die Bestimmung dieses Graphen anhand von (diskreten, vollständigen) Daten mit partiellem oder gänzlich ohne Vorwissen über die Struktur des Graphen und der zugehörigen Wahrscheinlichkeitsverteilungen ist eine der entscheidenden Fragestellungen bei der Anwendung von Bayesschen Netzen.

2.4.1 Repräsentation

Bayessche Netze werden auch als *bayesian belief networks, probabilistic networks, causal networks* oder als *kausalprobabilistische Netze* bezeichnet. Sie repräsentieren konditionale Unabhängigkeiten und Verbundwahrscheinlichkeiten.

Struktur

Die Zufallsvariablen, die Ereignisse oder Entscheidungen repräsentieren, bilden die *Knoten* des Bayesschen Netzes.
Verbunden sind diese mit *gerichteten Kanten*, die den direkten Einfluß in Pfeilrichtung symbolisieren. Der Weg zeigt von der Ursache zur Wirkung.
Mit jedem Knoten des Netzes wird eine *Tabelle der bedingten Wahrscheinlichkeiten (CPT)* assoziiert, die den Effekt der Eltern auf den Knoten quantifiziert.
Das Netz bildet einen *gerichteten azyklischen Graphen (directed acyclic graph, DAG).*

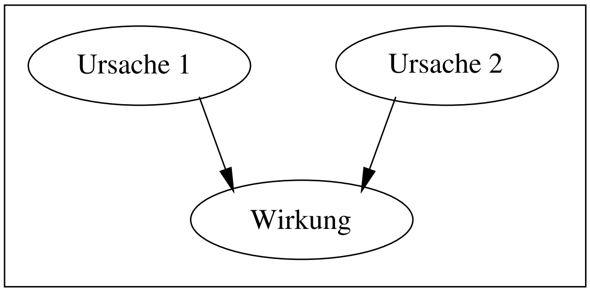

Abbildung 2.4: Ausschnitt der Grundstruktur eines Bayesschen Netzes

Konditionale Unabhängigkeit

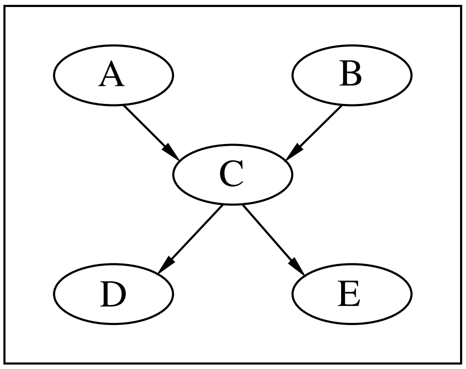

Abbildung 2.5: konditionale Unabhängigkeit

Intuitiv besagt der Pfeil in *Abbildung 2.5* von C nach E und das Nichtvorhandensein weiterer Pfeile nach E, daß C die einzige direkte Ursache von E ist. Dies bedeutet, daß E, gegeben C, bedingt unabhängig von allen anderen Variablen ist:

$$P(E \mid C, A) = P(E \mid C)$$

Bayessche Netze können als Menge von Unabhängigkeitsannahmen aufgefaßt werden.

Verbundwahrscheinlichkeit

In der Verarbeitung unsicheren Wissens wurde die Verbundwahrscheinlichkeit eingeführt. Nach der Produktregel gilt für die Wahrscheinlichkeit eines atomaren Ereignisses mit den Variablen $X_1, \ldots, X_n$ und den Werten $x_1, \ldots, x_n$:

$$\begin{aligned} P(x_1, \ldots, x_n) &= P(x_n \mid x_{n-1}, \ldots, x_1) \ldots P(x_2 \mid x_1) P(x_1) \\ &= \prod_{i=1}^{n} P(x_i \mid x_{i-1}, \ldots, x_1) \end{aligned}$$

Dabei sind $Eltern(X_i) \subseteq \{x_{i-1}, \ldots, x_1\}$.
Dann gilt:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Eltern(x_i))$$

Jeder Knoten ist damit unabhängig von den Vorgängern, bei gegebenen Eltern.
Im Beispiel bedeutet dies:

$$P(E \mid D, C, B, A) = P(E \mid C)$$

Mit der Netztopologie des Bayesschen Netzes und den Tabellen der bedingten Wahrscheinlichkeiten können die Verbundwahrscheinlichkeiten berechnet werden.

2.4.2 Einsatzgebiete

Die von Judea Pearl 1988 entwickelten Bayesschen Netze werden in Gebieten eingesetzt, die den Einsatz von Wahrscheinlichkeiten zur Modellierung des Grades von Überzeugung erfordern.
Den größten Bereich stellen die medizinischen Diagnosen. Hier gibt es für eine Vielzahl von Erkrankungen bereits kommerziell entwickelte und eingesetzte Systeme. Das System PATHFINDER deckt z.B. ca. 60 Lymphknotenkrankheiten und 100 Symptome und Testergebnisse ab. Weitere Anwendungsbereiche sind das Erlernen eines Grundrisses oder das Erlernen einer Sprache. In der heuristischen Suche wird auch auf Grundlage von Bayesschen Netzen verfahren.

2.4.3 Beispiel

Die aufgeführten Beispiele belaufen sich auf eine Größenordnung von 100 bis 1000 Knoten. Deshalb soll hier an einem kleinen Beispiel die Arbeits– und Anwendungsweise eines Bayesschen Netzes erläutert werden.
In *Abbildung 2.6* handelt es sich um den Alarm einer Alarmanlage, der entweder durch einen Einbruch oder durch ein Erdbeben ausgelöst wird. Der Alarm wiederum veranlaßt 2 Personen (A und B), einen Telefonanruf zu tätigen.
Die Wahrscheinlichkeiten sind in den Tabellen angegeben, wobei hier nur die positiven Wahrscheinlichkeiten berücksichtigt wurden. Die Wahrscheinlichkeiten der Komplementärereignisse ergeben sich aus $P(\neg X) = 1 - P(X)$, $(P(X) > 0)$.

2.4.4 Inferenzmechanismen

Durch Instantiieren einiger Variablen werden die Beobachtungen (Evidenzen) eingetragen und andere Knoten können abgefragt werden.
$P(Einbruch \mid Person1) =?$
Unter der Annahme, daß der Alarm sehr zuverlässig ist und Person A in 9 von 10 Fällen anruft, dann:
$P(PersonA \mid Einbruch) \sim 0.8$.
Im Beispiel ist die Evidenz, daß ein Einbruch nur alle 1000 Tage stattfindet, Person A aber

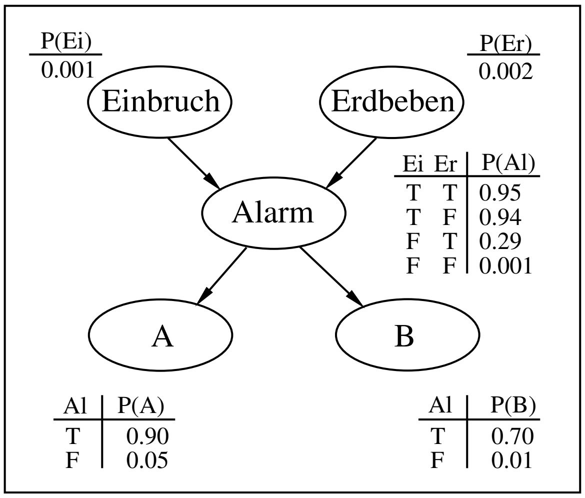

Abbildung 2.6: Beispiel Bayessches Netz

50 x in 1000 Tagen zum Telefonhörer greift.

$\Rightarrow P(Einbruch \mid PersonA)$

$$= \frac{P(Einbruch \wedge PersonA)}{P(PersonA)}$$

$$= \frac{P(PersonA \mid Einbruch)P(Einbruch)}{P(PersonA)}$$

$$= \frac{0.8 * 0.001}{0.05} = 0.016$$

Die Wahrscheinlichkeit eines Einbruchs, wenn Person A anruft, ist sehr gering.
Unter Inferenzmechanismen sind Schlußfolgerungen zu verstehen, die man in jegliche Richtung treffen kann. Die oben berechnete Schlußfolgerung entspricht einer diagnostischen Inferenz. Die Schlußrichtung ist von den Effekten zur Ursache. In den Bayesschen Netzen werden vier Inferenzmechanismen unterschieden (siehe *Abbildung 2.7*).

- *diagnostisch*: von Effekten zu Ursachen, z.B. $P(Einbruch \mid PersonA)$
- *kausal*: von Ursachen zu Effekten, z.B. $P(PersonA \mid Einbruch)$
- *interkausal (explaining away)*: zwischen Ursachen, z.B. $P(Einbruch \mid Alarm)$ und $P(Einbruch \mid Alarm, Erdbeben)$
- *gemischt*: Vermischung von positiven und negativen Ereignissen, z.B. $P(Alarm : \mid PersonA, \neg Erdbeben)$

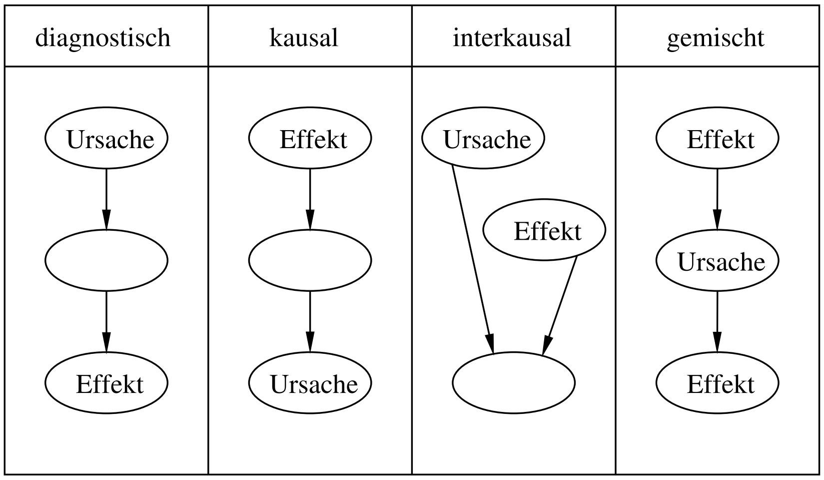

Abbildung 2.7: Inferenztypen

2.4.5 d–Separation

Aus den verschiedenen Inferenzmechanismen ergeben sich verschiedene Arten, wie Schlußfolgerungen stattfinden. Die d–Separation ist ein Mechanismus, der unabhängige Knoten in ein eigenes Bayessches Netz abtrennen kann (siehe *Abbildung 2.8*). Die Inferenz wiederum kann hier in kleinerem Rahmen stattfinden.
Eine Menge von Evidenzknoten E d–separiert die Mengen X und Y, falls jeder (ungerichtete) Pfad von X nach Y durch Z blockiert ist.

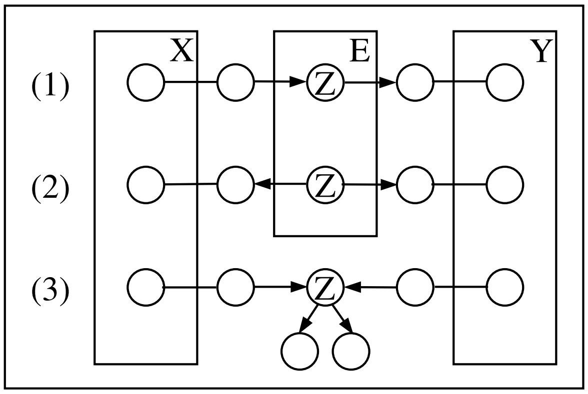

Abbildung 2.8: Unabhängigkeiten: d–Separation

$\Rightarrow$ X und Y sind bedingt unabhängig, gegeben E

Ein Pfad von X nach Y ist durch E blockiert, falls es einen Knoten Z auf dem Pfad gibt, so daß

1. Z in E liegt und Z einen eingehenden und einen herausführenden Teilpfad besitzt, oder
2. Z in E liegt und beide Teilpfade sind herausführend, oder

3. Z nicht in E liegt, beide Teilpfade eingehend sind und kein Nachfolger von Z in E liegt.

Wie in dieser Arbeit bereits aufgezeigt, bedarf die Anwendung von Bayesschen Netzen vor allem der Unabhängigkeit seiner Ereignisse und Tests. Um weitgehende Unabhängigkeiten zu erreichen und auf minimale Wahrscheinlichkeiten zu reduzieren, wird das Mittel der d–Separation verwendet.

Kapitel 3

Analyse der bestehenden Situation

In dieser Arbeit werden zwei Gebiete der Informatik miteinander verknüpft. Einmal der technische Aspekt des Netzmanagements, in dem Dynamik und Veränderungen eine große Rolle spielen. Hier wird der Intelligent Assistant als Werkzeug eingesetzt. Auf der anderen Seite der theoretische Bereich der Expertensysteme und der Wahrscheinlichkeitstheorie, insbesondere die Bayesschen Netze, die den Aspekt des unsicheren Wissens und dessen Modellierung vertreten.
Dazu werden die Problemstellung und die sich daraus ergebenden Anforderungen, die das Netzmanagement und der *Help Desk* an ein Werkzeug wie den Intelligent Assistant und dessen Expertenwissen stellen, erläutert.

3.1 Intelligent Assistant

Das Netzmanagementsystem am LRZ, in welchem der Intelligent Assistant eingesetzt wird, bildet einen Gesamtkomplex aus Werkzeugen, Datenbank, Expertensystem und graphischer Benutzerführung.
In diesem Spannungsfeld erwachsen für entscheidungstheoretische Grundlagen und die Modellierung unsicheren Wissens spezielle Anforderungen, die sich aus zwei Richtungen ergeben. Der Betreiber eines Rechnernetzes ist an den technischen Durchführungs– und Lösungsmöglichkeiten des Managements interessiert, der gesamte *Help Desk* an der Anwendung und einer einfachen Handhabung.

Im Hinblick auf eine leichte, komfortable, unproblematische Umstrukturierung des Expertenwissens im Intelligent Assistant werden die Anforderungen des Netzmanagements und des *Help Desk* im Folgenden analysiert.

3.1.1 Architektur

Der Intelligent Assistant ist am LRZ in eine Client–Server Managementumgebung eingebettet (*siehe Abbildung 2.3)*. Startet ein Host über eine graphische Schnittstelle (z.B. WWW–Browser) den Assistant, wird der Host zum Intelligent Assistant Client. Die Interaktion zwischen dem Benutzer und dem Intelligent Assistant Server wird aktiviert. Der IA Server besitzt Schnittstellen zu Netzmanagementwerkzeugen und Datenbanken, die managementrelevante Informationen enthalten. Der Vorteil dieser Client–Server Architektur liegt vor allem darin, daß eine hohe Anzahl von Benutzern betreut werden kann und daß sie eine gute Skalierbarkeit des Assistenten garantiert. Die vom Benutzer durchgeführten Aktionen werden auf dem Client lokal durchgeführt.

3.1.2 Klassifikation der Problembereiche

Derzeit ist der Intelligent Assistant für die Problembereiche Verbindung, Mail und Durchsatz realisiert.

Allgemeiner Ablauf

Die erste Maske, die dem Benutzer angezeigt wird, dient der Identifizierung. Derzeit findet keine Authentifizierung des Benutzers statt. Der Benutzer gibt in den vorgesehenen Eingabefeldern seinen Namen und die E–Mailadresse ein. Diese Angaben sind nötig, um den Nutzer später für Rückfragen der Experten erreichen zu können und ihm automatisch den Stand der Bearbeitung seines Trouble–Tickets zu melden. In der nachfolgenden Maske wählt der Benutzer bereits einen der drei Problembereiche aus.

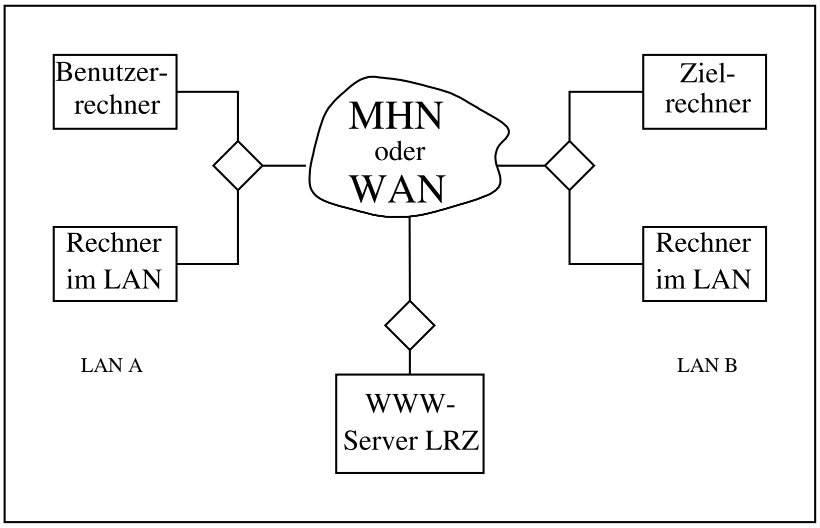

Abbildung 3.1: Testumfeld im Verbindungsproblem

Verbindungsproblem

Wählt der Benutzer diesen Problemfall, wird er gebeten, die Rechnernamen oder die IP–Adressen des betroffenen Clientrechners und des zweiten beteiligten Rechners einzugeben. Im Falle des Rechnernamens wird im Hintergrund der Befehl *nslookup* auf dem Domain Name Server (DNS) ausgeführt und die zugehörige IP–Adresse ermittelt. Im folgenden werden durch die Testroutinen *ping* und *traceroute* die Erreichbarkeiten überprüft. Entscheidet sich der Benutzer für den manuellen Testbereich, so wird er durch eine genaue Anleitung aufgefordert, die Diagnose selbst durchzuführen und anschließend das Ergebnis mit den Vorgaben im Intelligent Assistant zu vergleichen. Im automatischen Testbereich werden vom WWW–Server aus die oben genannten Testroutinen gestartet. Anschließend kann der Benutzer mit den bis dahin gesammelten Daten ein Trouble–Ticket erstellen lassen.

Durchsatzproblem

In diesem Fehlerfall wird der Benutzer ebenfalls gebeten, Rechnernamen oder IP–Adressen einzugeben. Im folgenden ist die Information über den Zugang relevant, die als Auswahl zwischen Wählmodem, Standleitung oder Wide Area Network (WAN)–Zugang zur Verfügung steht. In der anschließenden Abfrage wird der Wert des momentanen Durchsatzes erfragt. Dieser ermittelte Wert wird mit dem für die Quell–IP–Adresse und den entsprechenden Routerport zu erwartenden Durchsatzwert verglichen. Ist der gemeldete „Fehlerdurchsatz“ nicht um das 1000fache geringer als der Sollwert, so liegt kein Durchsatzfehler vor. Der Benutzer kann hier ebenfalls ein Trouble–Ticket erstellen lassen.

Mailproblem

Meldet der Benutzer ein Problem in diesem Bereich, wird zuerst die Betriebssystemkonfiguration ermittelt. Der Nutzer hat die Wahl zwischen UNIX–Rechner im Münchener Hochschulnetz (MHN), PC im MHN mit TCP/IP, PC mit Modem über Point to Point Protocol (PPP) und einer sonstigen Konfiguration. Sofern der Meldende auf keinem UNIX–Rechnersystem arbeitet, ist der Experte üblicherweise an der Konfiguration des Mailsystems interessiert. In allen Fällen wiederum ist die Frage interessant, ob in dieser Rechnerkonfiguration die Mailfunktion bereits erfolgreich verwendet wurde. Da Konfigurationsänderungen am Host die Ursache für ein Mailproblem sein können, ist die Angabe des vollständigen Rechnernamens oder der IP–Adresse notwendig. Diese wird dazu verwendet, im Hintergrund per *traceroute* die Erreichbarkeit dieses Rechners zu testen.

Da ein Großteil der Benutzer in der Regel Mail verwendet, fallen die meisten Fehlermeldungen in diesen Bereich. Der Benutzer besitzt jedoch nicht das Fachwissen, bereits in diesem Anwendungsstadium zwischen einem Mail– oder Verbindungsproblem zu unterscheiden. In den überwiegenden Fällen erweist sich ein Mailproblem als Verbindungsproblem.

3.1.3 Realisierung am Leibniz–Rechenzentrum

Expertenwissen im Intelligent Assistant

Grundlage für das Expertenwissen in Entscheidungsbäumen bilden die Hilferufeliste und eine Liste, in welcher *Frequently Asked Questions (FAQ)* am LRZ aufgeführt sind. In der Hilferufeliste werden einzelne Themengebiete den einzelnen Mitarbeitern zugeordnet. Diese Liste ist notwendig, um bei Fehleranfragen oder Problemstellungen den entsprechenden Experten ausfindig machen zu können.
Die *FAQ*–Liste beinhaltet diejenigen Fragen, die am LRZ am häufigsten gestellt werden. Die Antworten auf diese Fragen wiederum stehen ebenfalls zur Verfügung.

Entscheidungsbaum im Intelligent Assistant

Die jeweils zuständigen Experten wurden in einer vorhergehenden Arbeit *[Mode 94]* bezüglich der Vorgehensweise zur Fehlerlokalisierung befragt. In dieser Wissensakquisition wurde für jedes Themengebiet anhand eines Fragenkataloges die gestellten Fragen und das entsprechend verwendete Werkzeug ermittelt. Aus dieser Struktur und den Ergebnissen der Befragung wurden sogenannte Entscheidungsbäume entwickelt. Der Entscheidungs– und Diagnosegraph ist im Intelligent Assistant über verknüpfte HTML–Seiten realisiert. Der Benutzer wird mit den oben erwähnten Eingabefeldern (Buttons, Textfelder) durch den Graphen geführt. Die nachfolgenden Seiten werden durch C–Programme dynamisch erzeugt und diese Programme wiederum durch CGI–Skripte angestoßen. Die Diagnoseregeln, die die Reihenfolge im Graphen bestimmen, sind in C–Code implementiert.

Struktur

Das Wissen der Experten wird in ein Regelwerk implementiert, das die computertechnische Repräsentation des menschlichen Denkens und Handelns in Grundzügen simulieren kann und soll.
Ein Entscheidungsbaum M wird definiert als ein Baum $T = (V, E, L)$ mit:

- Knotenmenge V,
- Kantenmenge $E \subseteq V \times V$ und
- Beschreibungsfunktion $L : V \cup E \to A \cup C \times \mathrm{N}$
  A ist die endliche Menge von Aktionen, C eine endliche Menge von Klassenbezeichnungen.

Die Grundstruktur eines Entscheidungsbaumes verläuft dabei nach dem *WENN–DANN*–Prinzip.
Die Fragen werden auf die Weise formuliert, daß Antworten nur mit *ja* oder *nein* erfolgen können. Wird eine gestellte Frage mit *ja* beantwortet, so folgt eine andere Aktion oder Frage oder ein anderer Test, als wenn die anfangs gestellte Frage mit *nein* beantwortet wurde. Für den Graphen ergibt sich eine baumartige Struktur, dessen Wurzel den Problembereich und dessen Blätter die erstellten Trouble–Tickets darstellen.

3.1.4 Beispiel: Entscheidungsbäume aus dem Intelligent Assistant

Durchsatzproblem

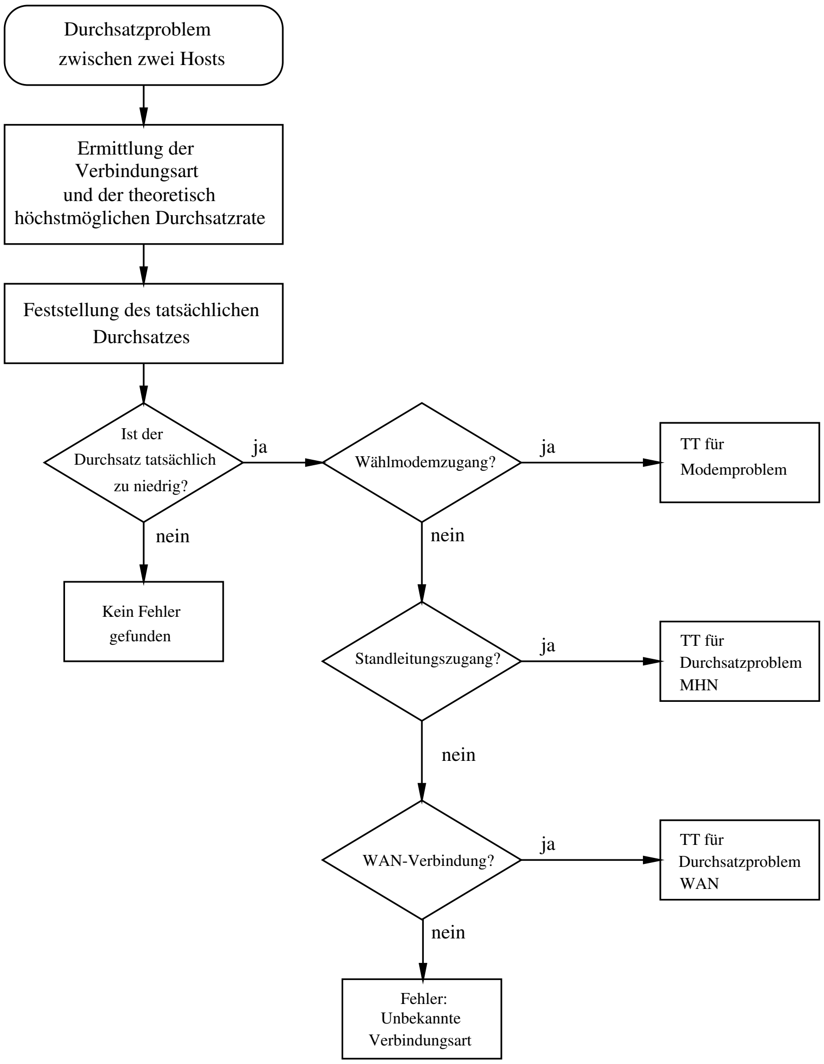

Abbildung 3.2: Entscheidungsbaum: Durchsatzproblem

Mailproblem

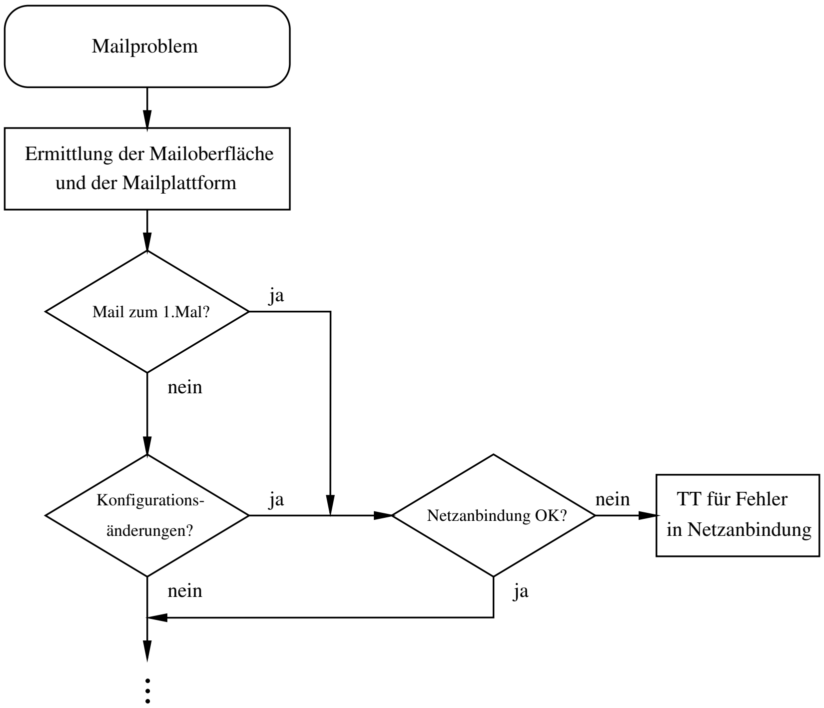

Abbildung 3.3: Entscheidungsbaum (Ausschnitt): Mailproblem

Verbindungsproblem

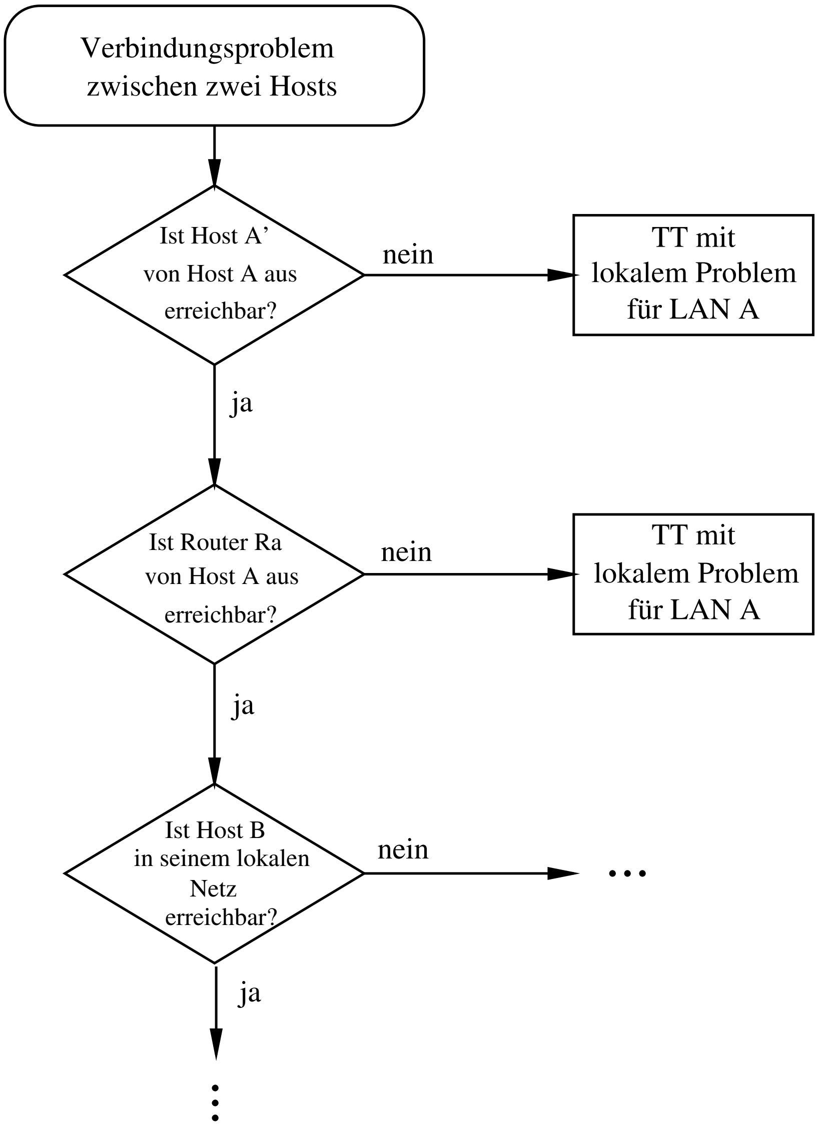

Abbildung 3.4: Entscheidungsbaum (Ausschnitt): Verbindungsproblem

3.1.5 Realisierte Schnittstellen

Im folgenden werden die Schnittstellen die der Intelligent Assistant anbietet aufgeführt.

Zum Benutzer

Im Intelligent Assistant wird die Benutzerschnittstelle in einer dynamischen Variante realisiert. Dabei wird der Benutzer zur Interaktion aufgefordert. Neben Einträgen in ein Formular kann der Benutzer Testaktionen und Datenbankabfragen durchführen.
Dem Benutzer wird die Funktionalität des Intelligent Assistant über eine WWW–Browser–

Schnittstelle zur Verfügung gestellt. Über diese Schnittstelle präsentiert sich der Entscheidungsbaum. Dabei bestimmt der Benutzer durch seine Antworten und Angaben den Verlauf der Diagnose. In Auswahlfenstern wird der Benutzer nach nötigen Informationen gefragt. Um Testaktionen durchzuführen, wird er in der lokalen Ausführung angeleitet und zu den Ergebnissen wiederum befragt. Testaktionen, die vom Intelligent Assistant Server aus gestartet werden, werden automatisch in eine Historie eingefügt. Der Benutzer bestimmt nach Durchlauf eines Entscheidungsbaumes, ob ein Trouble–Ticket erstellt wird.

Zu den Managementwerkzeugen

Die Durchführung verschiedener Aktionen und Tests, getriggert durch den Entscheidungsbaum, erfordert eine Schnittstelle zu Managementwerkzeugen. Realisiert ist dies im Verbindungsproblem. *Ping* und *traceroute* wurden als Testaktionen manuell in den Entscheidungsbaum eingebaut. Der Ablauf dieser Aktionen erfolgt automatisch.

Entwicklungsumgebung für Expertenwissen

Im Intelligent Assistant wird eine separate Entwicklungsumgebung zur Verwaltung von Expertenwissen (derzeit in Entscheidungsbäumen realisiert) zur Verfügung gestellt. Zu diesem Zweck existiert ein Java–Werkzeug, das es neben der manuellen Einpflege der Entscheidungsbäume mit einem Texteditor ermöglicht, diese auch graphisch zu erzeugen. Damit können Änderungen und Anpassungen unabhängig von der Standard–Programmierumgebung durchgeführt werden. Somit ist auch sichergestellt, daß in der Entwicklungsumgebung einzelne Wissensmodule flexibel ausgetauscht und erweitert werden können.

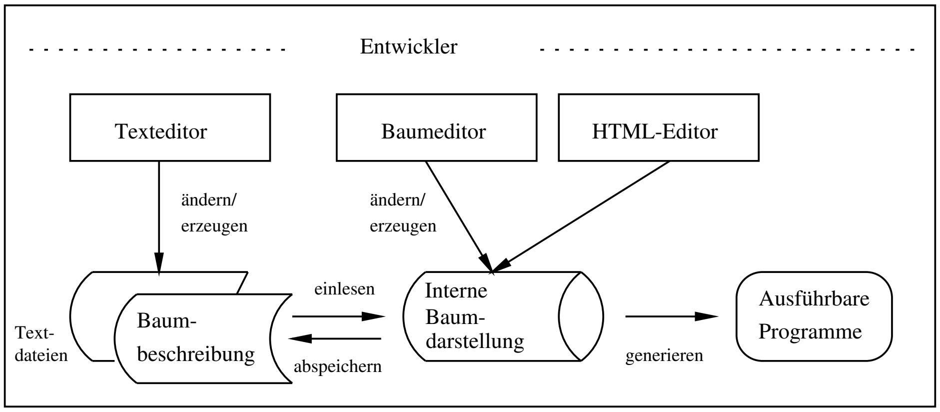

Abbildung 3.5: Entwicklungsumgebung für den Intelligent Assistant

3.2 Anforderungen aus dem Netzmanagement

3.2.1 Support

Für die Lokalisierung und Behebung von Fehlern in einzelnen Bereichen in einem komplexen und heterogenen System eines Kommunikationsdienstanbieters sind unterschiedliche Experten zuständig. In der Regel handelt es sich bei Experten um Einzelpersonen, deren Wissen sich

zumeist auf ein spezielles Fachgebiet bezieht. Um sämtliche Fachgebiete in aller Komplexität und Schwierigkeit abzudecken, bedarf es eines Zusammenschlusses der Experten. Die Arbeit der Experten sollte dabei aus Kosten– und Zeitgründen zu einem Großteil in der Behandlung von Problemen, die ihr Fachgebiet betreffen, liegen.
Ein Benutzer, der Hilfestellung bei Problemen benötigt oder einen Fehler meldet, trifft als erste Anlaufstelle auf den *Help Desk*. Als Organisationseinheit im Management vernetzter Systeme ist dieser überwiegend modular aufgebaut. Für bestimmte Fehler– oder Problembereiche gibt es jeweils zuständige Experten.
Die Experten versuchen, dem Benutzer, der eine Anfrage stellt oder ein Problem meldet, möglichst schnell und gezielt weiterzuhelfen. Die Verarbeitungsroutinen der Anfragen der Benutzer ähneln sich sehr stark. Kein Benutzer weiß von einer bereits behandelten vorhergehenden Problemstellung und will individuell behandelt werden. Für den Experten ist dies Routinearbeit. Die Experten, die im Rahmen des *Help Desk* dem Benutzer Hilfestellung leisten, werden durch den Intelligent Assistant von diesen Routineaufgaben befreit.
Zudem besteht die Möglichkeit, außerhalb der üblichen Arbeitszeiten Angestellte aus anderen Fachgebieten zu beschäftigen. Diese sollen aber Auskunft und Hilfestellung in allen Bereichen geben können. Deshalb wird ein Werkzeug zur Verfügung gestellt, welches es ermöglicht, ohne tiefgreifende Kenntnisse diese Maßnahmen durchführen zu können.
Die Aufteilung und Modularisierung des Hilfedienstes erfolgt sowohl personell als auch wissensspezifisch. Dies betrifft somit vor allem das dazugehörige implementierte Expertenwissen. Bezieht sich das Problem auf ein fremdes Fachgebiet, muß auch im Expertenwissen des Intelligent Assistant die Möglichkeit bestehen, eine automatische Erweiterung der Diagnosemöglichkeiten vorzunehmen. Zudem muß auch die Möglichkeit bestehen, neue Technologiebereiche zu integrieren.

3.2.2 Dienstspezifische Wissensmodule

Eine dezentrale Realisierung der Dienste in einem vernetzten und verteilten System wird häufig durchgeführt. Wird z.B. der E–Mail Dienst verteilt und deshalb auf mehreren E–Mail Servern realisiert, so übernimmt bei Ausfall eines Servers ein anderer Server diesen Dienst.
Das verteilte Erbringen eines Dienstes stellt für die Fehlerlokalisierung und Fehlerdiagnose eine erhebliche Schwierigkeit dar. Hier sind die Spezialisten betreffend des lokalen Dienstes sowie für verteilte Dienste nötig.
In der Realität sind diese beiden Wissensgebiete auf verschiedene Experten verteilt und das Spezialwissen wird um den Aspekt erweitert, daß es sich über ein gesamtes vernetztes System erstreckt. Hierbei sind zusätzliche Aufgaben notwendig, um die Dienstqualität des Netzes sicherzustellen und damit gleichzeitig die Qualität der Fehlerlokalisierung. Hohe Anforderungen werden hier an das implementierte Expertenwissen gestellt. Die Koordination der beiden Gebiete — verteilte Diensterbringung und der Dienst selbst — ist insbesondere in der Fehlerlokalisierung ein nennenswerter Faktor. Das Wissen um die auftretenden Fehler in einem derartigen Bereich muß durch Wissen in verteilten Anwendungen und Diensterbringungen ergänzt werden.

3.2.3 Schnittstellen zu Informationsquellen und Werkzeugen

Die realisierten Schnittstellen des Intelligent Assistant und seines Expertenwissens wurden bereits aufgeführt.

Ein Experte benutzt, in einem typischen Verlauf während einer Fehlerbearbeitung, von der ersten Meldung bis zur Behebung der Störung, verschiedene Werkzeuge und andere Informationsquellen. Die Werkzeuge liefern managementrelevante Informationen, die dem Experten in seinem Diagnoseverlauf hilfreich sind und ihn bei seinen Entscheidungen unterstützen. Ein Netzoperateur oder Experte, der die Fehlermeldungen eines Kunden bearbeitet, ist daran interessiert, weiteren Zugang zu möglichst vielfältigen und informationsreichen Quellen zu haben:

- Informationsquellen
  - Fehlerdokumentation
    In einem ersten Schritt zur Fehlerdiagnose sucht der Experte nach bereits dokumentierten ähnlich oder gleich gelagerten Fehlerfällen und deren Lösungswegen.
  - topologische Information
    Dies ist die Information über die physikalische Struktur und des Aufbaus eines Netzwerkes. Mit dieser Information kann der Experte die Lokalität eines Gerätes feststellen, z.B. an welchem Routerport in welchem LAN ein Hostrechner betrieben wird.
  - betriebliche Information
    Dies sind aktuelle Informationen, die der Kommunikationsdienstanbieter selbst eintragen kann. Wird z.B. ein Server oder eine andere Komponente im Netzwerk repariert, so wird deren Ausfallzeit bereits im voraus eingetragen. Bei Fehlermeldungen, die diese Komponenten betreffen, kann der Experte die Diagnose sofort stellen. Die Zuordnung der laufenden Anwendungen zu Geräten und sonstigen verwaltungstechnischen Hierarchien fällt ebenfalls in diese Rubrik.
  - Konfigurationsinformation
    Hier ist die physikalische Information hinterlegt, wie einzelne Netzkomponenten und Netzwerkverbindungen konfiguriert sind. Die Art des Gerätes, Standort, IP–Adresse, Routing.
  - Personendatenbank
    Diese Daten sind zweigeteilt. Einerseits die betreuenden Personen und Experten, die diverse Zuständigkeiten für bestimmte Fach– und damit Problemgebiete besitzen. Andererseits die Benutzerdaten, die für eine Problemlösung von Bedeutung sind, wie z.B. die Rechte eines Benutzers. Wird ein Fehler gemeldet, so kann der Experte feststellen, inwieweit der Benutzer den Anspruch hat, daß diese Fehlfunktion beseitigt wird.
- Werkzeuge
  Der Experte hat, wie schon erwähnt, auch Zugriff auf Werkzeuge, die managementrelevante Informationen liefern. Dabei kann er flexibel zwischen verschiedenen Werkzeugen entsprechend dem Problemfall wählen. Die Bedienung dieser Werkzeuge ist ihm vertraut, und er entscheidet sich je nach Schwierigkeitsgrad des Problemfalles für das qualitativ beste Werkzeug. Im Netzmanagement am LRZ wird überwiegend der HP OpenView Node Manager und im Systemmanagement HP OpenView IT/Operations eingesetzt.

Die Experten selbst stellen die Schnittstellen zwischen den Informationsquellen und den Werkzeugen dar. Das zusätzliche Wissen um Komponenten und Topologien und gewisse regelmäßige

Abläufe ist einer Bearbeitung der Fehlermeldungen förderlich. Da der Intelligent Assistant und das implementierte Expertenwissen einen zuständigen Experten simulieren, bestehen hohe Anforderungen an die Schnittstellen des Expertenwissens. Schnittstellen zu den oben aufgeführten Quellen und Werkzeugen sollen vorhanden sein und automatisch eingebunden werden. Weitere Werkzeuge sollen hinzugefügt werden können.

3.2.4 Plattformen

Die Erstellung des Expertenwissens in Bayesschen Netzen und der anschließende Einsatz muß möglichst plattformunabhängig erfolgen.
Die Diagnose eines gemeldeten Fehlers wird, auch wenn der Benutzer den Intelligent Assistant bedient, auf dem Intelligent Assistant Server durchgeführt. Dieser unterstützt gängige Plattformen wie UNIX und Windows NT.
Die Bayesschen Netze sollten aus Effizienz– und Übersichtlichkeitsgründen mit einer graphischen Oberfläche generiert werden. Dieses Front–End, eine Forderung die sich aus dem *Help Desk* ergibt muß plattformunabhängig einsetzbar sein.
Auch das Produkt zum Einsatz der Bayesschen Netze muß auf verschiedenen Plattformen eingesetzt werden können und über eine API–Schnittstelle verfügen.

3.2.5 Dynamik des Netzwerkes

Vernetzte Systeme besitzen eine hohe Dynamik. Die Netze werden an die jeweiligen Erfordernisse und Bedürfnisse der Nutzer angepaßt. Netz– und Systemkomponenten werden ausgetauscht oder neue Versionen einer Anwendungs– oder Systemsoftware installiert oder aktualisiert.
Damit werden hohe Anforderungen an die Darstellung hinsichtlich der Dynamik, der Anwendung und des Zugriffs auf das Expertenwissen gestellt.
Wird z.B. in einem Netzwerk eine neue Komponente eines anderen oder bereits verwendeten Herstellers installiert, können neuartige, bisher unbekannte Fehler auftreten. Hierunter sind sowohl Fehlersymptome zu verstehen, die die gleichen oder ähnlichen Fehlerursachen haben, oder auch Fehlermeldungen, die bisher unbekannt waren. Das Expertenwissen muß darauf entsprechend reagieren können und fähig sein, durch Rückkopplung mit Hilfe bisher aufgetretener und behobener Fehler das Wissen zu erweitern und damit den unbekannten Fehler zu diagnostizieren.

3.2.6 Zeitpunkt des Fehlers

Für eine Fehlerlokalisierung und Fehlerbehebung in einem Netzwerk ist der Zeitpunkt und die Dauer eines aufgetretenen Fehlers von Bedeutung. Das zusätzliche Wissen um diesen Zeitpunkt kann zur Fehlerbehebung beitragen. Ist z.B. dem Experten bekannt, daß ein Server zu bestimmten Zeiten ausgeschaltet oder neu installiert wird, so kann dem Benutzer eine schnelle und zufriedenstellende Antwort gegeben werden.

3.2.7 Verkehrsbelastung

Die Testroutinen wie *ping* oder *traceroute*, die der Intelligent Assistant zur Fehlerlokalisierung anstoßen kann, und auch die Benutzung des Intelligent Assistant selbst können unter Umständen neue Netzlast erzeugen. Dabei wird durch einen ungünstig durchgeführten Test nicht gewährleistet, daß ein Fehler vor allem schnell lokalisiert wird. Vielmehr wird durch neue Tests weiterer Verkehr hervorgerufen und damit das Netz zusätzlich belastet. Einen Test mit günstigen Konditionen auszuwählen, ist hier sinnvoll.

3.2.8 Geschwindigkeit

Die Schnelligkeit in der Fehlerlokalisierung, –diagnose und der anschließenden –behebung ist ein entscheidendes Kriterium für die Qualität eines Kommunikationsdienstanbieters. Dazu trägt ein automatisiertes Fehlererfassungsverfahren ebenso bei wie eine qualifizierte Fehlerbeschreibung wie sie der Intelligent Assistant mittels eines Trouble–Tickets erstellt. Hier können die entscheidenden Faktoren in der Fehlerdiagnose liegen, die durch weniger überflüssige Fragen und Tests schnell auf den Punkt kommt. Der Benutzer erfährt eine schnelle Problemlösung.

3.3 Anforderungen aus dem *Help Desk* Szenario

Wie bereits dargestellt, besteht der *Help Desk* eines Kommunikationsdienstanbieters überwiegend aus den Experten, die dem Benutzer eine erste Anlaufstelle in Problemfällen bieten. Zur Diagnose und Lösung derartiger Fälle benützen die Experten Hilfsmittel, die auch auf implementiertem Expertenwissen basieren. Um diese Werkzeuge schnell, effektiv und ohne lernintensive Schulungen erstellen, anwenden und bedienen zu können, werden Anforderungen an derartige Werkzeuge und das zugrundeliegende Expertenwissen gestellt. Hinsichtlich der Implementierung des gesamten Expertenwissens in Bayesschen Netzen werden die Anforderungen untersucht.

3.3.1 Wissensakquisition

Aufgrund der inhärenten Eigenschaften der Bayesschen Netze benötigt die Konstruktion eines oder mehrerer Bayesscher Netze das Wissen eines oder mehrerer Experten aus verschiedenen Bereichen. Traditionell werden die zu erstellenden und konstruierenden Netze von allen betroffenen Experten in großer Runde erarbeitet. Diese Vorgehensweise erfordert einerseits hohen Zeitaufwand und andererseits Kosten. Letzteres angesichts der Tatsache, daß Zeit, die Experten aufwenden müssen, sehr teuer ist.
Eine andere Methode ist, daß eine einzelne Person ein derartiges Netz in elektronischer Form produziert und dieses dann zur Bearbeitung und Überprüfung weiterreicht. Voraussetzung dafür ist wiederum, daß beide Beteiligten die gleiche Editiersoftware oder zumindest kompatible Werkzeuge für derartige Netze besitzen. Dieser Weg basiert auf elektronischem Kommunikationsaufwand und asynchroner Bearbeitung. Der Synergieeffekt, den ein reales Treffen zwischen Experten mit sich bringt, geht hier verloren.
Bayessche Netze entstehen durch Zusammenarbeit verschiedener Personenkreise. Menschen, die die Fähigkeit besitzen, Wissen zu akquirieren, den Experten Ihr Wissen zu „entlocken“, zu formulieren und zu formalisieren, zählen zu der bedeutendsten Gruppe. Eine weitere Gruppe besitzt das Wissen um die Feinheiten der Bayesschen Netze und deren Modellierung. Diejenigen Experten, deren Wissens– und Fachgebiet modelliert wird, stellen die dritte Personengruppe dar.
Zur Konstruktion des Bayesschen Netzes für den Intelligent Assistant werden Vertreter aus

diesen drei Gruppen benötigt. Somit stellt diese Vorgehensweise einen erheblichen Kostenfaktor dar, der reduziert werden muß.
Die Möglichkeit, bereits vorhandenes Expertenwissen zu nutzen, um einer neuen Implementierung und Strukturierung des Wissens in einer anderen Form gerecht zu werden, sollte bedacht werden. Die Existenz eines Verfahrens, das entweder eine erneute Akquisition überflüssig macht oder eine unumgängliche Wissensakquisition effektiv gestaltet, ist eine Forderung aus dem *Help Desk*. Nicht nur in der Anfangsphase der Konstruktion der Bayesschen Netze ist es wichtig, den Experten zu unterstützen. Auch in der Anwendungsphase ist der Experte in der weiteren Pflege und Erweiterung der Netze zu unterstützen. Eine Automatisierung ist hier ebenso wünschenswert.

3.3.2 Einfaches Frontend

Da im First Level Support der Benutzer selbst operieren kann, sowie auch weniger erfahrene Experten zur Bearbeitung eingesetzt werden, ist das zu bedienende System möglichst einfach zu halten. Das Frontend soll keine übermäßig hohen fachlichen und psychologischen Hürden darstellen. Der Ablauf soll klar und einfach sein und die Bedienung nicht viel Zeit in Anspruch nehmen. Voraussetzung dafür ist ein nicht übermäßig strukturiertes, dem Werkzeug zugrunde liegendes Expertenwissen. Es werden nur routinemäßig oberflächliche Fragen an den Benutzer gestellt und einfache Tests im Hintergrund angestoßen, um ein grundlegendes Fehlergebiet zu orten, soweit als möglich einzuschränken und dem Experten die weitere Fehlerlokalisierung und Fehlerbehebung zu erleichtern.

3.3.3 Ankopplung an ein Trouble–Ticket–System

Das *integrierende* Werkzeug *[HAN 99a, S. 315]* des Fehlermanagements und des Workflow–Managements spielt im gesamten Fehlerbearbeitungsprozeß eine Rolle. Die automatische oder durch den Experten veranlaßte Generierung eines derartigen Trouble–Tickets stellt einen wichtigen Schritt dar.
Die Experten können sich anhand der Historie der bereits durchgeführten Tests, derer Ergebnisse und den Antworten auf Routinefragen ein erstes Bild über die mögliche Fehlerursache machen. Damit kann die Reaktionszeit auf Fehler verringert werden.
Eine direkte Ankopplung des Expertenwissens an das Trouble–Ticket–System als passives Managementwerkzeug *[HAN 99a, S.69]* hat auch eine flexible Anpassung des strukturierten und implementierten Expertenwissens zur Folge — das Netz „lernt".
Fehler, die oft auftreten, können schnell und exakt identifiziert werden. Ein Rückgriff auf dynamisches Expertenwissen, das zugleich als Datenbank, Historie und Netzdokumentationssystem dient, kann als intelligent bezeichnet werden.
Um ein Bayessches Netz zu erhalten, das diesen Anforderungen genügt, ist ein Algorithmus notwendig, der diese Lernaufgaben übernimmt.

3.3.4 Entlastung der Experten

Die oben genannten Punkte ergeben eine Entlastung der Experten. Diese Gesamtforderung wird speziell auf das implementierte Expertenwissen erweitert. Je mehr Erfahrungen und Wissen in ein derartiges Netz eingebaut werden, um so mehr Entlastung erfahren die Experten, und sie können sich um grundlegende Problemfälle kümmern. Auch dem Kommunikationsdienstanbieter erwächst daraus ein wirtschaftlicher Vorteil, da auch Personal mit keinem solch großen Erfahrungssschatz eingesetzt werden kann und dem Anbieter keine hohen Personalkosten entstehen.

Kapitel 4

Anforderungen an die Bayesschen Netze

Die in *Kapitel 3* vorgestellten Anforderungen, die sich sowohl aus dem Netzmanagement als auch aus dem *Help Desk* Szenario ergeben, werden, in der Vorbereitung zur Modellierungsphase, zu Anforderungen an die Bayesschen Netze zusammengeführt. Der bisherige Einsatz in Entscheidungsbäumen zeigt den aktuellen Stand des Expertenwissens im Intelligent Assistant.
Die Forderungen, die an die Form des implementierten Expertenwissens im Intelligent Assistant gestellt werden, liegen in Bereichen, die den Entwurf, die Realisierung sowie die Eigenschaften der Bayesschen Netze betreffen.

4.1 Entwurf

4.1.1 Graphischer Editor

Bayessche Netze stellen eine Modellierungsmöglichkeit für Wissens– und Expertengebiete dar, die durch Unsicherheiten gekennzeichnet sind. Diese Unsicherheiten können in einem unvollständigen Wissen betreffend einen Fachbereich liegen oder in Zufallsmechanismen, die das Verhalten eines Systems betreffen.
In den Bayesschen Netzen werden kausale und probabilistische Zusammenhänge in einem gerichteten azyklischen Graphen dargestellt. Diese Tatsache bietet die Möglichkeit, derartige Netze nicht nur graphisch darzustellen und über eine graphische Oberfläche zu nutzen. Bereits in der Entwicklungs– und Umsetzungsphase des Expertenwissens kann ein Editor, basierend auf graphischer Darstellung, den Experten einen Einstieg in die Materie bieten. Ein Editor, der in der Anfangs– bzw. Modellierungsphase eingesetzt wird, sollte aus diesem Grund schon in über eine graphische Schnittstelle und Oberfläche verfügen.
Dies hat den Vorteil, daß die Netze einheitlich erstellt werden. Dabei wird den Experten und Fachkräften die Imagination und die Umsetzung ihres Wissens erleichtert, da Zusammenhänge graphisch darstellbar sind. Intern können bereits Konsistenzprüfungen stattfinden, die eine unwirkliche, irreale und zu komplexe Struktur vermeiden helfen. Anschließend erstellt der Editor automatisch ein Bayessches Netz, auf das über eine *Application Programming Interface* API–Schnittstelle zugegriffen werden kann.

Dies erfordert für einen Programmierer und für die Experten eine Einarbeitung sowohl in den Editor in der Anwendung als auch in die Schnittstellenkonfiguration, die je nach eingesetztem Werkzeug oder Editor unterschiedlich im Zeitaufwand ausfallen kann.

4.1.2 Automatische Umsetzung der Entscheidungsbäume

Die bereits vorhandenen Entscheidungsbäume beruhen auf dem Aspekt des Expertenwissens, der sich an einer von Experten vorgegebenen Reihenfolge und Qualifizierung der Fragen und Tests orientiert *(siehe Entscheidungsbäume in Abbildung 3.2, 3.3 und 3.4)*.
Dazu haben die Experten in einer Wissensakquisition durch Befragung Daten und Informationen angegeben und bestimmt, die sie zur Diagnose eines Fehlers benötigen. Diese Daten wurden soweit als möglich formalisiert und in das Schema der Entscheidungsbäume umgesetzt. Die Vorgehensweise nach dem WENN–DANN Prinzip war hier ausschlaggebend.
In die Bayesschen Netze kann dies so direkt nicht umgesetzt werden. Die Bayesschen Netze konzentrieren sich vor allem auf mögliche Ursachen eines Fehlers und die Symptome, die er auslösen kann. Auch die Reihenfolge der Abläufe wird durch die propagierten Wahrscheinlichkeiten bestimmt. Dadurch ist es möglich, Entscheidungen hinsichtlich ihres Informationsgehaltes zu treffen. Welche Entscheidung hat einen schnellen und positiven Diagnoseverlauf zur Folge.
An einem Beispiel wird dies deutlich (siehe *Abbildung 4.1*).

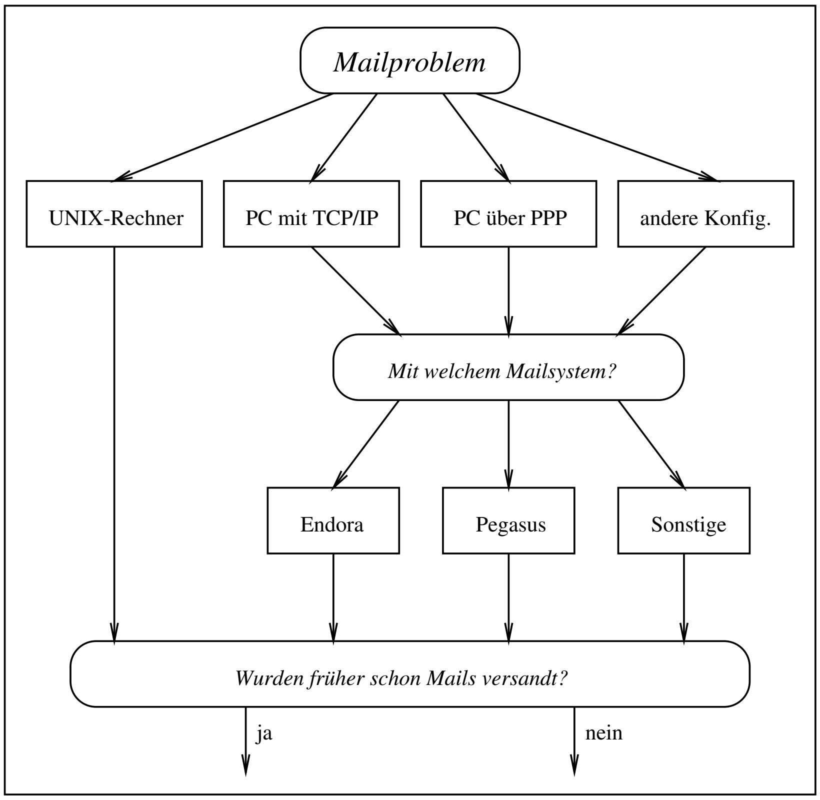

Abbildung 4.1: Ablauf Informationsbeschaffung im Mailproblem

Im Entscheidungsbaum des Mailproblems *siehe Abbildung 3.3* wird zuerst nach der Rechner-

UNIX	*ja*			*nein*		
Mailkonfiguration	Endora	Pegasus	Sonst	Endora	Pegasus	Sonst
Mailproblem *nein*	1.0	1.0	1.0			
Mailproblem *ja*	0.0	0.0	0.0			

Tabelle 4.1: bedingte Wahrscheinlichkeitstabelle für das Mailproblem

konfiguration und der Mailkonfiguration des Benutzers gefragt.
In *Bild 4.1* ist der Ablauf, wie ihn der Benutzer sieht, dargestellt. Er hat die Auswahl zwischen *UNIX*, *PC mit TCP/IP Verbindung*, *PC mit Modem über PPP* oder einer *sonstigen Konfiguration*. Diese Auswahl zeigt auch, daß in den Entscheidungsbäumen diese Informationen immer eingeholt werden.
Wählt der Benutzer des Intelligent Assistant die Rechnerkonfiguration *UNIX*, so erübrigt sich durch verwendete Standardprotokolle die Frage nach dem verwendeten Mailsystem. In den Bayesschen Netzen kann dieser Ausschluß nicht vergleichbar direkt getroffen werden.
Dieses Wissen, der direkte Übergang von der Rechnerkonfiguration auf die Frage, ob *bereits Mails versandt* wurden, kann im Bayesschen Netz indirekt durch die bedingten Wahrscheinlichkeitstabellen in den Knoten dargestellt werden. Die Reihenfolge in den Bayesschen Netzen wird durch die propagierte und bedingte Wahrscheinlichkeit bestimmt.
Das Gerüst einer Tabelle für dieses Problem ist in *Tabelle 4.1* als Beispiel aufgeführt.

Die bedingten Wahrscheinlichkeiten werden von den Experten festgelegt und manuell während der Entwicklungsphase in die Tabellen eingetragen. Um die Experten von diesem manuellen Eintrag zu entlasten, sollte ein Werkzeug oder Algorithmus zur Verfügung stehen, das oder der diese sicheren Annahmen (Rechnerkonfiguration *UNIX* bedarf keiner Frage nach der Mailkonfiguration) automatisch in die Wahrscheinlichkeitstabellen einträgt.
In diesem Fall kann auf eine weitere Wissensakquisition, die von vornherein auf die Bayesschen Netze abzielt, vermieden werden. Der Intelligent Assistant bleibt in dieser Situation dann einer Implementierung des Expertenwissens in Entscheidungsbäumen gleichwertig.

4.1.3 Äquivalenz

Wird Expertenwissen in eines oder mehrere Module von Bayesschen Netzen gepackt, stammt dieses Wissen von verschiedenen Experten und Fachkräften. Jeder Experte und auch jede Fachkraft wiederum besitzt unterschiedliche Arbeitsmethoden, um ein Problem zu analysieren, zu quantifizieren und zu lösen. Dies birgt die Gefahr, daß jeder Experten „sein" Bayessches Netz, betreffend sein Fachgebiet, anders konstruieren würde und die Abhängigkeiten sich auf andere Weise ergeben würden.
Werden für verschiedene Problembereiche verschiedene Bayessche Netze erstellt, sollten diese Netze äquivalent sein, um die gleiche Qualität der Diagnose in jedem Fachgebiet sicherstellen zu können. Dies wird vor allem durch eine einheitliche Struktur der Bayesschen Netze ermöglicht. Damit wird auch garantiert, daß die Netze kopplungsfähig, skalierbar und erweiterbar sind.
In den einzelnen Modulen der Bayesschen Netze, die sich zusammenfügen lassen, soll ein vergleichbarer und äquivalenter Wissensstand modelliert und repräsentiert werden.
Erreicht wird dies durch eine möglichst einheitliche und strukturierte Vorgehensweise in der

Modellbildung. So kann die oben erwähnte unterschiedliche Vorgehensweise verschiedener Experten auf ein einheitliches Niveau gestuft werden.
Durch die Einbindung des Expertenwissens in die vorhandenen Entscheidungsbäume wurde ein gewisses Level geschaffen. Da aber eine automatische und direkte Umsetzung der Entscheidungsbäume in die Bayesschen Netze nicht möglich ist, ist eine erneute Strukturierung des Expertenwissens erforderlich. Demzufolge kann in der Konstruktion der Bayesschen Netze eine nochmalige kostenintensive Befragung der Experten mit einer Einstufung der Wahrscheinlichkeiten nicht vermieden werden.

4.2 Realisierung und Anwendung

4.2.1 Struktur

Bayessche Netze stellen als graphisches Modell die Abhängigkeiten von Variablen auf wahrscheinlichkeitstheoretischer Basis dar. Dabei existieren verschiedene Abhängigkeiten zwischen den einzelnen Variablen. Im Zusammenhang mit statistischen Daten hat ein graphisches Modell diverse Vorteile. Ein derartiges Modell kann vor allem kausale und probabilistische Zusammenhänge zwischen Variablen und Daten darstellen. Die reale Welt, bestehend aus Expertenwissen und Daten, wird in das Modell der Bayesschen Netze abgebildet.
Um dabei das gesamte Bayessche Netz oder Teilnetze für die Experten und Fachkräfte sowohl in der Realisierung als auch in der Anwendung übersichtlich und überschaubar zu halten, ist es notwendig, daß die Struktur nicht zu komplex wird. Voraussetzung dafür ist ein Bayessches Netz, das nicht zu groß und nicht überfüttert ist. Eine zu große Anzahl an Abhängigkeiten und Verbindungen sollte vermieden werden.
Da im Bayesschen Netz bzw. in den Bayesschen Netzen für den Intelligent Assistant das Expertenwissen für die Erstversorgung der Benutzer modelliert und realisiert werden soll, lassen sich die bewertungsfähigen Variablen eingrenzen. Vergleichbar ist dies mit den Entscheidungsbäumen, die nur die ersten Schritte der Experten automatisieren.

4.2.2 Implementierung

Die Implementierung der Bayesschen Netze über eine graphische Schnittstelle besitzt überwiegend Vorteile in der Bearbeitung und Anwendung durch die Experten und Fachkräfte. Dabei besteht die Gefahr, daß die Graphiken erheblich Speicherplatz benötigen und die Abarbeitung der Fragen und Tests verlangsamen.
Die Experten sollten dies durch ihre Implementierung und die Wahl einer einfachen Struktur des Netzes beeinflußen und unnötige Fragen und Tests vermeiden.
Mit der Wahl eines entsprechenden Editors und Werkzeugs, die einfache Repräsentationsformen der Graphiken zur Speicherung wählen, kann dieser Punkt ebenso beeinflußt werden.

4.2.3 Werkzeugunterstützte Integration

Die Realisierung und der anschließende Einsatz und die Anwendung der Bayesschen Netze ist die konsequente Fortführung der Vorgehensweise, das Expertenwissen des *Help Desk* am LRZ über kausale und probabilistische Eigenschaften in den Intelligent Assistant einzubinden.
Wie bereits im Entwurf sollte auch in der Anwendung den Experten und Fachkräften

ein Werkzeug mit graphischer Schnittstelle zur Verfügung stehen. Die ihnen vertrauten Mechanismen aus dem Entwurf erleichtern das Verständnis und die Anwendung, erniedrigen die Hemmschwelle und zeigen den Nutzen auf. Das universelle Werkzeug soll eine Schnittstelle API besitzen, die es zum Einsatz im Intelligent Assistant integrationsfähig macht. Dabei soll es auch möglich sein, ein Werkzeug produkt– und herstellerunabhängig einsetzen zu können.
Da das Expertenwissen voraussichtlich nur in Teilen und zu unterschiedlichen Zeiten in Bayessche Netze umgesetzt werden kann, ist es erforderlich, parallel, aber in verschiedenen Fachgebieten, die Entscheidungsbäume und die Bayesschen Netze als implementiertes Expertenwissen einzusetzen. Diese Möglichkeit besteht über die bereits erwähnte API.
Um auch in der Anwendung eines bereits in der Enwicklung eingesetzten Editors, den Experten, Fachkräften und Benutzern den Umgang zu vereinfachen, sollten die Tabellen, die die Wahrscheinlichkeitswerte enthalten, versteckt werden. Im Normalfall wird dabei eine ODER–Beziehung zwischen zwei Ursachen angenommen. Ein Symptom tritt auf, wenn Ursache 1 *oder* Ursache 2 vorliegt.
Wo dennoch Wahrscheinlichkeitswerte notwendig sind, wie bei den a priori Wahrscheinlichkeiten, sollten dem Anwender Kategorien wie *sehr häufig* bis zu *sehr selten* angeboten werden.

4.3 Eigenschaften

4.3.1 Erweiterung mit neuen Fehlerquellen

Wie bereits erwähnt, besitzen Netzwerke eine hohe Dynamik. Durch An– oder Abkopplung von Geräten oder Installation neuer Anwendungen entstehen neue Fehlersituationen. Dies bedeutet auch, daß ein bereits vorhandener und im Bayesschen Netz berücksichtigter Fehler neue und weitere Ursachen haben kann. Ebenso kann ein derartiger Fehler neue Symptome hervorrufen und die Kostenstruktur eines Netzes verändern.
In diesen Fällen muß es möglich sein, das bzw. die Bayesschen Netze um diese Daten zu erweitern.

4.3.2 Kopplung

Im Intelligent Assistant werden derzeit drei Problembereiche bearbeitet. Das Expertenwissen jedes dieser Bereiche kann in separaten Bayesschen Netzen implementiert werden. Wie im Falle des Verbindungsproblems, das sich als Mailproblem erweist, erwähnt wurde, ist es durchaus notwendig, von einem Fehlerbereich in den nächsten zu wechseln.
Die Simulation des Übergangs von einem Fachgebiet in ein weiteres kann durch die im Einzelfall notwendige Kopplung realisiert werden.

4.3.3 Transparenz

Der *Help Desk* am Leibniz–Rechenzentrum setzt sich aus einer größeren Gruppe von Experten und Fachkräften zusammen. Jeder Experte ist für ein spezielles Fachgebiet verantwortlich. Tritt ein schwerwiegender Fehler in einem der Fachgebiete auf, so wird das Trouble–Ticket an den zuständigen Experten weitergeleitet oder er wird von einem weiteren bearbeitenden Experten konsultiert.
Damit findet eine Vermischung der Arbeits– und Wissensgebiete statt.

Der Intelligent Assistant hat vor allem die Aufgabe, die Experten von Routineaufgaben zu befreien und damit eine einheitliche Grundlage und einen gleichwertigen Einstieg in alle Problembereiche zu bieten.
Das Expertenwissen, auf das der Intelligent Assistant in den Bayesschen Netzen zurückgreift, sollte deshalb in seiner Anwendung und Realisierung nachvollziehbar (transparent) sein. Eine ausführliche Dokumentation erleichtert dabei die Aktualisierung und die Pflege der Bayesschen Netze erheblich.

4.3.4 Abarbeitungsreihenfolge und Geschwindigkeit

In den bereits implementierten Entscheidungsbäumen *(siehe Abbildung 3.2, 3.3 und 3.4)* ist durch die feste Struktur die Reihenfolge der Abarbeitung festgelegt. Dies führt dazu, daß die gesamte Fehlerdiagnoseroutine durchgearbeitet wird und durch eine mögliche rechtzeitige Kenntnis der Fehlerursache die Routine nicht vorzeitig beendet werden kann.
In den Bayesschen Netzen wird die Reihenfolge durch die berechneten Wahrscheinlichkeiten bestimmt. Zu den Entscheidungsknoten wird zusätzlich noch der Faktor des Informationsgehaltes eingeführt.
Zu jeder Variablen werden Kosten definiert, die den Aufwand beschreiben, der anfällt, wenn der Benutzer diese Frage beantwortet.
Der Geschwindigkeitsvorteil besteht überwiegend darin, daß durch einen Fragealgorithmus diejenige Frage ausgewählt wird, die die kleinsten Kosten erwarten läßt. Dies kann und muß jedoch nicht immer der Fall sein. Da der Algorithmus sich für die minimalsten Kosten im Verhältnis zum erwarteten Nutzen ausspricht, ist es möglich, daß der Informationsgehalt der Antworten bis zur letzten Frage nicht ausreichend ist und konsequent alle Fragen gestellt werden.
Der Test *traceroute* hat einen hohen Informationsgehalt, dadurch, daß er explizit den letzten erreichbaren Rechner nennt. Dieser Test besitzt auf der anderen Seite einen hohen zu erwartenden Kostenfaktor.
Die Testroutine *ping* hat im Gegensatz dazu einen geringen Informationgehalt, da sie nur prüft, ob ein angegebener Rechner erreichbar ist. Es ist nicht gewährleistet, daß nach einem erfolgten Test *ping* nicht noch ein *traceroute*–Test ausgeführt werden muß, da die erhaltenen Informationen nicht ausreichend waren.

4.3.5 Skalierbarkeit

Ein Bayessches Netz, das die Eigenschaft der Skalierbarkeit besitzt, startet mit einigen wenigen Knoten und kann leicht zu einem System mit mehreren tausend Knoten expandiert werden. Das Kriterium der Skalierbarkeit sorgt dafür, daß ein derartig aufgebautes Netz nicht ausufert und daß es sowohl in einem kleineren wie größeren Umfeld eingesetzt werden kann.
Dies ist für Rechnernetze erforderlich, da das Management derartiger Netze, flexibel auf die Veränderungen im Rechnernetz reagieren muß. Die Vielzahl unterschiedlicher Geräte und Komponenten sowie Dienste und Anwendungen erfordert dies. Im technologischen Fortschritt sowohl der Komponenten als auch der Dienste expandiert ein solches Rechnernetz sehr schnell.
Die Forderung nach Skalierbarkeit der Bayesschen Netze ist in dem Sinne zu verstehen, daß das Expertenwissen des Intelligent Assistant auch auf ein derart expandierendes Netzwerk anwendbar ist.

4.3.6 Automatisches Erstellen von Trouble–Tickets

Ein Experte ist zur Fehlerbehandlung an möglichst vielfältiger Unterstützung auf dem Diagnoseweg interessiert. Erste Anhaltspunkte liefern ihm die bereits im Trouble–Ticket–System hinterlegten Trouble–Tickets. Diese enthalten nicht nur Problembeschreibungen, sondern zeigen auch eine oder mehrere bereits erfolgreich angewandte Lösungsstrategie(n) auf.
Wird ein Fehler gemeldet, so trägt der Experte alle zur Fehlerbehebung beigetragenen Überlegungen und seine durchgeführten Tests und deren Ergebnisse in das Ticket ein. Die in der Fehlerdiagnose auf diese Weise gewonnene Historie ist der Kern eines qualifizierten Trouble–Tickets. Anhand dessen kann ein weiterbearbeitender Experte oder eine Arbeitskraft, die mit diesem Fachgebiet nicht allzusehr vertraut ist, die bereits durchgeführten Tests und weitere Lösungsstrategien einsehen und die entsprechenden Erstmaßnahmen oder fortführenden Maßnahmen zur Fehlerbehebung einleiten.
In den bereits vorhandenen Entscheidungsbäumen wird diese Historie automatisch im Hintergrund der Abfolge der Tests und Fragen erstellt.
Dies ist auch eine Forderung an die Bayesschen Netze. Über eine Schnittstelle sollte ebenfalls eine automatische Dokumentation des Ablaufes der Test– und Frageroutinen erfolgen, um die Nachvollziehbarkeit einer Diagnose zu garantieren.
Diese Schnittstelle kann durch den schon erwähnten Editor bzw. Werkzeug zur Verfügung gestellt werden.

4.3.7 Lernfähigkeit

Prinzipiell soll nicht bei jeder Fehlerdiagnose das einzelne Bayessche Netz neu initialisiert und berechnet werden. Die „Intelligenz" des Intelligent Assistant soll unter anderem auch darauf basieren, daß das Netz die Ergebnisse vorhergehender Abläufe noch präsent hat, welche Fehler gerade besonders häufig auftreten.
Demzufolge kann der Intelligent Assistant mit der *richtigen* Frage beginnen, falls überhaupt noch eine Frage notwendig ist.
Dies wird an einem Beispiel aufgezeigt:
Melden immer mehr Benutzer den gleichen Fehler oder ergeben die ausgestellten Trouble–Tickets eine Tendenz, daß eine Komponente defekt ist, so sollen sich in diesem Algorithmus auch die Wahrscheinlichkeiten verschieben. Wendet sich ein weiterer Benutzer an den *Help Desk*, so kann ihm schnell und effektiv weitergeholfen werden. Der Algorithmus kann entweder allein durch die Angaben des Benutzers (z.B. der Rechnernamen) feststellen, daß dieser von jenem Problem betroffen, ist oder durch eine einzelne Frage bereits die Fehlerkategorie festlegen.
Diese Lernfähigkeit wird in einem Lernalgorithmus repräsentiert.

4.3.8 Modularität

Im medizinischen Bereich zeichnet ein komplexes Bayessches Netz ein Krankheitsbild auf. Ein Beipsiel ist das System *MUNIN* für die Diagnose bei sechs Muskeln und acht Nerven und einer Knotenanzahl von mehr als 1000. Ein weiteres medizinisches Expertensystem *PATHFINDER* deckt ca. 60 Lymphknotenkrankheiten und 100 Symptome und Testergebnisse ab.
Wie in der medizinischen Diagnose der Mensch in Teilbereiche und damit in Teildiagnosen aufgeteilt wird, muß es im Rechnernetzmanagement die Möglichkeit geben, Teildiagnosen zu stellen.

Die Generierung einzelner unabhängiger Module in den Bayesschen Netzen ergibt sich auch aus der Forderung nach einer einfachen Struktur. Diese müssen auch einzeln diagnostizierbar sein, für jeden Modulbereich eigene Diagnosen stellen können.

Kapitel 5

Modellierungsaspekte

Die Anforderungen, die in *Kapitel 4* an die Bayesschen Netze für den Einsatz im Intelligent Assistant gestellt werden, wurden erläutert und motiviert. Der momentane Wissensstand in den implementierten Entscheidungsbäumen wurde dabei berücksichtigt.
In der hier folgenden Modellierungsphase werden die Schritte von der Erstellung bis zur Realisierung derartiger Netze das Gerüst darstellen, um einzelne Anforderungen in Lösungsansätze und –ideen umzuwandeln.

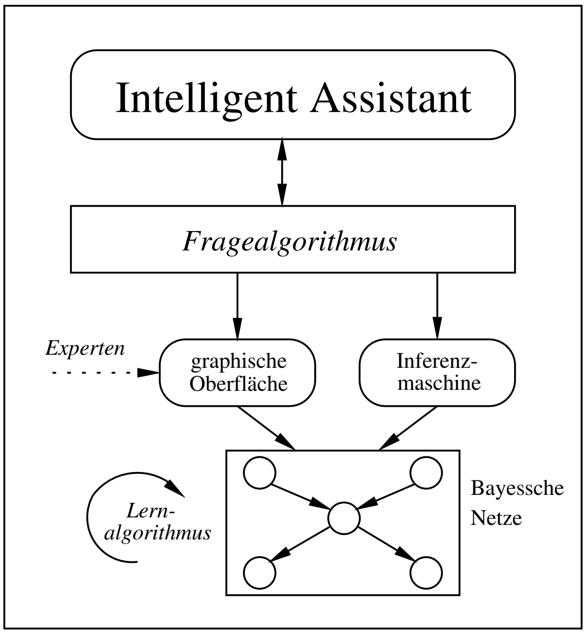

Abbildung 5.1: Einordnung der Werkzeuge und Algorithmen

In *Abbildung 5.1* werden die im folgenden erläuterten Zusammenhänge graphisch dargestellt.

5.1 Problembeschreibung

Bei Bayesschen Netzen handelt es sich um gerichtete azyklische Graphen. Die Knoten des Graphen enthalten in Tabellen Informationen über die Wahrscheinlichkeiten und bedingten Wahrscheinlichkeiten des Zutreffens bestimmter Ereignisse. Die im folgenden als Hypothesen bezeichneten Knoten, stellen die Annahmen bzw. die Variablen eines Bayesschen Netzes dar. Insofern repräsentieren die Knoten eines Bayesschen Netzes das unsichere und auch unvollständige Wissen über das Zutreffen dieser Hypothesen.
Die Kanten des Graphen repräsentieren Abhängigkeiten zwischen den Hypothesen, die ihrerseits in verschiedenen Knoten dargestellt werden. Im Detail handelt es sich um Abhängigkeiten zwischen den Wahrscheinlichkeiten des Zutreffens unterschiedlicher Ereignisse.
Ein Bayessches Netz im Intelligent Assistant soll den Bereich des Diagnoseverlaufs eines Experten am Leibniz–Rechenzentrum simulieren. Die Knoten dieses Netzes sind die Informationen, die der Experte zu einer Problemlösung benötigt. Die zugehörigen Abhängigkeiten legt der Experte während der Wissensakquisition und –modellierung fest.

5.2 Wissensbeschaffung

5.2.1 Fragebögen

Das Fachwissen der Experten kann effektiv in einer ersten gemeinsamen Runde und anhand eines vorgegebenen Ablaufs erarbeitet werden. Hier bieten sich als Hilfsmittel Fragebögen an, in welchen die Experten die für den jeweiligen Problembereich notwendigen Informationen bestimmen. Hier besteht auch die Möglichkeit, daß von den einzelnen Experten bereits Vorarbeit geleistet wird, und die Erarbeitung nicht viel Zeit in Anspruch nimmt.
Die Experten betreiben Ursache–Wirkung Forschung.
Dabei ist es von Bedeutung, daß der Zweck und das Einsatzgebiet der Bayesschen Netze bedacht wird. Das Expertenwissen für den Intelligent Assistant wird auf der Grundlage der Modellierung des Routineablaufes in einem *Help Desk* erstellt:

- Welche Fehlergebiete werden modelliert?
- Welche Informationen werden für jede mögliche Fehlermeldung in jedem der Bereiche benötigt?
- Wie werden diese Informationen in Fragestellungen ausgedrückt?
- Wie können diese bei Experten erfragt werden?
- Kann die ein oder andere Information durch Tests festgestellt werden? Wenn ja, wie?
- Welche Entscheidungen sind zu treffen?
- Welchen Nutzen bzw. Kosten haben die einzelnen Fragen und Tests?
- Welche Restkosten ergeben sich, wenn eine Frage positiv oder negativ beantwortet wird?

Den einzelnen Fragen bzw. Informationen werden durch die jeweiligen Experten Prioritäten zugewiesen. Für die Experten ist damit intuitiv eine Reihenfolge der Fragestellungen verbunden.
In einem weiteren Schritt bestimmen die Experten die Abhängigkeiten der Informationen und der Fragen. Wird beispielsweise eine Frage durch den Benutzer positiv beantwortet, so kann sich eine weitere Frage erübrigen oder in einem anderen Fall, sich noch als dringend notwendig erweisen. Diese Modellierung der Abhängigkeiten kann bereits in graphischer Weise erfolgen und gleicht (allein nur mit den Abhängigkeiten) einem Bayesschen Netz.

5.2.2 Hypothesen

Ein weiterer Schritt besteht darin, diese hypothetischen Ereignisse festzulegen. Diese werden mit Kostenfaktoren verbunden, die festlegen, wie hoch der Aufwand für den Rest der Diagnose voraussichtlich ist, wenn dieses Ereignis zutrifft. Die Hypothesen werden in Mengen von Variablen aufgegliedert, die jeweils gegenseitig sich ausschließende Ereignisse enthalten. Danach werden die Modellierungsvariablen in kausalen Zusammenhang gebracht. Dabei ist zu beachten, welche Ereignisse eine direkte Auswirkung auf weitere Ereignisse haben werden. Dieses Konzept bietet die Möglichkeit, sich verändernde Zustände in einem Abbild der realen Welt darzustellen.
Für jede Variable trifft genau eines ihrer Ereignisse zu. Registriert wird dies in einer (bedingten) Wahrscheinlichkeitstabelle, die voneinander unabhängige Ereignisse mit voneinander abhängigen Ereignissen repräsentiert.

5.2.3 Einbinden der Entscheidungsbäume

Wie in dieser Arbeit in *Punkt 3.3.4* bereits erwähnt wurde, ist die Entlastung der Experten vorrangiges Ziel des First Level Support Systems am LRZ. Dies bedeutet auch, daß das bereits vorhandene Expertenwissen in den Entscheidungsbäumen genutzt wird, um einer neuen Implementierung und Strukturierung des Expertenwissens in Form der Bayesschen Netze gerecht zu werden. Eine nochmalige Wissensakquisition, mit einer zusätzlichen Wahrscheinlichkeitseinstufung und Beurteilung für die a priori Wahrscheinlichkeiten der Knoten, stellt einen weiteren Arbeitsaufwand für die Experten dar.
Eine direkte Umsetzung der Entscheidungsbäume in Bayessche Netze ist nicht möglich. Um dennoch dieses vorhandene Expertenwissen nutzen zu können, kann ein Verfahren entwickelt werden, das das Wissen der Entscheidungsbäume interpretiert und den Bayesschen Netzen in passender Form zur Verfügung stellt.
In den Entscheidungsbäumen, wie dies auch in den Beispielen in *Abbildung 3.2, 3.3 und 3.4* dargestellt wird, ist die Reihenfolge der Fragen festgelegt. Wird eine Frage mit *ja* beantwortet, nimmt die Diagnose einen anderen Verlauf, als wenn die Frage mit *nein* beantwortet wird.
In den Bayesschen Netzen ist dieser vorzeitige Ausschluß einer Diagnoserichtung nicht gegeben. Hier werden alle Ursachen betrachtet und durch Propagierung wird eine spezielle Richtung eingeschlagen. Die Reihenfolge in den Entscheidungsbäumen kann während der Laufzeit nicht verändert werden.
Ein Verfahren, das die Informationen aus den Entscheidungsbäumen direkt in Bayessche Netzeinformation umsetzt, kann deshalb nicht realisiert werden. Vielmehr ist es notwendig, die Informationen in den Entscheidungsbäumen zu werten und in einer Formation zusammenstellen, daß für ein Bayessches Netz die Knoten (a priori oder bedingte Wahrscheinlichkeitsknoten) in

geeigneter Darstellung übernommen werden können.
Es ist fraglich, inwieweit dieses Verfahren einer neuen Wissensakquisition ähnelt und nicht komplett durch eine solche ersetzt werden kann.

5.2.4 Algorithmus zur Erstellung der Wahrscheinlichkeitstabellen

Das Bayessche Netz erhält über einen Editor seine Wahrscheinlichkeiten und bedingten Wahrscheinlichkeiten. Dieser Vorgang des Eintragens kann auch als Lernen bezeichnet werden.
Lernen ist dabei ein semi–automatisches Verfahren, um ein Modell zu konstruieren oder zu modifizieren. Unter qualitativem Lernen versteht man die Konzeption und die Erstellung der Struktur eines Netzmodells. Im Gegensatz dazu steht das quantitative Lernen, das die Spezifikation der bedingten Wahrscheinlichkeiten betrifft.
Die Abschätzung dieser bedingten Wahrscheinlichkeiten (im voraus) ist immer subjektiv.
Die Experten weisen den Knoten des Netzes, die keine Vorgänger besitzen, sogenannte a priori Wahrscheinlichkeiten zu.
Für jeden weiteren Knoten wird eine bedingte Wahrscheinlichkeitstabelle erstellt, in der jeder Knoten mehrere verschiedene Zustände besitzen kann, und die Vorgängerknoten die Bedingungen dafür darstellen. Dies wird in einer Tabelle berücksichtigt, wie es in *Tabelle 4.1* zu sehen ist. Die Abhängigkeiten werden dann durch die Werte der bedingten Wahrscheinlichkeiten repräsentiert. In der Bestimmung dieser Wahrscheinlichkeiten werden den Knoten für jede Kombination in der Tabelle Wahrscheinlichkeitswerte zugewiesen.
Wahrscheinlichkeiten, die die Experten aus ihrem Erfahrungsschatz bestimmen können, werden von ihnen eingetragen.
Für die restlichen besteht die Anforderung an das Werkzeug bzw. den Editor im Entwurf der Bayesschen Netze, für das Ausfüllen der Wahrscheinlichkeitstabellen einen Algorithmus zur Verfügung zu stellen. Dieser fragt die modellierenden Experten, welche Kombinationen und Abhängigkeiten im Expertenwissen auf keinen Fall zugelassen sind und trägt die bedingten Wahrscheinlichkeiten mit einer Sicherheit (Evidenz) von 0% automatisch ein.

5.2.5 Kosten

Zu jeder Variable bzw. Knoten werden zusätzlich Kosten definiert, die anfallen, wenn der Benutzer diese Frage beantworten muß oder der angebotene Test durchgeführt wird. Sie sollen den Aufwand beschreiben, den es erfordert, diese Frage zu beantworten oder den Test durchzuführen. Der Aufwand wird dabei jeweils für den Rest des Diagnosevorgangs bestimmt. Die erste Schrittweite wird aus realen Werten berechnet und für die weiteren Schrittweiten, bis zum Ende der Diagnose, werden geschätzte Werte angenommen. Diese Werte berechnen sich aus den noch nicht veränderten bedingten Wahrscheinlichkeiten, da die Werte der bedingten Wahrscheinlichkeiten sich erst durch die Propagierung, die Nutzung des Knotens im ersten Berechnungsschritt, ändert. Dies dient vor allem dazu, dem Fragealgorithmus, der die Reihenfolge der Fragen und Tests bestimmt, die Auswahl zu erleichtern.
Bewerkstelligt wird dies, indem der eingesetzte Editor die Möglichkeit bietet, Kosten und auch weitere wichtige Informationen wie Fragetext dieses Knotens und Knotenname zu hinterlegen. Die Kosten entsprechen nicht den Wahrscheinlichkeiten, da sie separat in einer Eigenschaftstabelle, die für jeden Knoten angelegt wird, gehalten werden und auch für den Anwender nicht sichtbar sind. Niedrige Kosten und eine hohe Wahrscheinlichkeit haben allerdings den Effekt, den Knoten bzw. die Frage für das weitere Vorgehen und den Ablauf

attraktiv zu machen.
Die Kosten eines Knotens geben damit an, wie teuer und aufwendig es ist den Zustand des Knotens zu ermitteln. Dies geschieht hinsichtlich der Fragen und durchzuführenden Tests dadurch, daß z.B. eine Frage direkt mit *ja* oder *nein* beantwortet werden kann oder als intensive Maßnahme ein Test wie *traceroute* durchgeführt werden muß.

5.3 Werkzeug zur Konstruktion und Anwendung

Um Bayessche Netze aus vorhandenen Daten und aus bereits in Auszügen vorhandenem Expertenwissen konstruieren und modellieren zu können, bedarf es einer strukturierten Vorgehensweise. Diese kann in der Konstruktion und Erstellung eines Bayesschen Netzes durch einen Editor, wie er bei Siemens in der Abteilung ZT IK 4 entwickelt wurde, bewerkstelligt werden. Es existieren auch kommerzielle Produkte und Werkzeuge, die das Erstellen über eine graphische Oberfläche ermöglichen. Für die Anwendung der Netze zur Propagierung der Wahrscheinlichkeiten stellen sie über eine Schnittstelle (API) ihre Inferenzmaschine zur Verfügung.
Im folgenden soll kurz auf das kommerzielle Produkt HUGIN der Fa. HUGIN Expert A/S, Dänemark, eingegangen werden, um die Arbeits– und Einsatzgebiete der Bayesschen Netze vor Augen zu führen.

5.3.1 Produkt

Das Softwaresystem HUGIN besteht aus zwei separaten Komponenten. Die Anwendung „HUGIN Runtime“ dient dazu, graphische Modelle für Entscheidungsprozesse zu erstellen, die mit Unsicherheiten behaftet sind. Die graphischen Modelle dabei sind Bayessche Netze und ihre Erweiterung zu Influenzdiagrammen. Das Runtime–Modul ist ein einfaches graphisches Werkzeug. Es enthält den Graphikeditor, den Compiler und das Basissystem. In dieser Entwicklungsumgebung lassen sich *kausal probabilistische Netze* erstellen, editieren und testen. Im Set mit dem eigenständigen Produkt *HUGIN API* können Entwickler die Inferenz–Maschine für kundenspezifische Anwendungen verwenden. Diese Schnittstelle ist eine C– oder C++–Bibliothek, um Bayessche Netze (mit diskreten und stetigen Variablen) und Influenzdiagramme in Anwendungen zu integrieren.
Die für diese Diplomarbeit verwendete Software (HUGIN Lite Version 5.3, Dec 7 1998) ist eine frei erhältliche Version, deren Funktionalität im wesentlichen dem Runtime–Modul entspricht. Die Anzahl der Knoten ist dabei auf 200 beschränkt.

5.3.2 Historie

Ursprünglich wurde für das Projekt ESPRIT zur Diagnose von neuromuskularen Krankheiten das Bayessche Netz MUNIN entwickelt. Eine Forschungsgruppe entwickelte an der Universtität von Aalborg Methoden, um Diagnoseprobleme kalkulieren zu können.
Diese Methoden wurden in ein Werkzeug integriert, das für jeden Entwickler, der mit der Materie nicht allzu sehr vertraut ist, verständlich und leicht zu bedienen ist.
HUGIN wurde im Laufe der Jahre durch die Möglichkeiten der Influenzdiagramme erweitert. Desweiteren stellten die Forscher die Option zur Verfügung, stetige Knoten durch die Gaußsche Normalverteilung in die Modelle einzubinden.

5.3.3 Technologie

Bayessches Netz

Wie durch die Entwicklungsstufen deutlich wurde, unterscheidet HUGIN zwei Arten von Netzen. In einer ersten Stufe wird das Bayessche Netz erstellt. Das Ergebnis ist ein gerichteter azyklischer Graph, dessen Knoten jeweils eine Zufallsvariable und dessen Kanten wahrscheinliche Abhängigkeiten zwischen den Knoten repräsentieren. Quantifiziert werden diese Abhängigkeiten durch Tabellen, in welchen für jeden Knoten die bedingten Wahrscheinlichkeiten notiert werden.

Influenzdiagramm

In einem weiteren Schritt wird dieses Netz mit Entscheidungs– und Nützlichkeitsknoten ausgestattet.
Dabei legen Kanten, die in Entscheidungsknoten hineinführen, den zeitlichen Ablauf fest: Führt eine Kante von einem Zufallsvariablenknoten in einen Entscheidungsknoten, so ist zum Zeitpunkt der Entscheidung der Wert der Zufallsvariablen bekannt. Eine Kante von einem Entscheidungsknoten in einen weiteren legt die chronologische Ordnung der Entscheidungen fest. Aus diesem Grund muß auch das Netz azyklisch sein und es muß ein direkter Pfad existieren, der alle Entscheidungsknoten beinhaltet.

5.3.4 Konstruktion und Arbeitsweise

Bayessches Netz

Für jeden der im Netz befindlichen Knoten wird eine Wahrscheinlichkeit festgelegt. Die Knoten, zu denen keine kausale Abhängigkeit besteht, erhalten eine a priori Wahrscheinlichkeit. Für die weiteren Knoten erstellt HUGIN das Schema der Tabelle der bedingten Wahrscheinlichkeiten automatisch, während die Abhängigkeiten modelliert werden. Hier müssen nur noch die entsprechenden Zahlenwerte von den Experten eingetragen werden. Im Ablauf einer Propagierung (Berechnung) werden die Zustände der jeweiligen Knoten mit 100% festgelegt und somit wird als Ergebnis geliefert, wie hoch die Wahrscheinlichkeit für die jeweiligen Ergebniszustände ist.

Influenzdiagramm

Im Übergang zu den Influenzdiagrammen wird das Bayessche Netz gespiegelt, um für die Gegenwart und die Zukunft zwei Teilnetze zu erhalten.
Die Gegenwartsknoten werden jeweils mit ihren Zukunftsknoten verbunden. Die Pfeilrichtung verläuft dabei vom Gegenwarts– zum Zukunftsknoten bis auf diejenigen Knoten, in denen es sehr wahrscheinlich ist, daß sich der Zustand dieses Knotens auch in Zukunft nicht ändert — diese werden nicht verbunden.
Die Zukunftsknoten erhalten den gleichen Namen wie die Gegenwartsknoten, sie werden nur durch spezielle Zusatzzeichen als solche gekennzeichnet. Auch innerhalb des Zukunftsnetzes bleiben die Abhängigkeiten, wie sie im Gegenwartsnetz bestehen, erhalten. Die Addition des Zukunftsnetzes zum Gegenwartsnetz ermöglicht es, die Erwartungen und Wahrscheinlichkeiten für die Zukunft zu modellieren. Wird nun im Wechsel von der Gegenwart zur Zukunft eine Entscheidung getroffen, so hat diese Auswirkungen auf die Zukunft. Wird eine Entscheidung

bzw. ein Entscheidungsknoten eingefügt, so hat diese Entscheidung Einfluß auf einen oder mehrere Knoten im Zukunftsnetz.
Zusätzlich kann die jeweilige Entscheidung mit einem Kostenknoten verbunden werden, mit dem die Kalkulation der Entscheidung berechnet werden kann.
Durch Treffen der Entscheidungen werden die Wahrscheinlichkeiten des Netzwerkes verschoben und neu berechnet (propagiert). Dadurch können für sämtliche Alternativen der zu treffenden Entscheidung die erwarteten Kosten berechnet werden.
Unter der Annahme, daß alle zukünftigen Entscheidungen optimal getroffen werden, zusätzlich mit der Annahme, daß sämtliche Hinweise für eine Entscheidung berücksichtigt wurden, kalkuliert HUGIN die Alternativen der ersten getroffenen Entscheidung. So wird mit jeder folgenden zu treffenden Entscheidung verfahren.

5.4 Darstellung

Die Anwendung im Intelligent Assistant kann direkt per graphischer Oberfläche erfolgen. Dabei ist das in Punkt *5.3.1* vorgestellte Werkzeug eine Möglichkeit, den Editor einzusetzen.

5.4.1 Graphisch

Wählt der Benutzer den entsprechenden Fehlerbereich aus, so wird ihm im Browser das graphische Modell des Bayesschen Netzes angezeigt. Zu jedem Knoten existiert eine (bedingte) Wahrscheinlichkeitstabelle, die in der Graphik neben die jeweiligen Knoten plaziert werden. Hier füllt er die entsprechenden Wahrscheinlichkeiten aus (z.B. Durchsatzproblem: Zugangsart WAN = 100% und Durchsatz < *vorgegebener Wert*). Nach der Propagierung, dem Fortschreiben der Wahrscheinlichkeiten, wird aus der Tabelle ersichtlich, daß ein reales Durchsatzproblem besteht und es wird ein Trouble–Ticket erstellt. Der Eintrag der Wahrscheinlichkeiten ist für den Benutzer des Intelligent Assistant mehr verwirrend als klärend und untersttzend. Die Anforderung nach einem einfach zu bedienenden Frontend kann hier nicht erfüllt werden.

5.4.2 Tabellarisch

Der Benutzer trägt in tabellarischer Form seine Erkenntnisse ein. Dies bedeutet, er kreuzt zum Beispiel in einer ihm dargebotenen Tabelle an, welchen Zugang er besitzt und welchen momentanen Durchsatz er feststellen kann.
Die Bedienfreundlichkeit ist hier nicht zu unterschätzen. Für den Benutzer stehen sämtliche Auswahlmöglichkeiten zur Verfügung und er muß sich für eine Auswahl entscheiden.

5.4.3 Entscheidungsorientiert

Die Oberfläche des Intelligent Assistant wird dabei nicht verändert. Die Entscheidungen, die in den Bayesschen Netzen über die Eingaben des Benutzers getroffen werden und welche Fragen ihm gestellt werden, sieht der Benutzer nur anhand der ihm präsentierten Maske.
Die Oberfläche ist dem Kunden, sofern er bereits Fehler mit dem Intelligent Assistant diagnostiziert hat, vertraut. Er kann den Einsatz Bayesscher Netze lediglich in einer Abänderung der Fragenreihenfolge feststellen.

5.5 Fragealgorithmus

5.5.1 Motivation

In den vorhandenen Entscheidungsbäumen wird die Abfolge der Fragen und Tests durch die jeweilige Antwort bzw. durch das jeweilige Ergebnis eines Tests bestimmt.
In den Bayesschen Netzen werden diese unsicheren Entscheidungen zur Fehlerbehandlung über Wahrscheinlichkeiten berechnet. Dazu wird ein Inferenzalgorithmus verwendet, um die Wahrscheinlichkeiten der Fehlerfälle in den Systemkomponenten zu bestimmen. Die bedingten Unabhängigkeiten, die in den Bayesschen Netzen repräsentiert werden, ermöglichen diese Berechnungen für die meisten Problemfälle, die in derartigen Netzen dargestellt werden können. Zu berücksichtigen ist, falls eine fehlerhafte Komponente repariert wird und möglicherweise zusätzliche Erkenntnisse gewonnen werden, daß das Bayessche Netz nur zur Berechnung der Wahrscheinlichkeiten der Fehlerkomponenten dient. Die vorhergehenden Wahrscheinlichkeitswerte können aufgrund eines Statuswechsels in der Komponente ungültig werden.
Die Darstellung der Ergebnisse dieses Algorithmus kann auch über eine graphische Oberfläche erfolgen.

5.5.2 Fehlersuchplan

Im Bereich der Fehlersuche in Systemen bedarf es einer systematischen Vorgehensweise. Die hierfür existierenden Pläne zur Fehlersuche und Reparatur in einem System werden von Unsicherheitsfaktoren beeinflußt. Das Geflecht zwischen Geräten und Komponenten, Beobachtungen sowie Wirkungen auf Aktionen und Veränderungen der Gerätezustände, kann in einem Bayesschen Netz dargestellt werden. Eine optimale Fehlersuche und Fehlerdiagnose beruht dabei auf einem Plan, der minimale Kosten in der Reparatur der betroffenen Komponenten verursacht.
Dieser Plan berücksichtigt hinsichtlich der Unsicherheiten die Abhängigkeiten zwischen den Komponenten und deren Status, die Beobachtungen, die über die betroffenen Komponenten gemacht werden und die Wirkung, die Aktionen auf den jeweiligen Status haben.

5.5.3 Bayessches Netz

Ein Bayessches Netz repräsentiert die oben bezeichneten unsicheren Beziehungen zwischen den einzelnen differenzierten Beobachtungen. Es existieren in kommerziellen Werkzeugen Inferenzalgorithmen, auf die über eine Schnittstelle zugegriffen werden kann. Sie dienen der Berechnung der Wahrscheinlichkeiten von Fehlerkomponenten und eindeutigen Beobachtungen. Weiterhin bestehen Möglichkeiten, Aktionen und sich daraus ergebende Veränderungen der realen Welt, in der Modellwelt der Bayesschen Netze darzustellen. Es ist möglich, den Zustand eines Systems im Vorfeld einer durchzuführenden Aktion mit einem gewissen Maß an Unsicherheit zu bestimmen.

5.5.4 Algorithmus

Ein derartiger Plan beinhaltet die Vorgabe, nach welcher Struktur vorgegangen wird und wie ein Fehler zu suchen ist. Demzufolge dient der Algorithmus dazu, die Wahrscheinlichkeiten zu berechnen, durch das Netz zu propagieren und eine Entscheidung zu treffen bezüglich der weiteren Schritte und Vorgehensweise.

In einer Vorgängerarbeit *[Klin 98]* wurde dieser, von Siemens in der Abteilung ZT IK 4 in Anlehnung an *[HeBr 95]* und mit einer Benutzerschnittstelle über das Internet ausgestattete und implementierte, Algorithmus vorgestellt. Anwendung findet er in einem Modell für die Reparatur und Wartung der Siemens Telefonanlagen. Dabei wurde die Vorausplanung verallgemeinert und erfolgt in mehr als einem Schritt. Zusätzlich besitzt die Siemens–Variante noch eine Benutzerschnittstelle über das Internet.
Der Algorithmus wird in Bereichen eingesetzt, in welchen die Fehlerursachen auch direkt abgeprüft werden können. In einem Rechnernetz kann z.B. der Ausfall einer Komponente durch *ping* oder *traceroute* überprüft werden. Die Konfiguration einer Komponente, die sich in einem Rechnernetz befindet, kann durch weiteren Managementeinsatz ermittelt werden. Es können auch mehrere Variable für eine Komponente definiert werden, die einzelne Konfigurationsparameter beschreiben.
Wenn der Anlaß für die Diagnose ein Verbindungsproblem zwischen zwei Komponenten ist, gibt es die Möglichkeit, die Variablen noch aufzuspalten, in: z.B. *Konfiguration des Startpunktes* und *Konfiguration des Zielpunktes*. Andernfalls kann die Einbindung der Konfiguration der Rechnernetzekomponenten das Bayessche Netz ausufern lassen.
Zu jeder fragbaren Variable werden Kosten definiert, die anfallen, wenn der Benutzer diese Frage beantworten muß. Sie sollen den Aufwand für den Rest der Diagnose beschreiben, den es erfordert, diese Frage zu beantworten.
Der Fragealgorithmus stellt von allen möglichen Fragen diejenige, mit dem kleinsten Erwartungswert der Kosten bis zum Abschluß der Diagnose. Um diese zu ermitteln, wird für jeden Zustand jeder fragbaren Variable berechnet, welche Rest–Kosten (im Schnitt) später noch anfallen werden, wenn diese Variable erfragt wurde, und die dem Zustand entsprechende Antwort gegeben wurde. Die erwarteten Restkosten werden über die möglichen Antworten zu einer Frage gemittelt, gewichtet nach der Wahrscheinlichkeit, daß die entsprechende Antwort gegeben wird (durch Inferenz aus den bisher gegebenen Antworten).
Die Frage, bei der die Summe aus den eigenen Kosten und den zu erwartenden Restkosten minimal ist, wird gestellt.
Um die Restkosten zu ermitteln, kann man neu propagieren und dann wieder alle möglichen Fragen durchspielen, diese immer wiederholen, solange bis man die Ursache des Fehlers in der durchgespielten Variante gefunden hat, oder alles gefragt hat.
Da dies aufgrund der großen Anzahl von Schritten zu langsam ist, bricht man das Durchspielen nach wenigen Schritten ab (geringe Suchtiefe) und schätzt die Restkosten dadurch ab, daß man nur noch die Fehlerursachen in der Reihenfolge ihrer Wahrscheinlichkeit, unter Berücksichtigung der bisherigen Evidenz, also sicheren Annahmen, durchspielt.
Können die Ursachen, wie etwa Krankheiten in der Medizin, nicht erfragt werden, dann wird zu jeder Frage und jeder möglichen Antwort die verbleibende Entropie (Restkosten) auf den Ursachen abgeschätzt. Die Frage, die in Erwartung die Entropie am stärksten reduziert, wird gestellt.
Bei Ursachenknoten, die nicht abgefragt werden können, ändert sich am Aufbau des Bayesschen Netzes nichts. Über einen modifizierten Fragealgorithmus werden sehr hohe Kosten für die entsprechenden Ursachen–Knoten angesetzt. Damit wird diese Ursache erst dann gefragt, wenn sie eindeutig identifiziert ist, und die Kosten sich lohnen oder keine weitere Frage die Eindeutigkeit eines Ergebnisses verbessern kann.
Der Algorithmus sieht insgesamt sechs Schritte vor, um für eine Anzahl von Variablen die Wahrscheinlichkeiten zu berechnen, wenn bei gegebenem Zustand Aktionen und Tests durchgeführt bzw. Fragen beantwortet werden:

1. In einem ersten Schritt wird ein Bayessches Netz konstruiert. Jede Variable wird in der Weise gekennzeichnet, ob sie für einen durchzuführenden Test oder Frage verantwortlich ist oder nicht.

2. Werden für jede Variable ein Abbildungsknoten $x_i(\prod_i)$ eingeführt, der die möglichen Abbildungen von den Elternknoten von x_i auf sich selbst repräsentiert. Wird festgestellt, daß $\prod_i$ Ursachen der Variablen x_i sind, ist jede derartige Abbildungsvariable nicht für eine Aktion verantwortlich.

3. Werden die wahrscheinlichen Abhängigkeiten aller unverantwortlicher Variablen, einschließlich der Abbildungsvariablen abgeschätzt.

4. Jede verantwortliche Variable wird kopiert, ähnlich den Influenzdiagrammen in HUGIN. Die erste und zweite Instanz jeder Variable zeigt den Zustand vor und nach einer durchzuführenden Aktion.

5. Für die zweite Instanz, das gespiegelte Netz, werden die gleichen Abhängigkeiten wie im originalen Netz hergestellt.

6. Im letzten Schritt werden in dem Netz der zweiten Instanz die Knoten identifiziert, die von der durchgeführten Aktion direkt betroffen waren. Ihre Verbindung zu den Elternknoten wird getrennt und die Zustände dieser Knoten werden auf die Werte gesetzt, die die Aktion festgelegt hat.

Die so entwickelten Bayesschen Netze werden als Antwortnetze, in HUGIN z.B. als Inferenzdiagramme, bezeichnet. Dabei ist zu beachten, daß die Bayesschen Netze nicht nur Abhängigkeiten hinsichtlich einer Wahrscheinlichkeit aufzeigen, sondern auch kausale Wechselbeziehungen. Das Problem in der Konstruktion derartiger Antwortnetze liegt in der Anzahl der Zustände eines jeden einzelnen Abbildungsknotens. Diese kann sehr groß werden und damit die Einschätzung der Wahrscheinlichkeiten unhandlich machen.

5.5.5 anwendbare Implementierung

Ein solcher Fragealgorithmus ist im Werkzeug HUGIN nicht vorhanden. Bei Siemens, Abteilung ZT IK 4, wurde jedoch ein dieser Funktionalität entsprechender Fragealgorithmus implementiert.
Die Inferenzmaschine, die HUGIN über seine Schnittstelle bietet, wird als Teilschritt im Fragealgorithmus benötigt, um die Wahrscheinlichkeiten nach den bisher gegebenen und im Durchspielen als gegeben angenommenen Antworten zu berechnen.
Implementiert wurde der Algorithmus in der Programmiersprache C, in Anlehnung an *[HeBr 95]*, unter Benutzung der HUGIN–API.

5.6 Inferenz

Die zentrale ursprüngliche Idee der Künstlichen Intelligenz war die Trennung der Wissensdarstellung von den Methoden, das Wissen zu nutzen. Inferenz ist der Begriff für die Schlußfolgerung, die das vorhandene Wissen und hinzugewonnene Erkenntnisse nutzt.
In dieser Kombination müssen die Wahrscheinlichkeiten berechnet und propagiert werden,

sobald ein Testergebnis oder die Antwort auf eine Frage die Wahrscheinlichkeiten durch Evidenzeinträge verschiebt. Als Teilproblem tritt beim Fragealgorithmus genau diese Inferenz auf. Es werden Schritt für Schritt Wahrscheinlichkeiten verändert, indem neue Beobachtungen gemacht werden.
Dafür liegt ein Inferenzmechanismus vor, den der Fragealgorithmus über die HUGIN Schnittstelle benutzen kann.

5.7 Lernalgorithmus

Das Wissen, das zur Initialisierung eines Netzes (sogenanntes Erstlernen) benötigt wird, unterscheidet sich von der Adaption, mit der ein Modell Schritt für Schritt den Gegebenheiten angepaßt wird.
Die „Intelligenz“ des Intelligent Assistant besteht auch darin, daß Bayessche Netze von den vorhergehenden Abläufen wissen, welche Fehler besonders häufig auftreten. Mit diesem Zusatzwissen wird die Frage ausgewählt, die den kleinsten Erwartungswert der Kosten bis zum Ende der Diagnose besitzt. Dafür ist kein neuer Algorithmus zu implementieren, vielmehr ist der Aufbau Bayesscher Netze passend zu machen. Es muß ein Netz sein, bei dem es Sinn macht, die Evidenzen aus den vorhergehenden Diagnosesitzungen zu übernehmen.
Die „historischen“ Wahrscheinlichkeiten müssen in den Knoten des Bayesschen Netzes hinterlegt werden. Dabei ist zu unterscheiden zwischen globalen Variablen, deren bedingte Wahrscheinlichkeiten übernommen werden und lokalen Variablen, deren bedingte Wahrscheinlichkeiten in jeder Sitzung und bei jedem Benutzer neu berechnet werden.
Ein Problem ist dabei aufzuzeigen:
Durch die Konzentration der Wahrscheinlichkeit in einem Zustand eines Knotens kann es im Falle der Behebung des Fehlers einige Zeit dauern, bis sich die Wahrscheinlichkeit wieder umverteilt und normalisiert. Dieser Nachlauf der Wahrscheinlichkeiten bezüglich der Realität kann nur dadurch vermieden werden, daß für jede Fehlerdiagnose mit neutralen Wahrscheinlichkeiten, wie dies in den lokalen Variablen geschieht, eine neue Berechnung begonnen wird.

Kapitel 6

Lösungskonzepte im Intelligent Assistant

6.1 Allgemein

Derzeit wird im Intelligent Assistant Expertenwissen in Entscheidungsbäumen realisiert. Dafür existiert ein Java–Tool, das es ermöglicht, die Entscheidungsbäume graphisch zu erstellen. Dies wird „online" erledigt. Weitere Unterstützung von Problemen oder Fehlersituationen wird während Wartungsarbeiten eingepflegt, da es derzeit keinen für den Intelligent Assistant verantwortlichen Experten gibt.
Die Umsetzung des Expertenwissens in Bayessche Netze erfordert, wie in dieser Arbeit aufgezeigt wurde, eine Reihe von Schritten von der Wissensakquisition bis zur Realisierung.
Im folgenden werden Lösungsansätze aus den Anforderungen an Bayessche Netze in *Kapitel 4* und aus den Modellierungsaspekten in *Kapitel 5* für den Intelligent Assistant dargestellt. Dabei werden die erforderlichen Tabellen im Durchsatzproblem beispielhaft im Text aufgeführt. Die Tabellen des Mailproblems befinden sich im *Anhang A*. Die bedingten Wahrscheinlichkeitstabellen des Verbindungsproblems werden skizziert.
Ausgangspunkt für diese Lösungskonzepte war das bereits in den Entscheidungsbäumen vorhandene Expertenwissen. Damit sollte eine erneute Befragung der Experten vermieden werden und nur auf der Basis des vorhandenen Wissens Lösungsmöglichkeiten überdacht werden.

6.2 Anwendung

Bayessche Netze werden verwendet, um Unsicherheiten in der Wissensrepräsentation zu modellieren. In der graphischen Darstellung stellt jeder Knoten eine Zufallsvariable dar, die es zu entscheiden gilt, und besitzt eine Tabelle der bedingten Wahrscheinlichkeiten. Diese bedingten Wahrscheinlichkeiten werden aus den Zuständen der Elternknoten berechnet.
Der Aufbau Bayesscher Netze für den Intelligent Assistant erfolgt hinsichtlich der für ein beschriebenes Problem notwendigen Informationen. Der Experte benötigt diese im Trouble–Ticket zur Weiterverarbeitung und die Tickets entlasten ihn wiederum von seiner Routinearbeit.
Die Wahrscheinlichkeiten im Bayesschen Netz verändern sich dynamisch durch Propagierung

über die Kanten. Gilt eine Annahme als sicher, so wird deren Wahrscheinlichkeitswert in der bedingten Wahrscheinlichkeitstabelle auf einen Wert möglichst nahe 1 gesetzt. Da eine der Annahmen eine höhere Wahrscheinlichkeit besitzt, beeinflußt sie die abhängigen Wahrscheinlichkeiten und es kann eine Tendenz festgestellt werden. Ergeben sich noch weitere sichere Annahmen in dieser Abhängigkeit, so wird sich eine weitere Frage– oder Testrichtung herauskristallisieren. Dies ist diejenige Richtung, in welche ausgedrückt wird, wie wahrscheinlich diese spezielle Information noch benötigt wird. Dabei sind auch die Restkosten, die sich auf diesem Weg für die gesamte Diagnose noch ergeben können, berücksichtigt. Die Angabe der Restkosten erfolgt immer für den wahrscheinlichsten Fall.

6.3 Einsatzgebiete

Der Intelligent Assistant wird im *Help Desk* des LRZ für die Problembereiche Durchsatz, Mail und Verbindung eingesetzt. Jedes dieser Gebiete wird im Intelligent Assistant als individueller Bereich realisiert. Da in jedem Problembereich zusätzlich eigene Kriterien gelten und andere Informationen wichtig sind, werden im folgenden Lösungskonzepte für Durchsatz, Mail und Verbindung vorgestellt.

6.3.1 Durchsatzproblem

Entwurf

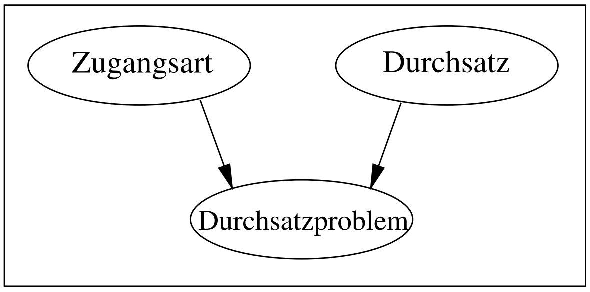

Abbildung 6.1: Bayessches Netz für das Durchsatzproblem

Von den Entscheidungsbäumen ist bekannt, daß für diese Fehlerart die Art des Zugangs und der momentane Durchsatz relevante Faktoren sind. Das Netz besitzt Abhängigkeiten von den beiden Knoten *Durchsatz* und *Zugangsart* zum Knoten *Durchsatzproblem*. Die Experten teilen die beiden erstgenannten Knoten in realitätsnahe Zustände ein (siehe *Tabelle 6.1 und Tabelle 6.2*).
Im Leibniz–Rechenzentrum existiert die Zugangsart der Wählverbindung, der Standleitung und des WAN–Zugangs. Jede dieser Zugangsarten besitzt eigene Kategorien an Durchsatzgeschwindigkeiten. So ist ein Durchsatz von 64 kBit/s oder 2 MByte für Standleitungen üblich. In Wide Area Network Verbindungen und bei Wählleitungsverbindungen reichen die Durchsätze von 9600 Byte bis 56000 Byte. Insbesondere ISDN–Verbindungen teilen sich in 64 kBit/s und 128 kBit/s ein. Der Knoten *Zugangsart* erhält die drei genannten Zugangsarten als Zustände. Der Knoten *Durchsatz* wird von den Experten mit mathematischen Ausdrücken belegt. Jeder

Zugangsart	
Standleitung	0.x
WAN	0.x
Wählverbindung	0.x
$\sum$	1

Tabelle 6.1: a priori Wahrscheinlichkeiten: *Zugangsart*

Durchsatz	
$\ll$ 64 kBit/s	0.x
$\ll$ 128 kBit/s	0.x
$\sum$	1

Tabelle 6.2: a priori Wahrscheinlichkeiten: *Durchsatz*

Zustand deckt einen Wertigkeitsbereich der Durchsatzzahlen ab ($\ll$ Durchsatz). Um die Tabelle nicht zu groß werden zu lassen, wurden nur zwei Bereiche ausgewählt (siehe *Tabelle 6.3*). Für den Knoten *Durchsatzproblem* wird aufgrund der Abhängigkeiten das Schema einer Tabelle erstellt, in das die Experten die Wahrscheinlichkeiten eintragen. Diese Tabelle enthält in einer Matrix sämtliche mögliche Kombinationen der Zustände der Knoten *Zugangsart* und *Durchsatz*.

Wahrscheinlichkeiten

Für das Durchsatzproblem sollen hier *Tabelle 6.1* und *Tabelle 6.2* beispielhaft aufgeführt werden.
Die a priori Wahrscheinlichkeiten für die Knoten *Zugangsart* und *Durchsatz* werden anhand der bekannten Verteilung (prozentualer Anteil der jeweiligen Zugangsart an der Gesamtanzahl der Zugänge und prozentualer Anteil des jeweiligen Durchsatzbereiches am Gesamtdurchsatz) festgelegt. Die Summe ergibt 100%.
Danach bestimmen die Experten in der Tabelle des Knotens *Durchsatzproblem* die Felder, die als Kombination nicht realistisch sind oder nicht existieren.
Im Beispiel ist dies der Bereich ISDN, dessen Durchsatz nicht größer als 128 kBit/s sein kann. Demzufolge kann bei einem derartigen Durchsatz kein Verbindungsproblem auftreten und die Wahrscheinlichkeit in *Durchsatzproblem nein* auf 0.9 gesetzt werden. Gleichzeitig dazu im Feld *Durchsatzproblem ja* kann die Wahrscheinlichkeit auf 0.1 gesetzt werden. Eine sichere Annahme, die in der Summe 1 ergibt.

	Standleitung		WAN		Wählverbindung	
	$\ll$ 64 kB	$\ll$ 128 kB	$\ll$ 64 kB	$\ll$ 128 kB	$\ll$ 64 kB	$\ll$ 128 kB
Durchsatzproblem *ja*	0.9	0.9	0.9	0.9	0.9	0.1
Durchsatzproblem *nein*	0.1	0.1	0.1	0.1	0.1	0.9

Tabelle 6.3: bedingte Wahrscheinlichkeiten: *Durchsatzproblem*

Unterschreitet dagegen der Durchsatz bei ISDN 64 kBit/s, liegt sicherlich ein Durchsatzproblem vor. Die Wahrscheinlichkeiten für *Durchsatzproblem ja/nein* kann dann entsprechend belegt werden.
Das Expertenwissen, das sich in diesem Bayesschen Netz verbirgt, ist die Korrelation, wie hoch der Durchsatz für die jeweilige Zugangsart ist und wo die Grenzen liegen, um als Fehler erkannt zu werden.
Die in *Tabelle 6.3* aufgelistete Aufteilung ist nur ein Beispiel dafür, wie die Aufteilung der Wertebereiche im Durchsatzknoten erfolgen kann. Wählt man mehrere Bereiche, so spiegelt sich dies auch in der Größe der Tabelle der bedingten Wahrscheinlichkeiten wieder.

Bewertung

Das modellierte Bayessche Netz ist sehr einfach in seinem Aufbau und in seiner Struktur. Auch mit den Wahrscheinlichkeitstabellen wird deutlich, wie Wissen der Experten ausgedrückt und formuliert werden kann.
Ein einfaches Problem, wie es der Intelligent Assistant durch Vergleich der Durchsatzwerte bewerkstelligt, kann mit einem einfach strukturierten Bayesschen Netz dargestellt werden. Dem Wissenstand, den die Entscheidungsbäume repräsentieren, ist das Netz gleichwertig, da es nicht mehr Informationen enthält und verarbeitet.
Es kann als sinnvolle Grundlage dienen, um ein komplexes Expertensystem für den Intelligent Assistant auf– und ausbauen zu können. Für ein derartiges System ist der Einsatz Bayesscher Netze gerechtfertigt.

6.3.2 Mailproblem

Informationen

Um ein Mailproblem diagnostizieren zu können, benötigen die Experten einige Informationen vom Benutzer des Intelligent Assistant oder, falls dies telefonisch geschieht, vom Fehlermeldenden:

- Angaben über die Adresse des Senders und des Empfängers
- Konfiguration des Senderechners, wobei folgende Auswahlmöglichkeiten derzeit zur Verfügung stehen:
  - PC im Münchener Hochschulnetz (MHN) mit TCP/IP
  - PC mit Modem über PPP–Verbindung
  - UNIX Rechner im Münchener Hochschulnetz (MHN)
  - andere Konfiguration
- Welches Mailsystem wird vom Senderechner benutzt:
  - Endora
  - Pegasus
  - Sonstige
- Wird zum ersten Mal mit diesem System eine Mail versandt?

- Haben Konfigurationsänderungen am Rechner stattgefunden?

Entwurf 1

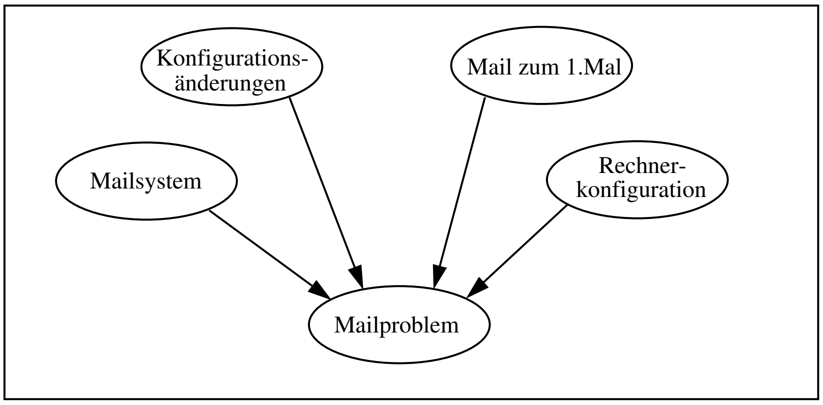

Abbildung 6.2: Bayessches Netz für den 1.Entwurf des Mailproblems

Außer den Angaben über die Adressen oder Namen der betroffenen Rechner, kristallisieren sich vier Elternknoten oder Frageknoten heraus:

- Rechnerkonfiguration
- Mailsystem
- Mail zum 1. Mal versenden
- Konfigurationsänderungen

Der Aufbau der Tabellen für die a priori Wahrscheinlichkeiten erfolgt in den Knoten *Konfiguration* und *Mailsystem* nach der oben aufgezeigten Auswahl. Die Knoten *Mail zum 1. Mal* und *Konfigurationsänderungen* erhalten boolesche Tabellen mit den Antwortmöglichkeiten *ja* oder *nein*.
Die vier Ursachenknoten sind gleichberechtigt und führen damit zu einer Tabelle der bedingten Wahrscheinlichkeiten im Knoten *Mailproblem*.
In den Entscheidungsbäumen wird auf die Antwort, daß sich der Rechner als UNIX–Rechner im MHN befindet, die Frage nach dem Mailsystem nicht gestellt. Diese Tatsache wird in der bedingten Wahrscheinlichkeitstabelle im Knoten *Mailproblem* durch die entsprechende Wertigkeit der Wahrscheinlichkeit modelliert. Durch diese Aussage sind die Kombinationen von UNIX mit dem jeweiligen Mailsystem in der Zeile *Mailproblem ja* mit 0% und in der Zeile *Mailproblem nein* mit 100% Wahrscheinlichkeit auszufüllen, da dies sichere Annahmen (Evidenzen) sind.

Entwurf 2

Der Entwurf 1 wirft hinsichtlich der zu erstellenden bedingten Wahrscheinlichkeitstabelle im Knoten *Mailproblem* Schwierigkeiten auf. Die Größe der Tabelle ist nicht übersichtlich und kann eine wesentliche Hürde für den Experten darstellen.

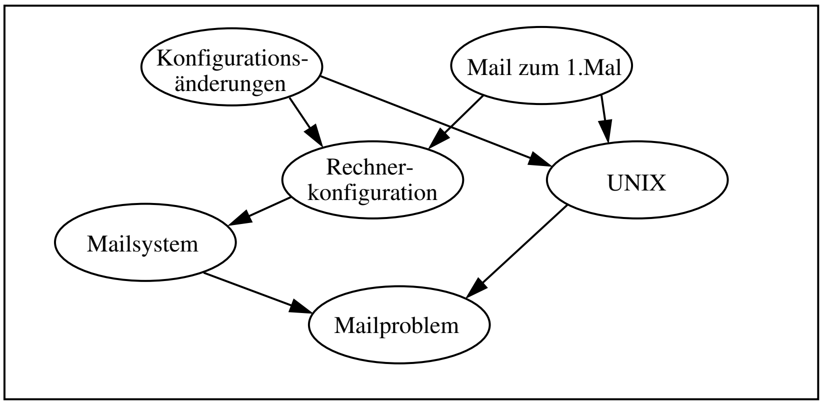

Abbildung 6.3: Bayessches Netz für den 2.Entwurf des Mailproblems

Da bekannt ist, daß im Falle eines UNIX–Rechners das verwendete Mailsystem als Information nicht notwendig ist, läßt sich das oben entworfene Bayessche Netz für das Mailproblem ummodellieren.
In diesem Entwurf werden die beiden Knoten *Mail zum 1. Mal* und *Konfigurationsänderung* zu Eltern– bzw. Ursachenknoten. Diese beiden Knoten sind jeweils Ursache für einen *UNIX*–Knoten und einen *Mailsystem*–Knoten. Die verschiedenen Mailsysteme können nur für die Rechnerkonfigurationen, die nicht UNIX–Rechner sind, eine Fehlerursache darstellen. Die beiden Rechnerkonfigurationsknoten wiederum führen zum *Mailproblem*–Knoten.

Ausbau

Da von den Experten und aus dem bisherigen Einsatz des Intelligent Assistant bekannt ist, daß ein Mailproblem in der überwiegenden Anzahl von Fällen letztendlich ein Verbindungsproblem ist, muß hier noch eine Lösung gefunden werden. Ein Vorschlag ist, durch einen weiteren spezifischen, zusätzlich zu den bereits vier vorhandenen gleichberechtigten Knoten, weitere Frageknoten einzuführen. Z.B. den Test *traceroute*, der den Kontakt zum Verbindungsproblem bzw. zum entsprechenden Bayesschen Netz herstellt.

Bewertung

Durch die Modellierung der beiden Bayesschen Netze wird deutlich, wie das Mailproblem aus den Entscheidungsbäumen unterschiedlich erarbeitet werden kann.
Der *Entwurf 2* konnte deshalb modelliert werden, da durch den Ablauf des Entscheidungsbaumes (siehe *Abbildung 4.1* neue Informationen gewonnen wurden. Die Information, daß für die *Rechnerkonfiguration UNIX* keine Aussagen über ein verwendetes *Mailsystem* notwendig sind, kann in einen separaten Knoten ausgelagert werden. Damit ist *UNIX* nur von *Konfigurationsänderungen* und *Mail zum 1. Mal* abhängig. Die restlichen Rechnerkonfigurationen werden in einem weiteren Knoten berücksichtigt und in den Zuständen repräsentiert.
In diesem Entwurf entstehen Abhängigkeiten zwischen den Rechnerkonfigurationsknoten und den beiden a priori Knoten. In diesem Entwurf können die Abhängigkeiten modelliert werden, da sie in *Entwurf 1* eine lineare Kette ergeben.
Der Wissensstand entspricht wiederum dem des Entscheidungsbaumes.

Das Ergebnis einer Diagnose im Bayesschen Netz des *Entwurf 2* gibt an, mit welcher Wahrscheinlichkeit ein Mailproblem besteht.
Für einen Experten sind die Aussagen *es besteht zu 60% Wahrscheinlichkeit ein Mailproblem* (Bayessches Netz) und *es besteht ein Mailproblem* (Entscheidungsbaum) gleichwertig. Er wird in jedem Fall ein Mailproblem annehmen.

6.3.3 Verbindungsproblem

Informationen

Das Verbindungsproblem wurde in den Entscheidungsbäumen so konstruiert, daß nach Angabe des Quell– und Zielrechners der Benutzer im Intelligent Assistant lediglich die Auswahl hat zwischen den Tests *ping* oder *traceroute*. Bei letzterem wird dem Benutzer die Anwendung erleichtert, indem dieser Test vom Intelligent Assistant Server ausgeführt wird. Den Test *ping* hat der Anwender lokal auf seinem Rechner durchzuführen.
Das Wissen, das sich in dieser Vorgehensweise im Verbindungsproblem verbirgt, besteht lediglich aus der Anwendung und den Ergebnissen der beiden Testvarianten. Nach Ablauf einer der oben aufgeführten Testroutinen wird das Ergebnis des durchgeführten Tests entweder automatisch oder manuell durch den Benutzer in ein Trouble–Ticket eingetragen.
Das „Wissen“, das sich im Verbindungsproblem verbirgt, besteht darin, daß der Experte im Trouble–Ticket die gesammelten Informationen einsehen kann. Die geleistete Vorarbeit erleichtert dem Experten die Diagnose insofern, als daß bereits ein Test durchgeführt wurde.
Klinnert hat in seiner Arbeit *[Klin 98]* ein Bayessches Modell vorgestellt, das es erforderlich macht, jede mögliche Fehlerkomponente eines Rechnernetzes im Bayesschen Netz als Knoten zu spiegeln. Dies hat den Nachteil, daß der in dieser Arbeit erwähnten Dynamik der Rechnernetze in puncto Komponentenerweiterung wenig Rechnung getragen wird. Sobald dem Rechnernetz eine weitere Komponente hinzugefügt wird, muß das Netz um Knoten für Ursachen und Abhängigkeiten erweitert werden.

Entwurf

Der Entwurf eines Bayesschen Netzes kann sich hier nur an den Testroutinen, die durchgeführt werden können und an den Zusatzinformationen, die an seinen Schnittstellen zur Verfügung stehen, orientieren. Die Testroutine *traceroute* kann nur vom Intelligent Assistant Server aus durchgeführt werden. Ihr Testergebnis ist erfolgreich, wenn der Rechner, in diesem Falle kommen Quell– und/oder Zielrechner und sämtliche dazwischen liegenden Knoten in Frage, erreicht wird. *Ping* kann auch vom Server aus durchgeführt werden, um zu prüfen, ob der Benutzerrechner erreichbar ist. Will der Benutzer im Verlauf des Intelligent Assistant umgekehrt den Server erreichen, muß *ping* lokal ausgeführt werden.

Wahrscheinlichkeiten

Wird das Bayessche Netz, wie in *Abbildung 6.4* dargestellt, realisiert, können die einzelnen Teilstrecken einer getesteten Verbindung als Zustände der Testknoten eingeteilt werden. Für den Test *ping* bieten sich die Strecken Quelle–Ziel, Quelle–Server, Server–Quelle und Server–Ziel an. Jede dieser Richtungen kann korrekt arbeiten oder nicht.
Für *traceroute* können die Verbindungen Server–Quelle und Server–Ziel gewählt werden,

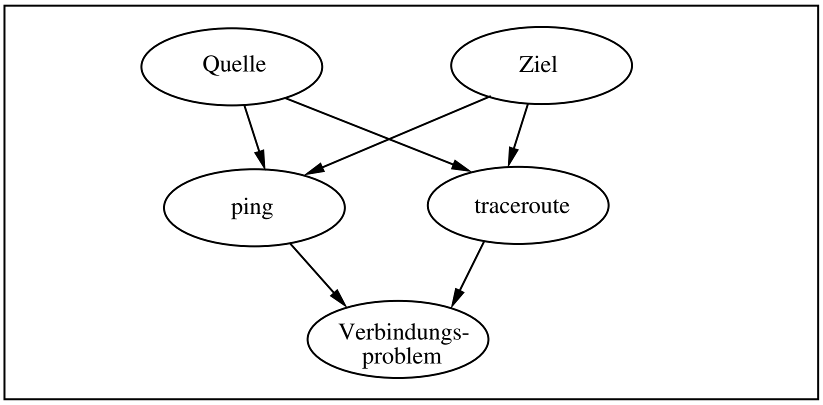

Abbildung 6.4: Bayessches Netz für das Verbindungsproblem

ping		
	Quelle	Ziel
Q–Z OK	0.x	0.x
Q–Z NOK	0.x	0.x
Q–S OK	0.x	0.x
Q–S NOK	0.x	0.x
S–Q OK	0.x	0.x
S–Q NOK	0.x	0.x
S–Z OK	0.x	0.x
S–Z NOK	0.x	0.x

Tabelle 6.4: bedingte Wahrscheinlichkeiten *ping*

wiederum mit jeweils korrekter oder inkorrekter Funktionsweise. Diese Teilstrecken werden auch in *Abbildung 3.1* im Testumfeld des Verbindungsproblems, das dem Benutzer des Intelligent Assistant angezeigt wird, dargestellt.
Quelle und Ziel sind in jedem Fall bekannt, da diese Informationen bereits im ersten Schritt der Informationssammlung im Intelligent Assistant bekannt gemacht werden müssen.
Für den Knoten des Verbindungsproblems ergibt sich dann die bedingte Wahrscheinlichkeitstabelle aus der Kombination der beiden Tabellen der Knoten *ping* (siehe *Tabelle 6.4*) und *traceroute* (siehe *Tabelle 6.5*).

Bewertung

Ausgehend von einem sehr komplexen Bayesschen Netz aus der Vorgängerarbeit *[Klin 98]*, wurde versucht, das Verbindungsproblem zu vereinfachen.
Die Entscheidungsbäume für das Verbindungsproblem enthalten die Form des Wissens lediglich in der Reihenfolge der Abarbeitung der Informationsbeschaffung. Während einer Diagnose wird eine der Testroutinen *ping* oder *traceroute* durchgeführt und deren Ergebnis protokolliert. Die Umsetzung des Wissens kann nur mit den Informationen der Testroutinen und den An-

traceroute		
	Quelle	Ziel
S–Q OK	0.x	0.x
S–Q NOK	0.x	0.x
S–Z OK	0.x	0.x
S–Z NOK	0.x	0.x

Tabelle 6.5: bedingte Wahrscheinlichkeiten *traceroute*

gaben über Quell– und Zielrechner erfolgen.
Um die Anforderung einer Historie für jeden möglichen Quell– und Zielrechner zu erfüllen, muß in das Bayessche Netz für jeden Rechner ein Knoten eingeführt werden. In diese Knoten kann der Lernalgorithmus die zugehörigen Wahrscheinlichkeiten, die sich aus vorhergehenden Diagnosen ergeben, eintragen.

Kapitel 7

Zusammenfassung und Ausblick

7.1 Zusammenfassung

Diese Arbeit umfaßt die Untersuchung des Einsatzes Bayesscher Netze im Intelligent Assistant, wie er am Leibniz–Rechenzentrum in München eingesetzt wird. Derzeit ist das Expertenwissen für den First Level Support mit dem Intelligent Assistant in Entscheidungsbäumen implementiert. Aus der Analyse der bestehenden Situation ergeben sich Anforderungen für den Aufbau und den Einsatz der Bayesschen Netze.
Beginnend mit der Wissensakquisition und weiter im Entwurf der Netze und der Umsetzung des Wissens, sollten die Experten durch graphische Editoren unterstützt werden.
Weitere Unterstützung sollte in der Anwendung erfolgen, insbesondere über eine Schnittstelle zu einer Inferenzmaschine, um die Berechnung und Propagierung der Wahrscheinlichkeiten und bedingten Wahrscheinlichkeiten zu automatisieren. Die Inferenzmaschine ist ein Algorithmus.
Um dies realisieren zu können, sind drei weitere Algorithmen notwendig. Ein erster Algorithmus sollte dem Experten das Eintragen der Wahrscheinlichkeiten in die bedingten Wahrscheinlichkeitstabellen erleichtern und automatisieren. Der Fragealgorithmus, der sich wiederum des Inferenzalgorithmus zur Berechnung bedient, entscheidet, welche Frage oder welcher Test als nächstes durchgeführt wird. Ein Lernalgorithmus sorgt dafür, daß das Bayessche Netz ein Erinnerungsvermögen besitzt, um das Bayessche Netz lernfähig zu machen.
Für die Realisierung der Anforderungen an Bayessche Netze stehen kommerzielle Werkzeuge zur Verfügung, die aber an diese speziellen Bedürfnisse noch angepaßt werden müssen, oder es können nur Teile des Werkzeuges verwendet werden, wie es bei Siemens im Einsatz ist.
Ein Ansatz zur Lösung in den drei Problembereichen des Intelligent Assistant wird im vorhergehenden *Kapitel 6* dieser Arbeit aufgezeigt.

7.2 Designentscheidungen

Die Aufgabenstellung wurde in dieser Weise interpretiert, daß ein einfaches Werkzeug wie der Intelligent Assistant, durch einfache Mittel und Strukturen unterstützt werden soll. Ein Vergleich:
Ist an einem Fahrzeug die Scheinwerferglühbirne defekt, so ist es nicht notwendig, als

Zusatzinformation zur Reparatur den Ölstand zu kontrollieren.
Dieser Vergleich soll, im übertragenen Sinn, eine pragmatische Lösung des Einsatzes der Bayesschen Netze im Intelligent Assistant motivieren. Er zeigt, daß durch eine einfache Strukur und Aufbau der Bayesschen Netze ein erfolgreicher und für die Experten zufriedenstellender Einsatz erreicht werden kann, da eine Entlastung der Experten durch Automatisierung der Trivialfragen und –probleme möglich ist.
Das Expertenwissen aus den Entscheidungsbäumen im Intelligent Assistant, gekoppelt mit den Informationen aus den bestehenden Datenbanken des *Help Desk*, kann in Bayessche Netze umgesetzt werden.

7.3 Herausforderungen

Ziel dieser Arbeit war es, das Spannungsfeld zwischen den verschiedenen Aspekten des Netzmanagements am LRZ und dem komplexen Feld der Künstlichen Intelligenz mit Expertensystemen und seinen Wissensbasen zu analysieren. Die Anwendung Bayesscher Netze als mögliche Form des implementierten Expertenwissens im Intelligent Assistant sollte aufgezeigt werden. Eine Reihe von Aspekten wurde motiviert und daraus in Teilen Lösungsvorschläge und Einsatzbereiche erarbeitet.
Gerade diese Kombination aus technischem und dynamischem Einsatzgebiet im Management von Rechnernetzen und dem theoretischen Gebiet der Künstlichen Intelligenz mit seinen komplexen Expertensystemen bietet verschiedene Ansätze zur Umsetzung der aufgezeigten Anforderungen.

7.4 Fazit

Wie aufgezeigt wurde, stammen Bayessche Netze aus einem sehr komplexen Bereich der Wahrscheinlichkeitstheorie und Kognitionswissenschaft. Ihr häufigstes Einsatzgebiet ist derzeit die Medizin, ein Gebiet das nicht allzu dynamisch ist, da beim Menschen nicht jeden Tag neue Krankheiten entdeckt werden, die es zu diagnostizieren gilt.
Im Gegensatz dazu steht das Netzmanagement, das durch ständige Veränderungen in der Topologie und in der Infrastruktur gekennzeichnet ist. Neue Komponenten, neue Anwendungen, die installiert werden, müssen im Netzmanagement berücksichtigt und integriert werden.
Werden Bayessche Netze wiederum in diesem Gebiet eingesetzt, so ist das auch unter einem kritischen Aspekt zu betrachten.
Die Experten werden zuerst in einer Wissenakquisition befragt, sie bestimmen die Abhängigkeiten der einzelnen Informationen, die für eine Diagnose notwendig sind und sie haben Wahrscheinlichkeiten einzuschätzen, die durchaus subjektiven Charakter haben. Die Reibungsfläche zwischen den einzelnen Experten dabei ist groß.
Als Beispiel sei hier das Bayessche Netz für das System *PATHFINDER* erwähnt. Dieses Expertensystem deckt ca. 60 Lymphknotenkrankheiten und 100 Symptome und Testergebnisse ab. Die Experten benötigten dabei für die Festlegung des Vokabulars 8 Stunden, für die Abhängigkeitstopologie 35 Stunden und für die Schätzung von 14000 Wahrscheinlichkeiten 40 Stunden.
Für die Automatisierung im Intelligent Assistant müßte kein derart großes und komplexes

Netz erstellt werden. Dieses Beispiel soll lediglich aufzeigen, daß für jeden Diagnosebereich in Expertensystemen adäquate Mittel existieren und angewandt werden. Deshalb ist in der Erstellung und Modellierung Bayesscher Netze für den Intelligent Assistant auf die Dimension zu achten. Hier kann einmal ein sehr komplexes Bayessches Netz für ein einfaches Problem entstehen. Der Aspekt der einfachen Struktur kann auf der anderen Seite zeigen, daß es weitere einfache Möglichkeiten gibt, das Expertenwissen im Intelligent Assistant zu implementieren.
Hierzu Göran Kauermann (wissenschaftlicher Assistent im Institut für Statistik, LMU) über den Einsatz der Bayesschen Netze im Rechnernetzmanagement: „Mit einem Mercedes SL nur durch 30km–Zonen fahren".
Gerechtfertigt wird der Einsatz der Bayesschen Netze in komplexen Expertensystemen wie es *PATHFINDER* darstellt. Der Intelligent Assistant hingegen ist ein Werkzeug für den First Level Support. Der Aufwand zur Erstellung eines komplexen Expertensystems zum Einsatz Bayesscher Netze ist hoch. Die Anwendung des Intelligent Assistant ist eine „pragmatische" (Zitat Prof. Dr. Heinz–Gerd Hegering) und kann mit entsprechenden und bereits existierenden Mitteln umgesetzt werden.
Der Experte „Mensch" stellt letztendlich immer den Ansprechpartner für Probleme und auftretende Fehler dar. Seine Entlastung ist gerechtfertigt und notwendig. Dies sollte aber mit adäquaten Mitteln erfolgen.

Abkürzungsverzeichnis

Abkürzung	**Bedeutung**
A	
API	Application Programming Interface
C	
CPT	Conditional Probability Table
CGI	Common Gateway Interface
D	
DFN	Deutsches Forschungsnetz
DAG	*directed acyclic graph* (Gerichteter Azyklischer Graph)
DNS	Domain Name Service
DNS	Domain Name Server
F	
FTP	File Transfer Protocol
FAQ	Frequently Asked Questions
H	
HTML	Hypertext Markup Language
I	
IA	Intelligent Assistant
ISDN	Integrated Services Digital Network
IP	Internet Protocol
IT	Informationstechnologie
K	
KI	Künstliche Intelligenz
L	
LRZ	Leibniz–Rechenzentrum der Akademie der Wissenschaften
M	
MNM–Team	Münchener Netzmanagement–Team
MHN	Münchener Hochschulnetz

P

ping	packet internet groper
PPP	Point to Point Protocol
PC	Personal Computer

Q

QoS	Quality of Service

S

SLA	Service Level Agreement

T

TTS	Trouble–Ticket–System
TT	Trouble–Ticket
TCP	Transmission Control Protocol
ttl	time to live

W

WWW	World Wide Web
WAN	Wide Area Network

Literaturverzeichnis

[BaBa 97] Baras, J. und Ball, M. und Gupta S. und Viswanathan P. und Shah P.: *Automated Network Fault Management. Military Communication Conference MILCOM 97*, Monterey, 1997.

[BaLi 98] Baras, J. und Li, H. und Mykoniatis G.: *Integrated, Distributed Fault Management for Communication Networks.* Technical Report, University of Maryland, Center for Satellite and Hybrid Communication Networks, Dept. of Electrical Engineering and Institute for Systems Research, 1998.

[Ball 92] Ball, L.: *Cost–Efficient Network–Managment.* McGraw–Hill, 1992.

[BoKr 98] Borgelt, C. und Kruse, R. und Lindner G.: *Lernen probabilistischer und possibilistischer Netze aus Daten: Theorie und Anwendung.* KI: Künstliche Intelligenz — Forschung, Entwicklung, Erfahrungen, Themenheft: Data Mining(1), März 1998.

[Bree 92] Breese, J.: *Construction of Belief and Decision Networks.* Computational Intelligence, 8(4):624 – 647, 1992.

[BrHe 94] Breese, J. und Heckerman, D.: *Decision–Theoretic Case–Based Reasoning.* Technical Report, MSR–TR–95–03, Microsoft Research, Advanced Technology Division, November 1995 revised.

[BrKo97] Breese, J. und Koller, D.: *Bayesian Networks and Decision Theoretic Reasoning for Artificial Intelligence. American Conference for Artificial Intelligence AAAI 97*, Juli 1997.

[CFSD 90] Case, J. D., M. Fedor, M. L. Schoffstall und C. Davin: *RFC 1157: Simple Network Management Protocol (SNMP).* Technischer Bericht Mai 1990.

[Char 91] Charniak, E.: *Bayesian Networks without Tears.* Artificial Intelligence Magazine, 12(4):50 – 63, 1991.

[Chyl 96] Chylla, P.: *Integrierte Managementkonzepte für Betreiber von IT–Plattformen.* it+ti, Informationstechnik und Technische Informatik, 38(6):43 – 47, 1998.

[Dreo 97] Dreo Rodosek, G.: *Entwicklung des „Intelligent Assistant“ am LRZ.* Projektpapier, Leibniz–Rechenzentrum München, 1997.

[DrKa 98] Dreo Rodosek, G. und Kaiser, T.: *Intelligent Assistant: User–Guided Fault Localization.* Technischer Bericht, Leibniz–Rechenzentrum München, 1998.

[Gamm 95] Gamerman, A. (Herausgeber): *Probabilistic Reasoning and Bayesian Belief Networks.* Alfred Waller Limited, 1995.

[Görz 95] Görz, G. (Herausgeber): *Einführung in die künstliche Intelligenz.* Addison–Wesley, 2. Auflage, 1995.

[HAN 99a] Hegering, H.-G. und Abeck, S. und Neumair B.: *Integriertes Management vernetzter Systeme — Konzepte, Architekturen und deren betrieblicher Einsatz.* dpunkt-Verlag, 1999.

[HeBr 94] Heckerman, D. und Breese, J. und Rommelse K.: *Troubleshooting under Uncertainty.* Technical Report, MSR–TR–94–07, Microsoft Research, Advanced Technology Division, Januar 1994 revised.

[HeBr 95] Heckerman, D. und Breese, J. und Rommelse K.: *Decision–Theoretic Troubleshooting.* Communications of the ACM, Uncertainty in AI, 38(3):49 – 57, März 1995.

[Heck 96] Heckerman, D.: *A Tutorial on Learning with Bayesian Networks.* Technical Report, MSR–TR–95–06, Microsoft Research, Advanced Technology Division, November 1996 revised.

[HeMa 95] Heckerman, D. und Mamdani, A. und Wellman M.: *Real–World Applications of Bayesian Networks.* Communications of the ACM, Uncertainty in AI, 38(3):26, März 1995.

[HeNe 98] Hegering, H.-G. und Neumair, B.: *Managementarchitekturen als Lösungsansätze für ein Netz– und Systemmanagement in heterogener Umgebung.* it+ti, Informationstechnik und Technische Informatik, 38(6):7 – 16, 1998.

[HeUh 97] Henke von, F. und Uhrmacher, A.: *Grundlagen der Künstlichen Intelligenz.* Vorlesungsskript, Universität Ulm, Fakultät für Informatik, Abt. Künstliche Intelligenz, 1997.

[HeWe 95] Heckerman, D. und Wellman M.: *Bayesian Networks.* Communications of the ACM, Uncertainty in AI, 38(3):27 – 30, März 1995.

[HoTr 96] Hofmann, R. und Tresp, V.: *Discovering Structure in Continous Variables Using Bayesian Networks.* In: *Advances in Neural Information Processing Systems 8.* MIT Press, Cambridge (MA), 1996.

[HUGIN98] *HUGIN GUI Version 5.2 Lite und Manuals der Fa. Hugin Expert A/S, Niels Jernes Vej 10, DK–9220 Aalborg.* freie Version, Dezember 1998.

[Jens 96] Jensen, F. V.: *An Introduction to Bayesian Networks.* Springer NY, 1. Auflage, 1996.

[Kaue 97] Kauermann, G.: *Probabilistische Expertensystem.* Vorlesungsskript, *vormals* TU Berlin, *jetzt* Institut für Statistik LMU, 1997.

[KiBr 95] Kipper, B. und Brants, T. und Plach M.: *Bayessche Netze: Ein einführendes Beispiel.* Bericht Nr. 4, Graduiertenkolleg Kognitionswissenschaft, Universität des Saarlandes, Fachbereich 14, 1995.

[Klin 98] KLINNERT, P.: *Methoden der Neuroinformatik im Netz– und Systemmanagement.* Diplomarbeit, Technische Universität München, Mai 1998.

[Klus 97] KLUSSMANN, N.: *Lexikon der Kommunikations– und Informationstechnik: Telekommunikation, Datenkommunikation, Multimedia, Internet.* Hüthig–Verlag, 1997.

[Lien 96] LIENEMANN, G.: *TCP/IP–Praxis: Design, Management, Analyse.* Verlag Heinz Heise GmbH & Co KG, 1996.

[Mode 94] MODES, A.: *Entwurf und Realisierung eines "Intelligent Assistant" für ein Trouble-Ticket-System.* Diplomarbeit, Technische Universität München, Mai 1994.

[Pearl 88] PEARL, J.: *Probabilistic Reasoning in Intelligent Systems, Networks of Plausible Inference.* Morgan Kaufman Publishers Inc., 2. Auflage, 1988.

[Pupp 91] PUPPE, F.: *Einführung in Expertensysteme.* Springer–Verlag, 2. Auflage, 1991.

[Rose 93] ROSE, M.: *Einführung in die Verwaltung von TCP/IP–Netzen: Netzwerkverwaltung und das Simple Network Management Protocol.* Hanser–Verlag, 1993.

[Stin 90] STINE, R. H.: *RFC 1147: FYI on a network management tool catalog: Tools for monitoring and debugging TCP/IP internets and interconnected devices.* Technischer Bericht April 1990.

[Thij 87] THIJS, W. L.: *Fault Management.* Dissertation, Technische Universität Delft, Januar 1987.

[Wale 99] WALES, E.: *Helpdesks fördern die Selbsthilfe der Kunden.* Computer Zeitung, (Nr. 8):30, 25. Februar 1999.

[Weis 96] WEISHAUPT, O.: *Implementierung des Intelligent Assistant für das LRZ.* Fortgeschrittenenpraktikum, Technische Universität München, August 1996.