



Otto-Friedrich-Universität Bamberg
Lehrstuhl für Praktische Informatik



Bachelor

im Studiengang Wirtschaftsinformatik
der Fakultät Wirtschaftsinformatik und Angewandte Informatik
der Otto-Friedrich-Universität Bamberg

Zum Thema:

Automatisierung der Datenerfassung für Wissensdatenbanken im technischen Kontext

Vorgelegt von:
Vasilyev Petr

Themensteller:
Prof. Dr. Guido Wirtz

Abgabedatum:
31.03.2017

Inhaltsverzeichnis

1	Einführung	1
1.1	Motivation	1
1.2	Verwandte Arbeiten	1
1.3	Zielsetzung	2
1.4	Aufbau der Arbeit	2
2	Grundlagen von Expertensystemen	3
2.1	Begriffsdefinition	3
2.2	Architektur eines Expertensystems	5
2.3	Wissensbasis	7
2.4	Wissensakquisitionskomponente	10
3	Automatisierung der Wissenserfassung	12
3.1	Die Wissensträgerschnittstelle	12
3.2	Maschineller Wissenserwerb	14
3.3	Schnittstelle für Validierung und Speicherung	15
4	Implementierung	16
4.1	Ansatz 1	16
4.2	Ansatz 2	16
4.3	Ansatz 3	16
5	Evaluation und Fazit	16
5.1	Test 1	16
5.2	Test 2	16
5.3	Fazit	16
6	Ausblick	16
	Literaturverzeichnis	17

Abbildungsverzeichnis

1	Begriffsabgrenzung, [Mat00, S.30]	4
2	Expertensystems nach Haun, [Mat00, S.126]	5
3	Phasen der Expertensystementwicklung, [GD94, S.138]	7
4	Expertensystems nach Beierle und Kern-Isberner, [CG14, S.18]	8
5	Wissensakquisitionskomponente	11
6	Die Struktur und die Informationsflüsse im DSS, [SK09, S.98]	13

Abkürzungsverzeichnis

IaaS Infrastructure-as-a-Service

PaaS Platform-as-a-Service

SaaS Software-as-a-Service

DSS Decision Support System

1 Einführung

1.1 Motivation

Die Idee von wissensbasierten Systemen entstand aus dem Anliegen, ein intelligentes System zu schaffen, das mithilfe des spezifischen Wissens die Menschen bei den Problemlösungen unterstützt [RP10, S.18]. Darin besteht der Unterschied zwischen einem konventionellen Informationssystem und einem wissensbasierten System. Informationssysteme sind schließlich datenbasiert und konzentrieren sich auf die Datenverarbeitung [RP10, S.19]. Eine Spezialisierung der wissensbasierten Systeme stellen Expertensysteme dar, in denen das Wissen letztendlich von Experten stammt. Ein wissensbasiertes System bzw. ein Expertensystem ist durch die Trennung zwischen der Wissensdarstellung eines Problembereichs und der Wissensverarbeitung gekennzeichnet [CG14, S.11].

Die Beschreibung des Wissens von einem wissensbasierten bzw. Expertensystem erfolgt in der Wissensbasis, die in Form einer Wissensdatenbank realisiert wird. Die initiale Datenbank wird in der Regel mithilfe manueller Datenerfassung aufgebaut [Kur92, S.70]. Ein Interview zwischen einem Wissensingenieurs und einem Wissensträger wird beispielsweise oft eingesetzt [GTW90, S.76]. Allerdings sind die manuellen Wissenserwerbsmethoden für die Weiterentwicklung der Wissensbasis hinsichtlich der Erweiterung und Aktualisierung der Wissensdatenbank eher schlecht geeignet, da sie fehleranfällig, kosten- und zeitintensiv sind. Aus diesem Grund wird seit langem versucht, den Prozess der Datenerfassung zu automatisieren. Eine Auswahl an bestehenden Ansätzen, die bezüglich der vorliegenden Arbeit relevant sind, wird im Abschnitt zu den verwandten Arbeiten gegeben. Trotz der vielen Herangehensweisen gibt es kein einheitliches Konzept, das einen allgemeinen Rahmen für die Automatisierung der Datenerfassung bildet.

1.2 Verwandte Arbeiten

Ein Konzept der Automatisierung der Wissenserfassung wird in [SK09] vorgestellt. Die Entwicklung der Wissensbasis umfasst dabei drei Phasen. In der ersten Phase wird die initiale Wissensdatenbank aufgebaut, die unvollständig und teilweise widersprüchlich sein kann. Die zweite Phase umfasst inkrementelle Erweiterung und Verbesserung der Wissensbasis. Dabei werden Methoden des maschinellen Lernens eingesetzt. Im Laufe der dritten Phase wird die Wissensbasis in Bezug auf Effizienz optimiert [Ghe92, S.1444]. Gheorghe Tecuci betont, dass bei jeder Phase eine Kooperation zwischen dem fachlichen Experten und dem System notwendig ist. Nach der maschinellen Datenerschließung werden die Ergebnisse manuell kontrolliert und anschließend in der Wissensbasis gespeichert. In dieser Weise lassen sich die Vorteile maschineller und manueller Datenerfassung optimal kombinieren [GD94, S.137], [Ghe92, S.1445].

Ein weiterer Ansatz wird in [SK09] thematisiert. Dabei handelt es sich um ein datenbasiertes Decision Support System (DSS), das die Unternehmensführung bei den Entscheidungen in Bezug auf die Produktionsoptimierung unterstützen soll. Allerdings werden die Störungen in der Produktion von Anlagenbedienern (Experten) mittels Erfahrungswissen intern behoben, ohne dieses Wissen weiterzugeben. Als Folge hat die Unternehmensführung kein umfangreiches Bild der Produktion und kann keine optimalen Entscheidungen treffen. Aus diesem Grund erweitern Gebus und Leiviskä das bestehende DSS mit einer Schnittstelle, um das Expertenwissen in die Datenbank zu integrieren [SK09, S.94]. All-

gemeiner betrachtet geht es um die Transformation eines datenbasiertes System in ein wissensbasiertes System. Auch hier wird versucht, die Daten mithilfe der engen Kooperation zwischen dem Menschen und dem System zu erfassen.

1.3 Zielsetzung

Das Ziel der vorliegenden Arbeit ist die Erarbeitung eines allgemeinen Konzeptes zur Automatisierung der Datenerfassung. Da die komplett automatisierte Datenerfassung sehr schwierig umzusetzen ist, liegt der Schwerpunkt dieser Arbeit auf der Kombination zwischen den manuellen und maschinellen Vorgehensweisen.

Beim technischen Kontext wird angedeutet, dass die Umsetzung im Rahmen eines bestimmten Anwendungsbereichs erfolgt. Es wird also kein Allgemeinwissen wie in [Nik16], sondern ein anwendungsbezogenes Wissen betrachtet. In diesem Zusammenhang wird das Konzept auf Basis eines Expertensystems entwickelt, da die Wissensbasis eines Expertensystems spezifischer ist als bei einem wissensbasierten System.

Die praktische Umsetzung soll auf Basis von *PaaSfinder*¹ erfolgen. Bei *PaaSfinder* handelt es sich um eine Web-Anwendung, die über eine Wissensdatenbank im Bereich Platform-as-a-Service (PaaS) verfügt. PaaS gehört neben Infrastructure-as-a-Service (IaaS) und Software-as-a-Service (SaaS) zum Themengebiet von Cloud Computing [PT11] und soll die Anwendungsentwicklung erleichtern, indem die Laufzeit- oder Entwicklungsumgebungen von einem PaaS-Anbieter dem Kunden vorkonfiguriert angeboten werden [Geo08, S.14]. Da *PaaSfinder* ein Open-Source Projekt ist, kann jeder der Datenbank von *PaaSfinder* beitragen. Allerdings ist die Mitwirkung mit einem hohen Aufwand verbunden und setzt Informatikkenntnisse voraus. Außerdem werden die Daten von *PaaSfinder* hauptsächlich auf den Webseiten von PaaS-Anbietern manuell erfasst. Dies stellt ein hohes Entwicklungspotential, die Erfassung von solchen Daten zu automatisieren.

1.4 Aufbau der Arbeit

Die vorliegende Arbeit ist wie folgt aufgebaut. In Kapitel 2 wird der Begriff und die grundlegende Architektur eines Expertensystems erläutert. Darauf folgend werden die Bestandteile, die für das Konzept relevant sind, näher betrachtet. Anschließend wird schematisch das Kontext der Datenerfassung dargestellt und Automatisierungsmöglichkeiten angesprochen. In Kapitel 3 wird genauer auf die Automatisierungsmethoden bei der Datenerfassung eingegangen. Dabei werden bestehende Forschungsergebnisse vorgestellt und konzeptuell verallgemeinert. Kapitel 4 umfasst die praktische Umsetzung der Erkenntnisse, in Kapitel 2 und 3 gewonnen wurden. In Kapitel 5 wird die Implementierung evaluiert und die Ergebnisse der Arbeit zusammengefasst. Anschließend wird in Kapitel 7 ein Ausblick für zukünftige Forschung im betrachteten Bereich gegeben.

¹<https://paasfinder.org>

2 Grundlagen von Expertensystemen

2.1 Begriffsdefinition

Ursprünglich waren Expertensysteme Anwendungsprogramme, die logische Schlussfolgerungen aus einer Wissensbasis ziehen konnten. Außerdem konnten sie überprüfen, ob eine Aussage aus einer vorhandenen Wissensbasis abgeleitet werden kann [Pet10, S.75]. Daher handelt es sich in der früheren Literatur meist um Anwendungen, die ihr Wissen in Form von logischen Ausdrücken darstellen und in der Lage waren, neue Erkenntnisse vom bestehenden Wissen abzuleiten [Ghe92]. Im Laufe der Zeit hat sich das Konzept eines Expertensystems auf andere Anwendungsbereiche verbreitet. Aus diesem Grund gibt es mehrere Definitionen, die im Allgemeinen ähnlich sind und im Spezifischen Merkmale des zugehörigen Anwendungsbereichs beinhalten.

Allgemein lässt sich sagen, dass ein Expertensystem ein Computersystem (Hardware und Software) ist, das in einem bestimmten Bereich Wissen und Schlussfolgerungsfähigkeit eines menschlichen Experten nachbildet [CG14, S.12]. Aus Sicht der Wirtschaftsinformatik zielen Expertensysteme drauf ab, das Expertenwissen menschlicher Fachleute in der Wissensbasis eines Computers abzuspeichern und für eine Vielzahl von Problemlösungen zu nutzen [PFW⁺12, S.59]. Im Weiteren gehen Beierle und Kern-Isberner auf die Eigenschaften ein, die ein Expertensystem aufweisen soll [CG14, S.12]. Im Rahmen dieser Arbeit sind folgende Eigenschaften besonders relevant:

- Anwendung des Wissens eines oder mehrerer Experten, um Probleme in einem bestimmten Anwendungsbereich zu lösen.
- Leicht lesbare Wissensdarstellung.
- Möglichst anschauliche und intuitive Benutzerschnittstelle.
- Leichte Wartbarkeit und Erweiterbarkeit des Wissens im Expertensystem.
- Unterstützung beim Wissenstransfer vom Experten zum System.

Es ist auch wichtig anzumerken, dass die Begriffe “Künstliche Intelligenz“, “Wissensbasiertes System“ und “Expertensystem“ in einer engen Beziehung zueinander stehen. Haun gibt eine systematische Abgrenzung dieser Begriffe, die sich folgendermaßen beschreiben lässt [Mat00, S.30]:

- *Künstliche Intelligenz* stellt den Oberbegriff dar und bildet den theoretischen Rahmen für die Entwicklung von Wissensbasierten Systemen und Expertensystemen.
- *Wissensbasierte Systeme* sind eine Teilmenge der Anwendungen der künstlichen Intelligenz. Sie wenden die Wissensverarbeitung auf ein konkretes Aufgabengebiet an und verwalten Allgemeinwissen explizit und getrennt vom Rest des Systems.
- *Expertensysteme*, die ein Teilbereich der Wissensbasierten Systeme sind, stellen eine Spezialisierung von Wissensbasierten Systemen dar. Sie verwalten spezifisches Expertenwissen, das von einem Experten stammt und auf praxisbezogene Probleme angewandt wird.

Graphisch lässt sich die vorliegende Abgrenzung in der Abbildung 1 darstellen:



Abbildung 1: Begriffsabgrenzung, [Mat00, S.30]

Nach dieser Abgrenzung lässt sich feststellen, dass der Unterschied zwischen einem Wissensbasierten System und einem Expertensystem darin besteht, dass das Wissen im Endeffekt von einem Experten stammt. Allerdings ist dieses Kriterium zu einfach und nicht besonders aussagekräftig. Beierle und Kern-Isberner weisen drauf hin, dass nach diesem Kriterium viele existierenden Wissensbasierten Systeme als Expertensysteme bezeichnet werden könnten [CG14][S.11]. Aufgrund dessen stellen die Autoren die Eigenschaften eines Experten dar, die sich folgendermaßen zusammenfassen lassen:

- Experten sind selten und teuer.
- Experten sind nicht immer verfügbar.
- Leistungsfähigkeit der Experten ist nicht konstant, sondern kann nach Tagesverlauf schwanken.
- Expertenwissen kann oft nicht als solches weitergegeben werden.
- Expertenwissen kann verloren gehen.

Ein gutes Beispiel hinsichtlich der Gefahr, dass Expertenwissen verloren gehen kann, wird in [SK09][S.94] vorgestellt. Gebus nimmt den Bezug auf die sogenannte Baby-Boomgeneration, die bald in die Rente geht. Infolgedessen verlassen gleichzeitig viele Experten Unternehmen. Damit geht ebenso das Expertenwissen verloren.

Zusammenfassend lässt sich sagen, dass die Entwicklung eines Expertensystems eine hohes Potenzial besitzt. Allerdings kann ein Expertensystem nicht als Ersatz eines menschlichen Experten betrachtet werden. Vielmehr geht es um eine Erfassung, Darstellung und Pflege des Expertenwissens in einem Expertensystems, um die Arbeitsprozesse effizienter zu gestalten und sowohl erfahrene als auch neue Anwender in einem bestimmten Wissensbereich bei der Aufgabenabwicklung zu unterstützen.

2.2 Architektur eines Expertensystems

Beierle und Kern-Isberner betonen, dass die Trennung zwischen der Darstellung des Wissens (Wissensbasis) und der Wissensverarbeitung (Wissensverarbeitungskomponente) der wichtigste Aspekt eines Wissensbasierten Systems ist. [CG14, S.11]. Die Wissensbasis kann man sich als eine Art Datenstruktur vorstellen, in der das benötigte Wissen gespeichert wird. Die Wissensverarbeitungskomponente umfasst eine Menge von anwendungsunabhängigen Algorithmen, die mithilfe der Wissensbasis eine Lösung für ein gegebenes Problem erarbeiten. Somit stehen die Wissensbasis und die Wissensverarbeitungskomponente in einer engen Beziehung zueinander [Kur92, S.18].

Allgemein umfasst ein Expertensystem folgende Bestandteile [Pet10, S.75]:

- *Wissensbasis*, die Expertenwissen in Form von Fakten in einer bestimmten Sprache speichert sowie Regeln zur Wissensorganisation beinhaltet.
- *Inferenzmaschine*, die unter Berücksichtigung des zugrunde liegenden Wissensbedarfs die Wissensbasis durchsucht bis das System einen Problemlösungsvorschlag erarbeitet hat oder herausfindet, dass keiner existiert.
- *Dialogkomponente*, die eine Schnittstelle zwischen dem Nutzer und dem System darstellt.
- *Erklärungskomponente*, die dem Benutzer erläutert, warum und auf welche Weise eine bestimmte Lösung gefunden bzw. nicht gefunden wurde [Mat00, S.126].
- *Wissensakquisitionskomponente*, die den Entwickler des Expertensystems bei der Erweiterung, Änderung und Wartung der Wissensbasis unterstützt.

Laut Tecuci stellen Wissensbasis und Inferenzmaschine grundlegende Bestandteile eines Expertensystems dar und bilden damit den Kern des Expertensystems [Ghe92, S.1444]. Dialogkomponente, Erklärungskomponente und Wissensakquisitionskomponente gehören zur sogenannten Schale und sind für die Kommunikation zwischen dem Systemverwalter und dem Nutzer zuständig (siehe Abbildung 2).

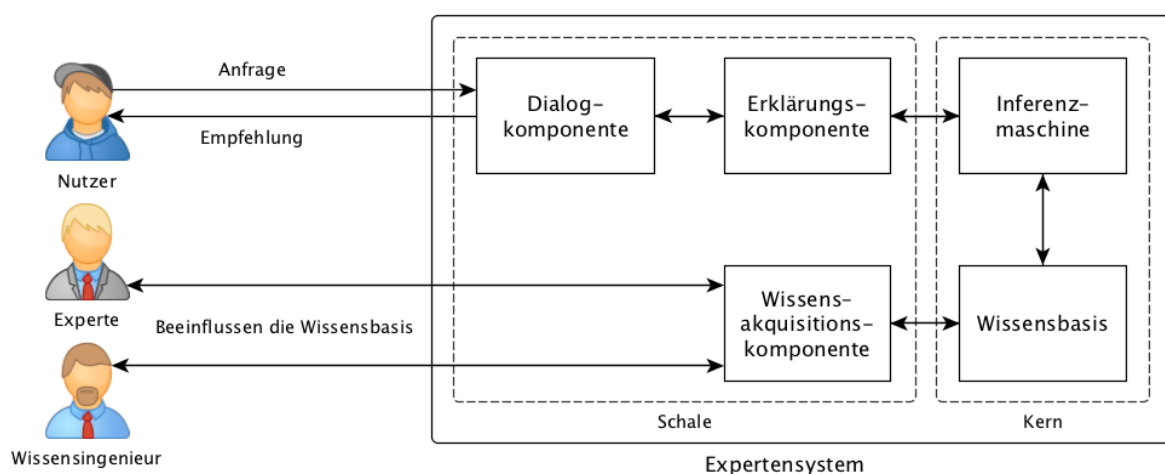


Abbildung 2: Expertensystems nach Haun, [Mat00, S.126]

Im Hinblick auf die Interaktion gibt es drei Gruppen, die mit dem Expertensystem interagieren:

- *Nutzer*, der das Expertensystem zum Lösen eines Problems benutzt und mit der Dialogkomponente kommuniziert. Der Wissensingenieur und der Experte können ebenso als Nutzer auftreten [JFI93, S.758].
- *Wissensingenieur*, der sich mit dem Aufbau und Wartung der Wissensbasis beschäftigt. Unter anderem ist Wissensmodellierung ein wichtiger Aufgabenbereich eines Wissensingenieurs [JFI93, S.742].
- *Experte*, der über spezifisches Erfahrungswissen verfügt, das für das Expertensystem relevant ist.

Der Ablauf der Kommunikation zwischen dem Nutzer und dem Expertensystem sieht folgendermaßen aus:

- Der Nutzer schickt eine Anfrage an die Dialogkomponente des Expertensystems.
- Die Dialogkomponente übermittelt die Anfrage an die Inferenzmaschine.
- Die Inferenzmaschine erarbeitet eine Lösung für das gegebene Problem mittels der Wissensbasis und gibt das Ergebnis an die Dialogkomponente zurück.
- Anschließend teilt die Dialogkomponente dem Nutzer die Lösung des Problems mit. Falls keine Lösung zum Problem existiert, wird eine entsprechende Fehlermeldung angezeigt.

Auf der anderen Seite können die Inhalte der Wissensbasis von einem Wissensingenieur mithilfe der Wissensakquisitionskomponente beeinflusst werden. Der Wissenserwerb durch den Wissensingenieur ist die verbreitetste Vorgehensweise, neue Daten für ein wissensbasiertes System zu erschließen. Meistens handelt es sich um ein Interview zwischen dem Wissensingenieur und dem Experten [Pet10, S.76], [HDNR97, S.210]. Neben dem Interview kann der Wissensingenieur eine Recherche der verfügbaren Wissensquellen wie Text, technische Zeichnungen oder Web-Ressourcen durchführen. Anschließend werden die Daten vom Wissensingenieur formalisiert und in die Wissensbasis gespeichert.

Die Wissensbasis kann in einigen Fällen von einem fachlichen Experten beeinflusst werden. Dafür ist eine geeignete Expertenschnittstelle innerhalb der Wissensakquisitionskomponente notwendig, die den Experten ermöglicht, ihr Erfahrungswissen selbst zu formalisieren und gegebenenfalls zu warten [JFI93, S.743]. Die Überprüfung des Dateninputs ist ebenfalls die Aufgabe der Wissensakquisitionskomponente. Dies kann mittels Durchführung automatisierten Tests bei jeder Änderungsanfrage erfolgen, um die Konsistenz der Wissensbasis zu gewährleisten [JFI93, S.743].

Um ein geeignetes Konzept der automatisierten Datenerfassung zu entwickeln, ist ein grundlegendes Verständnis von der Struktur und Funktionsweise der Wissensbasis sowie der Wissensakquisitionskomponente erforderlich. Im weiteren Verlauf der Arbeit werden die Erkenntnisse über die Wissensbasis und die Wissensakquisitionskomponente erläutert, die in der Forschung von Expertensystemen entstanden sind.

2.3 Wissensbasis

Neben der Inferenzmaschine stellt die Wissensbasis den zentralen Teil eines Expertensystems, der die Daten des gesamten Systems beinhaltet [JFI93, S.754]. Im Folgenden werden der allgemeine Prozess der Wissensbasisentwicklung, der Inhalt der Wissensbasis und die Möglichkeiten der Wissensrepräsentation thematisiert. Gheorghe Tecuci beschreibt folgende Phasen bei der Entwicklung der Wissensbasis [Ghe92, S.1444]:

1. Systematische Erfassung vom Expertenwissen
2. Verfeinerung der Wissensbasis
3. Reorganisation der Wissensbasis

In der ersten Phase werden das Vokabular und die geeignete Wissensrepräsentation festgelegt. Gebus und Leiviskä betonen, dass die Wissensrepräsentation den entscheidenden Einfluss auf die Generierung und spätere Handhabung der Wissensbasis hat [SK09, S.95]. Die initialen Daten werden meistens im Rahmen eines Interviews zwischen dem Wissen-singenieur und dem Experten erfasst [Ghe92, S.1444]. Das Ergebnis der ersten Phase ist eine initiale Wissensbasis, die unvollständig und teilweise widersprüchlich ist. In der zweiten Phase wird die initiale Wissensbasis mithilfe der geeigneten Datenerfassungsmethoden solange erweitert und verbessert, bis sie vollständig und korrekt genug ist, um ein gegebenes Problem richtig zu lösen. In der dritten Phase wird die vollständige und korrekte Wissensbasis reorganisiert, um die Effizienz der Lösungsberechnung zu steigern [Ghe92, S.1445]. Zusammenfassend werden die Phasen in der Abbildung 3 dargestellt.

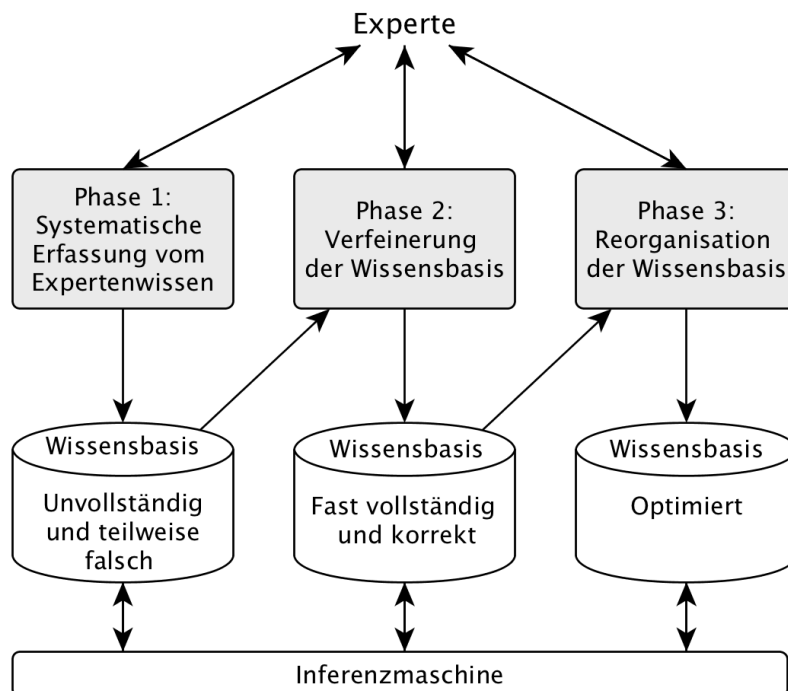


Abbildung 3: Phasen der Expertensystementwicklung, [GD94, S.138]

In der Abbildung 3 sieht man, dass der Autor dem Experten die gesamte Kontrolle über die Entwicklung der Wissensbasis zuweist. Allerdings ist diese Sichtweise nicht vollständig, da im Entwicklungsprozess der Wissensingenieur und der Systementwickler beteiligt sind und dementsprechend berücksichtigt werden sollen.

In Bezug auf den Inhalt der Wissensbasis unterscheiden Beierle und Kern-Isberner folgende Wissensarten [CG14, S.5]:

- *Fachspezifisches Wissen*. Dabei handelt es sich um das spezifischste Wissen, das sich nur auf den gerade betrachteten Problemfall bezieht. Das sind z.B. Fakten, die von Beobachtungen oder Untersuchungsergebnissen stammen.
- *Regelhaftes Wissen*, das den eigentlichen Kern der Wissensbasis darstellt. Dieses Wissen kann noch genauer differenziert werden:
 - *Bereichsbezogenes Wissen*, das sich auf den gesamten Problembereich beziehen. Das kann sowohl theoretisches Fachwissen als auch Erfahrungswissen sein. Anders gesagt handelt es sich um generisches Wissen.
 - *Allgemeinwissen*, das z.B. um generelle Problemlösungsheuristiken, Optimierungsregeln oder auch allgemeines Wissen über Objekte und Beziehungen in der realen Welt beinhaltet.

Unter Berücksichtigung der Differenzierung der Wissensarten innerhalb der Wissensbasis beschreiben die Autoren in [CG14, S.18] auf eigene Weise die Architektur des Expertensystems, die in der Abbildung 4 dargestellt wird.

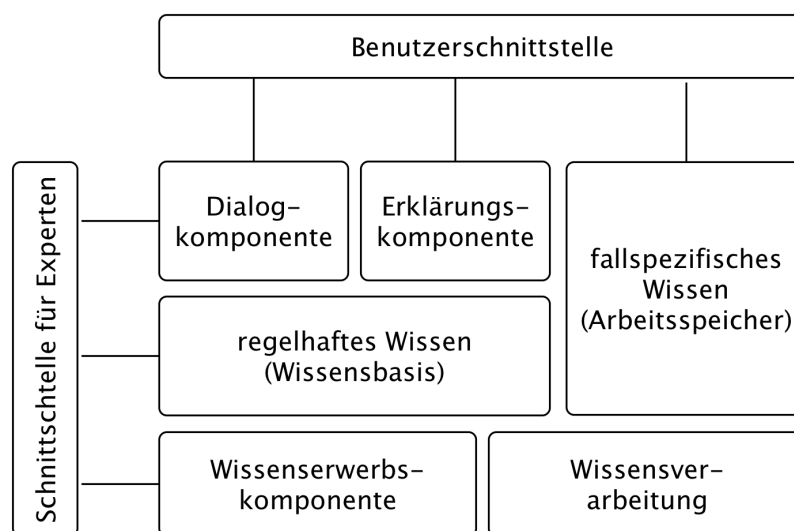


Abbildung 4: Expertensystems nach Beierle und Kern-Isberner, [CG14, S.18]

Laut Beierle und Kern-Isberner können verschiedene Wissensarten in einem wissensbasierten System je nach dem Anwendungsbereich unterschiedlich umfangreich auftreten. Ein hochspezialisiertes System kann beispielsweise über sehr wenig oder gar kein Allgemeinwissen verfügen. Auf der anderen Seite kann ein wissensbasiertes System den Schwerpunkt auf das gewöhnliche Alltagswissen legen [CG14, S.5-6].

Ein weiterer Aspekt beim Aufbau der Wissensbasis ist die Wissensrepräsentation. Die grundlegende Aufgabe der Wissensrepräsentation ist die Formularisierung von Wissen, um eine maschinelle Verarbeitung erst zu ermöglichen [Mat00, S.22]. Sinz und Ferstl unterscheiden folgende Formen der Wissensrepräsentation [OE13, S.366]:

- *Regelorientierte Darstellung*, in der das Wissen als (WENN, DANN)-Beziehungen beschrieben wird. Diese Darstellungsform wird beispielsweise bei Prolog-Regeln eingesetzt.
- *Objektorientierte Darstellung*, die das Konzept der Objekttypen übernimmt und mit deklarativen Operatorbeschreibungen verbindet.
- *Constraints Darstellung*, die Modellbeschreibungen aus dem Operations Research benutzt. Dabei handelt es sich um Lösungsräume durch Nebenbedingungen und Zielvorgaben.

Hinsichtlich der Wissensrepräsentation stellen Ferstl und Sinz imperative und deklarative Paradigmen gegenüber [OE13, S.366]. Ein Programm, das dem imperativen Paradigma folgt, besteht aus einer Folge von Befehlen, die nacheinander ausgeführt werden [OE13, S.341]. Bei einem deklarativen Programm handelt es sich um eine Beschreibung der Aufgabenaußensicht. Ein deklaratives Programm hat keine festgelegten Lösungsverfahren je Aufgaben. Stattdessen wird eine Lösung zum Zeitpunkt der Aufgabendurchführung mittels Inferenzmaschine abgeleitet [OE13, S.361].

Allgemein beziehen sich die Autoren darauf, dass an ein wissensbasiertes System nur geringe Anforderungen bezüglich Vollständigkeit, Widerspruchsfreiheit und Eindeutigkeit gestellt werden können. Aus diesem Grund ist das deklarative Paradigma für die Wissensrepräsentation besser geeignet. Folgende Gründe nennen die Autoren für die deklarative Umsetzung der Wissensbasis [OE13, S.366]:

- *Wissensdarstellung*: Da ein Mensch das Erfahrungswissen durch assoziative Beziehungsmuster aufbaut, ist die deklarative Wissensdarstellung eher geeignet.
- *Wissensauswertung*: Änderungen von Erfahrungswissen werden normalerweise in deklarativen Form erfasst.
- *Wissensverfügbarkeit*: Die Codewartung von einem imperativen Programm ist fehleranfällig, kosten- und zeitintensiv, da das Erfahrungswissen häufig geändert und aktualisiert werden muss.

Der objektorientierte Ansatz ist eine weitere Möglichkeit, das Wissen zu beschreiben. Ein Beispiel für die objektorientierte Implementierung wird in [KM90] vorgestellt. Die Wissensbasis wird dabei als eine Sammlung von Klassen, Objekten und Methoden definiert [KM90, S.40]. Der große Vorteil solcher Umsetzung besteht in der Modularität des Wissens. Das Wissen in unabhängige Module aufgeteilt wird. Da die einzelnen Module unabhängig voneinander sind, können sie getrennt getestet und modifiziert werden, ohne den Rest der Wissensbasis zu beeinträchtigen. Dies ermöglicht hohe Flexibilität bei der Wissensbasiserweiterung [KM90, S.43].

Neben der Implementierung der Wissensbasis ist eine geeignete Umsetzung der Wissensakquisitionskomponente erforderlich, um die Wissensbasis aktuell, möglichst fehlerfrei und konsistent zu halten. Im Folgenden wird die Wissensakquisitionskomponente in Hinsicht auf den allgemeinen Aufbau und Funktionen thematisiert.

2.4 Wissensakquisitionskomponente

Bei der Wissensakquisitionskomponente handelt es sich um einen Bestandteil eines Expertensystems, der den Wissensingenieur oder einen Experten beim Aufbau und späterer Erweiterung der Wissensbasis unterstützt [GTW90, S.18]. Allgemein umfasst die Wissensakquisitionskomponente zwei grundsätzliche Aufgaben, nämlich den Wissenserwerb und die Prüfung des Dateninputs auf Konsistenz, Vollständigkeit und Einschränkungen des Expertensystems [JFI93, S.759].

Unter dem Wissenserwerb wird eine Übertragung sowie Eingliederung von Wissen über Problemlösungsverfahren in ein Computerprogramm verstanden [GTW90, S.178]. Es werden folgende Grundarten des Wissenserwerbs unterschieden [JFI93, S.742]:

- *Indirekter Wissenserwerb:* Ein Wissensingenieur führt ein Interview mit einem Experten, oder allgemein mit einem Wissensträger. Die Analyse der Dokumente, die für das System relevant sind, gehört ebenso zur Aufgabe des Wissensingenieurs.
- *Direkter Wissenserwerb:* Ein Wissensträger gibt sein Wissen selbst mittels einer Schnittstelle ins Expertensystem ein.
- *Automatischer Wissenserwerb:* Die Wissensbasis wird entweder mithilfe der automatisierten Datenererschließung aus verfügbarer Dokumente oder Methoden des maschinellen Lernens erweitert.

Die Methoden des indirekten Wissenserwerbs lassen sich grundsätzlich in unstrukturierte und strukturierte Verfahren unterteilen. Das unstrukturierte Interview ist die am häufigsten verwendete Methode [GTW90, S.76]. Dabei stellt der Wissensingenieur dem Experten problembezogene Zwischenfragen, um ein möglichst vollständiges Bild des zur Problemlösung erforderlichen Wissens zu bekommen. Die Hauptschwierigkeit bei der Wissenserhebung durch ein Interview ist die Formulierung der Fragen. Wenn die Fragen zu spezifisch sind, können wichtige Informationen ausgelassen werden [SK09, S.95]. Eine strukturierte Vorgehensweise der Wissenserhebung ist die Protokollanalyse, wobei der Experte beim Lösen eines Problems aufgezeichnet wird. Um den Lösungsweg nachvollziehbar zu machen, kann der Experte die Aufgabe gezielt langsamer durchführen oder die Aufzeichnung mit Kommentaren versehen. Anschließend werden die erzielten Ergebnisse vom Wissensingenieur formalisiert und ins Expertensystem eingetragen.

Beim direkten Wissenserwerb soll eine geeignete Schnittstelle im Rahmen der Wissensakquisitionskomponente zur Verfügung gestellt werden, die es dem Wissensträger ermöglicht, sein Wissen ins System einzugeben. Die Schnittstelle soll eine dem Wissensträger bekannte Wissensrepräsentation verwenden und benutzerfreundlich bei der Dateneingabe sein [JFI93, S.743]. Ein Beispiel der Benutzerfreundlichkeit ist gleichzeitige Validierung der Benutzereingaben sowie eine Rückmeldung bei unzulässigen Aktionen. Der direkte Wissenserwerb hat den Vorteil, dass die Wissensbasis ohne den Wissensingenieur erweitert werden kann. Allerdings betonen die Autoren in [JFI93, S.765], dass das nur in gut verstandenen und strukturierten Anwendungsbereichen möglich sei.

Zu dem automatisierten Wissenserwerb gibt es am wenigsten Erkenntnisse, die allgemein anwendbar sind. Meistens handelt es sich um Lösungen, die nur innerhalb eines spezifischen Anwendungsbereichs funktionieren. Nichtsdestotrotz lässt sich sagen, dass das Wissen sich entweder aus vorhandenen Daten (maschinelles Lernen) oder aus verfügbaren Dokumenten (automatische Dokumentenanalyse) generieren lässt [GTW90, S.78]. Die Implementierung hängt jedoch vom Anwendungsgebiet ab. Ein Beispiel für den Fall großer

Datenmenge und Anwendung des maschinellen Lernens ist ein Diagnosesystem, das für mögliche Diagnose Fälle mit Symptomen enthält. Für die Textanalyse können beispielsweise Bedienungsanleitungen analysiert werden, wobei diese Vorgehensweise gewisse Einschränkungen ausweist. Die Schwierigkeit dabei besteht darin, dass das Erfahrungswissen nicht in den Textdokumenten zu finden ist [GTW90, S.79].

In allen Fällen des Wissenserwerbs werden die Daten an die zentrale Schnittstelle weitergereicht. Diese Schnittstelle beschäftigt sich mit der Zwischenspeicherung und der Prüfung der Datensätze auf Korrektheit, bevor die Daten endgültig in der Wissensbasis gespeichert werden. Zum Teil kann der Wertebereich direkt im einzelnen Modul des Wissenserwerbs eingeschränkt werden. Beispielsweise kann ein Eingabefeld in der Wissensträgerschnittstelle nur positive Zahlen zulassen. Für den restlichen Teil werden automatische Tests durchgeführt, die sicherstellen, dass neue Daten keine Inkonsistenzen in Bezug auf die Einschränkungen der Wissensbasis erzeugen [JFI93, S.765].

Zusammenfassend lässt sich die Wissensakquisitionskomponente schematisch in der Abbildung 5 wie folgt darstellen:

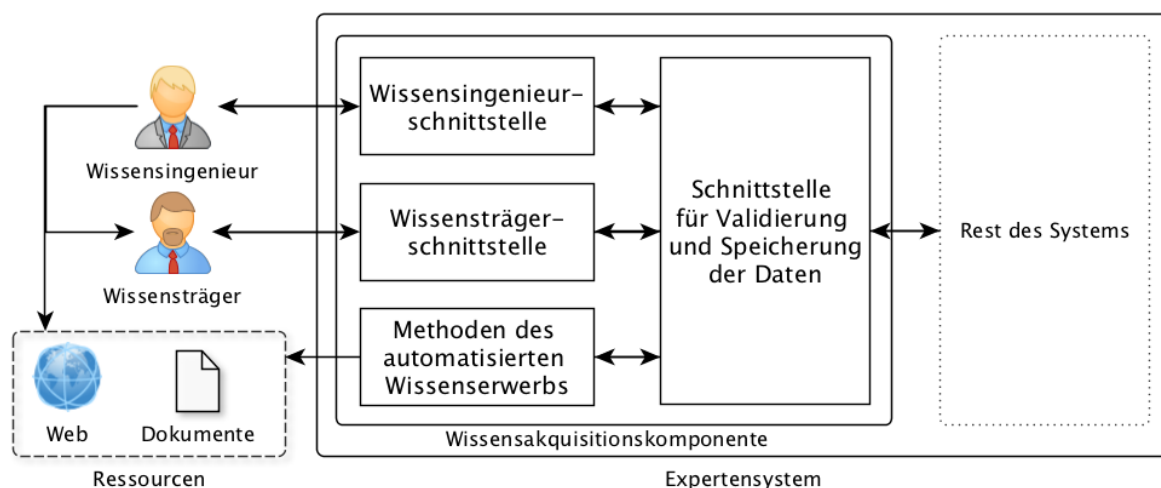


Abbildung 5: Wissensakquisitionskomponente

In der Abbildung 5 sieht man deutlich, dass die Wissensakquisitionskomponente modular aufgebaut ist. Allgemein kann dieses Modell in jedem Expertensystem eingesetzt werden, wobei die konkrete Umsetzung in Bezug auf den Anwendungsbereich spezifiziert wird. Dabei lässt sich der Automatisierungsgrad der Datenerfassung leicht anpassen, indem der Schwerpunkt auf gewünschte Bestandteile der Wissensakquisitionskomponente gelegt wird. Beispielsweise kann sich ein Expertensystem mit umfangreichen Falldaten auf das maschinelle Lernen konzentrieren. Ein System, dass auf der Datenerfassung mithilfe der Wissensträgerschnittstelle basiert, wird in [SK09, S.97] vorgestellt.

Im weiteren Verlauf der Arbeit werden die Wissensträgerschnittstelle, die Komponente mit Methoden des automatisierten Wissenserwerbs und die Schnittstelle für Validierung und Speicherung der Daten thematisiert, da sie ein Potenzial für die Automatisierung der Datenerfassung aufweisen.

3 Automatisierung der Wissenserfassung

Mit der Wissensakquisitionskomponente wurde bereits angedeutet, dass die Datenerfassung für Expertensysteme meistens nur teilweise automatisierbar ist. Die Autoren in [GD94] weisen ebenso darauf hin, dass manuelle und maschinelle Wissenserschließung jeweils eigene Stärke haben, die sich gegenseitig ergänzen [GD94, S.137]. Aus diesem Grund ist bei der Datenerfassung ein hybrides Modell sinnvoll, das die Vorteile manueller und maschineller Verfahren kombiniert. Aufgrund der Fragestellung dieser Arbeit wird es im Weiteren auf automatisierte und halb-automatisierte Bestandteile der Wissensakquisitionskomponente beschränkt.

3.1 Die Wissensträgerschnittstelle

Bei der Wissensträgerschnittstelle handelt es sich um eine Methode zur Datenerfassung, die einen direkten Wissenstransfer zwischen einem Wissensträger und dem Expertensystem ermöglicht. Da die Daten manuell eingegeben und maschinell verarbeitet werden, bezeichnet man die Wissensträgerschnittstelle als semi-automatisierte Wissenserfassungsmethode. Im Folgenden wird ein Anwendungsbeispiel aus der Praxis betrachtet und in Bezug auf die Erkenntnisse bei der Entwicklung und dem Einsatz erläutert.

Das Praxisbeispiel wird in der [SK09] vorgestellt und bezieht sich auf die Wissenserfassung aus der Produktion eines Elektrotechnikunternehmens. In der Fallstudie wird ein bereits bestehendes Decision Support System (DSS) betrachtet, das die Führungskräfte bei den Entscheidungen von nicht-strukturierten Problemen in der Produktion unterstützt [SK09, S.94]. Das DSS verfügt bereits über eine Datenbank, die allerdings nur Daten von Produkteigenschaften enthält. Die eigentlichen Prozesse werden allerdings von Experten gesteuert, die mithilfe des Erfahrungswissens Störungen in der Produktion beseitigen. Dieses Erfahrungswissen über Störungen wird im System nicht erfasst. Als Folge hat die Unternehmensführung einen begrenzten Überblick über die Situation in der Produktionsabteilung. Außerdem besteht die Gefahr, dass das spezifische Expertenwissen verloren geht, falls der Wissensträger das Unternehmen verlässt.

Die Zielsetzung von [SK09] ist die Erfassung des Expertenwissens aus der Produktion und die Integration dieses Wissens in die Entscheidungsprozesse auf der Organisationsebene. Um das Wissen aus der Produktionsabteilung zu erschließen, erweitern Gebus und Leiviskä das bestehende System um eine Schnittstelle für Anlagenbediener. Die Umsetzung erfolgt in Form von einem Prototyp. Der Entwicklungsprozess lässt sich allgemein wie folgt beschreiben:

1. Die Definition der Zielgruppe, die für die Schnittstelle relevant ist.
2. Ausgehen von der Zielgruppe werden die Informationsrepräsentation sowie funktionale und nicht-funktionale Anforderungen spezifiziert.
3. Die Umsetzung und Evaluation des Prototyps.

Im ersten Schritt definieren Gebus und Leiviskä die bestehenden Nutzergruppen des gesamten Systems [SK09, S.97]:

- Anlagenbediener (Experte), der sein Wissen zu den Störungen mittels der Wissens-trägerschnittstelle in die Datenbank eingibt.
- Administrator, der das gesamte System verwaltet.
- Qualitätsabteilung, die die Störungsstatistik analysiert, eine Qualitätsrückmeldung an die Produktion und einen Bericht an die Führungskraft übermittelt.
- Führungskraft, die eine umfangreiche Übersicht von der Qualitätsabteilung erhält und davon ausgehend Entscheidungen zur Prozessoptimierung trifft.

Im Rahmen der Fallstudie ist die Nutzergruppe der Anlagenbediener relevant, wobei das System jeder Nutzergruppe eine geeignete Benutzerschnittstelle zur Verfügung stellt. Schematisch lässt sich die Struktur und die Informationsflüsse im DSS in der Abbildung 6 vereinfacht nachbilden.

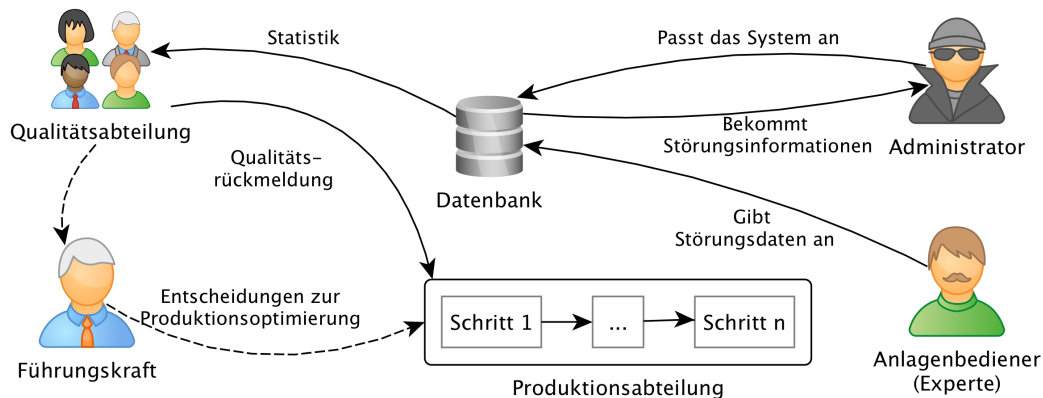


Abbildung 6: Die Struktur und die Informationsflüsse im DSS, [SK09, S.98]

Bei der Informationsrepräsentation werden die Anlagenbilder benutzt. Die Autoren betonen, dass die Expertenschnittstelle einfach und intuitiv wie möglich gehalten wurde, um die Dateneingabe zu einer alltäglichen Tätigkeit zu machen [SK09, S.97]. Im Hinblick auf die nicht-funktionalen Anforderungen werden beispielsweise Einstellungen und Maschinenbilder vom Administrator vorkonfiguriert und automatisch beim Hochfahren des Systems geladen. Die funktionalen Anforderungen umfassen die Erfassung einer Störung durch die Auswahl des passenden Anlagebildes und das Markieren der betroffenen Stelle. Darauf folgend wird eine Vorlage mit möglichen Ursachen zur Auswahl angezeigt. Zusätzlich gibt es ein Feld zur Freitexteingabe, wenn die vorliegende Störung im System noch nicht vorhanden ist [SK09, S.99].

In der Evaluation wurden drei benutzerorientierte Verbesserungspotentiale [SK09, S.100]:

- *Usability* (engl. *Gebrauchstauglichkeit*): wie einfach und intuitiv ist das System zu verwenden.
- *Usefulness* (engl. *Nützlichkeit*): wie nützlich ist das System für primäre (Anlagenbediener) und sekundäre (z.B. Qualitätsabteilung) Nutzer.
- *Usage* (engl. *Nutzung*): inwiefern wird das System verwendet.

Wenn es um die Butzerinteraktion mit einem System geht, spielt Usability eine zentrale Rolle. Gemäß ISO 9241-11 wird Usability durch folgende Aspekte definiert: Effektivität (wurde das Ziel erreicht), Effizienz (wie schnell wurde das Ziel erreicht) und Zufriedenheit (positive Erfahrung bei der Systemnutzung) [ISO98]. In Bezug auf Usefulness ist eine interessante Ansicht in [Ste14] gegeben. Steve Krug bezeichnet einen Gegenstand nützlich, wenn eine Person mit durchschnittlichen Fähigkeiten herausfinden kann, wie dieser Gegenstand zu bedienen ist, um ein Ziel zu erreichen. Dabei soll der Aufwand kleiner als der Wert des Ziels sein [Ste14, S.9].

Statistisch gesehen gab es in der betrachteten Testperiode 183 Anlagenausfälle. Allerdings wurden nur 70 Fälle kommentiert, was die Nutzungsrate von 38% ergibt. Laut Gebus und Leiviskä lag es daran, dass der Prototyp aus Usability-Perspektive nicht optimal war. Beispielsweise waren einige Elemente der Schnittstelle eher verwirrend. Nach den notwendigen Anpassungen konnte die Nutzungsrate fast bei 100% erreicht werden. Hinsichtlich der Nützlichkeit gab es positive Bewertungen. Es wurde festgestellt, dass die Umwandlung des Systems aus einer reinen Datensammlung über Störungen zu einem Informationsaustauschsystem Vorteile für alle Nutzergruppen bringt. In der Abbildung 6 sieht man deutlich, dass die Informationen, die in der Datenbank gespeichert werden, von allen Beteiligten genutzt werden. Der Administrator kann mithilfe der Daten das System entsprechend anpassen. Die Qualitätsabteilung setzt die Informationen zur Berichterstellung für die Führungskräfte und die Produktion. Die Unternehmensführung erhält ein umfangreicheres Bild über die Lade in der Produktion und kann aufgrund dessen bessere Entscheidungen bei der Produktionsoptimierung treffen.

Zusammenfassend lässt sich sagen, dass die Entwicklung der Wissensträgerschnittstelle aus drei Phasen besteht. In der ersten Phase wird die Zielgruppe definiert. Im Hinblick auf die Zielgruppe werden die geeignete Informationsrepräsentation sowie die funktionalen und nicht-funktionalen Anforderungen festgelegt. Die dritte Phase umfasst die Implementierung mit der darauffolgenden Evaluation, in dem die Schnittstelle nach drei Kriterien bewertet wird, nämlich Usability, Usefulness und Usage.

3.2 Maschineller Wissenserwerb

Eine Auswahl an generellen Methoden für den maschinellen Wissenserwerb

3.3 Schnittstelle für Validierung und Speicherung

Zweck, Konzept

4 Implementierung

4.1 Ansatz 1

4.2 Ansatz 2

4.3 Ansatz 3

5 Evaluation und Fazit

5.1 Test 1

5.2 Test 2

5.3 Fazit

6 Ausblick

Literatur

- [CG14] CHRISTOPH BEIERLE und GABRIELE KERN-ISBERNER: *Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen*. Vieweg, 2014.
- [GD94] GHEORGHE D. TECUCI und DAVID DUFF: *A framework for knowledge base refinement through multistrategy learning and knowledge acquisition*. Knowledge Acquisition, Seiten 137 – 162, 1994.
- [Geo08] GEORGE LAWTON: *Developing Software Online with Platform-as-a-Service Technology*. Computer, 2008.
- [Ghe92] GHEORGHE D. TECUCI: *Automating Knowledge Acquisition as Extending, Updating, and Improving a Knowledge Base*. Transactions on Systems, Man, and Cybernetics, Seiten 1444–1460, 1992.
- [GTW90] GEORG GOTTLÖB, THOMAS FRÜHWIRTH und WERNER HORN: *Expertensysteme*. Springer-Verlag Wien New York, 1990.
- [HDNR97] H. FUJIHARA, D. B. SIMMONS, N. C. ELLIS und R. E. SHANNONS: *Knowledge Conceptualization Tool*. Transactions on Knowledge and Data Engineering, 1997.
- [ISO98] ISO: *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability*. Technischer Bericht, International Organization for Standardization, 1998.
- [JFI93] JOSEF MEYER-FUJARA, FRANK PUPPE und IPKE WACHSMUTH: *Expertensysteme und Wissensmodellierung*, Kapitel 7, Seiten 714–766. Addison-Wesley, 1993.
- [KM90] K. S. LEUNG und M. H. WONG: *An Expert-System Shell Using Structured Knowledge: An Object-Oriented Approach*. Computer, 23(3):38–47, 1990.
- [Kur92] KURBEL, KARL: *Entwicklung und Einsatz von Expertensystemen: Eine anwendungsorientierte Einführung in wissensbasierte Systeme*. Springer-Verlag, 1992.
- [Mat00] MATTHIAS HAUN: *Wissensbasierte Systeme*. Expert Verlag, 2000.
- [Nik16] NIKET TANDON: *Commonsense Knowledge Acquisition and Applications*. Doktorarbeit, Saarland University, Saarbrücken, Germany, 2016.
- [OE13] OTTO K FERSTL und ELMAR J. SINZ: *Grundlagen der Wirtschaftsinformatik*. Oldenbourg Verlag, 2013.
- [Pet10] PETER ZÖLLER-GREER: *Künstliche Intelligenz: Grundlagen und Anwendungen*. Composita Verlag, 2010.
- [PFW⁺12] PETER MERTENS, FREIMUT BODENDORF, WOLFGANG KÖNIG, ARNOLD PICOT, MATTHIAS SCHUMANN und THOMAS HESS: *Grundzüge der Wirtschaftsinformatik*. Springer-Verlag, 2012.

- [PT11] PETER MELL und TIMOTHY GRANCE: *The NIST Definition of Cloud Computing*. NIST Special Publication 800 - 145, 2011.
- [RP10] RAJENDRA AKERKAR und PRITI SAJJA: *Knowledge-Based Systems*. Jones and Bartlett Publishers, 2010.
- [SK09] SÉBASTIEN GEBUS und KAUKO LEIVISKÄ: *Knowledge Acquisition for Decision Support Systems on an Electronic Assembly Line*. Expert Systems with Applications, Seiten 93–101, 2009.
- [Ste14] STEVE KRUG: *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. New Riders, 2014.

Ich erkläre hiermit gemäß §17 Abs. 2 APO, dass ich die vorstehende Bachelorarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Bamberg, den 31.03.2017

Petr Vasilyev