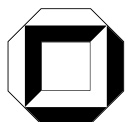


# Unternehmensspezifisches Klassifikationssystem zur effizienten Datenverwaltung (mit Anwendungs- szenarien aus der Praxis)



Abschlussbericht des Verbundprojektes  
„Klassifikationssysteme automatisch  
erstellen“ (KLASTER)

Gefördert vom Bundesministerium  
für Bildung und Forschung (BMBF)





**Unternehmensspezifisches Klassifikationssystem zur  
effizienten Datenverwaltung (mit Anwendungsszenarien  
aus der Praxis)**



# **Unternehmensspezifisches Klassifikationssystem zur effizienten Datenverwaltung (mit Anwendungsszenarien aus der Praxis )**

Abschlußbericht des Verbundprojektes  
„Klassifikationssysteme automatisiert erstellen“  
(KLASTER)

gefördert vom  
Bundesministerium für Bildung und Forschung  
(BMBF)

Autoren:

Dipl.-Ing. André Guldi, Eigner

Dipl.-Ing. Alexander Hoffmann, Arc Solutions

Dipl.-Inform. Alexander Mahl, RPK

Dr.-Ing. Stefan Sander, Atos Origin

Dr.-Ing. Hans-Eckhard Scholz, MAN TURBO AG

Dipl.-Ing. Paul Thierse, Aucoteam

Dr.-Ing. Jörg Weißkopf, simus systems



Weitere Projektmitarbeiter:

Dipl.-Ing. Sven Düselmann, MAN TURBO AG

Dipl.-Inform. Ingrid Gosheger, Eigner

Dipl.-Ing. Indra Kusumah, RPK

Dipl.-Inform. Iris Schumann, Atos Origin

Dipl.-Ing. Horst Uphoff, MAN TURBO AG

## **Impressum**

Universitätsverlag Karlsruhe  
c/o Universitätsbibliothek  
Straße am Forum 2  
D-76131 Karlsruhe

[www.uvka.de](http://www.uvka.de)

© Universitätsverlag Karlsruhe 2005  
Print on Demand

ISBN 3-937300-25-2

## Vorwort des Herausgebers

In Anbetracht der voranschreitenden Globalisierung sehen sich viele Unternehmen immer größerem Konkurrenzdruck ausgesetzt. Sie sind gefordert, die Entwicklungszeiten zu verkürzen, wodurch die schnelle Verfügbarkeit von Informationen eine strategische Bedeutung gewinnt. Erfahrungswissen wird mehr und mehr zum wichtigsten Kapital von Unternehmen. Aufgrund der wachsenden Datenbestände, wird jedoch das bedarfsgerechte Auffinden von Erfahrungswissen immer schwieriger. Das Erfahrungswissen ist in unterschiedlichen Dokumentenarbeiten abgelegt, wie z.B. 3D-CAD-Modelle, Arbeitspläne, technische Zeichnungen, Prüf- und Sachberichte etc. Können die Dokumente und Daten nicht in kurzer Zeit wiedergefunden werden, wenn es gilt eine gleiche oder ähnliche Aufgabenstellung zu bearbeiten, sind sie für das Unternehmen wertlos geworden.

Bereits in den frühen 60er Jahren wurde erkannt, dass Klassifikation ein geeignetes Mittel zur Strukturierung heterogener Datenbestände ist und somit den Zugriff auf Erfahrungswissen ermöglicht. Allerdings waren diese Ansätze durch großen manuellen Aufwand und Fehleranfälligkeit bei der Einordnung von Objekten in ein Klassifikationssystem geprägt. Ebenfalls entstehen bei einer manuellen Erstellung bzw. unternehmensspezifischen Anpassung eines Klassifikationssystems sehr hohe Aufwände. Werden bestehende Klassifikationssysteme nachträglich geändert, da sich beispielsweise die Produktpalette des Unternehmens geändert hat, müssten die Altdatenbestände nachklassifiziert werden, was in den meisten Fällen nahezu ausgeschlossen ist.

Bisherige Ansätze zur automatisierten Klassifikation, wie z.B. der Einsatz künstlicher Neuronaler Netze, genetische Algorithmen oder Fourier-Transformation führten nicht zu den gewünschten Resultaten. Wesentliches Problem hierbei war, dass die produktbeschreibenden Daten in einer heterogenen Systemlandschaft verteilt liegen. Weiterhin sind die Klassifikationsergebnisse für den Menschen nicht nachvollziehbar und können somit nicht verifiziert oder modifiziert werden.

Innerhalb des zweijährigen Verbundprojektes KLAUSTER (Klassifikationssysteme automatisiert erstellen) wurde aufbauend auf den Ergebnissen des Vorgängerprojektes PAK (Produkte automatisch klassifizieren) ein Konzept erarbeitet und validiert, das eine Automatisierung des Klassifizierungsprozesses erlaubt. Während innerhalb von PAK die automatische Einordnung von Objekten in eine existierende Klassenstruktur behandelt wurde, beschäftigt sich KLAUSTER mit der automatischen Erstellung eines unternehmensspezifischen Klassensystems. Der vorliegende Abschlussbericht beschreibt neben dem Konzept ausführlich Anwendungsbeispiele bei den beteiligten Projektpartnern.

Mein Dank gilt den ehemaligen Projektmitarbeitern und den Autoren der verschiedenen Projektpartner, die wesentlich zum Gelingen des Projektes beigetragen haben. Hierbei möchte ich insbesondere Herr Dr. Weißkopf erwähnen, der innerhalb seiner Zeit als Mitarbeiter des RPK die Koordination des Projektes KLAUSTER durchgeführt hat.

Besonderer Dank gilt dem Ministerium für Bildung und Forschung (BMBF) als Projektförderer sowie dem Projektträger Produktion und Fertigungstechnologien (PFT) am Forschungszentrum Karlsruhe (FZK). Hierbei möchte ich mich bei Herrn Edwin Steinebrunner sowie Herrn Andreas Gässler für die gute Zusammenarbeit bedanken.

Karlsruhe, im Januar 2005

Prof. Dr.-Ing. Dr. h.c. Hans Grabowski

Institut RPK

Universität Karlsruhe (TH)



**Inhalt**

<b>INHALT</b>	<b>I</b>
<b>ABBILDUNGSVERZEICHNIS</b>	<b>V</b>
<b>ABKÜRZUNGSVERZEICHNIS</b>	<b>IX</b>
<b>1 EINLEITUNG</b>	<b>11</b>
1.1 Zielsetzung	12
1.2 Inhalt und Aufbau des Abschlussberichtes	13
<b>2 KLASSENSYSTEME ALS BASIS FÜR EFFIZIENTES PLM</b>	<b>14</b>
2.1 Von PDM zu PLM	14
2.2 Probleme durch ineffizientes Datenmanagement in der Produkterstellung	15
2.2.1 <i>Mangelnde Möglichkeiten zur Wiederholteilsuche</i>	15
2.2.2 <i>Keine Unterstützung der Alternativteilsuche</i>	17
2.2.3 <i>Geringe Losgrößen</i>	17
2.2.4 <i>Inkonsistenzen zwischen Anwendungsprogrammen entlang der Prozesskette</i>	18
2.2.5 <i>Mangelnde Integration von Zulieferern</i>	19
2.3 Ansätze zur Teileverwaltung in der Produktentwicklung	19
2.3.1 <i>Ordnungsschemata und Klassifizierungssysteme</i>	19
2.3.2 <i>Sachmerkmalelisten</i>	21
2.3.3 <i>Klassensysteme</i>	22
2.4 Strukturierungsregeln für Klassensysteme	25
2.4.1 <i>Regel 1: Anwendungsgebiet</i>	26
2.4.2 <i>Regel 2: Höhere Klassifizierungsebenen</i>	26
2.4.3 <i>Regel 3: Untere Klassifizierungsebenen (Instanziierungsregel)</i>	27
2.4.4 <i>Regel 4: Homogenität einzelner Klassen</i>	28
2.4.5 <i>Regel 5: Variation der Perspektive auf Klassenhierarchien</i>	28
2.4.5.1 <i>Regel 5a: Maximale Anwendbarkeit</i>	28
2.4.5.2 <i>Regel 5b: Klassenmerkmale</i>	28
2.4.5.3 <i>Regel 5c: Ausprägung der Klassenmerkmale</i>	29
2.4.6 <i>Regel 6: Geeignete Merkmale für Klassen</i>	29
2.4.7 <i>Regel 7: Semantische Identität von Merkmalen</i>	30
2.4.8 <i>Regel 8: Instanzierbarkeit von vererbten Merkmalen</i>	31
2.4.9 <i>Beachtung der ist_ein Semantik</i>	32
2.4.10 <i>Klassifizierung vom Abstrakten zum Konkreten</i>	32
2.4.11 <i>Form der Klassenhierarchie</i>	33
2.4.12 <i>Sichten auf eine Klassenhierarchie</i>	33
2.5 Anwendungsszenarien für Klassensysteme im Produktlebenszyklus	34
2.5.1 <i>Unterstützung der frühen Phasen des Konstruktionsprozesses</i>	35
2.5.2 <i>Effizientere Wiederholteilsuche</i>	39
2.5.3 <i>Unterstützung der automatischen Produktkonfiguration</i>	40
2.5.4 <i>Zentraler Teilestamm</i>	41
2.5.5 <i>Suche nach alternativen Komponenten im Service</i>	43
2.5.6 <i>Reduzierung der Produktkomplexität durch Produktbereinigung</i>	44
2.5.7 <i>Unterstützung des Recycling</i>	46
2.6 Voraussetzungen für die Einführung eines Klassensystems	47

<b>3</b>	<b>AUTOMATISIERTE ERSTELLUNG VON UNTERNEHMENSSEZIFISCHEN KLASSIFIKATIONSSYSTEMEN</b>	<b>50</b>
3.1	Konzept zur automatisierten Erstellung von Klassifikationssystemen	50
3.1.1	<i>Datenanalysephase</i>	50
3.1.2	<i>Merkmalsermittlung</i>	51
3.1.3	<i>Aufbau eines Klassifikationssystems</i>	53
3.1.4	<i>Anwendung eines Klassifikationssystems</i>	55
3.2	Softwaretechnische Realisierung	55
3.2.1	<i>Systemarchitektur</i>	55
3.2.2	<i>Informationsmodell</i>	58
3.2.2.1	Informationsmodell zur Abbildung heterogener Datenstrukturen	58
3.2.2.2	Informationsmodell zur Abbildung und Verwaltung von Merkmalen	60
3.2.2.3	Informationsmodell zur Abbildung von Berechnungsvorschriften	61
3.2.2.4	Informationsmodell zur Abbildung von Klassen und Klassenhierarchien	62
3.3	Aufbau von unternehmensspezifischen Klassifikationssystem	64
3.3.1	<i>Installation und Starten der KLASSTER-Software</i>	64
3.3.2	<i>Anwendungsszenario</i>	65
3.3.3	<i>Nutzdatenquelle anbinden</i>	68
3.3.3.1	Typ der Nutzdatenquelle auswählen	68
3.3.3.2	Zugriffsparameter spezifizieren	70
3.3.3.3	Relationen verifizieren/modifizieren	72
3.3.4	<i>Merkmalsgenerierung</i>	73
3.3.4.1	Nutzdatenquelle(n) auswählen	73
3.3.4.2	Identifizierende(s) Merkmal(e) verifizieren/modifizieren	73
3.3.4.3	Klassifikationsrelevante Merkmale verifizieren/festlegen	74
3.3.4.4	Relevante Null/NOT-Null-Merkmale verifizieren/festlegen	75
3.3.4.5	Merkmale für Schlüsselwortgenerierung auswählen	75
3.3.4.6	Konfiguration des Schlüsselwortgenerators	75
3.3.4.7	Relevante Schlüsselworte auswählen	77
3.3.4.8	Merkmale für die Extraktion numerischer Anteile auswählen	77
3.3.4.9	Merkmale zur Generierung mathematischer Verknüpfungen auswählen	77
3.3.4.10	Weitere implizite Merkmale manuell generieren	77
3.3.5	<i>Klassensystem generieren und strukturieren</i>	79
3.3.5.1	Identifizierendes und klassifikationsrelevante Merkmale selektieren	80
3.3.5.2	Gewichtungen und Prioritäten verifizieren	80
3.3.5.3	Cluster-Analyse durchführen	81
3.3.5.4	MinObj/Level für Klassengenerierung festlegen	82
3.3.5.5	Generierte Hierarchie verifizieren	83
3.3.6	<i>Klassifikation durchführen</i>	84
3.3.6.1	Nutzdaten und Hierarchien wählen	84
3.3.6.2	Klassifikationsergebnis anzeigen	84
3.3.7	<i>KLASSTER Wizard</i>	85
<b>4</b>	<b>PRAXISEINSATZ UNTERNEHMENSSEZIFISCHER KLASSIFIKATIONSSYSTEME</b>	<b>94</b>
4.1	Klassifikation von Projektdaten in der Gebäudeautomation	94
4.1.1	<i>Problemstellung und Ziele</i>	94
4.1.1.1	Spezifische Problemstellung in der Automatisierungstechnik	94
4.1.1.2	Ziele der KLASSTER-Anwendung	95
4.1.2	<i>Einsatz der KLASSTER-Software</i>	96
4.1.2.1	Einsatzkonzept	96
4.1.2.2	Datenquellen	96
4.1.2.3	Merkmalsgewinnung aus DXF-Dateien	98

4.1.2.4	Merkmalsgewinnung aus Excel-Dateien	102
4.1.3	<i>Anwendungstest</i>	104
4.1.3.1	Organisatorische Vorgehensweise	104
4.1.3.2	Arbeitsablauf der KLAUSTER-Anwendung	105
4.1.3.3	Testdaten	110
4.1.4	<i>Ergebnis und zukünftige Nutzung</i>	114
4.1.4.1	Bewertung der Erprobungsergebnisse	114
4.1.4.2	Zukünftige Nutzungsszenarien für KASTER	114
4.2	Anbindung des CSM-Softwaresystems Remarc an die KLAUSTER-Software	116
4.2.1	<i>Problemstellung</i>	116
4.2.2	<i>Basisdaten</i>	117
4.2.3	<i>Aspekte der Realisierung</i>	118
4.2.4	<i>Ergebnisse</i>	124
4.3	Klassifikation von Maschinenelementen in der Produktentwicklung durch Anbindung an Unigraphics	124
4.3.1	<i>Problemstellung</i>	124
4.3.2	<i>Basisdaten</i>	125
4.3.3	<i>Vorgehensweise</i>	125
4.3.4	<i>Ergebnis / Nutzen</i>	132
4.4	Verwendung und Pflege von unternehmensspezifischen Klassifikationssystemen in SAP/R3	132
4.4.1	<i>Problemstellung</i>	132
4.4.2	<i>Sichten auf Stammdaten</i>	133
4.4.3	<i>Szenarienbeschreibung</i>	136
4.4.4	<i>KLAUSTER-Einsatz bei MAN TURBO AG</i>	138
4.4.5	<i>Technische Umsetzung der SAP Schnittstelle</i>	142
4.4.6	<i>Ergebnis/Nutzen</i>	148
4.5	Klassifikation von Teiledaten im Bereich Unterhaltungselektronik	150
4.5.1	<i>Problemstellung</i>	150
4.5.2	<i>Testdaten</i>	150
4.5.3	<i>Testdurchführung</i>	150
4.5.4	<i>Ergebnisse</i>	154
4.6	Kommerzielle Umsetzung der Ergebnisse aus KLAUSTER	154
4.6.1	<i>Übersicht</i>	154
4.6.2	<i>Ein Referenzprojekt der SIMUS-Systems GmbH, Karlsruhe</i>	154
<b>5</b>	<b>ZUSAMMENFASSUNG UND AUSBLICK</b>	<b>155</b>
<b>6</b>	<b>ANHANG</b>	<b>157</b>
6.1	Projektpartner	157
6.1.1	<i>Atos Origin</i>	157
6.1.1.1	Überblick	157
6.1.1.2	Product Lifecycle Management bei Atos Origin	157
6.1.1.3	Zielsetzung zur Nutzung von KLAUSTER	158
6.1.2	<i>Institut für Rechneranwendung in Planung und Konstruktion (RPK)</i>	159
6.1.2.1	Übersicht	159
6.1.2.2	Zielsetzung zur Nutzung von KLAUSTER	160
6.1.3	<i>Arc Solutions GmbH</i>	160
6.1.3.1	Übersicht	160
6.1.3.2	Zielsetzung zur Nutzung von KLAUSTER	161
6.1.4	<i>AUCOTEAM GmbH</i>	161
6.1.4.1	Übersicht	161
6.1.4.2	Zielsetzung zur Nutzung von KLAUSTER	162

---

6.1.5	<i>Das Unternehmen MAN TURBO AG</i>	163
6.1.5.1	Überblick	163
6.1.5.2	Motivation und Ausgangslage für die Teilnahme am KLAUSTER -Projekt	166
6.1.5.3	Zielsetzung zur Nutzung von KLAUSTER	167
6.1.6	<i>Firma EIGNER Germany GmbH</i>	167
6.1.6.1	Übersicht	167
6.1.6.2	Zielsetzung zur Nutzung von KLAUSTER	168
6.1.7	<i>Arc Solutions GmbH</i>	169
6.1.7.1	Übersicht	169
6.1.7.2	Zielsetzung zur Nutzung von KLAUSTER	169
<b>7</b>	<b>LITERATUR</b>	<b>170</b>

**Abbildungsverzeichnis**

Bild 2-1: Ordnungsschema .....	20
Bild 2-2: Verschlüsselung eines Rotationsteils nach Opitz .....	21
Bild 2-3: Sachmerkmaleiste nach DIN 4000 .....	22
Bild 2-4: Beispiel eines Klassensystems .....	23
Bild 2-5: Aufbau eines Klassensystems .....	24
Bild 2-6: Konzept der Referenzhierarchie .....	25
Bild 2-7: Unvollständiges Anwendungsgebiet .....	26
Bild 2-8: Überflüssige Klassen .....	27
Bild 2-9: Fehlende Klassen .....	27
Bild 2-10: Nicht anwendbares Merkmal .....	28
Bild 2-11: Maximierung der Anwendbarkeit vererbter Merkmale .....	28
Bild 2-12: Klassenmerkmale .....	29
Bild 2-13: Wahl geeigneter Merkmale .....	30
Bild 2-14: Erfüllung des Austauschbarkeitskriteriums .....	30
Bild 2-15: Nicht fakturierbares Merkmal .....	31
Bild 2-16: Klassifizierung gemäß der Konstruktionsphasen .....	33
Bild 2-17: Korrekte Form der Klassenhierarchie .....	33
Bild 2-18: Falsche Sichtenbildung .....	34
Bild 2-19: Anwendungsszenarien entlang des Produktlebenszyklus .....	34
Bild 2-20: Analogie Bauteilbibliothek -- Funktionsbibliothek .....	36
Bild 2-21: Produkt- und Katalogebene, Beziehungen .....	37
Bild 2-22: Suchmöglichkeiten in einem Klassensystem .....	40
Bild 2-23: Produktkonfiguration .....	40
Bild 2-24: Produktkonfiguration mit Hilfe von Merkmalen aus einem Klassensystem .....	41
Bild 2-25: Zentraler Teilestamm .....	42
Bild 2-26: Notwendige Beziehungen .....	42
Bild 2-27: Bauteil und Produktlebenszyklus .....	43
Bild 2-28: Beziehungen innerhalb eines Klassensystems .....	44
Bild 2-29: Datenmodell für die Produktbereinigung .....	45
Bild 2-30: Austausch von Zuliefererdaten .....	47
Bild 2-31: Aufbau und Integration eines Klassensystems .....	48
Bild 3-1: Übersicht über die vier Phasen des Lösungsansatzes [Wei-02] .....	50
Bild 3-2: Ermittlung nicht-klassifikationsrelevanter Merkmale [Wei-02] .....	52
Bild 3-3: Gewinnung zusätzlicher klassifikationsrelevanter Merkmale (1) .....	53
Bild 3-4: Verknüpfung kubischer Vektorräume [Wei-02] .....	54
Bild 3-5: Erzeugen hierarchischer Strukturen durch Clusterprojektionen [Wei-02] .....	55
Bild 3-6: Systemarchitektur .....	56
Bild 3-7: Informationsmodell der relationalen Sicht auf Nutzdatenquellen [Wei-02] .....	59
Bild 3-8: Informationsmodell zur Abbildung und Verwaltung von Merkmalen [Wei-02] .....	61
Bild 3-9: Informationsmodell zur Abbildung von Berechnungsvorschriften [Wei-02] .....	62
Bild 3-10: Informationsmodell zur Abbildung von Klassen und Klassenhierarchien .....	63
Bild 3-11: Nutzdatenquelle anbinden .....	65
Bild 3-12: Schlüsselwortgenerator konfigurieren .....	66
Bild 3-13: Schlüsselwortmerkmale ableiten .....	66
Bild 3-14: Ergebnis der Cluster-Analyse .....	67
Bild 3-15: Typ der Nutzdatenquelle auswählen .....	69
Bild 3-16: Anbindung der Nutzdaten .....	69
Bild 3-17: Laden des Schemas .....	70
Bild 3-18: Verwaltung von Nutzdatenquellen .....	71
Bild 3-19: Schema einer Nutzdatenquelle .....	72

Bild 3-20: Definierte Relationenverifizieren/modifizieren .....	73
Bild 3-21: Auswahl der Nutzdatenquelle(n) .....	73
Bild 3-22: Verifizierung/Modifizierung des identifizierenden Merkmals .....	74
Bild 3-23: Verifizierung der klassifikationsrelevanten Merkmale .....	74
Bild 3-24: Null/NOT-Null-Merkmale verifizieren/festlegen .....	75
Bild 3-25: Schlüsselwortgenerator konfigurieren .....	76
Bild 3-26: Auswahl der relevanten Schlüsselworte .....	77
Bild 3-27: Erstellung eines neuen Verzeichnis für weitere implizite Merkmale .....	78
Bild 3-28: Verfügbares Merkmal auswählen .....	79
Bild 3-29: Die Regel des impliziten Merkmals generieren .....	79
Bild 3-30: Selektion und Einfügen der Merkmale im VektorSpace .....	80
Bild 3-31: Gewichtung und Priorität definieren .....	81
Bild 3-32: Cluster-Analyse durchführen .....	82
Bild 3-33: Festlegung des MinObj/Level .....	83
Bild 3-34: Verifizierung der generierten Hierarchie .....	83
Bild 3-35: Nutzdaten und Hierarchien wählen .....	84
Bild 3-36: Klassifikationsergebnis .....	85
Bild 3-37: Klaster-Wizard .....	86
Bild 3-38: Bereiche im KLAUSTER-Wizard .....	87
Bild 3-39: Datenbanktyp zur Auswahl .....	87
Bild 3-40: Zugriffsparameter .....	88
Bild 3-41: Nutzdatenquelle .....	88
Bild 3-42: Identifizierendes Merkmal .....	89
Bild 3-43: klassifikationsrelevante Merkmale .....	89
Bild 3-44: Merkmale für Schlüsselwortgenerierung .....	89
Bild 3-45: Konfiguration für Schlüsselwortgenerierung .....	90
Bild 3-46: Schlüsselwörter auswählen .....	91
Bild 3-47: Null/Not-Null-Merkmale .....	91
Bild 3-48: Merkmale mit mathematischer Verknüpfungen .....	92
Bild 3-49: Generierung impliziter Merkmale .....	93
Bild 3-50: Merkmalübersicht .....	93
Bild 4-1: Beispiel eines R&I-Schemas .....	97
Bild 4-2: Symbole für Luft-/Wassertauscher und Wärmerückgewinnung .....	98
Bild 4-3: Heizregisterregelung mit Substruktur Luft-/ Wasser-Wärmetauscher .....	99
Bild 4-4: Heizregisterregelung mit den Substrukturen Luft-/ Wasser-Wärmetauscher und Wärmerückgewinnung (WRG) .....	99
Bild 4-5: Open-Source DXF-Viewer .....	100
Bild 4-6: Automatische Erkennung von Symbolen in DXF-Dateien .....	101
Bild 4-7: Anzahl und Position von Symbolen sind nutzbare Informationen zur Klassifikation .....	102
Bild 4-8: Beispiel einer Motorentabelle .....	103
Bild 4-9: Leistungsklassen der Motoren von ausgewerteten Motorenlisten .....	103
Bild 4-10: Datenfluss zur Datenaufbereitung und Durchführung der Klassifikation .....	104
Bild 4-11: Auswählen der zu klassifizierenden Dateien .....	105
Bild 4-12: geöffnete Nutzdatenquellen .....	106
Bild 4-13: Erkennung klassifikationsrelevanter Merkmale .....	106
Bild 4-14: KLAUSTER-Anzeige der visualisierte Regel für implizite Merkmale .....	107
Bild 4-15: Anzeige zum Klaster VectorSpace .....	108
Bild 4-16: KLAUSTER-Anzeige der generierten Klassenhierarchie .....	108
Bild 4-17: PAK-Anzeige der Klassifikationsergebnisse .....	109
Bild 4-18: Auswahl (1) Grundsymbole .....	110
Bild 4-19: Auswahl (2) Grundsymbole .....	111
Bild 4-20: Auswahl (3) Grundsymbole .....	112

Bild 4-21: Substrukturen zu Heizung, Lüftung, Klima .....	113
Bild 4-22: Anwendungsszenarium von KLAUSTER im Projektierungsprozess .....	116
Bild 4-23: Basisdaten DIN 4000 .....	118
Bild 4-24: XML- Normalisierung .....	119
Bild 4-25: Automatisierte Verknüpfung von DIN- & ERP-Daten.....	120
Bild 4-26: vorgegebenes Klassenschema .....	120
Bild 4-27: Clusterbildung für Schrauben .....	121
Bild 4-28: Automatisch generiertes, natives CAD-Part .....	122
Bild 4-29: geometrisches Merkmal „Bohrung-achsparell“ .....	126
Bild 4-30: geometrische Merkmale der Teileklasse „Labyrinthdichtung“ (Auszug) .....	126
Bild 4-31: Beispielteil der Klasse „Labyrinthdichtung“ in Unigraphics.....	127
Bild 4-32: Architektur der Schnittstelle .....	128
Bild 4-33: XML-Schema „AssemblComp“ .....	128
Bild 4-34: XML-Schema „CompFace“ .....	129
Bild 4-35: implizite KLAUSTER-Regel „Anzahl Querbohrungen mit Sackloch“ .....	130
Bild 4-36: Art.-Nr.: 1-11865394 (besitzt 8 gestufte Parallelbohrungen).....	131
Bild 4-37: Sichtenwahl .....	133
Bild 4-38: Nutzdatenquelle: CAD-Systeme .....	134
Bild 4-39: Nutzdatenquelle: ERP-System, Stückliste, Baugruppenstruktur (PSP).....	134
Bild 4-40: Nutzdatenquelle: Arbeitspläne, NC-Programme, Werkzeugverwaltung.....	135
Bild 4-41: Spezifische Klassifikationssysteme entsprechend den Anforderungen.....	135
Bild 4-42: Auswahl von SAP als Nutzdatenquelle .....	138
Bild 4-43: Ergebnis der Analyse der drei SAP Tabellen DRAD, MARA und DRAW .....	139
Bild 4-44: Ergebnis der Merkmalanalyse der SAP-Datenquelle .....	139
Bild 4-45: Definition der klassifikationsrelevanten Merkmale für die Klassifikation der SAP-Daten .....	140
Bild 4-46: Werteverteilung der Merkmale anzeigen und Gewichtung definieren.....	141
Bild 4-47: Automatisch erzeugtes Klassifikationssystem für SAP-Daten.....	142
Bild 4-48: Architektur der KLAUSTER - SAP R/3 Schnittstelle.....	143
Bild 4-49: Testtool für Low-Level Schnittstelle .....	145
Bild 4-50: Anzeige des BAPI zur Merkmalspflege im BAPI Explorer in SAP R/3 .....	146
Bild 4-51: Ergebnis der Zuordnung Merkmal zu Klasse in SAP R/3 .....	147
Bild 4-52: Das manuell gepflegte Klassifikationsschema am Standort Zürich ist funktionsorientiert .....	149
Bild 4-53: Benennung nicht bezeichneter Spalten (Bustaben A bis M) .....	151
Bild 4-54: Resultierende Merkmalsstruktur .....	152
Bild 4-55: Erzeugte Klassenhierarchie.....	153
Bild 6-1: Wiege der Ruhrindustrie - Stammhütte „Gute Hoffnung“, Oberhausen-Sterkrade 1834.....	163
Bild 6-2: Firmensitz und produzierende Standorte der MAN TURBO AG .....	165
Bild 6-3: Fertigung bei der MAN TURBO AG .....	166





## Abkürzungsverzeichnis

<b>API</b>	Application Programming Interface
<b>ASCII</b>	American Standard Code for Information Interchange
<b>B-REP</b>	Boundary Representation (engl.): Geometrisch-topologisches Strukturmodell
<b>CAD</b>	Computer Aided Design
<b>CAE</b>	Computer Aided Engineering
<b>CAM</b>	Computer Aided Manufacturing
<b>CORBA</b>	Common Object Request Broker Architecture
<b>DIN</b>	Deutsches Institut für Normung
<b>DK</b>	Dezimalklassifikation
<b>DV</b>	Datenverarbeitung
<b>EAN</b>	European Article Numbering
<b>EDM</b>	Engineering Data Management
<b>EDV</b>	Elektronische Datenverarbeitung
<b>ERP</b>	Enterprise Resource Planning
<b>FEM</b>	Finite Elemente Methode
<b>ISO</b>	International Standardization Organization
<b>IT</b>	Informationstechnik
<b>SI</b>	Système International d' Unités (franz.) – Internationales Einheitensystem
<b>KI</b>	Künstliche Intelligenz
<b>NAICS</b>	North American Industry Classification System
<b>NIGP</b>	National Institute of Government Purchasing
<b>PDM</b>	Product Data Management
<b>PLM</b>	Product Lifecycle Management
<b>PPM</b>	Produkt- und Produktionsmodell
<b>PPS</b>	Produktionsplanung und -steuerung
<b>SIC</b>	Standard Industry Classification
<b>SML</b>	Sachmerkmalreihe
<b>SQL</b>	Structured Query Language
<b>STEP</b>	Standard for the Exchange of Product model data (ISO-10303)
<b>UML</b>	Unified Modelling Language
<b>UN/SPSC</b>	United Nations Standard Products and Services Code
<b>UNO</b>	United Nations Organisation
<b>UPC</b>	Universal Product Code
<b>VDI</b>	Verein Deutscher Ingenieure
<b>VDMA</b>	Verein Deutscher Maschinen- und Anlagenbauer
<b>XML</b>	Extensible Markup Language



## 1 Einleitung

Angesichts steigenden Wettbewerbsdrucks und zunehmender Globalisierung kommt der schnellen Verfügbarkeit von Informationen zunehmend strategische Bedeutung zu. Durch die sich ständig verkürzenden Produktentwicklungszeiten sowie den massiven Einsatz rechnerunterstützter Verfahren steigt die Menge der verfügbaren Informationen sehr stark an. Die Produktion immer neuen Wissens führt jedoch ohne eine Systematisierung des vorhandenen Wissens in eine Sackgasse, da die jeweils relevanten Informationen in der ständig wachsenden Flut an Daten mit vertretbarem Aufwand nicht mehr auffindbar sind. Gespeicherte Dokumente und Daten sind im Grunde genommen wertlos, wenn sie nach Abspeicherung nicht in kürzester Zeit wiederzufinden sind und für aktuelle Aufgaben genutzt werden können. Dieser Zustand ist nur mit systemübergreifenden Ordnungssystemen zu verwirklichen, die alle verfügbaren DV-Anwendungen konsequent einschließen [Schö-99],[Wei-02].

Viele Unternehmen versuchen, diese Problematik durch die Einführung von Normteilibibliotheken, ein Stammdatensystem, ein ERP (Enterprise Resource Planning)- oder Datenmanagement-System in den Griff zu bekommen. Diese Systeme stammen zumeist von unterschiedlichen Anbietern und in der Regel handelt es sich hierbei um unverbundene Einzelsysteme. Z.B. betreibt ein großer deutscher Automobilzulieferer firmenintern mehr als fünf verschiedene Sachmerkmal-Systeme. Höhere IT-Kosten in Software-Anschaffung und Wartung sowie hohe Personalkosten sind die daraus unmittelbar resultierenden Folgen.

Dieser Aufwand löst jedoch nicht das Problem bei der Suche nach Standard- und Normbauteilen oder verfügbaren existierenden Konstruktionslösungen. Nun sind die Datenbestände mehrerer Softwaresysteme zu durchsuchen, wobei bereits eine Suchanfrage in einem System viel Zeit in Anspruch nimmt. An dieser Stelle entsteht oftmals ein Teufelskreis: Durch die zeitaufwändige und ineffiziente Suche wird ein Konstrukteur von seiner eigentlichen Aufgabe – Neues zu planen, zu entwickeln und zu entwerfen – abgehalten. Eine Suche nach bestehenden Konstruktionslösungen wird oft abgebrochen, da die Recherche zeitaufwändiger ist als eine Neukonstruktion des entsprechenden Teiles. Beispielsweise verbringen Ingenieure einer Konstruktionsabteilung eines Fertigungsunternehmens im Schnitt ca. 45% ihrer Arbeitszeit mit der Recherche nach Standardbauteilen, Baugruppen und existierenden Lösungen [VDI-2210].

Deshalb ist die automatische Klassifikation von Produkten ein schon lange gehegter Wunsch in der industriellen Praxis. Dieses Thema wurde innerhalb des Verbundprojektes „Automatische Klassifikation von Produkten“ (PAK) im Rahmenkonzept „Produktion 2000“ bearbeitet. Dieses Projekt hatte die Konzeption und Realisierung eines Programmsystems zum Ziel, welches die automatische Einordnung von Produkten in bestehende Klassifikationsschemata durchführt und damit die bisher manuell oder halbmaschinell durchgeführten Klassifikations-tätigkeiten ersetzt.

Die Resonanz seitens der Industrie zu den Ergebnissen des Projektes waren durchweg sehr positiv. Jedoch hat sich im Verlauf des Projektes gezeigt, dass die Einordnung in existierende Klassifikationssysteme dahingehend Probleme bereitet, dass solche Klassifikationssysteme in den Unternehmen vielfach nicht oder nur unzureichend entwickelt sind. Der Einsatz allge-

meingültiger Klassifikationssysteme, z.B. nach Opitz oder eCl@ss, wurde als nicht praktikabel erkannt. Sie genügen nicht den unternehmensspezifischen Anforderungen an die Klassifikation, weil sie beispielweise nicht auf das Produktspektrum eines Unternehmens abgestimmt sind. Die Entwicklung unternehmensspezifischer Klassifikationssysteme ist sehr zeit- und kostenaufwändig und zudem fehleranfällig. Zum Zweiten ist die Durchführung der Klassifikation, d.h. die Zuordnung von Objekten in ein Klassifikationssystem, aufwändig und fehleranfällig. Der Aufwand wird in [GaKn-86] mit 5 bis 10 Minuten für ein zu klassifizierendes Einzelteil quantifiziert. Die Fehleranfälligkeit einer manuellen Klassifikation liegt laut [GaKn-86] bei 25%, d.h. jedes vierte Einzelteil wird falsch klassifiziert.

Auf der anderen Seite ist das Nutzenpotenzial, das sich durch die sorgfältige Strukturierung und Klassifikation der Datenbestände eines Unternehmens ergibt, enorm. Erfahrungswissen kann verfügbar gemacht werden, die Wiederverwendungsrate steigt, die Teilevielfalt lässt sich reduzieren, etc. Um dieses Potenzial bei gleichzeitiger Minimierung des Aufwandes und der Fehleranfälligkeit zu erschließen, wurde das Nachfolgeprojekt „Klassifikationssysteme automatisch erstellen“ KLAster initiiert, das eine automatisierte Erstellung einer unternehmensspezifischen Klassifikation ermöglichen soll. Wie bereits beim Vorgänger bewerteten die beteiligten Industrievertreter die Projektergebnisse sehr positiv.

## 1.1 Zielsetzung

Inhalt des Verbundprojektes KLAster war die Konzeption und Realisierung eines Softwaresystems zur automatisierten Klassifikation von Produktdaten. Hierbei war zu berücksichtigen, dass die Datenbestände heutiger Unternehmen in der Regel in einer Vielzahl unterschiedlicher Softwaresysteme in einer heterogenen IT-Landschaft vorliegen. Dazu wurden Adapter für Repräsentanten von Anwendungstypen (z.B. CAD-System, PDM-System, SQL-Datenbanken) realisiert, die klassifikationsrelevante Produktdaten innerhalb eines Unternehmens verwalten.

Ziel des Projektes war es nicht, ein Softwaresystem zu implementieren, das ein Klassifikationssystem vollständig automatisch erstellt, sondern es soll den Anwender unterstützen. Beispielsweise muss das System potentiell klassifikationsrelevante Merkmale aus den verschiedenen Datenquellen extrahieren und dem Anwender vorschlagen. Basierend auf diesen Vorschlägen, wählt der Anwender die Merkmale aus, nach denen klassifiziert werden soll. Auf diese Weise kann das Klassifikationssystem entsprechend den Bedürfnissen des Unternehmens erstellt werden. Es ist auch möglich, innerhalb eines Unternehmens mehrere Klassifikationssysteme zu haben. Beispielsweise könnte für jede Fachabteilung eines Unternehmens (z.B. Einkauf, Service, Vertrieb, Konstruktion) ein eigenes Klassifikationssystem erstellt werden, das die unterschiedlichen Arbeitsabläufe unterstützt.

## 1.2 Inhalt und Aufbau des Abschlussberichtes

Im vorliegenden Abschlussbericht werden die Ergebnisse des Forschungsvorhabens zusammengefasst. In Kapitel zwei erhält der Leser eine thematische Einführung in das Problemumfeld der (automatisierten) Klassifikation und deren Nutzenpotential für die Unternehmen. Das dritte Kapitel beschreibt das theoretische Konzept sowie die softwaretechnische Umsetzung. Weiterhin wird die Anwendung der KLAUSTER-Software beschrieben, welche auf der Demo-CD mit diesem Buch mitgeliefert wird.

In Kapitel vier werden Anwendungsszenarien beschrieben, die den Einsatz der Software bei den im Projekt beteiligten Unternehmen beschreiben. Dies zeigt neben dem Nutzenpotential auch die Einsatzfähigkeit der erarbeiteten Projektergebnisse.

Die mitgelieferte CD enthält neben einer Demo-Version der Software und Benutzerhandbuch eine HTML-Dokumentation der gesamten Implementierung mit zugehörigen UML-Klassendiagrammen. Weiterhin sind die Präsentationen der Abschlussveranstaltung darin enthalten.

## 2 Klassensysteme als Basis für effizientes PLM

Der Übergang vom klassischen Produktdatenmanagement hin zum Product Lifecycle Management stellt sowohl Anforderungen an die Funktionalität der eingesetzten IT-Systeme, als auch an die enthaltenen Datenmodelle. Anhand von Beispielen aus verschiedenen Phasen des Produktlebenszyklus wird in diesem Kapitel gezeigt, welche Probleme durch ineffizientes Datenmanagement in der Produkterstellung entstehen können und welche Vorteile ein Klassensystem bei der Verwaltung von Teiledaten bietet.

### 2.1 Von PDM zu PLM

Die Philosophie und der Leistungsumfang klassischer PDM-Systeme hat sich in der Vergangenheit stark gewandelt. Während die ersten Systeme lediglich für die Zeichnungsverwaltung ausgelegt waren, kamen im Lauf der Zeit Funktionen für die Stücklistenverwaltung, die Produktstrukturierung und die Klassifizierung von Bauteilen hinzu. Ein weiterer Entwicklungsschritt war die Unterstützung von Prozessen wie beispielsweise Freigabe- und Änderungsprozess. Bei all diesen Erweiterungen blieb der Fokus jedoch stets auf die technische Auftragsabwicklung im Unternehmen gerichtet.

Der nächste Entwicklungsschritt im PDM-Bereich ist das Collaborative Product Definition Management (cPDM), das den Prozesskettengedanken und die überbetriebliche Zusammenarbeit in den Vordergrund stellt. Um den reibungslosen Datenaustausch zwischen kooperierenden Unternehmen im Sinne dieses neuen Ansatzes zu gewährleisten ist ein standardisiertes Format für Bauteildaten notwendig. Die ISO 15926 PLIB (Parts Library) ist ein internationaler Standard für den Aufbau von Klassensystemen [ISO-97], der allen Anforderungen des cPDM genügt. Es ist daher sinnvoll, Klassensysteme für die Bauteilverwaltung auf den Konzepten der PLIB aufzubauen.

Gegenwärtig steht die Thematik des Product Lifecycle Management (PLM) im Mittelpunkt des Interesses. PLM erweitert die klassische PDM-Funktionalität von der technischen Auftragsabwicklung hin zur Unterstützung des gesamten Produktlebenszyklus. Dazu muss der Leistungsumfang der Systeme erweitert werden. In späteren Lebensphasen sind beispielsweise Funktionen zur Unterstützung der Wartung oder des Recycling notwendig. Aufgrund der Erweiterung des Funktionsumfangs spielt die Integration verschiedener Softwaresysteme für effizientes PLM eine wichtige Rolle. Neben softwaretechnischen Gesichtspunkten ist eine einheitliche und erweiterte Datenlandschaft in kooperierenden Unternehmen die Grundvoraussetzung dafür. Wichtig Punkte sind in diesem Zusammenhang:

- Unterstützung des Konfigurationsmanagements
  1. Verwaltung von Sachnummern
  2. Verwaltung von Seriennummern

- Verwaltung verschiedener Produktstrukturen (BOM)
  1. „as\_designed“
  2. „as\_built“
  3. „as\_maintained“
- Verwaltung von Daten aus späteren Lebensphasen
  1. Wartungsinformationen
  2. Demontage-Informationen
  3. Recycling-Informationen, z.B. Altautoverordnung (verbotene Stoffe, Listenstoffe)
- Unterstützung von Beziehungen
  1. zwischen Klassenhierarchie und Produktstruktur (BOM)
  2. zwischen Klassenhierarchien
  3. innerhalb der Klassenhierarchie

## **2.2 Probleme durch ineffizientes Datenmanagement in der Produkterstellung**

Grundsätzlich können beim PLM die Bereiche Prozess- und Datenmanagement, sowie Zulieferer- und Kundenintegration unterschieden werden. Voraussetzung für das Funktionieren von PLM ist die Datenintegration auf Zulieferer-, Kunden- und interner Ebene. Gerade im Bereich dieser Integrationsschnittstellen gibt es heute in den Unternehmen jedoch noch einige Schwachpunkte.

### **2.2.1 Mangelnde Möglichkeiten zur Wiederholteilsuche**

Die Datenmengen in der Produktentwicklung wachsen stetig, da ständig neue Produkte entwickelt oder alte Produkte abgeändert werden. Oft gibt es kein zentrales Datenverwaltungssystem für die verschiedenen Bereiche eines Unternehmens und die einzelnen Konstruktionsabteilungen verwalten ihre technischen Unterlagen selbst, oft noch auf Ebene des Dateisystems. Ähnliches gilt für Arbeitsvorbereitung und Fertigung. Ein Werkzeug zur Datenintegration und einfachen Suche über alle konstruierten Teile im Unternehmen gibt es nicht und die Suche nach Teilen ist schwierig und zeitaufwändig. Die Neukonstruktion eines Bauteils hingegen ist heute durch leistungsfähige featurebasierte 3D-CAD-Systeme mit weit weniger Aufwand verbunden als in der Vergangenheit. Der Konstrukteur zieht es daher oft vor, ein Bauteil neu zu konstruieren anstatt nach einem bereits vorhandenen Teil zu suchen.

Die Auswirkungen dieser Vorgehensweise sind gravierend. Durch die wiederholte Konstruktion von gleichen Bauteilen werden wertvolle Ressourcen in der Konstruktionsabteilung ver-

schwendet. Das mehrfache Ablegen ein und derselben Daten führt zu so genannten Dubletten. Durch diese wächst die Datenmenge in der Produktentwicklung unnötig an und verschärft das eingangs beschriebene Problem der Unübersichtlichkeit der Datenbasis. Darüber hinaus verursacht jeder Datensatz Kosten für das Anlegen, sowie seine Verwaltung und Pflege. Die Angaben über deren Höhe schwanken: das Anlegen eines neuen Datensatzes wird zwischen € 2000.- und € 20.000.- beziffert, während sich die jährlichen Verwaltungskosten auf € 1000.- bis € 4000.- belaufen können. Dubletten wirken sich nicht nur im Bereich der Produktentwicklung negativ aus. Sie verursachen auch in den nachfolgenden Bereich des Produktlebenszyklus vermeidbare Kosten. So müssen in Fertigung und Montage für jedes Eigenfertigungsteil Dokumente und Fertigungsmittel in Form von z.B. Arbeitsplänen, NC-Programmen, Werkzeugen usw. bereit gehalten werden. Viele Bauteile müssen durch umfangreiche Tests auf ihre Zuverlässigkeit hin untersucht werden. Diese Tests müssen bei Dubletten unnötigerweise mehrfach vorgenommen werden.

Die Lösung des oben beschriebenen Problems liegt in der Verwendung von Wiederholteilen. Ein Wiederholteil ist ein Bauteil, das in vielen verschiedenen Produkten in immer gleicher Form zum Einsatz kommt. Der Vorteil dabei ist, dass die notwendigen Daten für dieses Teil nur ein einziges Mal gespeichert und verwaltet werden müssen. Zeichnungen, Stücklisten, Arbeitspläne, NC-Programme, Prüfpläne usw. liegen bereits vor und können sofort verwendet werden. Bei der Verwendung eines Wiederholteils ist von Anfang an sicher gestellt, dass das Bauteil im Unternehmen gefertigt werden kann und alle Qualitätsanforderungen erfüllt sind. Die Idee der Wiederholteilverwendung ist nicht neu. Ihre Realisierung scheitert jedoch oft daran, dass auch heute noch in vielen Unternehmen technische Daten in sehr unstrukturierter Form vorliegen, was eine effiziente Wiederverwendung nahezu unmöglich macht.

Voraussetzung für die Strukturierung der Datenbestände und damit auch für die Wiederholteilverwendung ist ein Teileverwaltungssystem in dem alle Daten über Bauteile und alle dazu gehörigen Dokumente zentral für das gesamte Unternehmen erfasst und in Form einer Klassifikationsstruktur verwaltet werden. Dieses System muss leistungsfähige Suchfunktionen bieten, sodass der Konstrukteur schnell zu einem Bauteil findet und auf die erneute Konstruktion verzichtet. Heutige Teileverwaltungssysteme haben jedoch oft gravierende Nachteile, die auf die Art der Datenverwaltung zurück zu führen sind. Insbesondere ist hier die mangelnde Beschreibungstiefe (die Genauigkeit und der Umfang der Beschreibung) zu nennen. Vor der Einführung des Rechners in den Unternehmen wurden Informationen über Bauteile auf Papier dokumentiert und bereitgestellt. Um einen möglichst schnellen Zugriff auf diese Informationen zu gewährleisten, mussten die Dokumente sinnvoll strukturiert und archiviert werden. Aus dieser Notwendigkeit entstanden Klassifizierungssysteme auf der Basis bestimmter Codes (numerisch oder verbal beschreibend), welche sowohl die Klassifizierung, als auch die Identifizierung und Beschreibung eines Bauteils enthielten. Ein Beispiel dafür ist der bekannte „Opitz-Schlüssel“ [Opi-71]. Aufgrund der begrenzten Stellenzahl in einem Code konnten nur wenige Merkmale eines Teils verschlüsselt werden, die Beschreibungstiefe dieser Verfahren ist also sehr gering.

Entsprechend gering sind auch die Möglichkeiten zur Auswertung der codierten Daten. So kann in einem solchen System nur auf Basis derjenigen Merkmale gesucht werden, die im



Code verschlüsselt sind. Darüber hinaus muss dem Konstrukteur für eine effiziente Suche die Art und Weise der Klassifizierung und Verschlüsselung bekannt sein, da die Schlüssel in der Regel nicht „sprechend“ sind, also nicht ohne weitere Kenntnisse interpretiert werden können. Auch nach der Einführung des Rechners blieben Klassifizierungssysteme auf Basis von Codes oftmals erhalten. Die Informationen werden nun nicht mehr auf Papier, sondern auf dem Rechner abgebildet. Die Nachteile sind dabei jedoch dieselben. Die Suche bleibt aufwändig und verlangt umfangreiche Vorkenntnisse über die zugrunde liegende Klassifizierung.

### **2.2.2 Keine Unterstützung der Alternativteilesuche**

Viele Produkte bestehen zum Großteil aus Komponenten, die wiederum in unterschiedlichsten anderen Produkten zum Einsatz kommen. Bei diesen Komponenten kann es sich um Norm- oder Standardteile handeln, die in gleicher Form von vielen Unternehmen verwendet werden oder um firmenspezifische Teile (Werknormteile) die nur in Produkten eines bestimmten Unternehmens zur Anwendung kommen. Um effizient Konstruieren zu können ist es für den Konstrukteur wichtig, sein umfangreiches Grundwissen über möglichst viele Lösungselemente zu besitzen. Da die Anzahl der entwickelten konstruktiven Lösungen und damit auch die Anzahl der verfügbaren Elemente ständig zunimmt, ist es für den einzelnen Konstrukteur heute sehr schwierig in diesem Umfeld den Überblick zu behalten. So ist es nicht verwunderlich, wenn Konstrukteure stets auf diejenigen Elemente zurückgreifen, die sie in der Vergangenheit bereits erfolgreich eingesetzt haben und über die sie das notwendige Hintergrundwissen besitzen. Durch diese Vorgehensweise werden jedoch Komponenten, die neu entwickelt wurden und möglicherweise für einen bestimmten Anwendungsfall besser geeignet sind, bei der Konstruktion nicht berücksichtigt. Um dieses Problem zu lösen ist es wichtig, dass die interne Datenverwaltung dem Konstrukteur die Möglichkeit bietet, zu einem bestimmten Teil Alternativen zu finden, die möglicherweise für den vorliegenden Anwendungsfall besser geeignet sind. Auf diese Art hat der Produktentwickler die Gelegenheit, die Qualität der Konstruktion zu verbessern.

### **2.2.3 Geringe Losgrößen**

Hohe Losgrößen, d.h. eine hohe Anzahl gleicher Teile, senken die Kosten in der Fertigung erheblich. Die Kostensenkung kommt dadurch zustande, dass die Rüstkosten auf eine größere Anzahl gefertigter Teile umgelegt werden können und Rohmaterial in größeren Mengen günstiger eingekauft werden kann. Auch die Kosten für notwendige Werkzeuge, die erst als Betriebsmittel gefertigt werden müssen können auf eine größere Anzahl gefertigter Einheiten umgelegt werden. Dadurch kann ein Fertigungsverfahren wie z.B. das Gießen, das bei geringen Stückzahlen oft unrentabel ist, wieder interessant werden. Der gleiche Effekt tritt auch bei Zulieferteilen auf. Die Kosten pro Stück werden mit zunehmender Stückzahl geringer, da die Fertigungskosten für den Zulieferer sinken.

Es ist also sowohl bei Eigenfertigung, als auch bei Zulieferteilen sinnvoll, die Losgrößen möglichst hoch zu halten. Durch die Neukonstruktion von Teilen, die an anderen Standorten des Unternehmens unter Umständen schon verwendet aber vom Konstrukteur nicht gefunden werden und durch die Verwendung unterschiedlicher Zulieferteile für ein und dieselbe Aufgabe sinken die Losgrößen jedoch sowohl bei Bestellungen, als auch in der Fertigung. Dies ist vor allem bei größeren Unternehmen mit mehreren Konstruktionsabteilungen an verschiedenen Standorten ein Problem. Bei geringeren Losgrößen steigt der Einkaufspreis und auch die Herstellkosten nehmen zu.

Eine einheitliche Datenbasis über alle Standorte des Unternehmens hilft dieses Problem zu lösen. Teile die auch an anderen Standorten verwendet werden, werden gefunden und können eingesetzt werden. Dadurch steigen die Losgrößen sowohl bei Eigenfertigung als auch bei Zulieferkomponenten. Die Vergleichbarkeit von Zulieferteilen trägt ebenfalls zur Lösung des Problems der geringen Losgrößen bei. Wird anstatt zweier verschiedener Zulieferkomponenten nur ein Typ verbaut, so steigt für diese Komponente die Losgröße und der Preis sinkt.

#### **2.2.4 Inkonsistenzen zwischen Anwendungsprogrammen entlang der Prozesskette**

In den verschiedenen Unternehmensbereichen kommen unterschiedliche Software-Systeme zur Datenverwaltung zum Einsatz. In der Konstruktionsabteilung ist dies in der Regel ein PLM-System, in den kaufmännischen Bereichen und der Fertigung ein ERP-System, in anderen Bereichen vielleicht zusätzlich noch ein Dokumentenmanagement-System. Die Daten in den einzelnen Systemen beziehen sich zum großen Teil auf dieselben Komponenten (Teile) und sind damit teilweise redundant (mehrfach vorhanden). Diese Redundanz wird dann zum Problem, wenn Änderungen die in einem System vorgenommen werden, in den anderen Systemen nicht nachgezogen werden. Die Daten sind dann nicht mehr konsistent, es können gravierende Fehler entstehen.

Ein Kunde wünscht z.B. ein anderes Anschlussmaß für ein Bauteil. Er teilt dies dem Vertrieb mit, der diese Information in das ERP-System einpflegt. Die Konstruktionsabteilung verwendet jedoch die Daten aus dem PLM-System, was zu fehlerhaften Fertigungsunterlagen und damit Kosten bei der Fehlerbeseitigung führt. Diese sind umso höher, je später der Fehler entdeckt wird. In diesem Zusammenhang gilt die „10er Regel“: Je später ein Fehler entdeckt wird, umso teurer wird seine Beseitigung – von Phase zu Phase im Produkterstellungsprozess etwa um den Faktor 10 [Ehr-95].

Eine Möglichkeit zum einfachen Abgleich der Daten aus verschiedenen Anwendungsprogrammen entlang der Prozesskette ist daher dringend erforderlich. Der Konstrukteur kann damit stets auf die aktuellsten Daten aus dem ERP-System zugreifen. Dies hilft, die Daten konsistent zu halten und Fehler durch Änderungen die nicht nachgezogen werden zu vermeiden. Dies setzt jedoch eine zentrale Datenhaltung und eine hinreichende Beschreibungstiefe voraus.

### **2.2.5 Mangelnde Integration von Zulieferern**

Um Kosten zu sparen reduzieren viele Unternehmen ihre Geschäftstätigkeit auf ihre jeweiligen Kernkompetenzen. Im Bereich der Produktentwicklung hat diese Strategie zur Folge, dass mehr und mehr Zulieferkomponenten eingesetzt und nur diejenigen Teile selbst konstruiert und gefertigt werden, für die im Unternehmen das notwendige Know How (die Kernkompetenz) verfügbar ist und bei denen sich dieses auch rentiert. Die notwendigen Informationen über Zulieferkomponenten bekommt der Konstrukteur zumeist aus Katalogen (auch online) oder auf spezifische Anfragen hin in Form von Datenblättern.

Ein Problem bei der Suche in Katalogen ist, dass verschiedene Zulieferer ihr Produktspektrum auf unterschiedliche Weise klassifizieren und beschreiben, auch dann, wenn ihre Produkte praktisch identisch sind. Sie werden mit unterschiedlich benannten Merkmalen beschrieben und sind so nicht sofort und direkt vergleichbar, vor allem nicht von einem Rechner. Daher ist es schwierig, zu einem bestimmten Produkt eines Anbieters Alternativen bei anderen Anbietern, die unter Umständen preisgünstiger sind, zu finden. Die Folge ist, dass Konstrukteure meist diejenigen Zulieferkomponenten verwenden, die sie schon kennen, obwohl es vielleicht bessere Alternativen bei anderen Zulieferern gibt.

## **2.3 Ansätze zur Teilverwaltung in der Produktentwicklung**

Die oben genannten Probleme gilt es durch eine effiziente Teilverwaltung im Unternehmen zu lösen. Die Datenhaltung im Bereich der Teilverwaltung ist in den meisten Unternehmen heute jedoch noch geprägt von Klassifizierungs- bzw. Nummernsystemen und Stammdatensätzen. Die Beschreibungstiefe auf Basis solcher Daten ist für viele Anforderungen des PLM jedoch zu gering. Um beispielsweise flexible Konfigurationslogiken für komplexe Produkte zu unterstützen ist ein merkmalsbasierter Ansatz für die Teilverwaltung unerlässlich. Mit diesem Ansatz lassen sich leistungsfähige hierarchisch strukturierte und vernetzte Klassensysteme aufbauen.

### **2.3.1 Ordnungsschemata und Klassifizierungssysteme**

Ordnungsschemata dienen zum Sammeln, Systematisieren und geordneten Darstellen von Informationen. Einen Überblick über ihre Möglichkeiten gibt [Drei-75]. Ein Ordnungsschema besteht üblicherweise aus Zeilen und Spalten denen ordnende Parameter zugeordnet werden. Ihre Einsatzmöglichkeiten sind vielfältig. Im Konstruktionsprozess können sie dazu verwendet werden, um Lösungen in Form von Konstruktionskatalogen strukturiert bereitzustellen [Rot-04] oder neue Lösungen durch Kombination vorhandener Elemente zu generieren. Im letzteren Fall spricht man von einem „morphologischen Kasten“ [Pah-97]. Ordnungsschemata bieten lediglich ein Werkzeug zur Strukturierung einer Informationsmenge. Allgemeingültige, ordnende Kriterien werden nicht vorgeschlagen. Sie müssen jeweils vom Anwender selbst festgelegt werden. Die Beschreibungstiefe der enthaltenen Informationen ist sehr gering. Sie

reduziert sich auf die vorhandenen Spalten- und Zeilenparameter. Ordnungsschemata eignen sich daher nicht zum Datenaustausch entlang des Produktlebenszyklus.

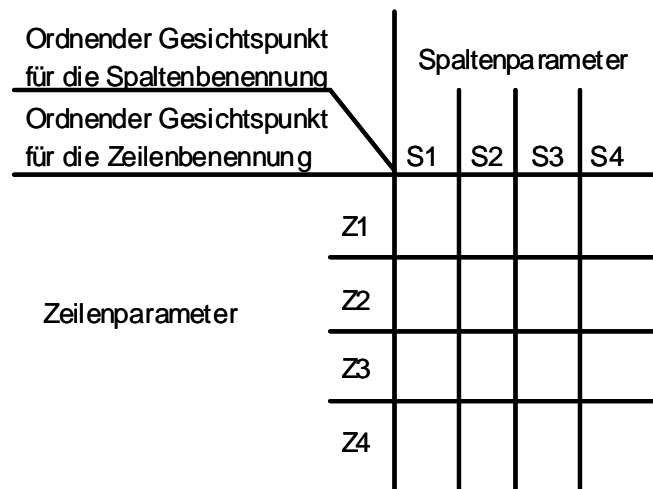


Bild 2-1: Ordnungsschema

Klassifizierungssysteme dienen dazu, Sachen und Sachverhalte nach bestimmten Gesichtspunkten zu ordnen und gleiche oder ähnliche Sachen zusammenzuführen. Sie werden zur Grobstrukturierung unterschiedlicher Gruppen von Objekten wie beispielsweise Halbzeugen, Vorrichtungen, Baugruppen, Bauteilen oder Produkten eingesetzt [Tho-90]. Charakterisiert werden Klassifizierungssysteme durch die beschriebenen Objekteigenschaften, bzw. das Unterscheidungskriterium zwischen den Klassen des Systems (den Diskriminator), die Art des Codes (numerisch, verbal) und dessen Aufbau (hierarchisch, unabhängig, gemischt). Eine gezielte Auswahl von Objekten ist lediglich hinsichtlich eines oder weniger bestimmter Kriterien möglich. Bauteile lassen sich in einem Klassifizierungssystem sammeln und ordnen, jedoch nicht vollständig beschreiben. Das Ziel von Klassifizierungssystemen ist in erster Linie die Identifikation und Strukturierung, für den Datenaustausch entlang des Produktlebenszyklus sind sie nicht geeignet.

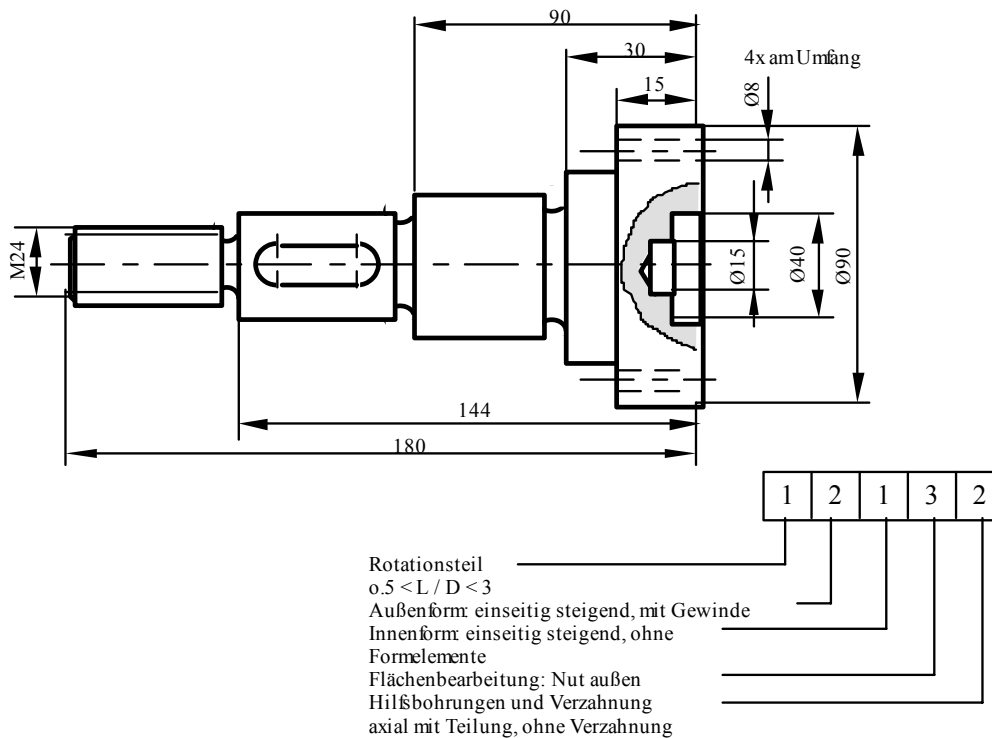


Bild 2-2: Verschlüsselung eines Rotationsteils nach Opitz

### 2.3.2 Sachmerkmalleisten

Sachmerkmal-Leisten (SML) dienen dem Zusammenfassen, Abgrenzen und Auswählen von genormten und nicht genormten, materiellen und nicht materiellen Gegenständen, die einander ähnlich sind. Sie ermöglichen das Dokumentieren, Speichern und Austauschen der Daten von Gegenständen mit Hilfe von informationstechnischen Verfahren [3]. Gestaltung und Hinweise zur Anwendung von Sachmerkmalleisten sind durch die DIN 4000 vorgegeben. Der Begriff des Sachmerkmals weist darauf hin, dass durch solche Merkmale (z.B. technische Eigenschaften) Gegenstände unabhängig von ihrem Umfeld (z.B. Verwendungszweck) beschrieben werden. Die Merkmalsausprägungen sind bei den SML gegenüber den Klassifizierungs-codes unverschlüsselt und die Merkmalsmenge wird den zu beschreibenden Teilen angepasst. Die Anzahl der festzulegenden Merkmale muss im Hinblick auf ein schnelles Auswerten eines Sachmerkmal-Verzeichnisses eingeschränkt werden und so groß sein, dass ein Bauteil gegenüber anderen Bauteilen ausreichend abgegrenzt werden kann.

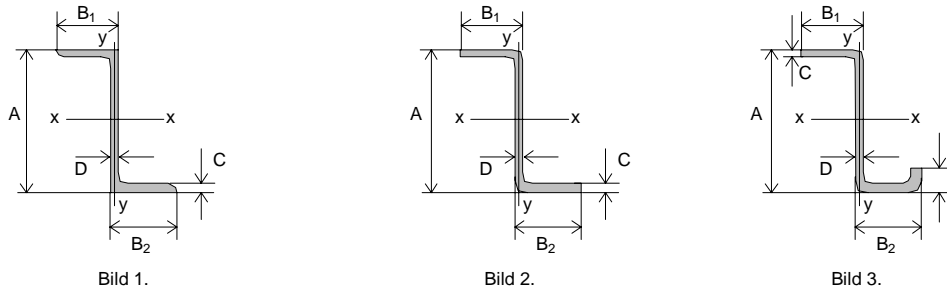


Bild 1.

Bild 2.

Bild 3.

1 von 1	Sachmerkmal-Leiste DIN 4000 - ... Profile									
Kennbuchstabe	BLD	A	B	C	D	E	F	G	H	J
Sachmerkmal-Benennung	Bildkennung	Höhe	Flanschbreite $B_1, B_2$	Flanschdicke	Stegdicke		Bördel- oder Abkantlänge	Widerstandsmoment $W_x, W_y$	Werkstoff	Oberfläche und/oder Schutzart
Referenzhinweis										
Einheit	-	mm	mm	mm	mm	-	mm	cm <sup>3</sup>	-	-

Bild 2-3: Sachmerkmal-Leiste nach DIN 4000

Die Beschreibungstiefe einer SML ist wesentlich größer als bei den zuvor dargestellten Verfahren, jedoch keinesfalls vollständig. Neben geometrischen Merkmalen können auch andere Eigenschaften dokumentiert werden. Ein Nachteil ist, dass keine hierarchische Struktur aufgebaut werden kann, innerhalb der auf jeder Ebene Merkmale vorhanden sind, die ausgewertet und nach unten weitervererbt werden können [Ond-98].

### 2.3.3 Klassensysteme

Ein Klassensystem vereinigt die Vorteile der hierarchischen Strukturierung eines Klassifizierungssystems mit denen der merkmalsorientierten Beschreibung einer Sachmerkmal-Leiste. Ähnliche Objekte mit gleichen Merkmalen, die sich nur in ihren Merkmalsausprägungen unterscheiden, werden in Klassen zusammengefasst. Die Klassen werden mit Merkmalen beschrieben und hierarchisch strukturiert. Innerhalb der Hierarchie trägt die Beziehung zwischen einer Ober- und Unterklasse die Semantik „ist\_ein“ („is\_a“). Jede Klasse schafft einen Definitionsraum für Merkmale, mit denen die zugehörigen Objekte beschrieben werden, d.h. die Semantik eines Merkmals ist innerhalb einer Klasse und allen Unterklassen für alle Objekte gleich. Dies ist bei Sachmerkmal-Leisten nicht immer der Fall. Ein Klassensystem folgt dem Paradigma der Objektorientierung, d.h. alle Merkmale einer übergeordneten Klasse werden an nachfolgende Klassen vererbt. Klassen, denen wiederum Unterklassen zugeordnet sind, werden als generische Klassen bezeichnet. Diesen generischen Klassen können generische Objekte bzw. Teile angehören. Dabei handelt es sich nicht um real existierende Bauteile oder Produkte, sondern um eine Abstraktion derselben. Im Sinne einer methodischen Vorgehensweise beim Konstruieren sind generische Teile beispielsweise dann nützlich, wenn der

Konstrukteur in einer frühen Phase des Konstruierens festlegt, an einer bestimmten Stelle ein Rotativlager zu verwenden, der genaue Typ aber noch unbestimmt ist. Er kann dann ein generisches Teil „Rotativlager“ einsetzen und später dafür eine konkrete Ausprägung eines bestimmten Typs von Lager auswählen (vgl. Bild 2-4).

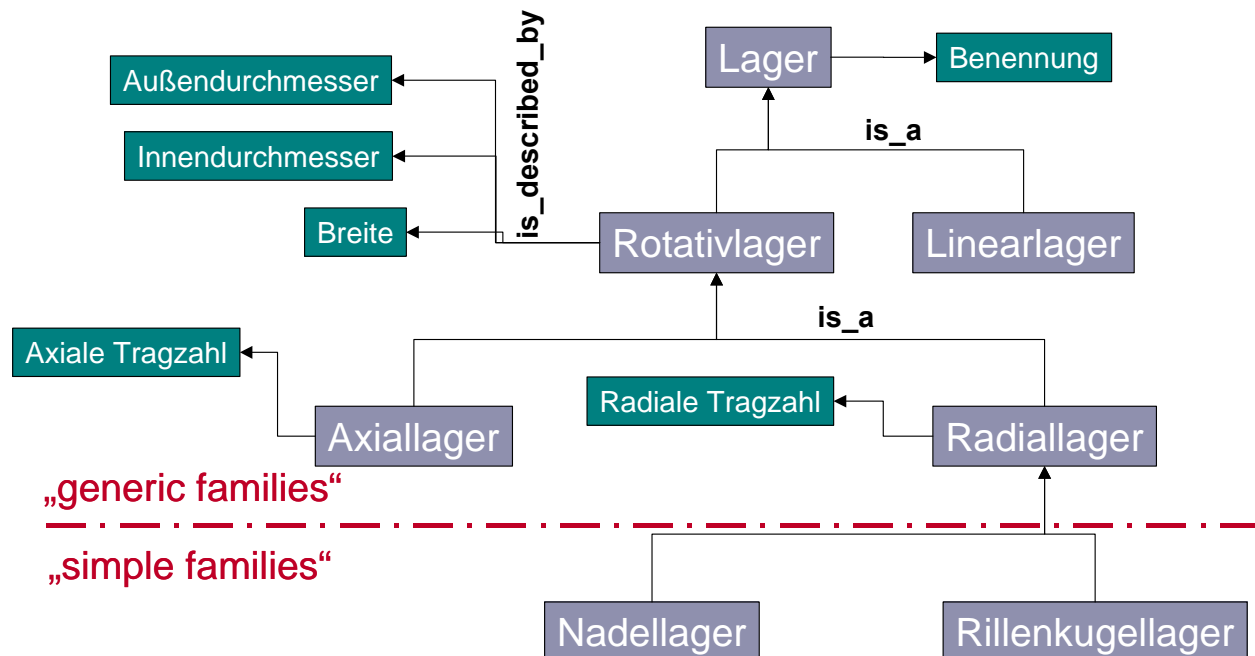


Bild 2-4: Beispiel eines Klassensystems

Die Verwendung eines Klassensystems zur Datenverwaltung in der Produktentwicklung und entlang des gesamten Produktlebenszyklus bietet verschiedene Vorteile. Die hierarchische Struktur sorgt für eindeutige Beziehungen zwischen den Klassen, bietet somit eine bessere Übersicht und schafft Ordnung im Datenbestand. Dies führt auch zu einer schnelleren und methodischen Suche und ermöglicht die Merkmalsvererbung. Vererbung sorgt für eine einheitliche Merkmalsbenennung innerhalb der Hierarchie. Sie spart Zeit bei der Merkmalsdefinition, da nicht für jede Klasse sämtliche Merkmale neu erfasst werden müssen und verhindert Redundanzen. Die Möglichkeit zur merkmalsorientierte Suche auf höheren Hierarchieebenen stellt einen weiteren Vorteil der Merkmalsvererbung dar.

Der grobe Aufbau eines Klassensystems sollte sich an den Konstruktions- bzw. Lebenszyklusphasen orientieren (vgl. Bild 2-5). Die Klassen im oberen Bereich der Hierarchie werden in erster Linie durch organisatorische Merkmale beschrieben, welche insbesondere in der Planungsphase relevant sind. Daran schließen sich Klassen an, die für die Konzeptionsphase wichtig sind. Typische Merkmale in diesem Bereich beschreiben Funktionen, Effekte oder Wirkprinzipien. Der nachfolgende Bereich enthält Klassen deren Merkmale die Grobgeometrie widerspiegeln und in der Entwurfsphase von Bedeutung sind. Der untere Bereich ist durch Klassen geprägt, die Merkmale zur Beschreibung der Feingeometrie zur Verfügung stellen. Durch die Merkmalsvererbung innerhalb der Hierarchie wird gewährleistet, dass die unteren Klassen eine vollständige Beschreibung ihrer Objekte liefern.

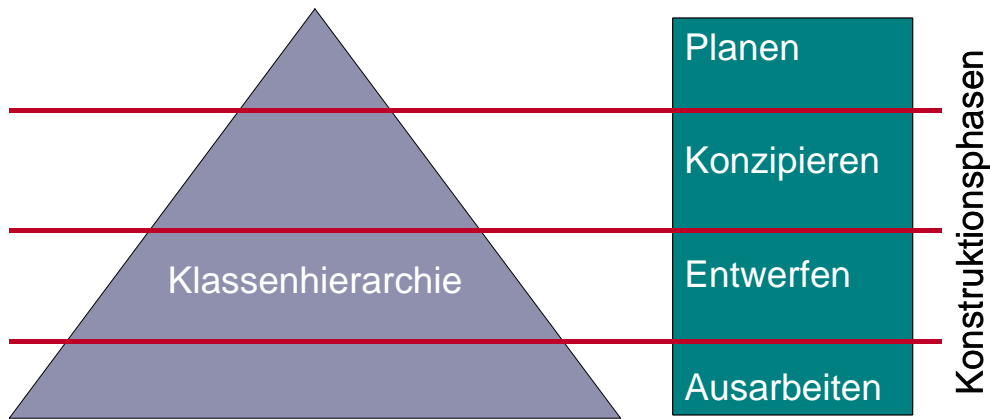


Bild 2-5: Aufbau eines Klassensystems

Eine Klassenhierarchie stellt durch die Art und Weise ihrer Strukturierung stets eine gewisse Sichtweise auf die enthaltenen Objekte dar. Verschiedene Unternehmensbereiche nehmen die Aufteilung der Klassen nach unterschiedlichen Gesichtspunkten (Diskriminatoren) vor. Für den Fertigungstechniker ist entscheidend, ob ein Teil spanlos oder spanend hergestellt und auf welcher Maschine es gefertigt wird. Für den Konstrukteur ist dies weniger wichtig. Er klassifiziert Teile vorrangig nach ihrer Funktion, während der Einkauf eher Fremdbezugs- oder Eigenfertigungsteile unterscheidet. Eine ähnliche Problematik ergibt sich auch beim Vergleich der Klassenhierarchien unterschiedlicher Hersteller. Der Aufbau der Hierarchie, sowie die Bezeichnung der Klassen und Merkmale ist für gleiche Produkte oft stark unterschiedlich. Eine Referenzhierarchie erlaubt es, verschiedene Klassensysteme aufeinander abzubilden und individuelle Strukturen für verschiedene Anwendungsgebiete zu realisieren [ISO-95].

Bild 2-6 zeigt die Situation am Beispiel von Elektromotoren. Kurzzeichen und Bauformen für umlaufende elektrische Maschinen sind in der DIN 34-7 genormt. Die Klassifizierung der Norm kann für Elektromotoren als Referenzhierarchie verwendet werden. Zulieferer klassifizieren ihr Produktspektrum jedoch in der Regel spezifisch. So findet sich der Elektromotor mit der Normbezeichnung IM 1001 bei Zulieferer A unter Antriebstechnik – Motoren – Reihe A, während Zulieferer B sich für die Einteilung Elektrotechnik – Antriebe – Bauform 2 entschieden hat. Um nun die Produkte von Zulieferer A und B vergleichbar zu machen, sind in der Bibliothek Referenzen vom Typ „is\_case\_of“ verwendet worden.



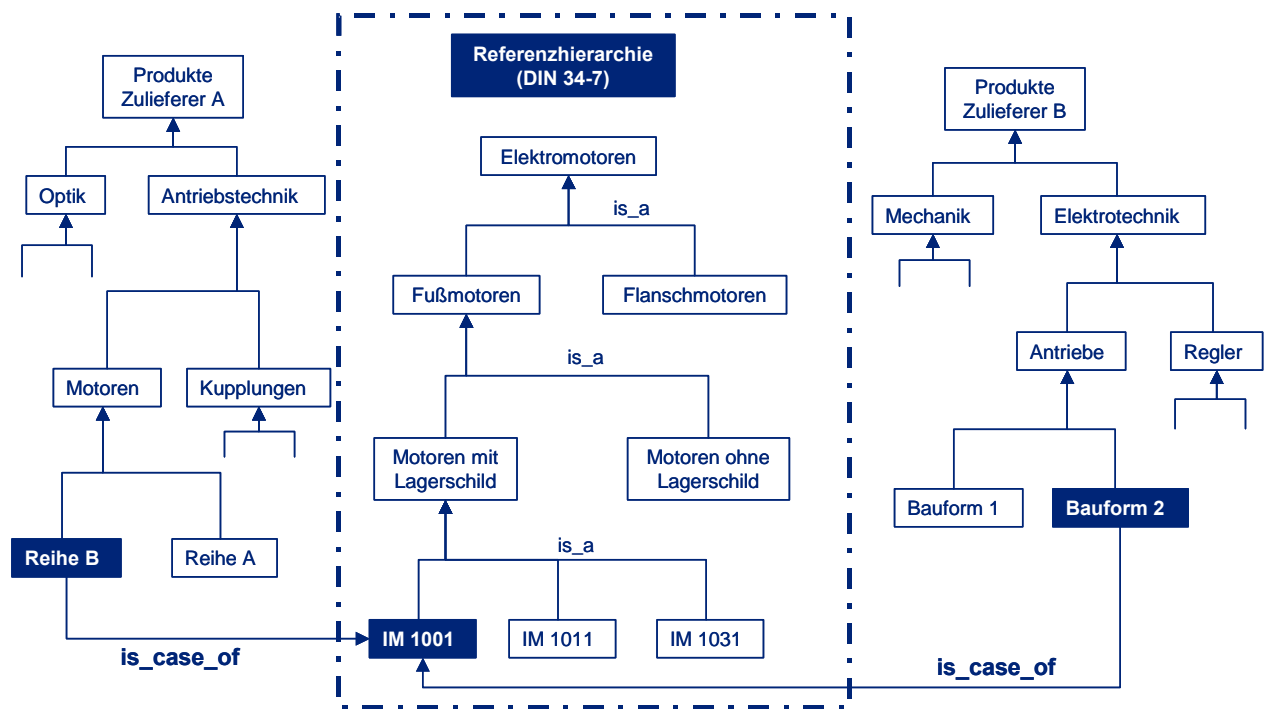


Bild 2-6: Konzept der Referenzhierarchie

Bild 2-6 zeigt lediglich Klassen. Die Problematik tritt allerdings auch bei Merkmalen auf. Merkmale zur Beschreibung von Klassen werden in der Regel zuliefererspezifisch bezeichnet. Ihre Bezeichnung weicht oft von Normen ab. Das Konzept der Referenzhierarchie kann auch beim Vergleich der Merkmale von Produkten verschiedener Zulieferer angewandt werden. Die Referenzhierarchie definiert die Merkmale einer Klasse mit einer neutralen Benennung und jeder Zulieferer kann über eine Relation von seiner Merkmalsdefinition aus auf die Merkmale der Referenzhierarchie verweisen.

## 2.4 Strukturierungsregeln für Klassensysteme

Das vorhergehende Kapitel hat gezeigt, dass Klassensysteme einen effizienten Ansatz zur Teileverwaltung darstellen, mit dessen Hilfe vielfältige Probleme gelöst werden können. Die Erstellung einer Klassenhierarchie ist jedoch ein komplexer Prozess, der tiefgreifendes Know How über die zu klassifizierenden Objekte und die Klassifizierung im Allgemeinen erfordert. Als Hilfestellung zur Erzeugung wohlstrukturierter Klassensysteme, definiert die ISO 13584 PLIB acht Regeln zur Strukturierung von Klassen, die dem Anwender eine schnelle Validierung des eigenen Ergebnisses erlauben [ISOa-97]. Diese Regeln befassen sich mit den folgenden Themen:

- Regel 1: Anwendungsgebiet – Möglichkeit der Beschreibung aller relevanten Teile
- Regel 2: Höhere Klassifizierungsebenen – Festlegung durch ICS (international classification of standards); Definitionsräume für Merkmale und Klassen

- Regel 3: Untere Klassifizierungsebenen – Definition wenn von Anwendungsprogramm unterstützt und benötigt
- Regel 4: Homogenität einzelner Klassen – Ähnlichkeit der Teile; Instanzierbarkeit von Merkmalen für jedes Teil
- Regel 5: Variation der Perspektive auf Klassenhierarchien – Sichten
- Regel 6: Geeignete Merkmale für Klassen – Auswahl von Teilen oder Klassen
- Regel 7: Semantische Identität von Merkmalen – Vererbung nur bei semantische Identität
- Regel 8: Instanzierbarkeit von vererbten Merkmalen – Unterscheidung visible/applicable

Die nachfolgenden Kapitel beschreiben die Regeln und deren Anwendung genauer.

#### 2.4.1 Regel 1: Anwendungsgebiet

Eine Klassenhierarchie oder standardisierte Referenzhierarchie zur Teileidentifikation soll alle Teile die in ISO oder IEC Standards vorkommen beinhalten. Eine Zuliefererhierarchie soll alle Produkte des Zulieferers beinhalten [ISOa-97].

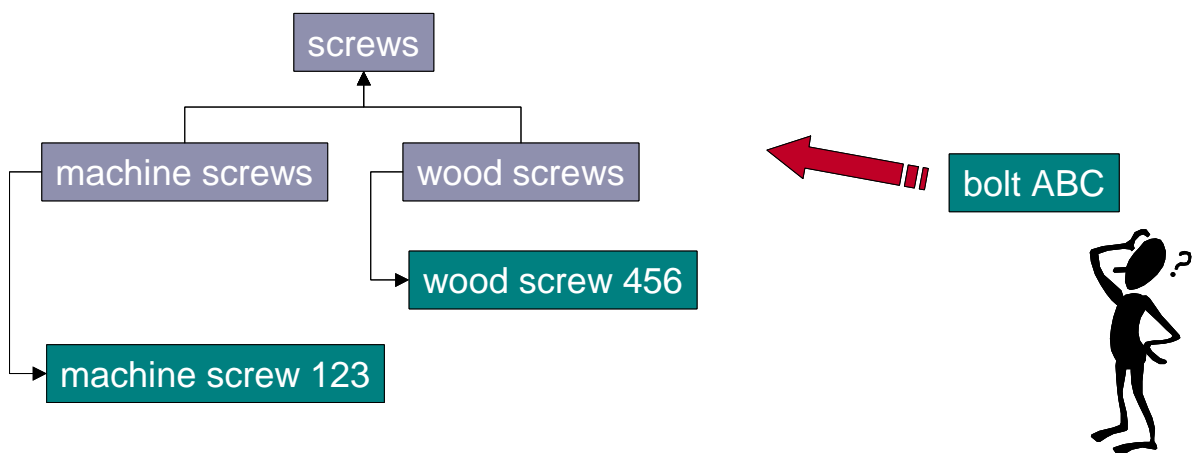


Bild 2-7: Unvollständiges Anwendungsgebiet

#### 2.4.2 Regel 2: Höhere Klassifizierungsebenen

Der höhere Teil der standardisierten Identifikationshierarchie (Klassenhierarchie) soll auf der internationalen Klassifikation von Standards (ICS) basieren.

Merkmale können den Klassen der höheren Klassifizierungsebenen gemäß der Fakturierungsregel (Regel 8) zugeordnet werden [ISOa-97].

### 2.4.3 Regel 3: Untere Klassifizierungsebenen (Instanziierungsregel)

Unterhalb der höheren Hierarchieebenen soll eine generische Klasse (Klasse, die kein Blatt ist) nur dann erzeugt werden, wenn es möglich und zweckmäßig ist, ihr ein funktionales Modell (z.B. geometrisches Modell, semantisches Modell) zuzuordnen, beispielsweise wenn ein Benutzer aus der Klasse ein generisches Bauteil auswählen könnte um einen bestimmten Zustand (Phase) seines Designprozesses zu unterstützen [ISOa-97].

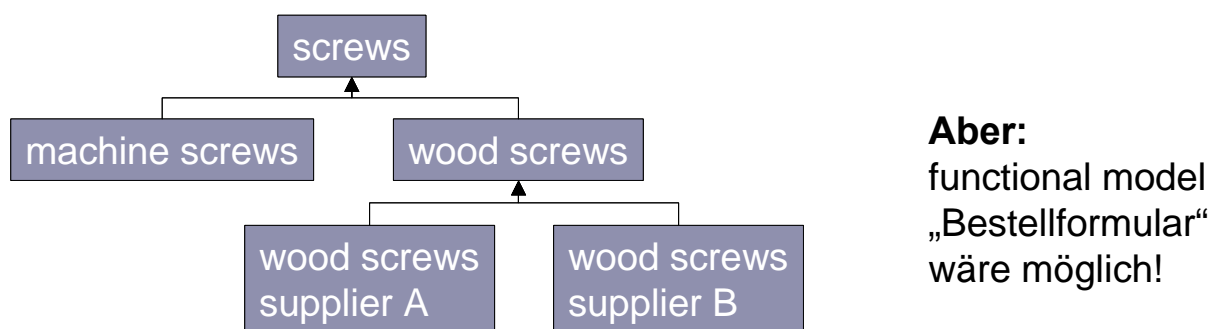


Bild 2-8: Überflüssige Klassen

Es ist beispielsweise widersprüchlich zur Instanziierungsregel Schrauben für Metall, Holz und Blech in eine Klasse „Schrauben“ zu gruppieren. Während des Design-Prozesses würde ein Designer nie eine generische Schraube auswählen ohne zu wissen ob diese Schraube später eine Maschinenschraube oder eine Holzschraube sein soll. Folglich ist es notwendig unter dem oberen Abschnitt „Bolzen/Schrauben/Stiftschrauben“ unterschiedliche generische Klassen für die verschiedenen Schraubentypen zu definieren [ISOa-97].

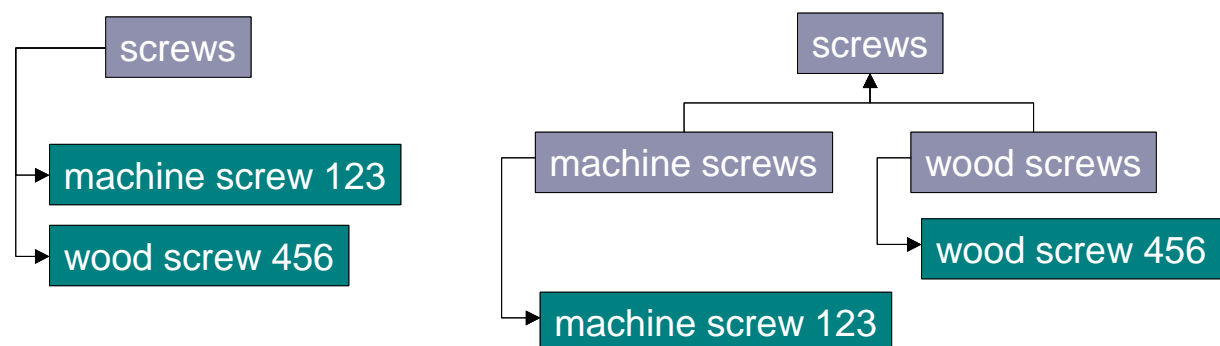


Bild 2-9: Fehlende Klassen

#### 2.4.4 Regel 4: Homogenität einzelner Klassen

Manchmal werden in Papierdokumenten, Standards oder Katalogen Mengen von Bauteilen auch dann gruppiert, wenn sich die Menge der Merkmale, die auf jedes Bauteil angewendet werden kann, unterscheidet. Das sollte für eine einfache Klasse (Blattklasse) verboten sein.

Jedes Merkmal, das einer einfachen Klasse (Blattklasse) zugeordnet ist, soll auf jedes Bauteil aus dieser Klasse anwendbar sein (einen Wert besitzen) [ISOa-97].

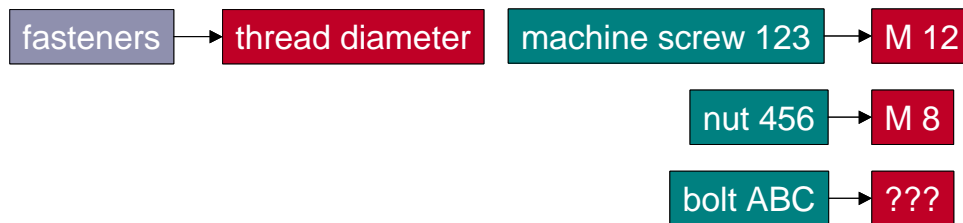


Bild 2-10: Nicht anwendbares Merkmal

#### 2.4.5 Regel 5: Variation der Perspektive auf Klassenhierarchien

Es kann auftreten, dass an einem bestimmten Punkt in der Hierarchie verschiedene Merkmale einer Klasse genutzt werden können, um eine Substruktur für diese Hierarchie zu definieren.

##### 2.4.5.1 Regel 5a: Maximale Anwendbarkeit

Wenn die Instanzierungsregel (Regel 3) in Abhängigkeit von der Sichtweise zu verschiedenen möglichen Strukturen in einer Hierarchie führt, sollte die Struktur gewählt werden, welche die maximale Anwendbarkeit der vererbten Merkmale sicherstellt [ISOa-97].

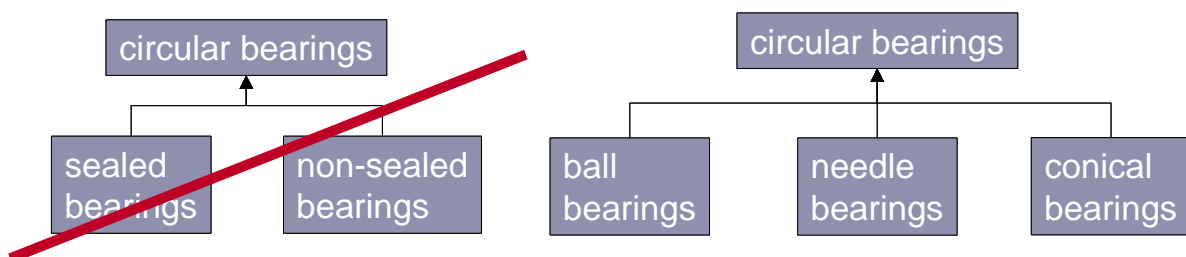


Bild 2-11: Maximierung der Anwendbarkeit vererbter Merkmale

##### 2.4.5.2 Regel 5b: Klassenmerkmale

Die anderen Sichtweisen, die als wichtig für z.B. die Suche betrachtet werden, und die gemäß Regel 5a nicht zur Strukturierung gewählt wurden, sollen durch Klassenmerkmale abgebildet

werden. Diese sollen vom Benutzer zur Suche entsprechend der anderen Sichtweise benutzt werden [ISOa-97].

### 2.4.5.3 Regel 5c: Ausprägung der Klassenmerkmale

Die Merkmalsausprägung eines Klassenmerkmals sollte auf Ebene der Klasse zugewiesen werden, bei der diese Eigenschaft auf jede einfache Klasse des korrespondierenden Unterbaums anwendbar ist, nicht jedoch auf die unmittelbare Vorgängerklasse [ISOa-97].

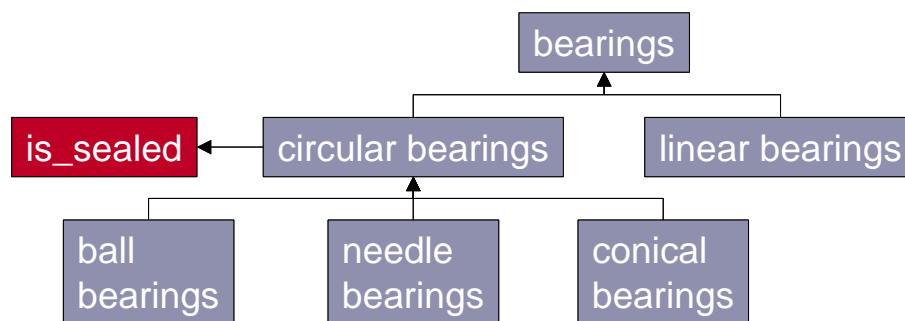


Bild 2-12: Klassenmerkmale

Die Klasse Radiallager kann z.B. sowohl in Kugel-, Nadel- und Kegelrollenlager, als auch in abgedichtete und nicht abgedichtete Lager aufgeteilt werden. Gemäß Regel 5a sollte die erste Sichtweise zum Aufbau einer Hierarchie verwendet werden. Ein boolesches Klassenmerkmal „abgedichtet“ kann dazu dienen, die andere Sichtweise abzubilden (Regel 5b). Dieses Merkmal soll auf der Ebene der Klasse Radiallager definiert werden, falls es z.B. für Linearlager nicht anwendbar ist. Dieses Merkmal soll auf der Ebene der Klasse Lager definiert werden falls es für alle Lager anwendbar ist (Regel 5c).

### 2.4.6 Regel 6: Geeignete Merkmale für Klassen

Zumindest diejenigen Merkmale, die eine generische Klasse beschreiben und zur Suche nach Teilen in allen Unterklassen der generischen Klasse verwendet werden können, sollen der standardisierten Identifikationshierarchie bzw. der Klassenhierarchie zugeordnet werden. Diejenigen Merkmale, die selten oder gar nicht zur Suche verwendet werden können nach Bedarf zugeordnet werden [ISOa-97].

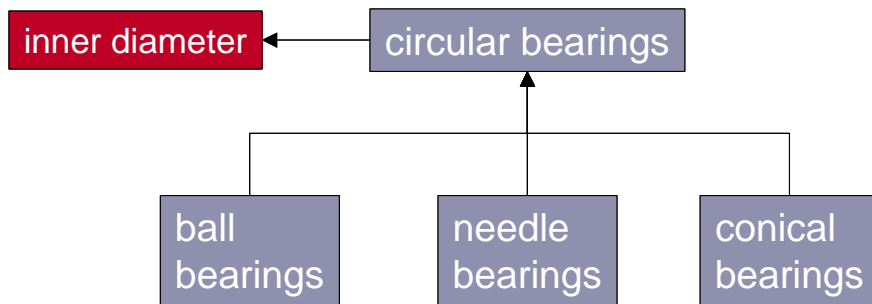


Bild 2-13: Wahl geeigneter Merkmale

#### 2.4.7 Regel 7: Semantische Identität von Merkmalen

Die folgende Regel definiert zwei Kriterien für die Entscheidung über die semantische Identität von Merkmalen.

Zwei Merkmale zweier verschiedener Klassen sollen zu einer höheren Ebene in der Hierarchie fakturiert werden dann und nur dann, wenn sie semantisch identisch sind. Die semantische Identität kann mit den nachfolgenden Kriterien ermittelt werden:

- Austauschbarkeitskriterium: Zwei Instanzen (Teile) der Klassen können ausgetauscht (alternativ verwendet) werden. Wenn sie ausgetauscht werden, dann haben die beiden betrachteten Merkmale denselben Wert.
- Homogenitätskriterium innerhalb von Prozessen: Die beiden Merkmale spielen innerhalb bestimmter Prozesse die gleiche Rolle.

Andernfalls sollen zwei verschiedene Merkmale definiert werden [ISOa-97]. Beispiele für die Anwendung der Regel::

- Beispiel 1: Das Merkmal „Gewindedurchmesser“ für Sechskantschrauben und Zylinderkopfschrauben erfüllt das Austauschbarkeitskriterium.

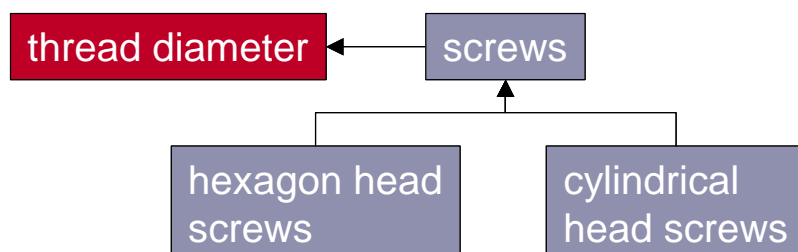


Bild 2-14: Erfüllung des Austauschbarkeitskriteriums

- Beispiel 2: Das Merkmal „Außendurchmesser“ für elektronische Bauteile erfüllt das Homogenitätskriterium im Hinblick auf die automatisierte Handhabung der Bauteile durch einen Greifer im Montageprozess.
- Beispiel 3: Das Merkmal „Gewindedurchmesser“ von Holzschrauben und Maschinenschrauben kann nicht fakturiert werden, da es keines der beiden Kriterien erfüllt.

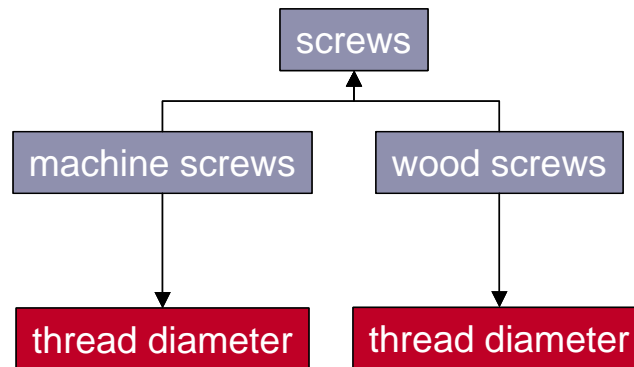


Bild 2-15: Nicht fakturierbares Merkmal

#### 2.4.8 Regel 8: Instanzierbarkeit von vererbten Merkmalen

Es ist nicht immer möglich eine Hierarchie zu definieren, in der:

- Jedes Merkmal das in verschiedenen Unterfamilien dieselbe Semantik trägt fakturiert (nur einmal definiert und dann vererbt) wird
- Jedes Merkmal das auf der Ebene einer generischen Familie definiert ist für jede Unterfamilie anwendbar ist (für jedes Teil einer Unterfamilie einen Wert hat)

Die Definition einer dazwischen liegenden Familie zum Zweck der Fakturierung eines Merkmals könnte durch die Instanzierungsregel (Regel 3) verhindert werden. Die folgende Regel gibt Hinweise zur systematischen Fakturierung von semantisch identischen Merkmalen unter Beibehaltung der Instanzierungsregel.

Zwei Merkmale zweier verschiedener Klassen, die gemäß Regel 7 semantisch identisch sind, sollen als ein einziges Merkmal der gemeinsamen übergeordneten Klasse zugeordnet werden. Falls dieses Merkmal für einige (nicht alle!) Unterklassen nicht anwendbar ist, so soll es als sichtbares Merkmal definiert werden („visible“). Es kann dann bei den entsprechenden Unterklassen als anwendbar („applicable“) definiert werden, bei denen es einen Wert besitzt. Die Definition des Merkmals soll so eindeutig sein, dass es bei keiner Klasse Zweifel gibt, ob das entsprechende Merkmal „anwendbar“ ist oder nicht (also nur „sichtbar“). Wenn es anwendbar ist soll es keinen Zweifel geben, welche konkrete Eigenschaft eines Teils das Merkmal beschreibt. [ISOa-97]

Beispiele für die Anwendung der Regel:

- Beispiel 1: Das Merkmal „Werkstoff“ ist definiert als „Werkstoff aus dem ein Teil gefertigt ist“. Dieses Merkmal kann nicht fakturiert werden, da die Definition dieses Merkmals beispielsweise für einen beschichteten Bohrer nicht eindeutig ist.
- Beispiel 2: Das Merkmal „einziger Werkstoff“ ist definiert als „Merkmal das nur für Teile anwendbar ist, die aus einem einzigen Werkstoff gefertigt sind. Die Ausprägung dieses Merkmals ist der eindeutige Code des Werkstoffs“. Dieses Merkmal kann gemäß Regel 7 fakturiert werden.

#### **2.4.9 Beachtung der ist\_ein Semantik**

Bei der Bewertung einer Klassenhierarchie muss die „ist\_ein“ Relation zwingend auf ihre Gültigkeit überprüft werden. Warum das so ist, kann leicht anhand von zwei Beispielen erläutert werden:

Die Klasse Kreis hat die Attribute Mittelpunkt  $m$  und Radius  $r_1$ . Wird nun von der Klasse Kreis durch hinzufügen des Attributs Höhe  $h$  ein Kreiskegel abgeleitet, so ist die „ist\_ein“ Semantik nicht mehr gültig. Das ist beispielsweise bei der Suche von Teilen problematisch, da in einer Klassenhierarchie niemand einen Kreiskegel unterhalb von Kreis suchen würde.

Fügt man zu den Kreisattributen  $m$  und  $r_1$  das Attribut  $r_2$  als zweite Halbachse dazu, erhält man eine Ellipse. Auch hier ist die „ist\_ein“ Semantik nicht gültig.

Fazit: Bei einem datengetriebenen bottom-up Ansatz besteht die Gefahr, dass obige Strukturen entstehen. Bei der Bewertung einer Klassenhierarchie muss die „ist\_ein“ Relation daher unbedingt auf ihre Gültigkeit hin überprüft werden.

#### **2.4.10 Klassifizierung vom Abstrakten zum Konkreten**

Die Erstellung einer Klassenhierarchie in der Konstruktion bewegt sich vom Abstrakten zum Konkreten. Begonnen wird mit der Beschreibung eines neuen Bauteils in der Planungsphase mit relativ wenigen abstrakten Merkmalen. In der Konzeption und im Entwurf bis hin zur Ausarbeitung wird die Beschreibung ständig weiter verfeinert. Das bedeutet zum Einen, dass die Beschreibung der Klassen immer konkreter wird, zum Anderen dass die Anzahl der Merkmale zunimmt je weiter der Konstruktionsprozess fortgeschritten ist.



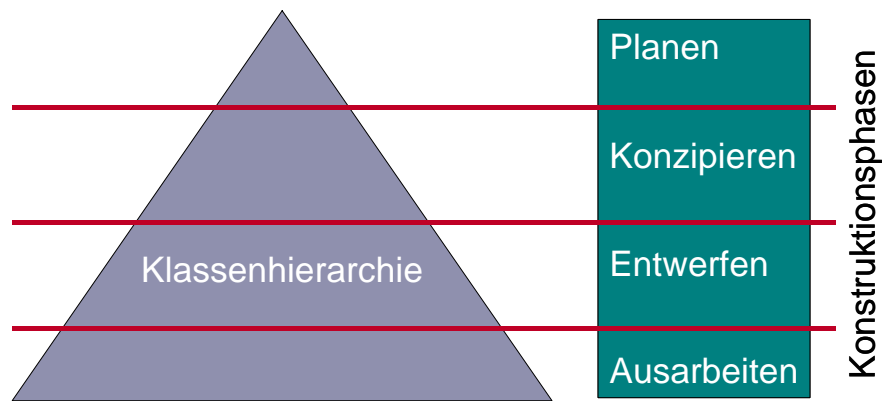


Bild 2-16: Klassifizierung gemäß der Konstruktionsphasen

### 2.4.11 Form der Klassenhierarchie

Die resultierende Klassenhierarchie sollte keine extreme Form annehmen:

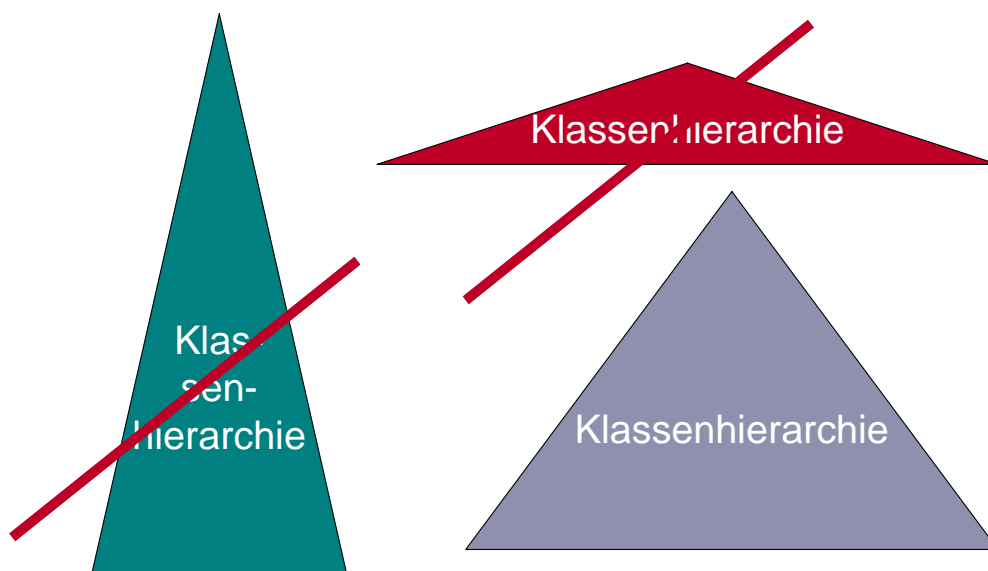


Bild 2-17: Korrekte Form der Klassenhierarchie

### 2.4.12 Sichten auf eine Klassenhierarchie

Eine Klassenhierarchie kann aus verschiedenen Sichten betrachtet werden. Eine Sicht verändert die Struktur der Klassenhierarchie aber nicht, sie stellt lediglich einen Filter dar der bestimmte Informationen ein- oder ausblendet. Würde die Struktur verändert werden wären die Vererbungsbeziehungen nicht mehr konsistent.

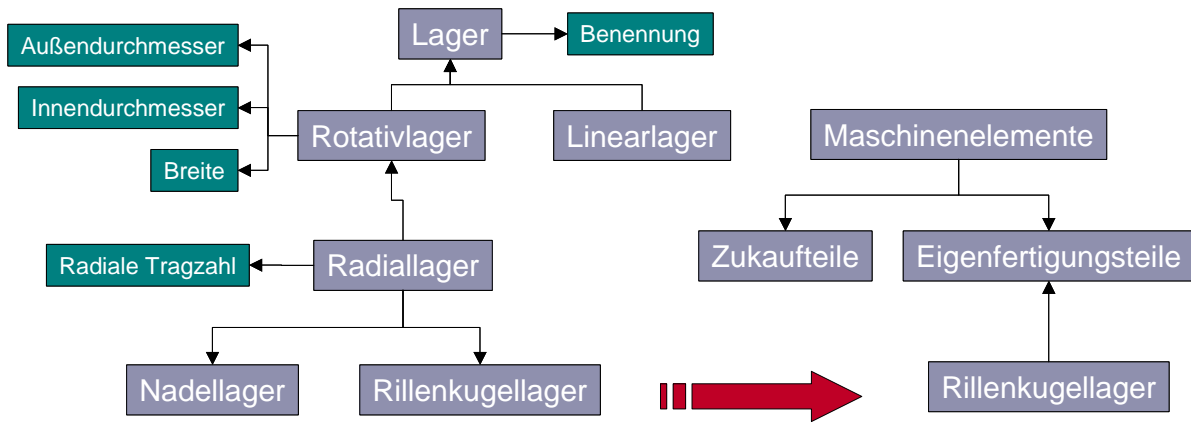


Bild 2-18: Falsche Sichtenbildung

### 2.5 Anwendungsszenarien für Klassensysteme im Produktlebenszyklus

Product Lifecycle Management stellt an die innerbetriebliche Datenverwaltung weitaus höhere Ansprüche als konventionelles PDM. Durch ein Klassensystem zur Teileverwaltung lassen sich viele dieser Ansprüche einfacher erfüllen. Anhand einiger Anwendungsszenarien wird in den folgenden Kapiteln gezeigt, wie ein Klassensystem die verschiedenen Prozesse entlang des Produktlebenszyklus unterstützen kann.

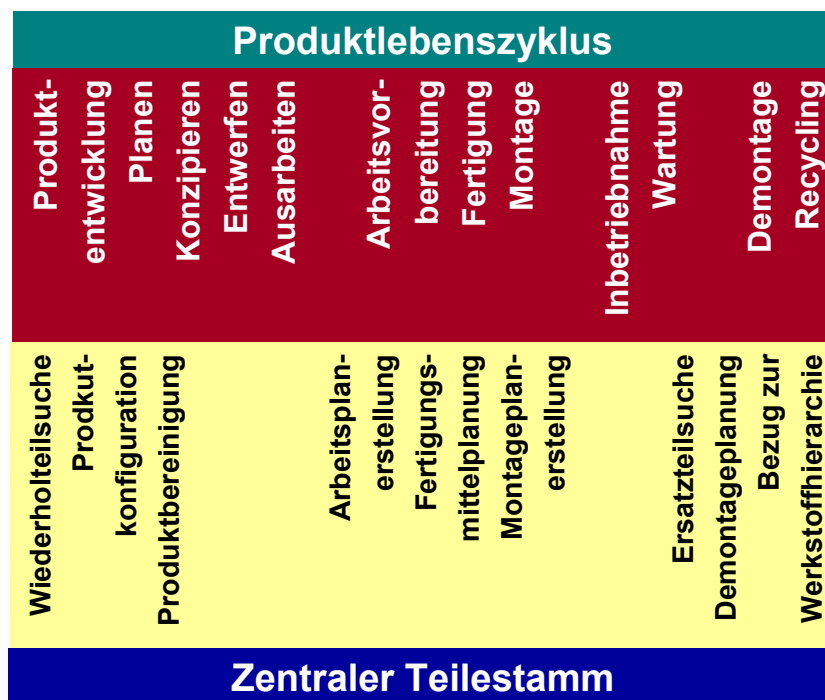


Bild 2-19: Anwendungsszenarien entlang des Produktlebenszyklus

### 2.5.1 Unterstützung der frühen Phasen des Konstruktionsprozesses

Für die Phase der Konzeption in der Produktentwicklung ist die Modellierung der Problemstellung auf der Basis unvollständiger Informationen charakteristisch. Die Lösungselemente der Konzeptphase werden in der Gestaltungsphase um Designmerkmale erweitert, sodass eine Beurteilung der Lösung bezüglich mehrerer Lebenszyklusphasen ermöglicht wird [Pah-97]. Roth erläutert in [Rot-04], dass das methodische Konstruieren zwar das Vorgehen innerhalb der Phasen erleichtert, jedoch gleichzeitig „ausgeprägte Übergänge“ zwischen den Phasen schafft, „die nur durch mehr oder weniger willkürliche Zuordnungen zu überbrücken“ sind. Eine methodische Lücke ergibt sich insbesondere beim Übergang von der Phase der Konzeption in die Phase der Gestaltung. Die frühen Phasen der Planung und Konzeption zeichnen sich hauptsächlich durch qualitative Daten aus, während in den späteren Phasen der Gestaltung und Ausarbeitung der Anteil quantitativer Daten stark zunimmt. Während der Konzeption ist es in der Regel nicht möglich, quantitative Aussagen über Kosten, Herstellbarkeit, oder Umweltgerechtigkeit zu machen, obwohl hier mit relativ geringen Änderungen weitgehende Verbesserungen hinsichtlich dieser Aspekte möglich wären. Zur Lösung dieser Problematik wird eine Erweiterung des Datenmodells vorgeschlagen [Osh-02]. Die meisten Untersuchungen auf diesem Gebiet deuten darauf hin, dass ein integratives, phasenübergreifendes Produktmodell eine Grundvoraussetzung für PLM ist.

Die frühen Phasen des Konstruktionsprozesses beeinflussen in hohem Maß Qualität und Wirtschaftlichkeit eines Produktes. Obwohl die Basis der methodischen Konstruktion für die Konzeptphase schon vor mehr als 30 Jahren gelegt wurde, gibt es bis heute noch keine praxisgerechten rechnergestützten Werkzeuge dafür. Eine Vielzahl von Gründen wird genannt: Unkenntnis der Methodik, zu allgemein, praxisfremd, unhandlich, unterschiedliche Ansätze, unterschiedliche Semantik. Aus diesen lassen sich Ziele und Aufgaben der Forschung ableiten. Die Repräsentation von Funktionen, zusätzlich zur Geometrie in CAD-Systemen, ein standardisiertes Darstellungsschema für Funktionen, sowie gemeinsame Begriffsdefinitionen für Funktionen mechanischer Systeme [Sto-97]. Als eine der wichtigsten Aufgaben, ist die formale Beschreibung von Funktionen aus verschiedensten Gründen unerlässlich:

- Reduzierung der Vieldeutigkeit (Unklarheit) in der frühen Modellierungsphase,
- Gewährleistung der Einzigartigkeit der Begrifflichkeiten auf konzeptioneller Ebene,
- Erhöhung der Gleichförmigkeit und Vergleichbarkeit der Information in Funktionsmodellen,
- Erhöhung der Aussagekraft und Erleichterung der Kommunikation in den frühen Phasen der Produktentwicklung,
- Dekomposition von Konstruktionsaufgaben in realisierbare Unteraufgaben,
- Systematische Suche nach Analogien zur Lösung von Konstruktionsaufgaben,

- Synthetisierung von Konstruktionslösungen mittels rechnergestützter Methoden [Bei-81], [Rot-04], [Sto-97].

Seit Jahren wird versucht eine Taxonomie der Funktionen im Konstruktionsbereich aufzustellen. Die Notwendigkeit einer generischen Taxonomie für Funktionen und Flüsse zieht sich entlang der Geschichte der methodischen Konstruktion. Verschiedene Ansätze werden in den Standardwerken vorgeschlagen [Alt-96], [Pah-97]. Neuere Ansätze versuchen eine Vereinigung dieser auf Basis der Gemeinsamkeiten [Ott-00]. Zur Unterstützung des PLM ist eine lebenszyklusbegleitende Funktionsbibliothek auf Basis der ISO 13584 PLIB sinnvoll.

Zentrale Bedeutung innerhalb der PLIB hat die Wissensstrukturierung mit Hilfe von Beziehungen. Die anbieter- bzw. zuliefererübergreifende, allgemeingültige Wissensstrukturierung erfolgt mit Hilfe einer Referenzhierarchie (vgl. Bild 2-6). Sie ist Teil des „dictionaries“ einer Bibliothek und ermöglicht es, Zulieferer-Wissen für verschiedene Produktkategorien von unterschiedlichen Zulieferern zu strukturieren und einheitlich sowie recheninterpretierbar abzubilden [San-01]. In Analogie lässt sich schließen, dass die Voraussetzung für die Vergleichbarkeit von Produkten anhand von Funktionsbeschreibungen über Relationserzeugung, das Vorhandensein einer Referenzbibliothek ist (vgl. Bild 2-20). Dies eröffnet auch Zulieferern die Möglichkeit, eine funktionale Beschreibung ihrer Produkte in ihrer domänenspezifischen Sprache vorzunehmen und diese in bestehende Funktionsbibliotheken der Kunden einzubinden oder als digitale Lösungsbibliotheken anzubieten.

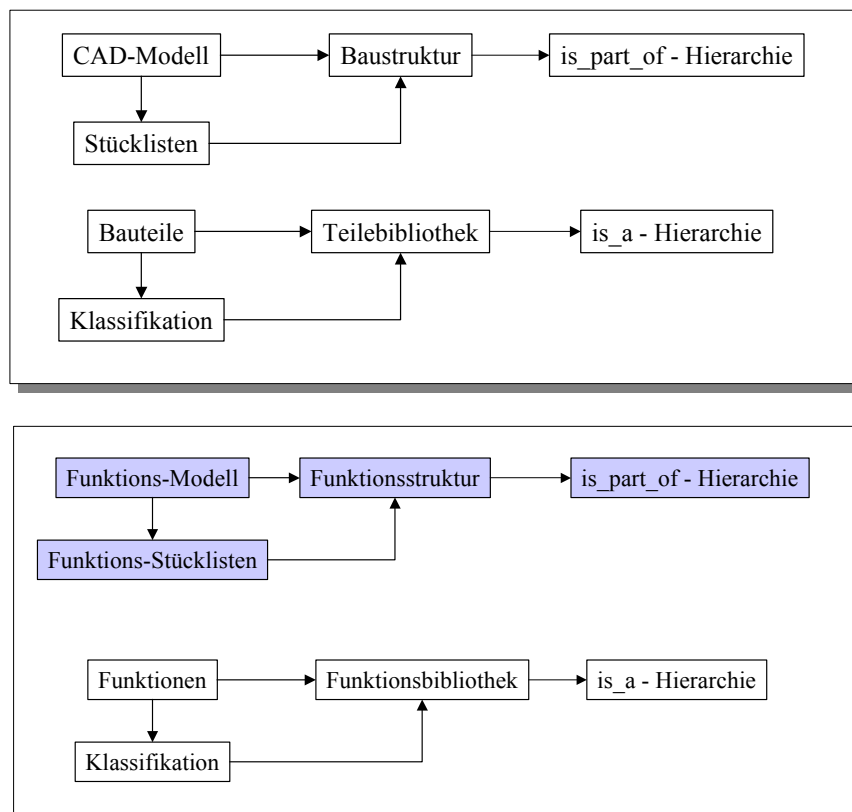


Bild 2-20: Analogie Bauteilbibliothek -- Funktionsbibliothek

Im Falle eines bereits realisierten Produktes ist dessen Struktur und vollständige Beschreibung anhand von Teilfunktionen sowie auch die Zuordnung dieser zur Baustruktur möglich. Die Verknüpfung der Funktionen zu Bauteilen, Baugruppen und komplexen Produkten, die im eigenen Unternehmen entwickelt worden sind, ist somit ohne methodische Schwierigkeiten durchführbar. Über eine Referenzhierarchie würde auch Zulieferern die Möglichkeit geboten, ihr Produktspektrum unabhängig von der firmeninternen Bezeichnung bei der funktionalen Beschreibung von Produkten einzubinden. Dies würde den „Einstieg“ der Zulieferer beim Kunden in einer frühen Phase der Produktentwicklung ermöglichen.

Aus dieser Aufgabenstellung entwickeln sich mehrere Herausforderungen:

- die Zuordnung von Bauteilen bzw. Baugruppen ist nicht immer auf einer Diskretisierungsebene der Funktionsstruktur möglich,
- die Einheitlichkeit der Beschreibung ist heute noch nicht gegeben (Referenzhierarchie notwendig) und
- Doppel- oder Mehrfachfunktionen von Funktionsträgern (Funktionsintegration) müssen gelöst werden.

Als Grundgerüst, zum Aufbau einer Referenzhierarchie können die von Stone in [Sto-97] vorgeschlagenen Grundfunktionen und Flüsse verwendet werden.

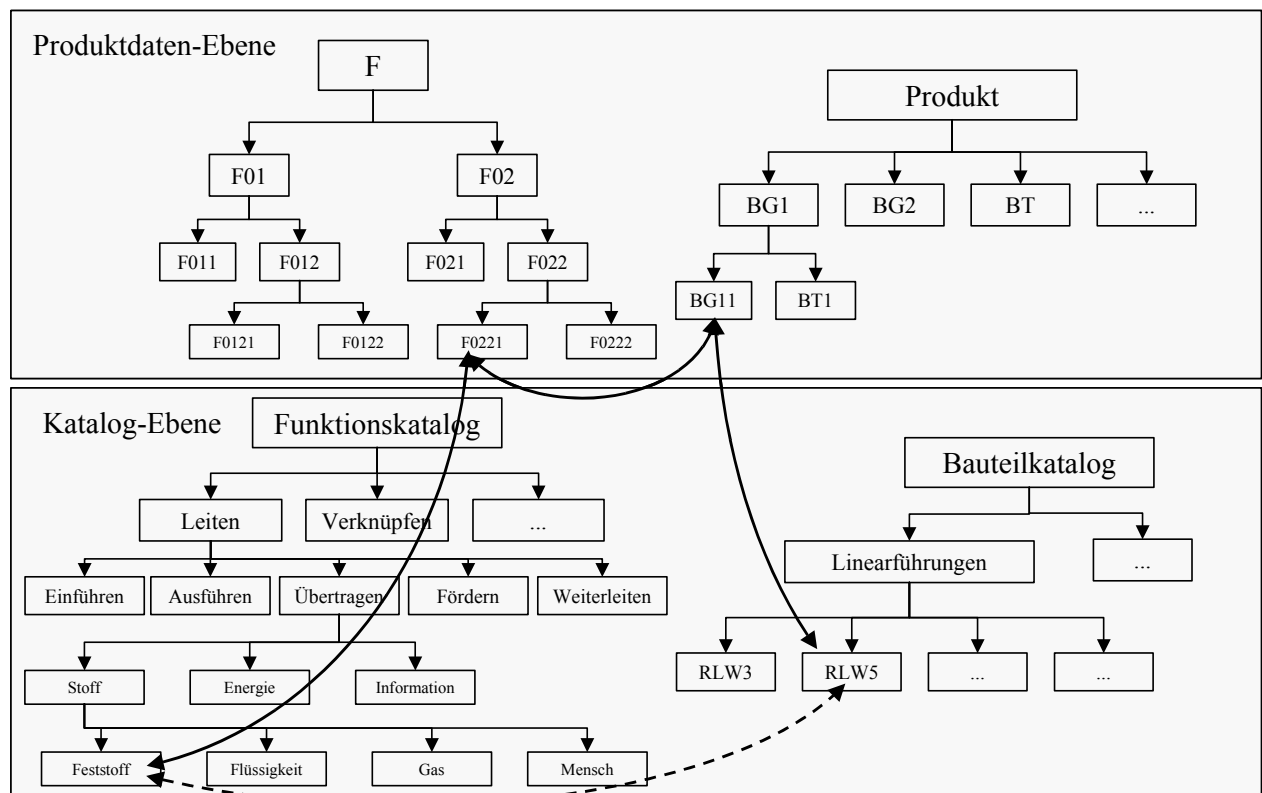


Bild 2-21: Produkt- und Katalogebene, Beziehungen

Der Einsatz einer vernetzten Funktions-Bauteilbibliothek und somit implizit der Funktionsstruktur als produktlebenszyklusbegleitendes Modell gemäß Bild 2-21 bietet mehrere Vorteile:

- bidirektionale Hilfestellung,
- vereinfachte Produktpflege und –speicherung anhand der Funktionsstruktur,
- Dekomposition von Funktionen bis zu einem Konkretisierungsgrad bei dem die Zuweisung von Bauelementen oder Baugruppen vorgenommen werden kann,
- Optimierung von Entwürfen durch Zurückführung in die frühen Phasen mittels der Relationen zwischen Bauteil bzw. Baugruppe und Funktion und
- Vergleichbarkeit der Funktionsträger anhand der Funktion ermöglicht dem Konstrukteur Alternativen zu einer bestimmten Lösung zu finden [San-01].

Einen weiteren entscheidenden Vorteil der Funktionsstruktur als lebenszyklusbegleitendes Modell stellt die Schnittstellenfunktion zwischen der physikalischen und wirtschaftlichen Ebene dar. Entscheidungen der betriebswirtschaftlichen Ebene können konsequent und effizient während des gesamten Produktlebenszyklus gefällt werden. Z.B. können Wertanalysen iterativ durchgeführt werden. Änderungen in der physikalischen Ebene wie etwa die Substitution eines Funktionsträgers durch einen Lieferantenwechsel können auf Basis der funktionalen Beschreibung und der Verknüpfung zur betriebswirtschaftlichen Ebene problemlos ausgewertet werden.

Lebenszyklusbegleitende funktionsorientierte Produktmodelle auf Basis von Funktionsklassen unterstützen das PLM aktiv:

- im Änderungs- und Konfigurationsmanagement,
- bei der Reduktion der Teileanzahl z.B. durch Funktionsintegration, größere Einkaufsvolumina, kleinere Lagerhaltungskosten,
- bei Plattformstrategien,
- bei der Identifizierung von Baukasten- und Baureihensystematiken anhand von Funktionsstrukturen,
- bei der Kontrolle der zu erreichenden Funktionalität des Produktes unter gleichzeitiger Berücksichtigung der Kosten (Wertanalyse).

### 2.5.2 Effizientere Wiederholteilsuche

Viele Teileverwaltungssysteme strukturieren die enthaltenen Informationen in einer hierarchischen Klassifizierung. Aufgrund der Menge an Daten kann diese Struktur sehr umfangreich und damit auch unübersichtlich werden. Ein weiterer Nachteil besteht darin, dass die Strukturierung immer nach einem bestimmten Gesichtspunkt (einem sog. Diskriminator) erfolgt. Der Diskriminator kann beispielsweise das Fertigungsverfahren für ein Teil sein (Einteilung in Blechbiegeteile, Schweißteile, Gussteile usw.), die Grobgeometrie (rotationssymmetrisch, eben, gebogen, kubisch) oder die Funktion (Lager, Verbindungselemente, Federn usw.). Jeder Unternehmensbereich möchte jedoch die für ihn relevanten Informationen auf eine andere Art strukturieren. Die Fertigung bevorzugt sicher die Einteilung nach Fertigungsverfahren oder Grobgeometrie, während die Konstruktion eine funktionsorientierte Strukturierung bevorzugen wird.

Bei der Suche über eine Hierarchie muss dem Suchenden die Strukturierung der Hierarchie bekannt sein, damit er zu den gewünschten Informationen gelangt. Dies ist schwierig, wenn es eine große Anzahl an Bauteilen gibt und verschiedene Arten der Strukturierung. Auch gibt es oft „Schubladen“ für Elemente, die sich nicht richtig in eine bestehende Klassifizierung einordnen lassen. Diese sind dann sehr schwer zu finden. Vor allem Anwender, die nicht ständig mit der Klassifizierung zu tun haben oder neu im Unternehmen sind, haben oft Schwierigkeiten Teile schnell und zielsicher zu finden. Daher werden oftmals Teile unnötig neu konstruiert, die Teilezahl und die damit verbundenen Kosten steigen. Ein weiterer Nachteil liegt darin, dass bei einem Suchvorgang nur jeweils ein Teil gefunden wird. Alternative Teile aus anderen Zweigen der Hierarchie bleiben unberücksichtigt, sofern kein weiterer zeitaufwändiger Suchvorgang durchgeführt wird.

Ein Klassensystem bietet die Möglichkeit sowohl über die Klassifikationsstruktur, als auch merkmalsorientiert oder über Relationen zu suchen (vgl. Bild 2-22). Bei der merkmalsorientierten Suche wird ein gesuchtes Teil auf der Ebene einer höher gelegenen Klasse mit denjenigen Merkmalen beschrieben, die es tragen soll. Ohne dass dem Anwender die Art der Klassifizierung bekannt ist, werden in einem Suchvorgang alle in Frage kommenden Teile gefunden, möglicherweise aus verschiedenen Klassen. Die Art und die Kenntnis der Klassifizierung sind damit für den Erfolg einer Suche nicht mehr ausschlaggebend. Auch weniger erfahrene Konstrukteure finden in kurzer Zeit alternative Teile aus verschiedenen Klassen.

- **Über Klassen**
- **Über Merkmale**
- **Über Relationen**
  - Innerhalb einer Hierarchie
  - Zwischen Hierarchien

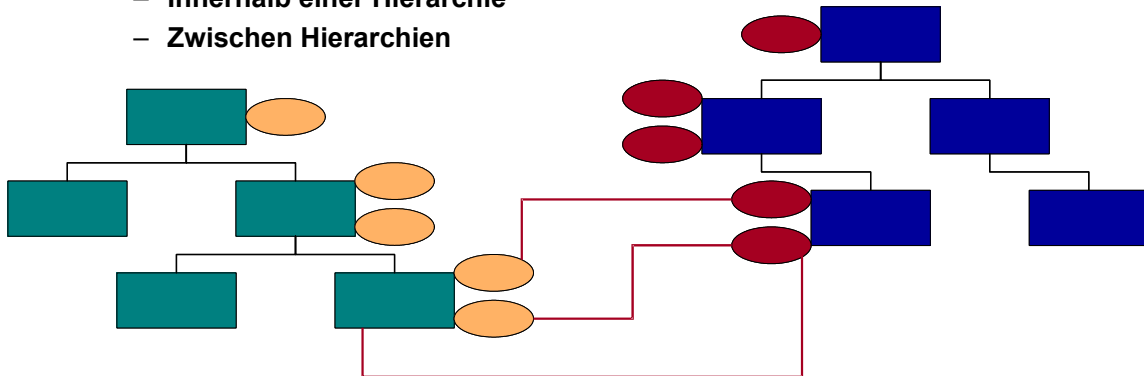


Bild 2-22: Suchmöglichkeiten in einem Klassensystem

### 2.5.3 Unterstützung der automatischen Produktkonfiguration

Die Anpassbarkeit an individuelle Kundenwünsche wird für den Markterfolg von Produkten zunehmend wichtiger. Online-Produktkataloge bieten potenziellen Kunden die Möglichkeit, die gewünschten Modifikationen selbst vorzunehmen und das Ergebnis ihrer Anpassungen zu überprüfen.

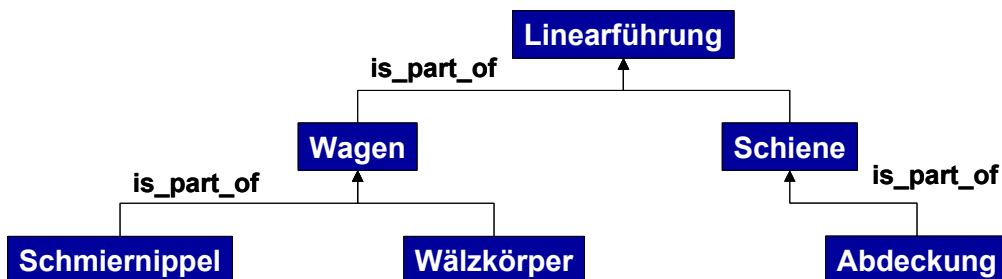
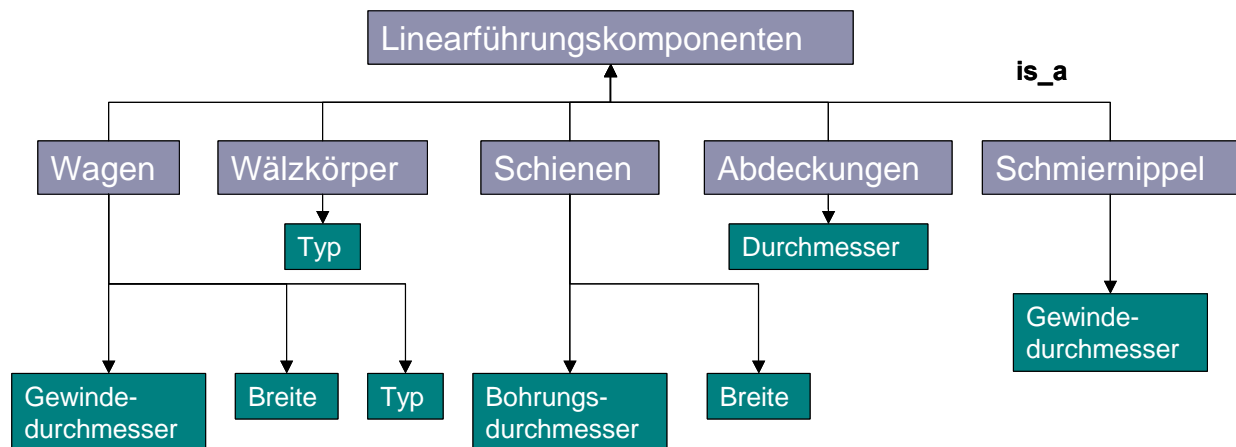


Bild 2-23: Produktkonfiguration

Dazu ist es notwendig, dass Modifikations- und Kombinationsmöglichkeiten in einem Baukasten mit Merkmalen und Regeln abbildet werden können (vgl. Bild 2-24). Dies erfordert für die Baukastenelemente (Bauteile) eine größere Beschreibungstiefe als z.B. Nummernsysteme sie bieten. Ein Klassensystem ist mit seiner merkmalsbasierten Beschreibung und den vielfältigen Relationen zwischen Klassen und ihren Merkmalen daher die ideale Grundlage für einen solchen Baukasten.





- **Wagen.Gewindedurchmesser = Schmiernippel.Gewindedurchmesser**
- **Wagen.Type = Wälzkörper.Type**
- **Wagen.Breite = Schienen.Breite**
- **Schienen.Bohrungsdurchmesser = Abdeckungen.Durchmesser**

Bild 2-24: Produktkonfiguration mit Hilfe von Merkmalen aus einem Klassensystem

#### 2.5.4 Zentraler Teilestamm

Die EDV-Landschaft in größeren Unternehmen ist zumeist historisch gewachsen. Nicht alle Standorte haben gleichzeitig den Rechner eingeführt und verwenden unter Umständen verschiedene Software-Systeme. Ähnlich verhält es sich auch mit der Klassifizierung von Teilestämmen. Ein Werk in dem gefertigt wird bevorzugt eine andere Struktur (eher fertigungsorientiert) als ein Werk bzw. ein Standort an dem in erster Linie konstruiert wird (hier wird eine funktionsorientierte Strukturierung bevorzugt).

Verschiedene Standorte eines großen Unternehmens wollen also eine eigene Klassifizierung, d.h. eigene Strukturdaten. Wenn diese nicht konsistent gehalten werden entstehen vielfältige Probleme. Teile werden mehrfach konstruiert weil Wiederholteile nicht gefunden werden. Die Losgrößen werden geringer, die Stammdaten nehmen unnötig an Umfang zu. Dadurch steigen die Kosten.

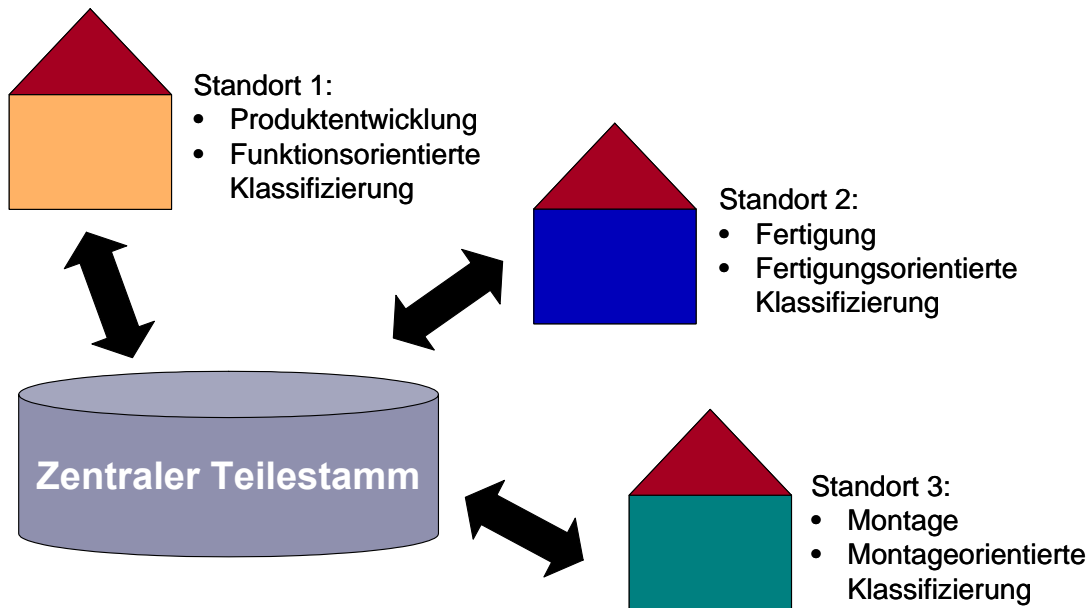


Bild 2-25: Zentraler Teilestamm

Eine Grundvoraussetzung zur Realisierung des zentralen Teilestamms sind die richtigen Beziehungen zwischen den Daten:

- **Beziehungen zwischen Hierarchien**
- **Beziehungen zwischen Merkmalen**
- **Beziehungen zwischen Verwaltungssystemen**

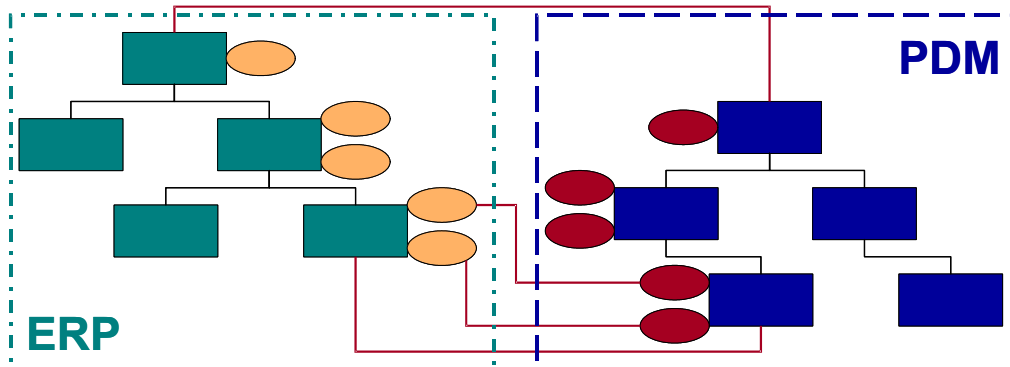


Bild 2-26: Notwendige Beziehungen

Die Vorteile des zentralen Teilestamms sind vielfältig:

- Reduzierung der Teilevielfalt
- Reduzierung der Lagerbestände
- Erhöhung der Losgrößen

- Erhöhung der Bestellmengen

### 2.5.5 Suche nach alternativen Komponenten im Service

Bei sehr langlebigen Produkten ist es möglich, dass der Produktlebenszyklus länger ist, als der Lebenszyklus bestimmter Komponenten oder Bauteile. Der Zugriff auf alternative Teile ist daher für den Service unumgänglich.

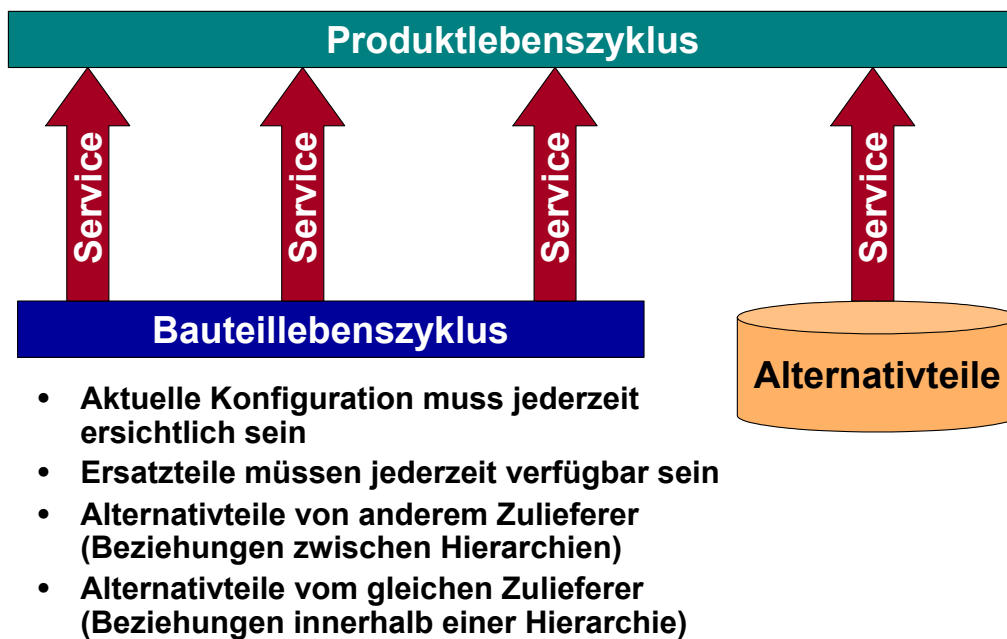


Bild 2-27: Bauteil und Produktlebenszyklus

Um diesen Zugriff zu ermöglichen sind Beziehungen zwischen Klassensystemen notwendig, die den Weg zu alternativ verwendbaren Komponenten oder Bauteilen aufzeigen:

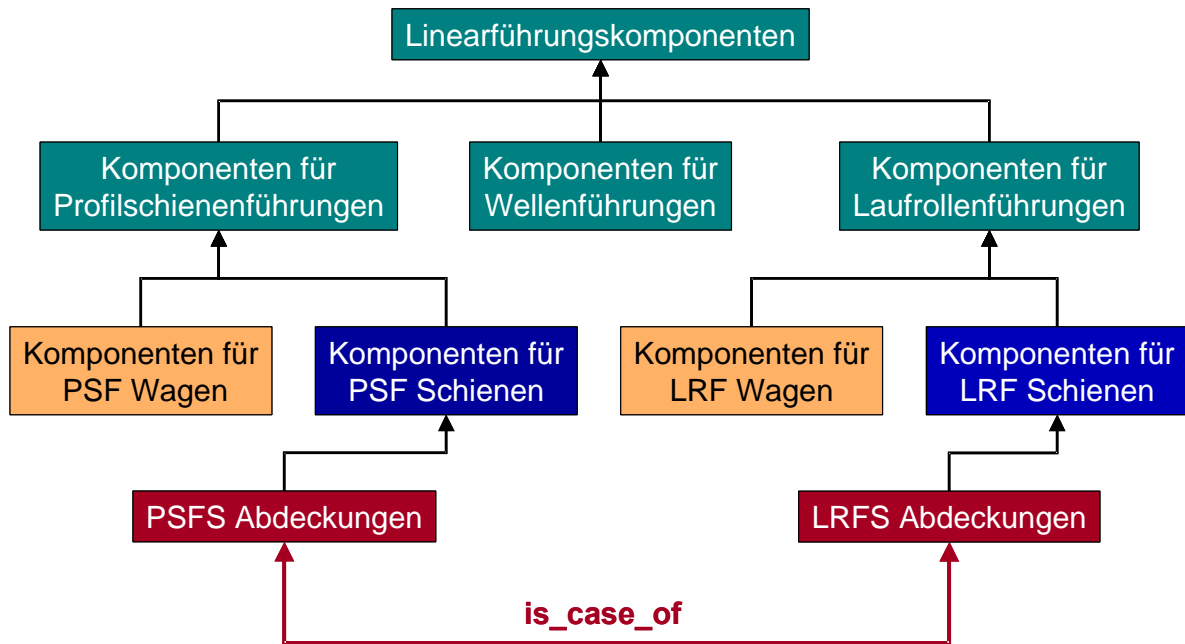


Bild 2-28: Beziehungen innerhalb eines Klassensystems

## 2.5.6 Reduzierung der Produktkomplexität durch Produktbereinigung

Das Ziel der Produktbereinigung ist es, unterschiedliche Teile eines Produkts, welche die gleiche Funktion erfüllen, zu vereinheitlichen, um dadurch die Teilevielfalt zu reduzieren [Mee-99]. Dabei können zwei Fälle unterschieden werden:

- die Teile gehören der gleichen Produktfamilie bzw. Klasse an (z.B. Sechskantschraube M8x20 und M10x20 sollen vereinheitlicht werden zu M10x20) oder
- die Teile gehören nicht der gleichen Klasse an (z.B. Schraubverbindung und Schnappverbindung sollen vereinheitlicht werden).

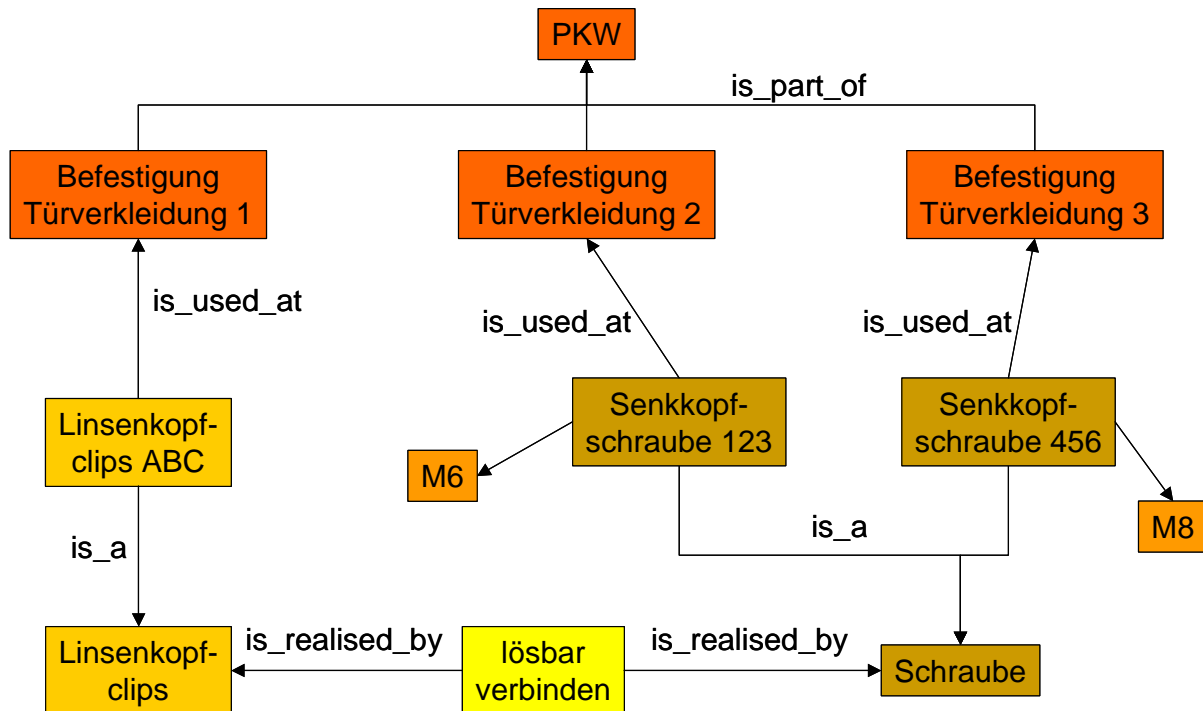


Bild 2-29: Datenmodell für die Produktbereinigung

Diese Aufgabe soll möglichst automatisiert vom Rechner durch Analyse von Produktstruktur und Teilstammdaten erfolgen. Grundvoraussetzung dafür ist die hierarchische Strukturierung und merkmalsbasierte Beschreibung der Teiledaten. Die Vereinheitlichung von Bauteilen gemäß Fall 1 lässt sich dann relativ einfach vom Rechner durchführen. Da alle Schrauben einer gemeinsamen Oberklasse angehören und sich nur durch das Merkmal Gewindedurchmesser unterscheiden, sind sie durch ihre Relation „is\_a“ zur gemeinsamen Oberklasse miteinander verknüpft. Es ist also für jede Schraube möglich, auf dem Weg über die Oberklasse ein alternatives Bauteil (z.B. eine Schraube mit anderem Gewindedurchmesser) zu finden.

Bei Verbindungselementen, die wie in Fall 2 auf einem unterschiedlichen Lösungsprinzip beruhen, ist dies jedoch in einer herkömmlichen Klassenstruktur nicht durch einfaches Auswerten der Vererbungsbeziehungen möglich. Es gibt bei konventionellen Teilstammdaten keine Information darüber, dass z.B. eine Schnappverbindung eine Alternative für eine Schraubverbindung darstellt. Diese Funktionsinformation kann jedoch in einem Klassensystem abgebildet werden. In Form einer Referenzhierarchie können hier Funktionsinformationen abgelegt werden. Jede Klasse von Verbindungselementen ist über eine Relation („is\_realized\_by“) mit der entsprechenden Funktion die sie erfüllt in der Referenzhierarchie verbunden. Ähnlich wie beim Vergleich der Produkte unterschiedlicher Zulieferer, kann jetzt auf dem Weg über die Referenzhierarchie ausgehend von einer Verbindungstechnik eine Alternative vom Rechner gefunden werden [Mee-99].

### 2.5.7 Unterstützung des Recycling

Jeder Hersteller eines Produktes ist aufgrund der nationalen und internationalen Umweltschutz-Gesetzgebung (z.B. EU-Altautoverordnung, Gefahrstoff-Gesetzgebung) für den gesamten Lebenslauf seiner Produkte verantwortlich (Herstellung, Be- und Verarbeitung, Vertrieb, Verwendung, Gebrauch, Wiederverwertung, Entsorgung). So muss die Automobilindustrie im Jahr 2015 gemäß der freiwilligen Selbstverpflichtung 95% eines Fahrzeugs wiederverwerten. Um dieser Anforderung in Zukunft gerecht zu werden, ist es bereits heute erforderlich, die Zusammensetzung der Produkte genau zu kennen. Auch die Rückverfolgung weit verbreiteter Materialien, die aufgrund neuer wissenschaftlicher Erkenntnisse nachträglich als gefährlich eingestuft werden, muss gewährleistet sein. Somit sind detaillierte Kenntnisse über die Zusammensetzung der verwendeten Materialien unbedingt erforderlich. Im Verband der Automobilindustrie e.V. (VDA) wurde daher beschlossen, den Erstmusterprüfbericht gemäß VDA-Handbuch „Qualitätsmanagement in der Automobilindustrie“ Band 2 „Sicherung der Qualität von Lieferungen“ um ein Blatt „Inhaltsstoffe in Zukaufteilen“ zu ergänzen. Man spricht hier vom sog. „Materialdatenblatt (MDB)“. In einem Gemeinschaftsprojekt der Firmen Audi, BMW, DaimlerChrysler, Ford, Opel, Porsche, Volvo, VW und EDS wurde ein EDV-technisches Konzept erarbeitet, das der Fülle von Daten in einem MDB gerecht wird. Dieses System findet Anwendung bei Erzeugnissen und bei produktbezogenen Materialien. Um bei Automobilherstellern und Zulieferbetrieben den Aufwand zu senken und eine möglichst effiziente Datenerfassung zu ermöglichen, wurde ein internetbasiertes internationales Materialdatensystem (IMDS) entwickelt. In weiteren Ausbaustufen soll IMDS nicht nur von der Automobilindustrie, sondern auch von anderen Branchen genutzt werden.

Die Datenbereitstellung erfolgt in Form von Stücklisten, Werkstoffbezeichnungen und Inhaltsstoffen. Die Stücklistenauflösung endet auf Einzelteil-Ebene. Für jedes Einzelteil dokumentiert der Hersteller die Werkstoffbezeichnung, die Inhaltsstoffe und die dazugehörigen Gewichtsangaben. Probleme bereitet das Zusammenstellen all dieser Daten, denn Unternehmen verwalten Stücklisten und die Zuordnung von Inhaltsstoffen zu Werkstoffen in der Regel in getrennten EDV-Systemen. Außerdem setzen verschiedene Hersteller unterschiedliche Anwendungssysteme ein. Es gilt nun, die Grenzen zwischen diesen Systemen aufzuheben und den Datenaustausch zum IMDS zu ermöglichen. Die Verwendung von einheitlichen Werkstoffbezeichnungen und die Vermeidung von Redundanzen ist eine weitere Herausforderung.

Zur Erfüllung der obigen Anforderungen, bietet sich der Einsatz eines Klassensystems an. Es bietet die Möglichkeit, Stücklisten-Stammdaten um Merkmale zu ergänzen, die zur Beschreibung und Identifikation von Werkstoffen und Inhaltsstoffen dienen. Darüber hinaus können mit einem Klassensystem Mengenangaben für Inhaltsstoffe hinterlegt werden. Auch die Verknüpfung firmenspezifischer Werkstoffbezeichnungen mit global gültigen Werkstoffbezeichnungen ist merkmalsbasiert möglich.

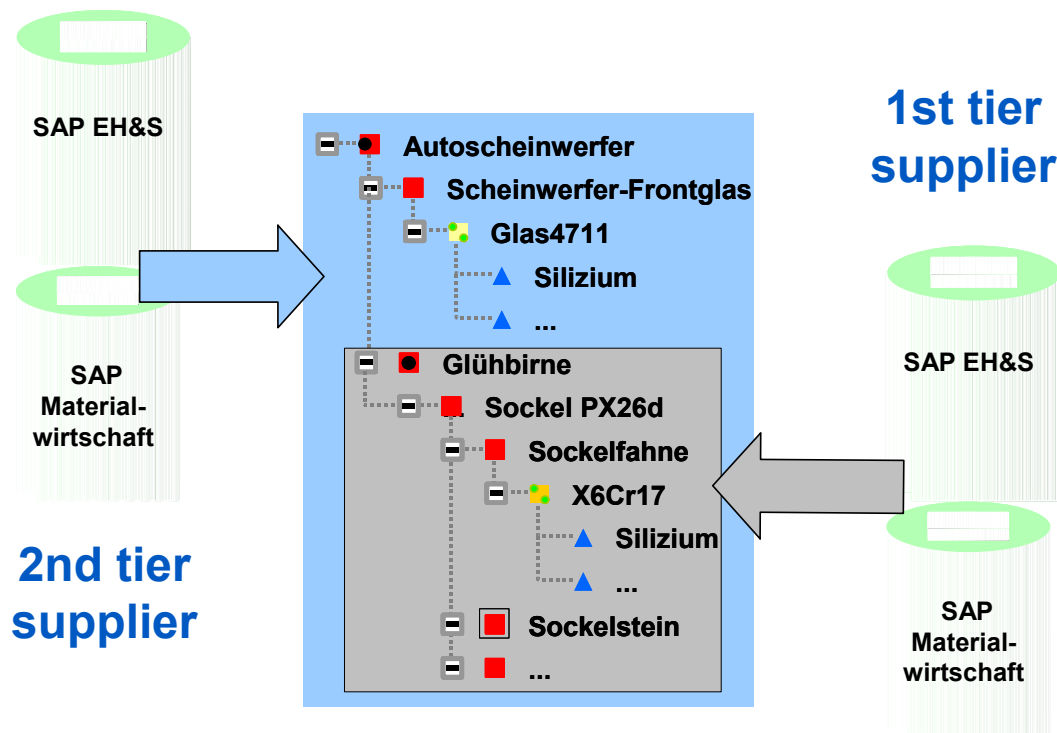


Bild 2-30: Austausch von Zuliefererdaten

Bild 2-30 zeigt, wie ein Klassensystem die Datenbereitstellung für Materialdatenblätter unterstützt. Im vorliegenden Fall wurde ein Klassensystem genutzt, das durch die Software SAP Environment, Health and Safety (EH&S) von SAP bereitgestellt wird. In diesem Modul wurden Klassen für Werkstoffe und Inhaltstoffe sowie deren Merkmale angelegt. Stücklisten für Eigenfertigungsteile werden aus der Materialwirtschaft heraus automatisiert übernommen, Zulieferteile können über eine IMDS spezifische Teilenummer referenziert werden. Eine Verknüpfung von Stücklisten und Werkstoffinformationen sichert die ständige Verfügbarkeit der Daten und ermöglicht auch eine Versionierung, sobald Varianten entstehen. Eine standardisierte XML-Schnittstelle überträgt die Daten zwischen EH&S und IMDS.

Das Ziel der integrierten EH&S-Lösung ist es, den Datenaustausch für Materialdatenblätter weitgehend zu automatisieren und den Aufwand bei den Zulieferern so gering wie möglich zu halten. Eine manuelle Befüllung von IMDS ist somit nicht notwendig. Für alle Zulieferer zugänglich, können einheitliche Werkstoffbezeichnungen in IMDS bereitgestellt werden. Durch geschlossene Prozessketten, wie im obigen Beispiel gezeigt, sollte es möglich sein, die Kosten und den Zeitaufwand für IMDS klein zu halten und von den Vorteilen eines solchen Systems zu profitieren. Es obliegt nun den Zulieferern, IMDS zu unterstützen.

## 2.6 Voraussetzungen für die Einführung eines Klassensystems

Um das Konzept der Klassensysteme in die betriebliche Praxis einzuführen ist es notwendig, dass die IT-Systeme (PLM-Systeme) das Klassenkonzept der OOP unterstützen. Während in der Vergangenheit die Teilverwaltung in den meisten Systemen von Nummernsystemen und

Stammdatensätzen geprägt war, unterstützen moderne PLM-Systeme diesen Ansatz vollständig. Zur Nutzung sämtlicher Vorteile eines Klassensystems entlang des kompletten Lebenszyklus eines Produkts sind Schnittstellen zu allen Systemen notwendig, die Daten aus dem Klassensystem verwenden.

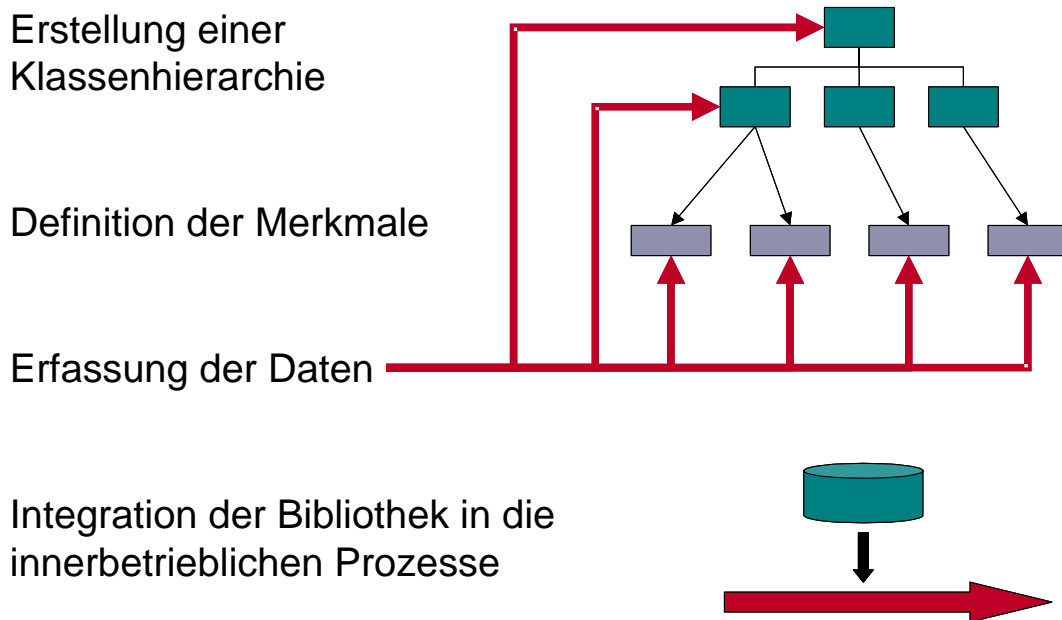


Bild 2-31: Aufbau und Integration eines Klassensystems

Neben den Voraussetzungen im Hinblick auf die IT ist auch tiefgreifendes Know How im Bereich der Klassifizierung notwendig. Hierbei gilt es ein umfangreiches Regelwerk bei der Definition und Strukturierung der Klassen zu berücksichtigen. Unter anderem müssen die folgenden Fragen geklärt werden:

- Welche Objekte sollen im Klassensystem verwaltet werden?
- Wie viele Klassen sollen maximal definiert werden?
- Wie viele Hierarchieebenen sind sinnvoll?
- Wie unterscheiden sich die Klassen auf einer Hierarchieebene (welcher Diskriminator wird gewählt)?
- Welche Merkmale dienen zur Beschreibung einer Klasse?
- Auf welcher Hierarchieebene soll ein bestimmtes Merkmal instanziiert werden?
- Wird eine Information als Merkmal oder durch Einführung einer neuen Klasse abgebildet?



- Welche Querbeziehungen zwischen Klassen („is\_case\_of“) gibt es?
- Welche Sichten auf das Klassensystem (unterschiedliche Strukturen) soll es geben?

Um diese Fragen zu beantworten, müssen die entsprechenden Daten und Prozesse des Unternehmens analysiert werden. In enger Abstimmung mit den Fachabteilungen können auf Basis der Analyse die Klassenstruktur und die beschreibenden Merkmale festgelegt werden. Der Abstimmungsprozess vollzieht sich sinnvoller Weise im Rahmen von Workshops, für deren Moderation und Durchführung in aller Regel externe Beratungsdienstleistung notwendig ist.

### 3 Automatisierte Erstellung von unternehmensspezifischen Klassifikationssystemen

#### 3.1 Konzept zur automatisierten Erstellung von Klassifikationssystemen

Das Lösungskonzept umfasst verschiedene Verfahren zur automatisierten Generierung von Metadaten aus gegebenen Nutzdatenbeständen. Es ist jedoch wichtig, dass jeder einzelne Schritt der automatischen Generierungsprozesse jederzeit nachvollziehbar und manuell modifizierbar ist. Das Konzept besteht aus vier Phasen. Angefangen mit den Phasen Datenanalyse, Merkmalgewinnung, Aufbau des Klassifikationssystems und schließlich der Einsatz des erstellten Klassifikationssystems zur automatischen Durchführung der Klassierung. Die Gesamtphase werden in Bild 3-1 dargestellt. Der Prozess ist prinzipiell sequenziell, kann jedoch auch Iterationen und Sprünge enthalten.

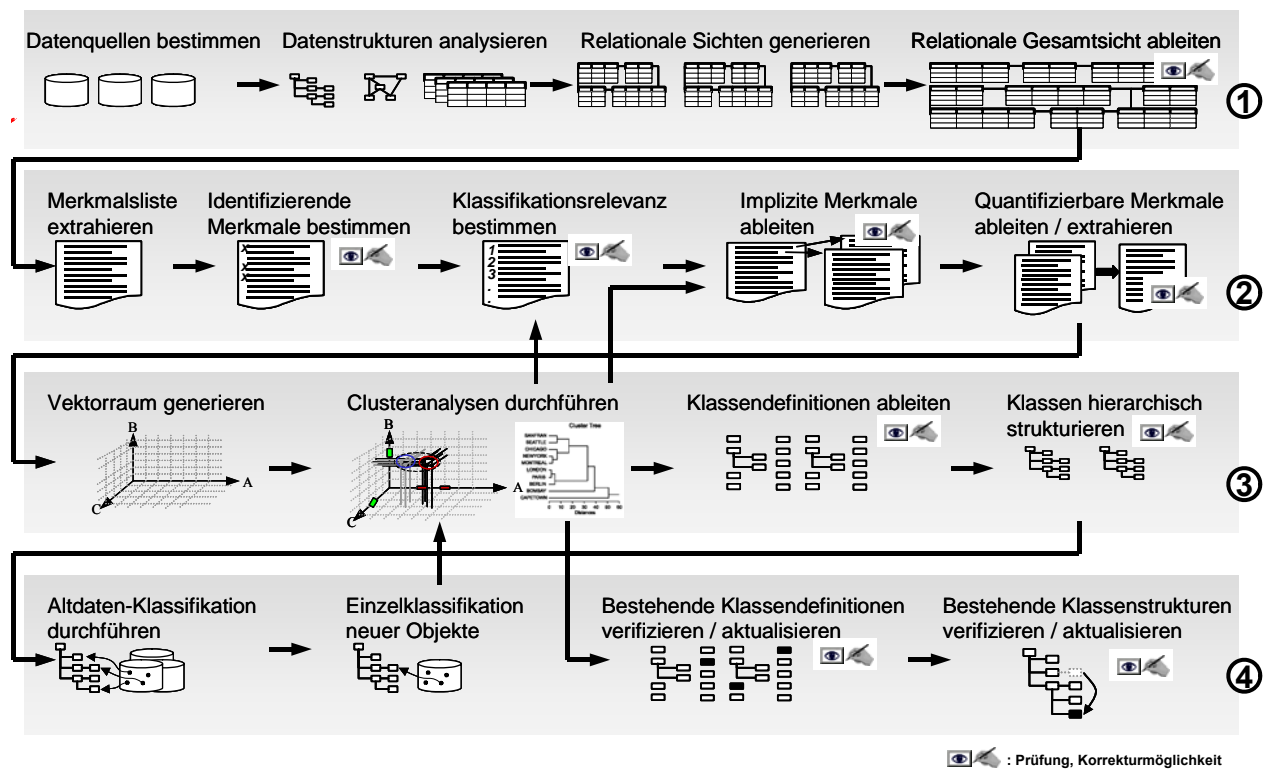


Bild 3-1: Übersicht über die vier Phasen des Lösungsansatzes [Wei-02]

In den nächsten Unterkapiteln werden die einzelne Phasen detailliert beschrieben.

#### 3.1.1 Datenanalysephase

In der ersten Phase werden zunächst die Datenquellen bestimmt, die zu klassifizierenden Datenbestände beinhalten. Die Datenquellen können hierbei unterschiedliche IT-Systeme sein,

die Produktdaten aus verschiedenen Produktentstehungsphasen beinhalten. Wichtig bei der Identifikation von Nutzdatenquellen ist, dass ein und dasselbe Softwaresystem mehrere Nutzdatenquellen „beinhalten“ kann. Zum Beispiel werden Arbeitspläne, Stücklisten und Anforderungslisten mit einem Tabellenkalkulationsprogramm erstellt. Wichtig ist zu beachten, dass jede Datei als eigene Nutzdatenquelle behandelt werden muss.

Die Summe aller Unterlagen beinhaltet alle Informationen, die benötigt werden, um ein Produkt in seinen Lebensphasen vollständig zu beschreiben. Dass hierbei auch redundante und teilweise sogar inkonsistente Informationen vorliegen können, ist ein weit verbreitetes Problem. Die automatische Klassifikation bietet hierfür ein Lösungspotenzial, indem durch Klassifikation Redundanzen und Inkonsistenzen erkannt und angezeigt werden können.

Die ausgewählten Datenquellen werden danach analysiert um festzustellen, zu welcher Datenmodellart sie gehört. Ein Datenmodell kann relational, hierarchisch, netzwerkartig oder objektorientiert sein. Die heute am meisten verbreitete Datenstruktur ist das relationale Datenmodell. Dieses Datenmodell liegt den meisten gängigen Datenbanksystemen, EDM/PDM-Systemen und ERP-Systemen zugrunde. Aber auch IT-Systeme, denen keine relationale Struktur zugrunde liegt, lassen sich relational beschreiben. Um einen systemübergreifenden Zugriff auf die unterschiedlichen Datenmodelle zu ermöglichen, ist die Bildung einer einheitlichen Sicht auf die verschiedenen Datenquellen erforderlich. Dem hier beschriebenen Lösungskonzept für die automatische Klassifikation wird daher ebenfalls eine relationale Sicht auf alle zu klassifizierenden Nutzdatenquellen zugrunde gelegt. Beim relationalen Datenmodell stehen als Strukturelemente ausschließlich Relationen zur Verfügung, die sich durch Tabellen darstellen lassen. Die Datensätze bilden die Zeilen, und die Merkmale des Objekts bzw. die Datenfelder entsprechen den Spalten der Tabelle. Beziehungen zwischen beliebigen Datensätzen werden über gleiche Feldinhalte hergestellt. Der Zugriff auf bestimmte Datensätze wird über die Feldinhalte ermöglicht.

### 3.1.2 Merkmalsermittlung

Jede Datenquelle besitzt eine individuelle Menge von Merkmalen. In diesem Schritt werden alle Merkmale ermittelt, die explizit in den Nutzdatenquellen verfügbar sind. Es werden danach diejenige Merkmale ermittelt, die einen identifizierenden Charakter haben, d.h. deren Ausprägungen ein zu klassifizierendes Objekt eindeutig identifizieren. Identifizierende Merkmale legen fest, welcher Typ von Objekten in einem Klassifikationssystem klassifiziert werden kann: Einzelteile, Baugruppen, Arbeitspläne, allgemein Dokumente, etc. Jedes klassifizierte Objekt muss für ein identifizierendes Merkmal eine eindeutige Ausprägung besitzen. Gegebenenfalls ergibt erst die Kombination mehrerer Nutzdatenattribute eine eindeutige Ausprägung, beispielsweise die ID eines Teilestammes in Verbindung mit einer Versionsnummer. In diesem Fall besteht das identifizierende Merkmal aus der Kombination der beiden Nutzdatenattribute ID und Version.

Der nächste Schritt ist die Sortierung der Merkmale entsprechend ihrer Klassifikationsrelevanz. Klassifikationsrelevante Merkmale sind solche, deren Ausprägungen die zu klassifizie-

renden Objekte gegeneinander abgrenzen. Mit einer Clusteranalyse wird es ermöglicht, nicht-klassifikationsrelevante Merkmale automatisch zu ermitteln. Entsprechend der Verfahrensweise der klassischen Clusteranalyse werden zunächst alle  $n$  explizit in den Nutzdaten verfügbaren Merkmale des zu klassifizierenden Produktspektrums auf Merkmalsachsen aufgetragen. Diese Merkmalsachsen spannen einen  $n$ -dimensionalen Vektorraum auf, in dem jedes Produkt durch einen Punkt repräsentiert wird. Die Punkte werden nun auf alle Merkmalsachsen projiziert. Aus der Verteilung der Punkte entlang einer Achse lassen sich erste Rückschlüsse bzgl. der Klassifikationsrelevanz dieses Merkmals ziehen. In Bild 3-2 ist ein idealisiertes Beispiel dargestellt, in dem die zu klassifizierenden Objekte durch drei Merkmale A, B und C beschrieben werden. Diese drei Merkmale spannen einen 3-dimensionalen Vektorraum auf.

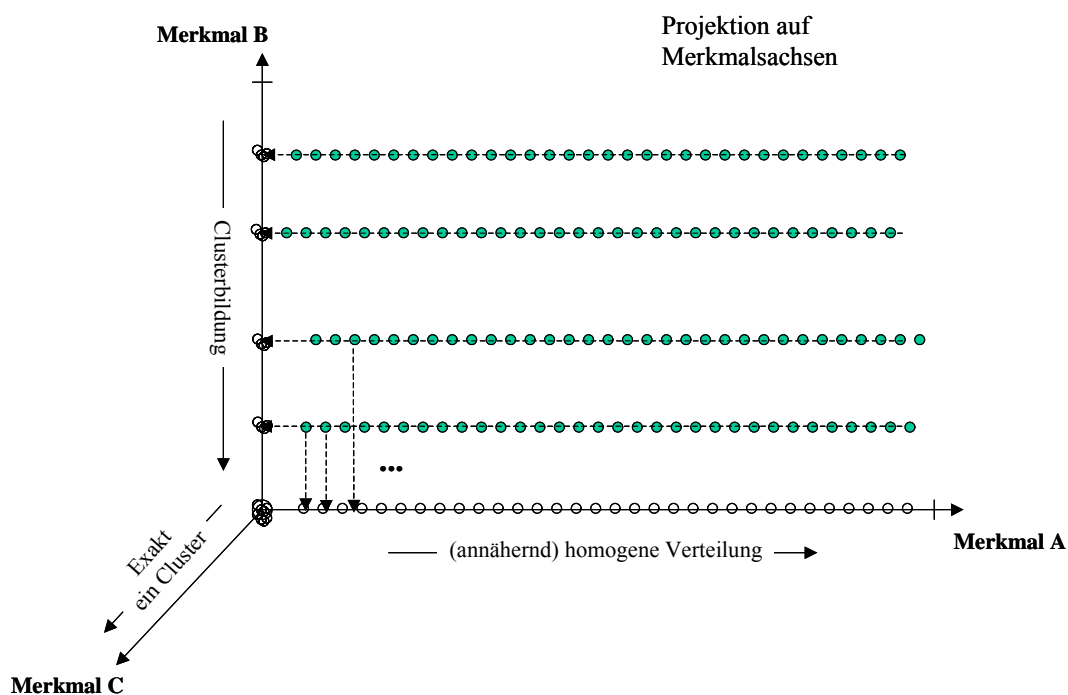


Bild 3-2: Ermittlung nicht-klassifikationsrelevanter Merkmale [Wei-02]

Bei annähernd homogener Verteilung (z.B. eine laufende Nummer) oder bei Bildung eines einzigen Clusters (z.B. bei nicht gepflegten Feldern einer Datenquelle) kann davon ausgegangen werden, dass dieses Merkmal für die Klassifikation nicht relevant ist. In dem in Bild 3-2 dargestellten Beispiel ergibt sich bei der Projektion aller Punktrepräsentationen auf den Merkmalsvektor A eine annähernd homogene Verteilung. Eine Projektion auf den Merkmalsvektor C führt zu genau einem Cluster. Die Merkmale A und C werden demnach als nicht-klassifikationsrelevant eingestuft. Nicht-klassifikationsrelevante Merkmale werden im weiteren Verlauf nicht berücksichtigt.

Merkmale können nicht nur explizit sondern auch implizit sein. Die impliziten Merkmale werden durch Verknüpfungsoperationen aus bereits verfügbaren Merkmalen definiert. Zum

Beispiel kann eine Welle als eine lange Welle oder kurze Welle definiert werden, indem man das Verhältnis zwischen deren Durchmesser und deren Länge berechnet.

Werden bei der Untersuchung, ob ein numerisches Merkmal für die Klassifikation relevant ist oder nicht, nur die Häufigkeitsverteilungen der Merkmalsausprägungen entlang der einzelnen Achsen des Vektorraumes untersucht, können für die Klassifikation wichtige Informationen verloren gehen. In Bild 3-3 ist ein Beispiel dargestellt, in dem alle Produkte sowohl hinsichtlich Merkmal A als auch hinsichtlich Merkmal B annähernd homogen im Vektorraum verteilt sind. Eine Projektion der Produkte auf die von A und B aufgespannte Ebene zeigt jedoch, dass die Anordnung der Punkte auf dieser Ebene nicht homogen ist.

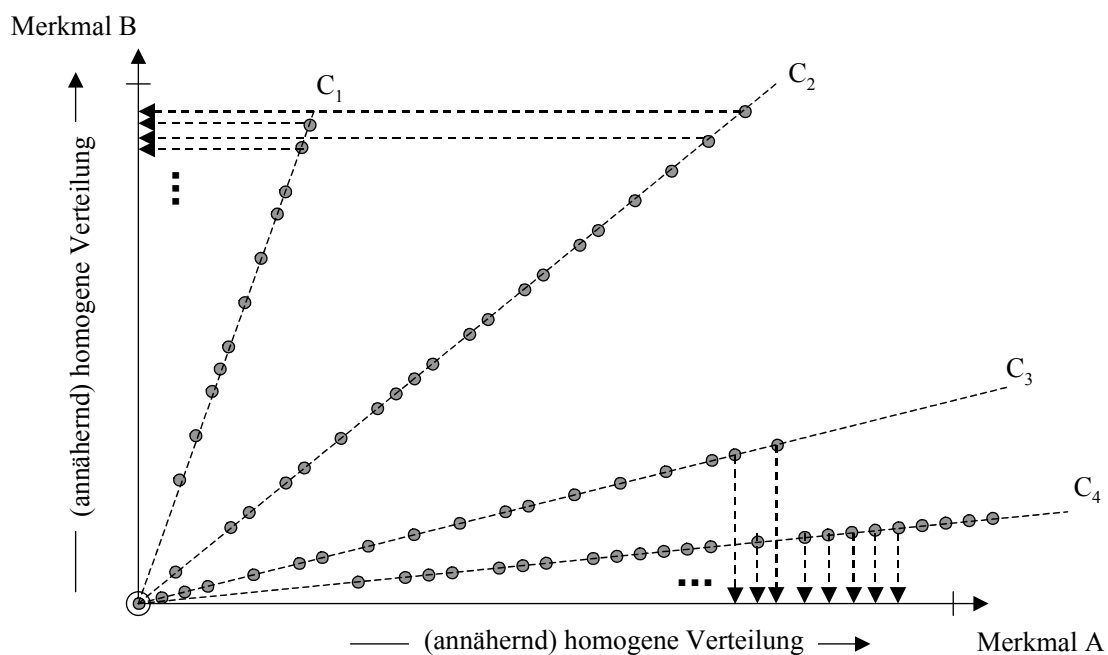


Bild 3-3: Gewinnung zusätzlicher klassifikationsrelevanter Merkmale (1)

Derartige inhomogene Verteilungen lassen darauf schließen, dass zwischen den betrachteten Merkmalen A und B eine Beziehung besteht, aus der sich ein weiteres, implizit verfügbares Merkmal ableiten lässt.

Die ermittelten klassifikationsrelevanten Merkmale werden abschließend quantifiziert, d.h. in numerische Merkmale überführt. Dieser Schritt ist erforderlich, da Clusteranalyse-Verfahren, mit deren Hilfe Metadaten automatisch ermittelt werden sollen, lediglich numerische Daten verarbeiten können.

### 3.1.3 Aufbau eines Klassifikationssystems

In der dritten Phase wird ein Klassifikationssystem durch Clusteranalysen erstellt. In Bild 3-4 ist ein Ansatz dargestellt, wie einfach nachvollziehbare, definierende Eigenschaften einer

Klasse abgeleitet werden können. Gemäß der klassischen Clusteranalyse-Verfahren werden hierbei Vektorräume generiert und Clusteranalysen durchgeführt. Für jedes Merkmal des Vektorraumes wird für einen Cluster ein Intervall gültiger Merkmalsausprägungen ermittelt. Liegt die Punktrepräsentation eines zu klassifizierenden Objektes innerhalb dieser Intervalle, wird es dieser Klasse zugeordnet. Die Clustergeometrien werden hierbei durch umhüllende, kubische Vektorräume angenähert. Diese Vereinfachung wird in vielen Fällen bereits zu befriedigenden Ergebnissen führen.

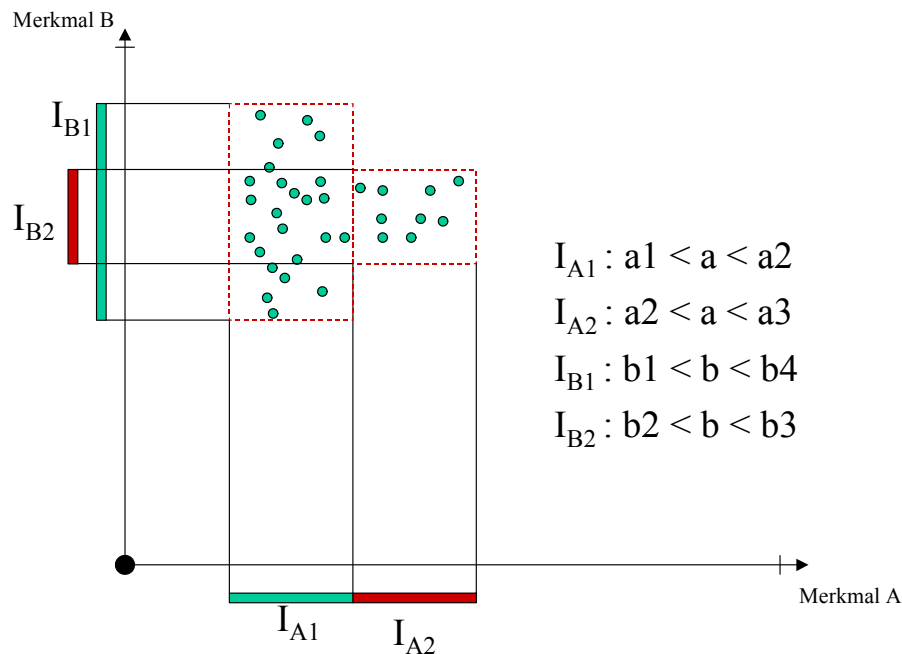


Bild 3-4: Verknüpfung kubischer Vektorräume [Wei-02]

Die Ergebnisse dieser Clusteranalysen werden aufbereitet, interpretiert und für den Aufbau des Klassifikationssystems (Klassendefinition und hierarchische Strukturierung der Klassen) verwendet. Bild 3-5 zeigt ein Beispiel, wie eine Hierarchie von einem Klassifikationssystem definiert wird. Zwei Objektklassen I.1 und I.2 sowie deren definierende Eigenschaften wurden ermittelt und im Bild dargestellt. Durch eine iterative Reduktion der Dimensionalität des Vektorraumes, d.h. einer Projektion der gefundenen Cluster in einen Vektorraum der Dimension  $n-1$ , lässt sich ermitteln, dass beide Cluster in einem um den Merkmalsvektor A reduzierten Vektorraum zu einem Cluster verschmelzen. Wird dieser Cluster in eine Objektklasse I überführt, so ist I eine Oberklasse von I.1 und I.2. Auf diese Weise lassen sich hierarchische Beziehungen zwischen Klassen ableiten, deren Cluster, aus denen sie abgeleitet wurden, im „vollständigen“ Vektorraum geometrisch weit voneinander entfernt liegen können.

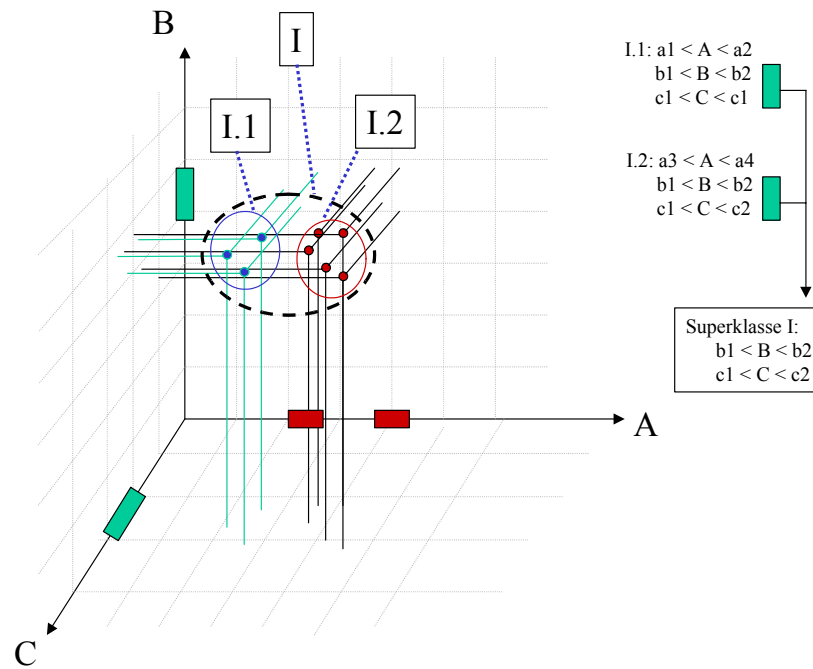


Bild 3-5: Erzeugen hierarchischer Strukturen durch Clusterprojektionen [Wei-02]

### 3.1.4 Anwendung eines Klassifikationssystems

Die letzte Phase ist der Einsatz und die Wartung eines Klassifikationssystems. Dies umfasst zunächst eine initiale Klassifikation, bei der die existierenden Datenbestände gemäß der in den vorangegangenen Phasen definierten Metadaten klassifiziert werden. Anschließend erfolgt bei jeder Änderung in den Datenquellen eine aktualisierende Klassifikation, bei der lediglich die geänderten bzw. neuen Datenbestände klassifiziert werden. Neue oder geänderte Datenbestände erfordern eine Verifikation und ggf. Anpassung des Klassifikationssystems, beispielsweise kann die Anzahl der Objekte, die einer Klasse zugeordnet sind, zu groß werden, was eine feinere Differenzierung der Klassifikation an dieser Stelle erforderlich machen würde. Ändert sich nicht nur der Inhalt, sondern auch die Struktur des Datenbestandes (z.B. durch eine Erweiterung des Produktspektrums oder durch die Einführung eines neuen IT-Systems), ist ein Rücksprung in die erste Phase erforderlich.

## 3.2 Softwaretechnische Realisierung

### 3.2.1 Systemarchitektur

In Bild 3-6 ist die Architektur des Systems zur automatischen Produktdatenklassifikation dargestellt. Die Architektur ist durch einen streng modularen Aufbau gekennzeichnet, was eine leichte Wartbarkeit und Erweiterbarkeit des Systems gewährleistet.

Kernkomponenten der Systemarchitektur sind die operativen Komponenten *Analyse-*, *Klassifizierungs-*, *Klassierungs-* sowie das *Berechnungsmodul*. Das Analysemodul erzeugt die rela-

tionale Gesamtansicht auf die zu klassifizierenden Datenbestände. Das Klassifizierungsmodul greift auf diese relationale Gesamtansicht über die *allgemeine Datenschnittstelle* zu, erzeugt auf Basis der beschriebenen Verfahren Merkmals- und Klassendefinitionen und führt eine Strukturierung der Klassen zu verschiedenen Klassifikationsschemata durch. Die vom Klassifizierungsmodul erzeugten Metadaten (Merkmals- und Klassendefinitionen, Klassenhierarchien, etc.) werden über die *Metadaten-schnittstelle* im Metadaten-speicher gemäß der beschriebenen Spezifikationen abgelegt.

Das Klassierungsmodul liest über die Metadaten-schnittstelle die spezifizierten Metadaten ein und baut analog der spezifizierten Klassenhierarchie eine *Metastruktur* auf, welche die Ergebnisse eines Klassifikationsprozesses aufnimmt. Anschließend ermittelt das Klassierungsmodul über die allgemeine Datenschnittstelle die zur Klassifikation erforderlichen Nutzdaten. Das Berechnungsmodul evaluiert die definierten Berechnungsvorschriften mit den ermittelten Nutzdaten und ermittelt so die Ausprägungen impliziter Merkmale. Durch Evaluierung der definierenden Eigenschaften der Klassen mit den ermittelten Nutzdaten führt das Klassierungsmodul die Zuordnung der zu klassifizierenden Objekte zu den einzelnen Klassen durch und trägt die ermittelten Merkmalsausprägungen für jedes klassifizierte Objekt in die Ausprägungsteile der Metastruktur ein.

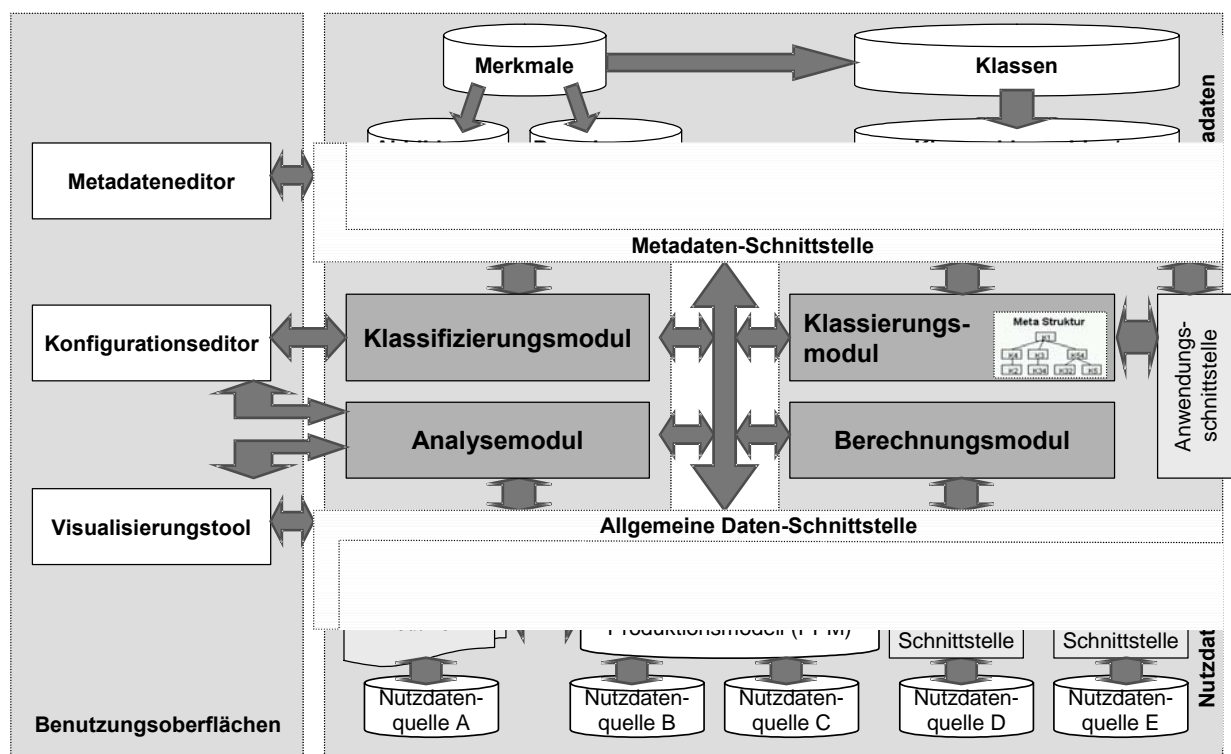


Bild 3-6: Systemarchitektur

Die Nutzdaten umfassen alle Datenquellen, die über *spezielle Datenschnittstellen*, *Interpretationsroutinen* oder direkt (integrierte Produktmodelle wie z.B. STEP [ISO-94] oder das integrierte Produkt- und Produktionsmodell (PPM) [GrMH-95], [HaMM-95]) an die allgemeine



Datenschnittstelle angebunden sind. Als Quellen für Nutzdaten wurde die Anbindung folgender Systeme realisiert:

- SAP R/3
- CADIM/EDB
- CAD-Schnittstellen:
- Pro/Engineer
- ACIS (3D-Autocad, Solid Edge)
- Oracle-Datenbank
- MsOffice
- Postgres-Datenbank
- XML
- mysql-Datenbank

Durch den modularen Aufbau des Softwaresystems ist die Anbindung weiterer Nutzdatenquellen leicht zu realisieren.

Die Steuerung des Klassierungsmoduls (Authentifizierung, Auswahl der Ordnungsschemata und der zu klassifizierenden Datenbestände, Start des Klassifikationsprozesses, etc.) sowie das Abrufen der Klassifikationsergebnisse erfolgt über die *Anwendungsschnittstelle*, welche die Anbindung externer Anwendungsapplikationen (z.B. ein PDM-System, dessen Sachmerkmale automatisch gepflegt werden sollen) ermöglicht. Der Funktionsumfang dieser Schnittstelle umfasst die folgenden Bereiche:

- Systemverwaltung:  
Zur Systemverwaltung gehören An- und Abmelfunktionalität, Authentifizierung sowie die Prozessverwaltung (jeder Benutzer ist in der Lage, beliebig viele Klassifikationsprozesse anzulegen).
- Metadatenabruf:  
Es sind alle spezifizierten Metadaten abrufbar: Physikalische Einheiten, Merkmalspezifikationen (inklusive aller Abbildungs- und Berechnungsvorschriften), Klassenspezifikationen, Klassenhierarchien sowie die verfügbaren Nutzdatenquellen (inklusive der relationalen Gesamtsicht).

- **Prozessspezifikation:**  
Bevor ein Klassifikationsprozess gestartet werden kann, muss dieser spezifiziert werden. Zur Spezifikation gehört: Festlegung der Klassenhierarchie(n), gemäß derer klassifiziert werden soll, sowie Festlegung der zu klassifizierenden Nutzdatenquellen (ggf. mit Einschränkung des zu klassifizierenden Datenbestandes). Es erfolgt automatisch eine Prüfung, ob die selektierten Nutzdatenquellen gemäß der ausgewählten Hierarchie klassifiziert werden kann.
- **Prozesssteuerung:**  
Zur Prozesssteuerung gehört neben dem Starten, Anhalten und Abbrechen eines Klassifikationsprozesses auch die Steuerung von Batch-Prozessen (zur Vorabplanung langwieriger Klassifikationsprozesse), Prozessstatus- und Fehlerhandhabung sowie die Festlegung von Callback-Routinen (Aktivitäten, die vom System beim Erreichen eines bestimmten Prozessstatus automatisch angestoßen werden).
- **Ergebnisabruf:**  
Die Ergebnisse eines Klassifikationsprozesses können klassenweise abgerufen werden, d.h., die Ausprägungen aller einer Klasse zugeordneten Klassenattribute werden für jedes Objekt zurückgeliefert, das dieser Klasse zugeordnet wurde.

Grafische Benutzungsoberflächen ermöglichen die Visualisierung der relationalen Gesamt-sicht auf den zu klassifizierenden Datenbestand, die Konfiguration des Analyse- und des Klassifizierungsmoduls (z.B. Auswahl der anzuwendenden Clusteranalyse-Verfahren, Ein-schränkung des zu berücksichtigenden Datenbestandes, etc.) sowie die Visualisierung, Verifi-zierung und Anpassung der erzeugten Metadaten.

Als Programmiersprache wurde JAVA gewählt, um eine plattform-unabhängige Implementie-rung des Softwaresystems zu gewährleisten und den Zugriff auf neueste Internet- und Netz-werk-Technologien sicherzustellen.

### 3.2.2 Informationsmodell

Voraussetzung für eine technische Realisierung der entwickelten Konzeption ist die Spezifi-zierung aller für die automatische Produktdatenklassifikation relevanten Informationsobjekte und ihrer Beziehungen zueinander. Konzeptionelles Kernstück ist dabei ein Informationsmo-dell, das mit Hilfe der *Unified Modeling Language* (UML) formal beschrieben wird.

#### 3.2.2.1 Informationsmodell zur Abbildung heterogener Datenstrukturen

In Bild 3-7 ist das Informationsmodell zur Abbildung der in Abschnitt 3.1.1 vorgestellten re-lationalen Sichten auf Nutzdatenquellen dargestellt. Ein *System* repräsentiert eine anzubin-

dende Nutzdatenquelle mit dem entsprechenden Systemtyp (z.B. Oracle-Datenbank, Pro/ENGINEER-Geometriedateien, etc.) sowie einer eindeutigen Bezeichnung der Datenquelle. Optional kann eine Instanz der Klasse System auch Login-Parameter (Benutzerauthentifizierung u.Ä.) verwalten. Ein System beinhaltet eine oder mehrere *Entitäten*, jeweils mit einer innerhalb eines Systems eindeutigen Bezeichnung. Eine Entität wiederum enthält ein oder mehrere *Attribute*, welche die typisierten, innerhalb einer Entität eindeutig bezeichneten Datenfelder einer Nutzdatenquelle repräsentieren.

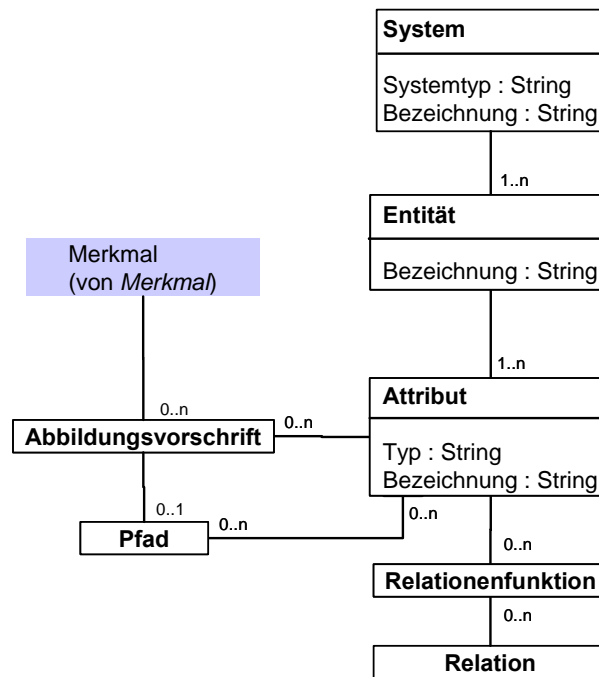


Bild 3-7: Informationsmodell der relationalen Sicht auf Nutzdatenquellen [Wei-02]

Mittels *Abbildungsvorschriften* kann ein Attribut mit beliebig vielen *Merkmalen* verknüpft werden. Diese Merkmale sind dann in der Lage, ihre Ausprägungen explizit aus der entsprechenden Nutzdatenquelle zu ermitteln. Es ist nicht erforderlich, alle verfügbaren Attribute durch Abbildungsvorschriften mit Merkmalen zu verknüpfen, auf diese Weise kann eine Vorselektion erfolgen, indem nur die für die Klassifikation relevanten Datenfelder auf Merkmale abgebildet werden. Andererseits können einem Merkmal beliebig viele Abbildungsvorschriften und damit Verknüpfungen mit Attributen zugeordnet werden. So ist es möglich, mehrere sinngleiche Attribute, z.B. aus verschiedenen Nutzdatenquellen, über ein einziges Merkmal zu referenzieren.

Sobald ein Attribut über eine Abbildungsvorschrift mit einem Merkmal verknüpft ist, kann diese Abbildungsvorschrift mit einem *Pfad* versehen werden. Ein Pfad bestimmt, auf welche Weise ein Merkmal seine Ausprägungen im Kontext eines bestimmten identifizierenden Merkmals bestimmt.

Die Abbildung von Beziehungen zwischen den Datenfeldern einer oder mehrerer Nutzdatenquellen erfolgt mittels *Relationen*. Alle Attribute, die derselben Relation zugeordnet werden,

stehen in direkter Beziehung zueinander. Die Verknüpfung eines Attributes mit einer Relation erfolgt mittels einer *Relationenfunktion*. Neben der Abbildung von Beziehungen zwischen Attributen über gleiche Feldinhalte können so auch Beziehungen über nicht exakt übereinstimmende Feldinhalte abgebildet.

### 3.2.2.2 Informationsmodell zur Abbildung und Verwaltung von Merkmalen

In Bild 3-8 ist das Informationsmodell zur Abbildung und Verwaltung von *Merkmalen* dargestellt. Merkmale werden spezialisiert in die typisierten Merkmale INTEGER-Merkmal, FLOAT-Merkmal, STRING-Merkmal, DATE-Merkmal, BOOL-Merkmal oder FUZZY-Merkmal. Alle Merkmale besitzen mindestens eine Bezeichnung, wobei beliebig viele synonyme Bezeichnungen möglich sind.

Einem Merkmal kann eine physikalische *Einheit* zugeordnet werden. Eine Einheit besitzt neben ihrer eindeutigen Bezeichnung ein Abkürzungszeichen sowie optional eine Formel, welche die Einheit mit anderen Einheiten verknüpft (z.B. Einheit „Meter“, Abkürzung „m“, Formel „1000\*mm“). Wenn erforderlich, können so Merkmalsausprägungen mit unterschiedlichen Einheiten entsprechend umgerechnet werden, beispielsweise bei der Evaluierung von Berechnungsvorschriften (z.B. „Fläche [mm<sup>2</sup>] = Länge[m] \* Breite[mm]“, hier wird die Einheit [m] dann durch die entsprechende Formel 1000\*mm ersetzt).

Einem Merkmal kann eine *Unschärfe* zugeordnet werden, beispielsweise wenn bekannt ist, dass es sich bei den Merkmalsausprägungen um geschätzte Werte handelt oder Merkmalsausprägungen tatsächlich Ausprägungsintervalle sind (z.B. Spannung = 220±10V). Je nach Merkmalstyp wird die Unschärfe spezialisiert in eine *numerische* oder eine *alphanumerische* Unschärfe. Eine alphanumerische Unschärfe wird durch einen regulären Ausdruck repräsentiert. Eine numerische Unschärfe beschreibt entweder ein absolutes oder ein relatives, d.h. prozentuales Ausprägungsintervall.

Einem Merkmal können beliebig viele *Wertebereiche* zugeordnet werden. Wertebereiche schränken die möglichen gültigen Ausprägungen eines Merkmals ein. Ein Wertebereich spezialisiert sich je nach Merkmalstyp in einen *numerischen* oder einen *alphanumerischen* Wertebereich.

FUZZY-Merkmalen kann darüber hinaus eine *Durchschnitts-* und eine *FUZZY-Funktion* zugeordnet werden, mit deren Hilfe die Ausprägung eines FUZZY-Merkmals aus anderen Merkmalen berechnet werden kann.

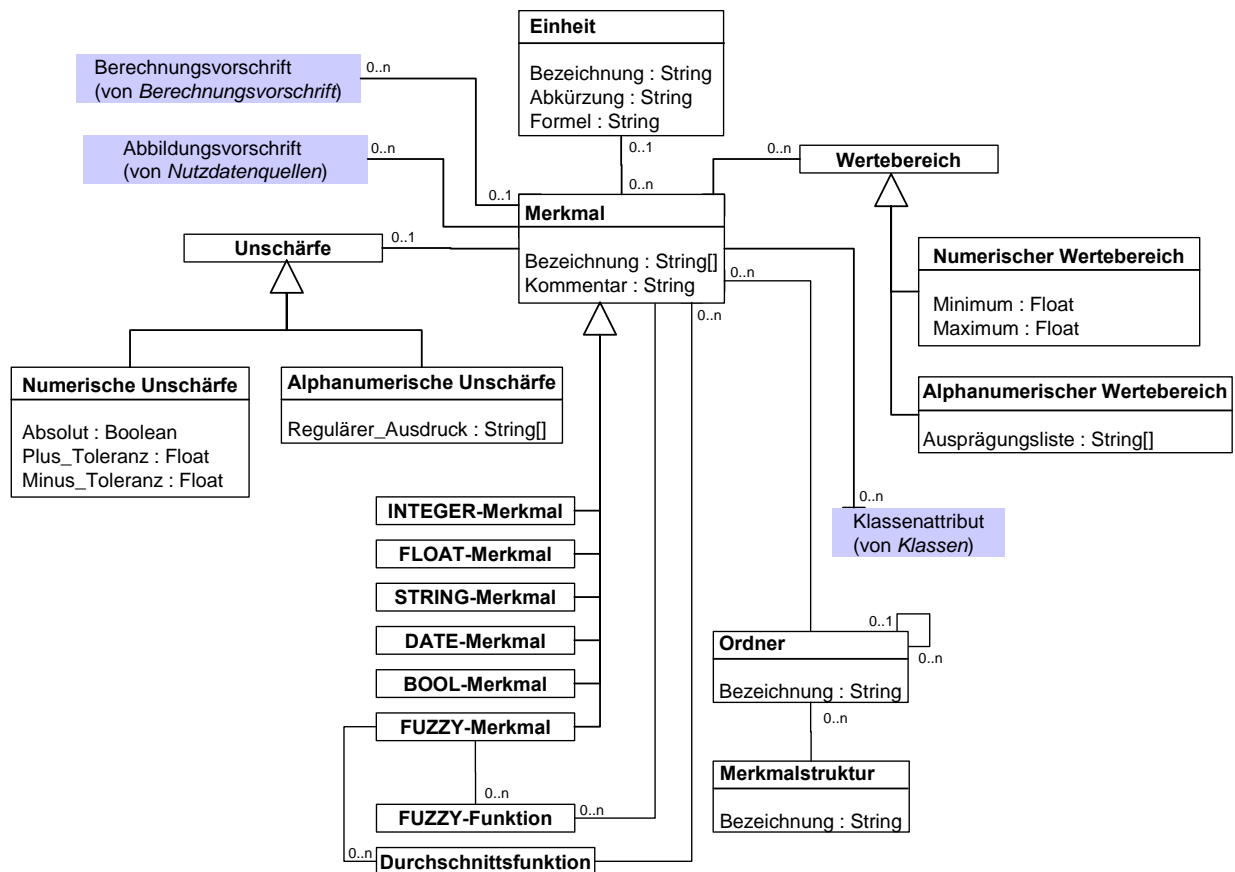


Bild 3-8: Informationsmodell zur Abbildung und Verwaltung von Merkmalen [Wei-02]

Die Verwaltung von Merkmalen erfolgt in *Ordnern*. Innerhalb eines Ordners muss jedes Merkmal eine eindeutige Bezeichnung haben. Ordner können beliebig hierarchisch strukturiert werden. Eine solche hierarchische Gliederung beliebig vieler Ordner wird als eine *Merkmalstruktur* verwaltet.

Neben der in Abschnitt 3.1.2 beschriebenen expliziten Verknüpfung eines Merkmals mit einem Nutzdatenattribut über eine Abbildungsvorschrift kann ein Merkmal seine Ausprägungen auch mittels einer oder mehrerer Berechnungsvorschriften aus anderen Merkmalen ermitteln. Die Abbildung solcher Berechnungsvorschriften ist Gegenstand des nächsten Abschnitts.

### 3.2.2.3 Informationsmodell zur Abbildung von Berechnungsvorschriften

In Bild 3-9 ist das Informationsmodell zur Abbildung von Berechnungsvorschriften dargestellt. Eine Berechnungsvorschrift ist aus einem Operator sowie einer beliebigen Anzahl Operanden aufgebaut. Operatoren spezialisieren sich in Textoperatoren, mathematische Operatoren, boolesche Operatoren, Vergleichsoperatoren oder Listenoperatoren.

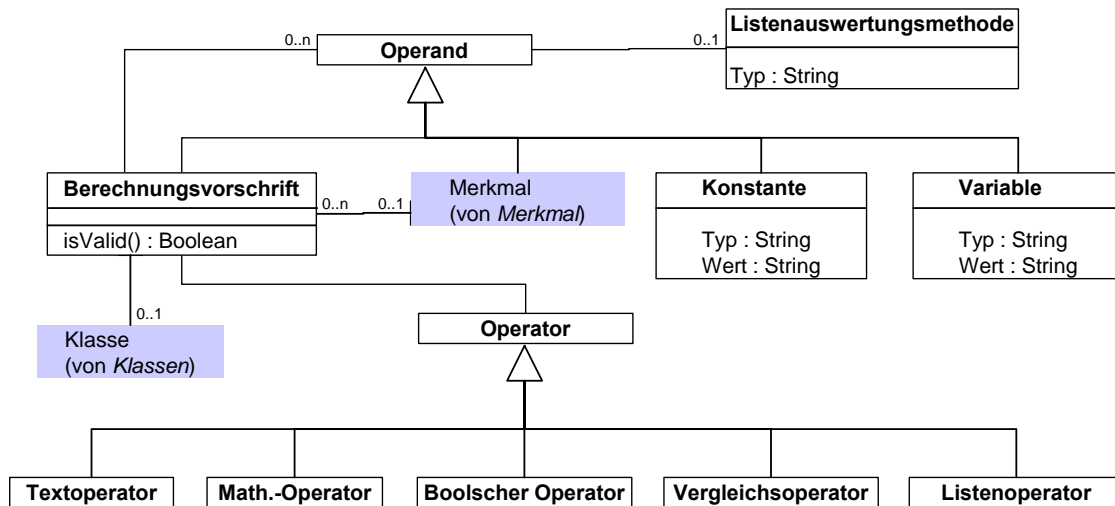


Bild 3-9: Informationsmodell zur Abbildung von Berechnungsvorschriften [Wei-02]

Operanden spezialisieren sich in Variablen, Konstanten, Merkmale oder wiederum Berechnungsvorschriften. Auf diese Weise ist die Definition beliebig komplexer Berechnungsvorschriften möglich. Eine Berechnungsvorschrift kann jederzeit auf ihre Gültigkeit überprüft werden. Beispielsweise ist es nicht zulässig, numerische Operanden mit booleschen Operatoren zu verknüpfen. Des Weiteren muss der Ergebnistyp einer Berechnungsvorschrift dem Typ des Merkmals entsprechen, dem die Berechnungsvorschrift zugeordnet ist.

Einem Operanden kann eine *Listenauswertungsmethode* zugeordnet werden, welche abhängig von ihrem Typ (SUM, AVERAGE, MIN, MAX, etc.) die Auswertung von Ausprägungslisten durchführt.

Berechnungsvorschriften mit dem Ergebnistyp Boolean oder Fuzzy können auch einer *Klasse* zugeordnet werden und bilden dann die *definierende Eigenschaft* dieser Klasse. Die Abbildung und Verwaltung von Klassen und Klassenhierarchien ist Gegenstand des folgenden Abschnitts.

### 3.2.2.4 Informationsmodell zur Abbildung von Klassen und Klassenhierarchien

In Bild 3-10 ist das Informationsmodell zur Abbildung von Klassen und Klassenhierarchien dargestellt. Jede Klasse ist mit einer Berechnungsvorschrift verknüpft. Diese Berechnungsvorschrift muss vom Rückgabebetyp Boolean oder Fuzzy sein und bildet die definierende Eigenschaft der Klasse. Das Evaluierungsergebnis dieser Berechnungsvorschrift für ein zu klassifizierendes Objekt wird mit dem Grenzwert der Klasse verglichen, liegt das Ergebnis über dem Grenzwert, so wird das Objekt dieser Klasse zugeordnet. Einer als abstrakt markierten Klassen können keine Objekte zugeordnet werden. Hierdurch kann z.B. erreicht werden, dass Objekte nur den Blättern einer Hierarchie zugeordnet werden, wie es z.B. gemäß [DIN-81] üblich ist.

Eine Klasse kann eine Codierung besitzen, die eine Verschlüsselung eines dieser Klasse zugeordneten Objektes gemäß dem in der *Klassenhierarchie* spezifizierten Codierungsformat erlaubt.

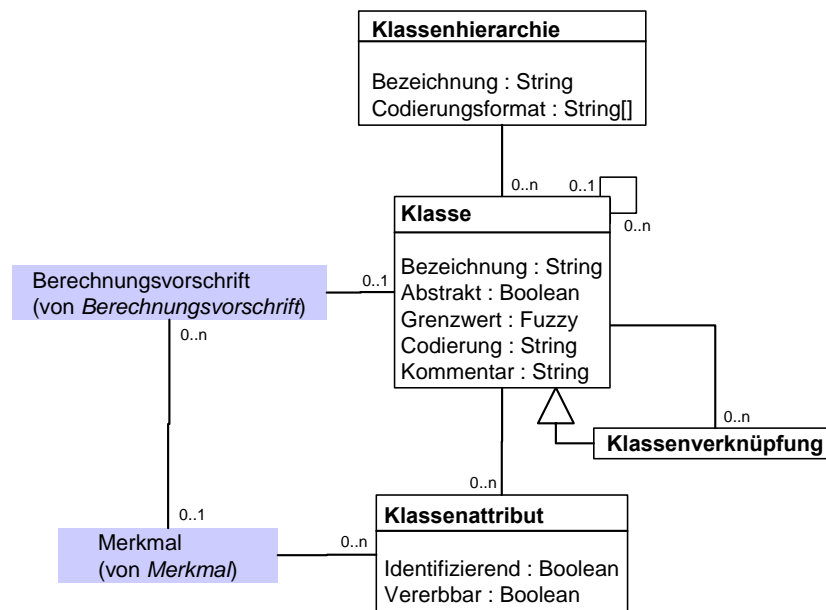


Bild 3-10: Informationsmodell zur Abbildung von Klassen und Klassenhierarchien

Eine Klassenhierarchie ist aus beliebig vielen Klassen aufgebaut, jede Klasse besitzt beliebig viele Unterklassen. An Stelle einer „echten“ Klasse, kann auch eine *Klassenverknüpfung* in die Hierarchie eingefügt werden. Eine Klassenverknüpfung besitzt alle Eigenschaften einer Klasse, verweist aber tatsächlich auf eine bereits an einer anderen Stelle definierten Klasse. Auf diese Weise kann eine einmal definierte Klasse beliebig oft in verschiedenen Hierarchien referenziert werden.

Einer Klasse können beliebig viele *Klassenattribute* zugeordnet werden, die jeweils mit genau einem Merkmal verknüpft sind. Ein Klassenattribut kann als identifizierend markiert werden. Die Vererbung eines Klassenattributes an die Unterklassen kann unterdrückt werden.

### 3.3 Aufbau von unternehmensspezifischen Klassifikationssystem

#### 3.3.1 Installation und Starten der KLAUSTER-Software

Mit diesem Buch wurde eine CD mitgeliefert, die eine Demo-Version der KLAUSTER-Software enthält. Diese Version ermöglicht die automatische Klassifikation von Daten, die in verschiedenen SQL-Datenbanksystemen (MS-Access, mSQL), MS-Excel-Dokumenten, XML-Dateien oder in beliebigen ODBC-Datenquellen gespeichert sein können. Schnittstellen zu weiteren Nutzdatenquellen wie z.B. PDM-Systemen, ERP-Systemen oder CAD-Systemen sind in der Demo-Version nicht enthalten.

Es ist zu beachten, dass es sich bei der Software um einen nicht kommerziellen Software-Prototypen handelt, der im Rahmen eines Forschungsprojektes entstand. Jede Haftung für Schäden oder Datenverluste, die sich mittelbar oder unmittelbar aus dem Einsatz der Software ergeben, wird ausgeschlossen.

Die Software wurde optimiert für den Einsatz unter dem Betriebssystem Microsoft Windows 2000, des weiteren wurde sie getestet unter den Betriebssystemen Microsoft Windows NT und XP. Der Einsatz der Software unter Unix-Betriebssystemen ist grundsätzlich möglich, erfordert jedoch Anpassungen an der Installation und wird von der Demoversion nicht unterstützt. Als Minimalkonfiguration zum Betrieb der Software wird ein Intel-kompatibles System mit einer Taktfrequenz von 500 MHz und 128 MB Arbeitsspeicher vorausgesetzt.

#### Installation/Deinstallation

1. Starten sie das Programm KLAUSTER\_DEMO\_R4.exe. Entpacken Sie dieses selbstextrahierende Archiv in ein beliebiges Verzeichnis. Im Installationspfad dürfen keine Leerzeichen enthalten sein.
2. Im Unterverzeichnis /KLAUSTER\_DEMO\_R4/Klassifikator/ finden Sie die Startdatei klaster.bat.
3. Deinstallation: Um die Software vollständig von Ihrem System zu entfernen, löschen Sie das Installationsverzeichnis. Das Programm nimmt keine Einträge in Windows-Systemdateien wie der Registry vor.

#### Start

1. Starten sie die KLAUSTER-Software durch einen Doppelklick auf die Datei /KLAUSTER\_DEMO\_R4/Klassifikator/klaster.bat.
2. Die Software wird ohne Metadatendatei initialisiert. Über das Menü „Datei“ → „Öffnen“ können Sie die Demo-Metadatendatei „DEMO.pak“ öffnen, oder Sie können über „Datei“ → „Speichern unter“ eine neue Metadatendatei anlegen, indem Sie im „Öffnen“-Dialog die Bezeichnung der neuen Metadaten-Datei eingeben und auf „Öffnen“ klicken.



### 3.3.2 Anwendungsszenario

Nach dem Start der KLAster-Software wie in Abschnitt 3.3.1 beschrieben binden Sie zunächst die Excel-Datei

```
\KLAster_DEMO_R4\Klassifikator\data\userDB\Excel_Verb\3-11-12a.xls
```

als neue Nutzdatenquelle an (vgl. Bild 3-11). Diese Exceltabelle beinhaltet Materialstammdaten über Verbindungselemente, die aus einem SAP R/3-System ausgeleitet wurden.

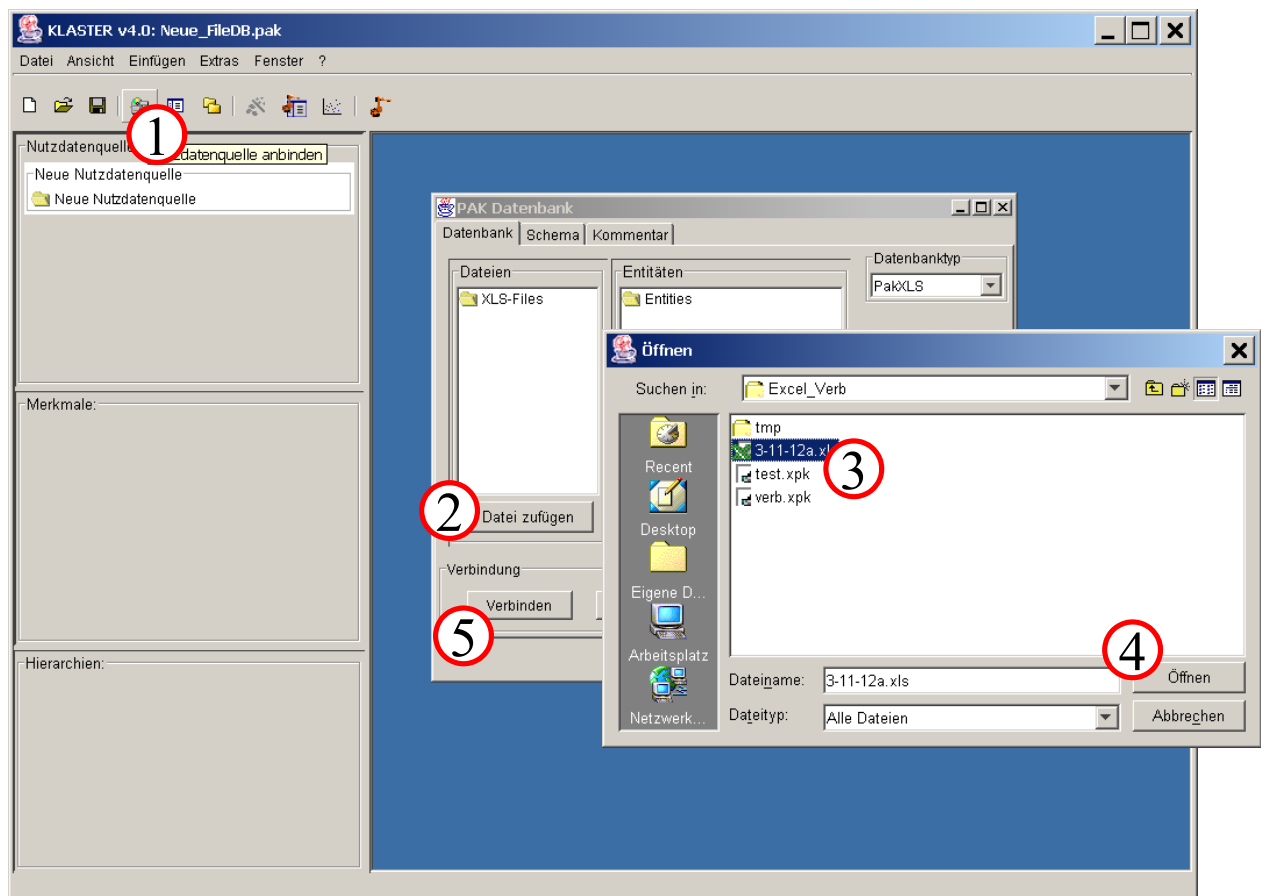


Bild 3-11: Nutzdatenquelle anbinden

Nach dem Einfügen der Datenquelle generieren Sie die ersten Merkmale mittels (Menü „Extras“ → „Initiale Merkmalsgenerierung“). Weiterhin können Sie den Schlüsselwortgenerator konfigurieren (Menü „Extras“ → Schlüsselwortgenerator konfigurieren“, siehe Bild 3-12).

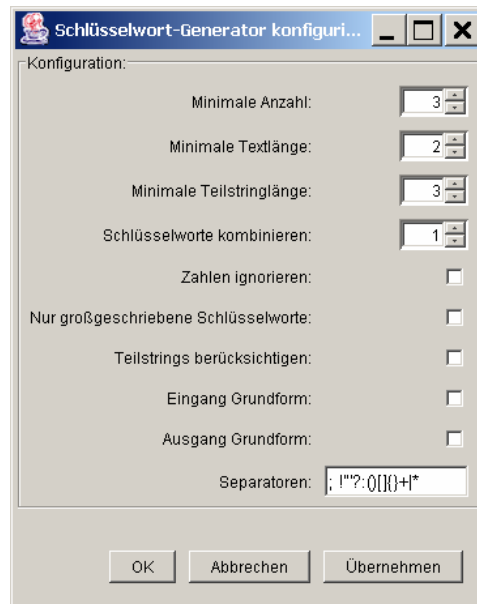


Bild 3-12: Schlüsselwortgenerator konfigurieren

Anschließend leiten Sie für das Merkmal „Norm“ Schlüsselmerkmale ab (siehe Bild 3-13) und speichern die Metadaten mittels Menü „Datei → „Speichern unter“ → z.B. test.pak.

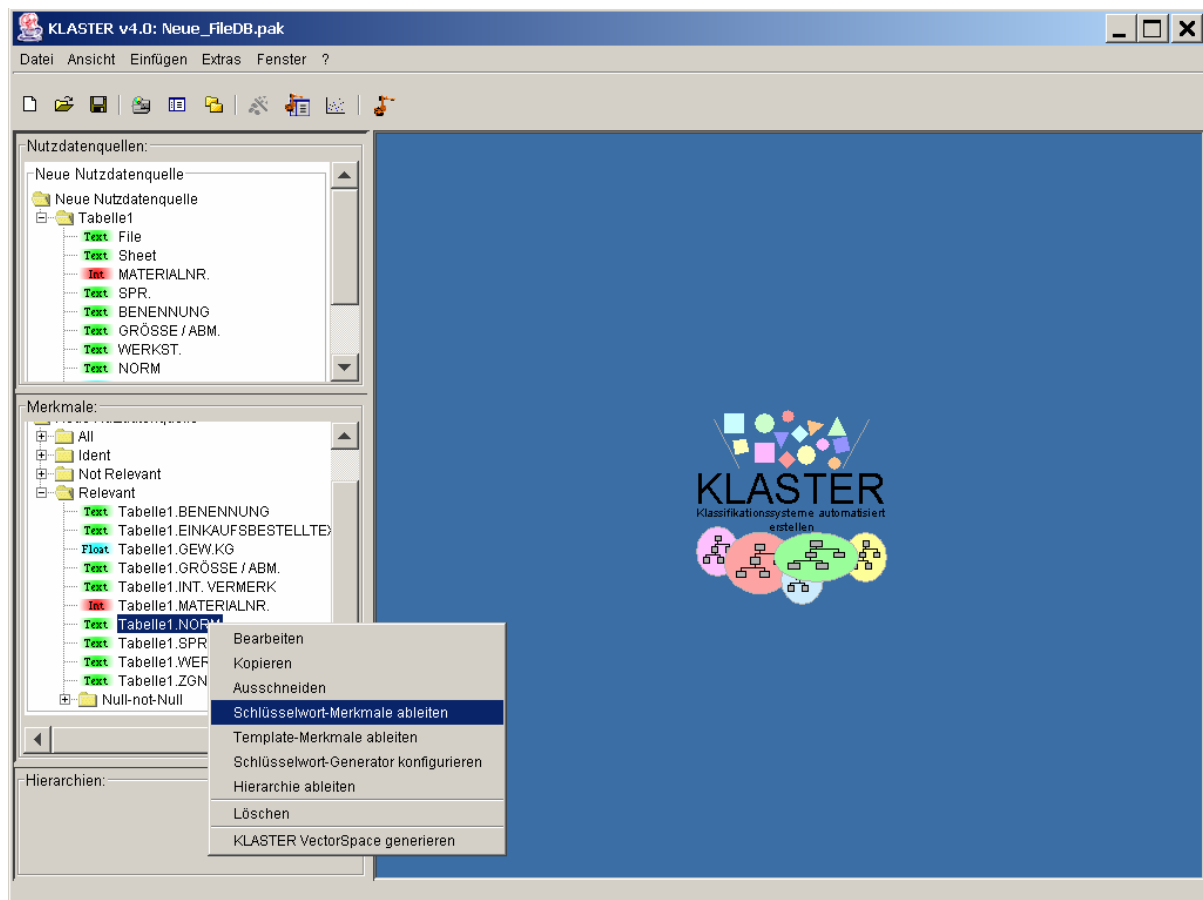


Bild 3-13: Schlüsselwortmerkmale ableiten

Mit „Extras“ → „Vektorraum erzeugen“ können Sie einen Vektorraum generieren, in dessen Fenster sie per Drag & Drop das Merkmal „Materialnr.“, das Merkmal „Benennung“ und den Ordner der Schlüsselwortmerkmale ziehen. Markieren Sie „Materialnr.“ als identifizierend. Die Cluster-Analyse können Sie über den Button „Cluster“ starten. Das Ergebnis wird anschließend dargestellt.

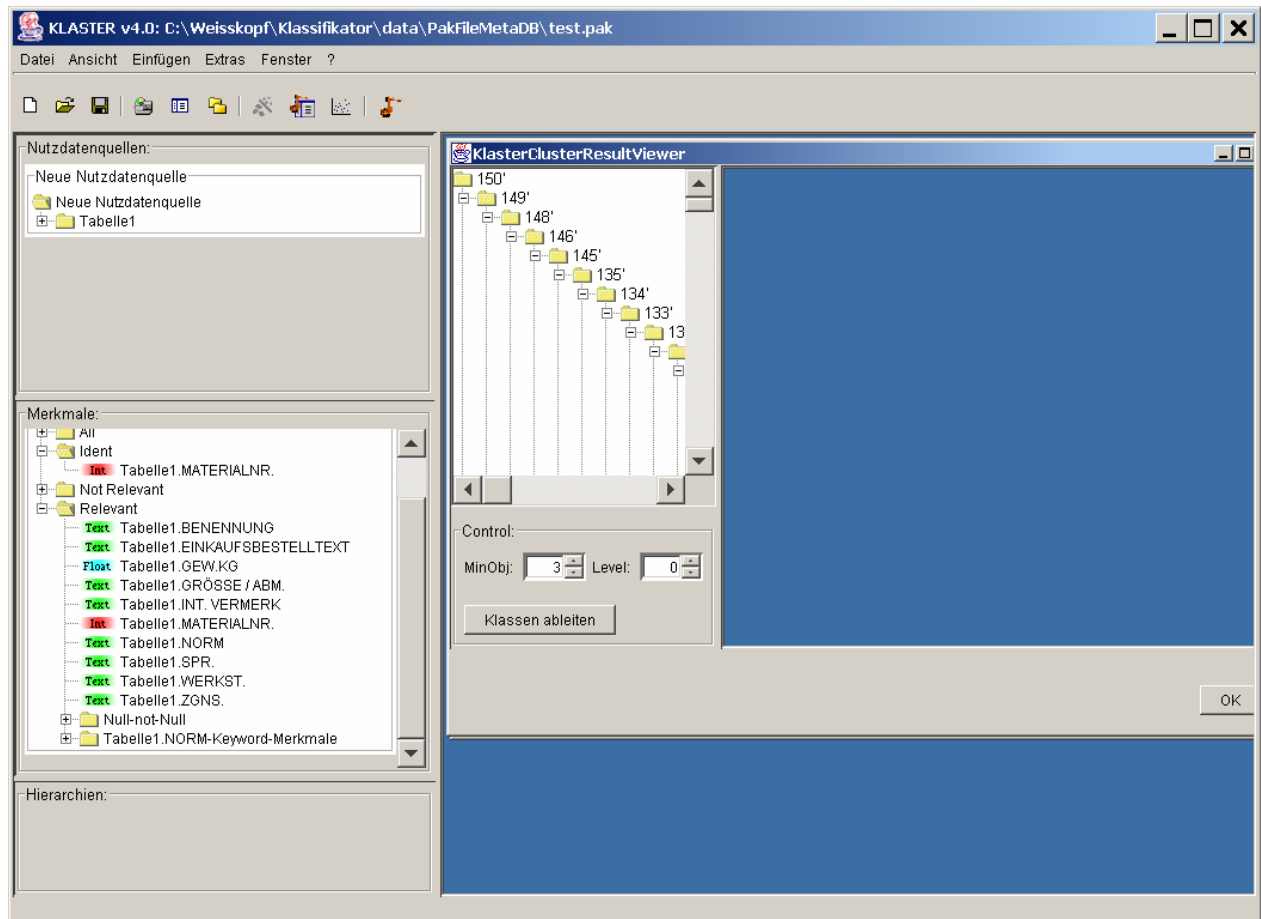


Bild 3-14: Ergebnis der Cluster-Analyse

Abschließend leiten Sie die Klassenhierarchie ab, indem Sie den Knopf „Klassen ableiten“ klicken, und klassifizieren Sie die Nutzdatenquelle gemäß dieser Hierarchie mit „Extras“ → „Klassifikation durchführen“.

Sie können analog diesem einfachen Szenario für andere alphanumerische Merkmale, z.B. der Benennung, Schlüsselwortmerkmale generieren, Cluster-Analysen durchführen (auch unter Verwendung weiterer Merkmale wie z.B. dem Gewicht) und Klassenhierarchien generieren. Ein vorgefertigtes Beispiel finden Sie in der Metadaten-Datei DEMO.pak.


### 3.3.3 Nutzdatenquelle anbinden

Zu Beginn der Erstellung eines Klassifikationssystems steht die Anbindung der Nutzdatenquellen, die die zu klassifizierenden Datenbestände beinhalten. Die Nutzdatenquellen können unterschiedlichste IT-Systeme sein, die Produktdaten aus verschiedenen Firmenabteilungen beinhalten. Schnittstellen für eine Vielzahl unterschiedlicher IT-Systeme sind verfügbar, wie z.B. für mSQL, Excel\_Dateien, PDM-Systeme, CAD-Systeme, etc. Diese Nutzdaten werden zur Erstellung der Metadaten verwendet.

#### 3.3.3.1 Typ der Nutzdatenquelle auswählen

Die zu klassifizierenden Nutzdatenquellen werden im Hauptfenster der KLAUSTER-GUI im Register *Nutzdatenquellen* verwaltet. Über das Menü „Einfügen“ können Nutzdatenquellen hinzugefügt werden. Über ein Kontextmenü (rechte Maustaste auf einer Nutzdatenquelle) können Nutzdatenquellen bearbeitet oder gelöscht werden.

#### Vorgehensweise:

Um Daten als Nutzdatenquelle einzubinden, klicken Sie auf „Einfügen“ und anschließend auf „Neue Nutzdatenquelle“. Dadurch wird eine neues Fenster „PakDatabase“ geöffnet. Ein Klick auf das Icon  bewirkt dieselbe Operation.

Das Bearbeitungsfenster einer Nutzdatenquelle besitzt drei Register: *Database*, *Schema* und *Kommentar*. Im Register **Database** werden die Verbindungsparameter für eine Nutzdatenquelle spezifiziert. Zunächst muss über das Auswahlfeld *Datenbanktyp* der Typ der anzubindenden Nutzdatenquelle festgelegt werden: mSQL-Datenbank, Excel-Dateien, PDM-System, CAD-System, etc.

Als Beispiel wählen Sie den Datenbanktyp „PakXLS“, wodurch sich im Register „Database“ das Fenster ändert.

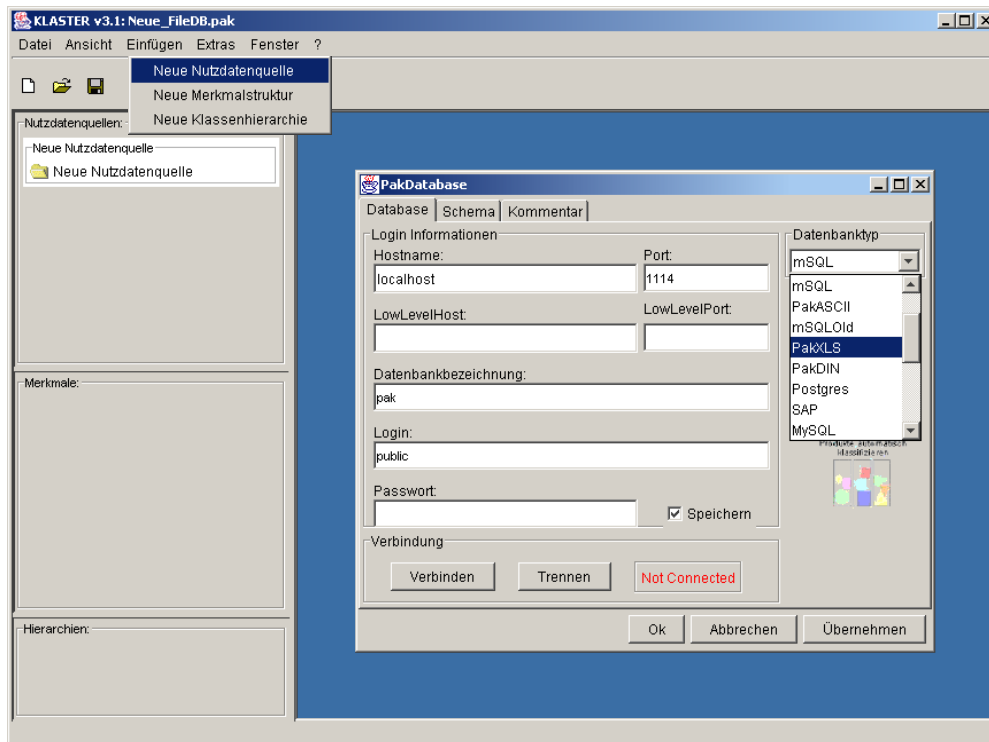


Bild 3-15: Typ der Nutzdatenquelle auswählen

Über den Button „Add Files“ können Sie die Daten anbinden. Es öffnet sich ein PopUp-Fenster, in dem Sie die anzubindenden Dateien auswählen können. Mehrere Dateien können nacheinander als Nutzdatenquelle hinzugefügt werden. Wenn alle Verbindungsparameter spezifiziert sind, stellen sie die Verbindung zu dieser Nutzdatenquelle her, indem Sie den Butten „Verbinden“ klicken.

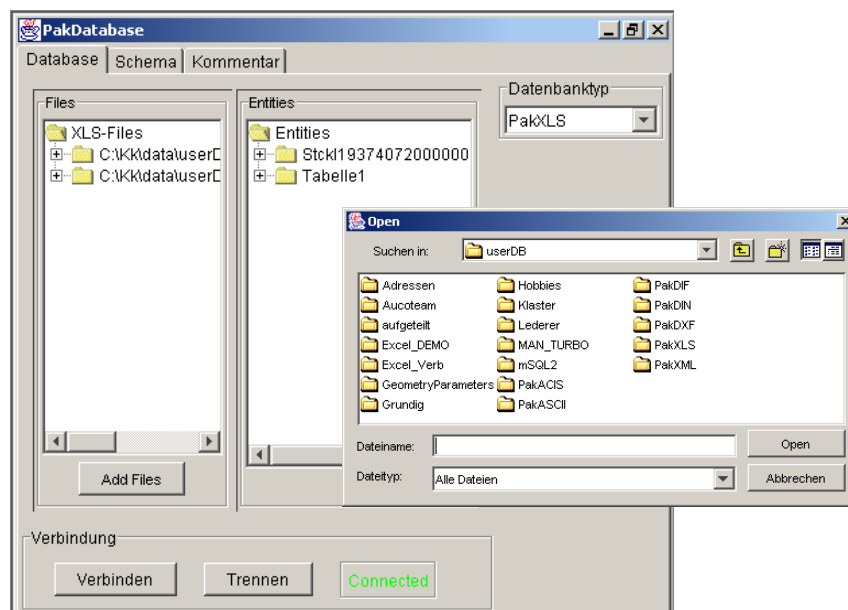


Bild 3-16: Anbindung der Nutzdaten

In Bild 3-19 ist das Register **Schema** einer Nutzdatenquelle dargestellt. Im Eingabefeld „Datenbankbezeichnung“ kann eine beliebige, aber eindeutige Bezeichnung der Nutzdatenquelle angegeben werden. Diese Bezeichnung wird im Register *Nutzdatenquellen* der KLAUSTER-GUI eingetragen.

Durch das Betätigen der Schaltfläche „Schema laden“ wird die relationale Sicht auf die Datenstruktur der Nutzdatenquelle automatisch ermittelt und in einer Baumstruktur dargestellt. Die Ordner repräsentieren hierbei die Entitäten, die Elemente in den Ordnern die den Entitäten zugeordneten Nutzdatenattribute. (Beim ersten Verbindungsaufbau zu einer Nutzdatenquelle das Schema automatisch geladen.) Vergessen Sie nicht, Änderungen zu speichern.

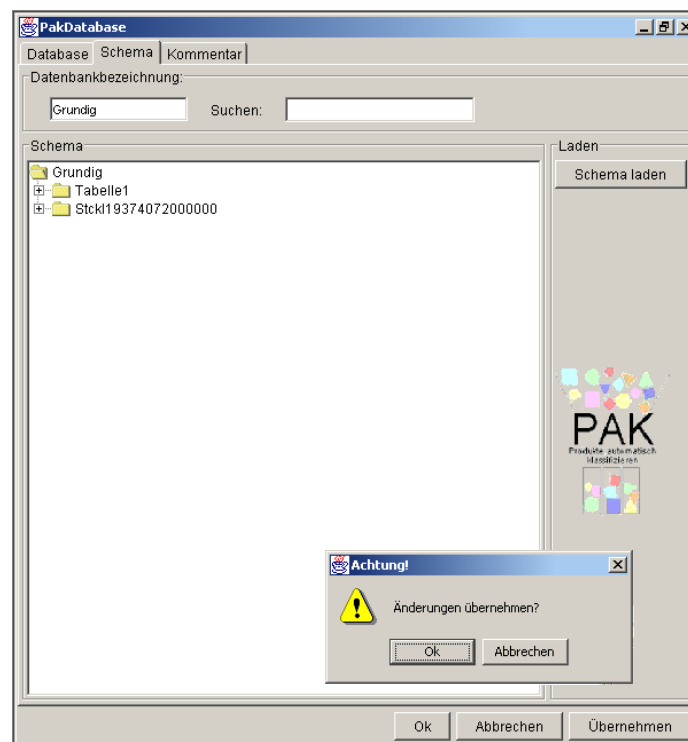


Bild 3-17: Laden des Schemas

### 3.3.3.2 Zugriffsparameter spezifizieren

Die Zugriffsparameter werden abhängig vom Typ der Nutzdatenquelle spezifiziert. Am Beispiel Excel müssen lediglich die anzubindenden Excel-Dateien angegeben werden. Für SQL-Datenbanken beispielsweise ist die Angabe weiterer Parameter erforderlich.

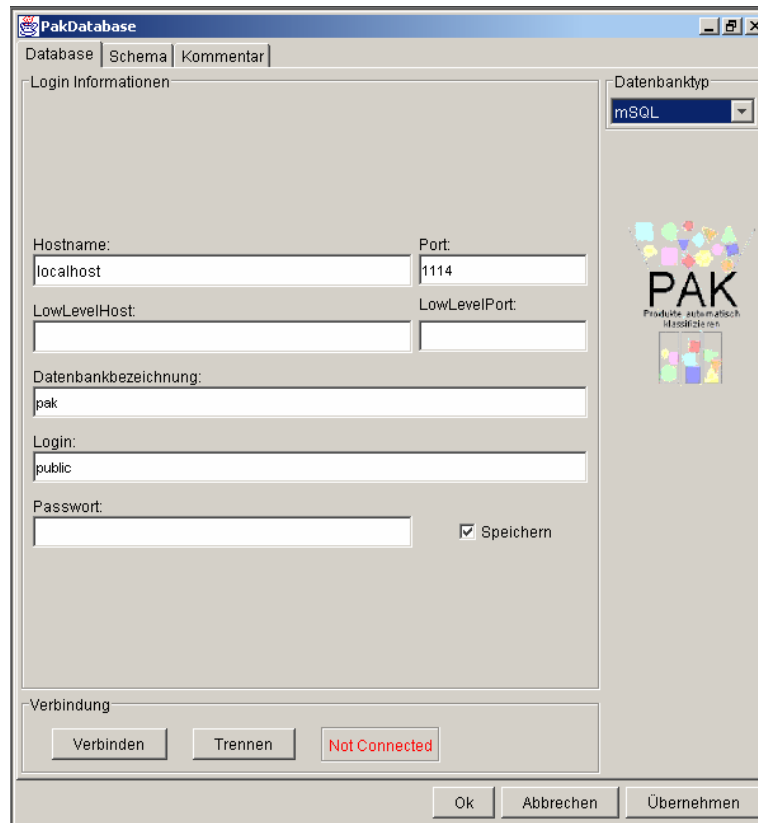


Bild 3-18: Verwaltung von Nutzdatenquellen

Im Eingabefeld *Hostname* wird die Bezeichnung des Rechners in einem Netzwerk eingetragen, über den der Zugriff auf die Nutzdatenquelle im Netzwerk direkt möglich ist. Bei SQL-Datenbanksystemen läuft auf diesem Rechner in der Regel ein „daemon“-Prozess, der einen bestimmten „Port“ (eine vierstellige Ziffernfolge) überwacht, über den Anfragen an das Datenbanksystem bearbeitet werden. Dieser *Port* ist im entsprechenden Eingabefeld einzutragen.

Viele Nutzdatenquellen, insbesondere diejenigen, bei denen die Daten in Dateien gespeichert werden, sind in der Regel nicht dazu vorgesehen, die Daten in einem Netzwerk zur Verfügung zu stellen (z.B. Excel, XML, CAD-Systeme). Sind diese Dateien tatsächlich auf dem lokalen Rechner gespeichert, auf dem auch das Metadatenmodul installiert ist, so ist im Eingabefeld *Hostname* als Rechnername ‚localhost‘ einzutragen (dies gilt auch für lokal installierte Datenbanksysteme). Sind solche Dateien jedoch auf einem anderen Rechner im Netzwerk gespeichert, so ist der Zugriff über einen PAK-LowLevelServer möglich, der auf dem entsprechenden Rechner installiert sein muss. In diesem Fall wird als *Hostname* ‚localhost‘, als *Port* ‚0000‘, als *LowLevel-Host* der entsprechende Rechner, auf dem die Dateien gespeichert sind, und als *LowLevel-Port* standardmäßig der Wert ‚5001‘ eingetragen.

Im Eingabefeld *Datenbankbezeichnung* wird die eindeutige Benennung der Nutzdatenquelle eingetragen. Für Datei-basierte Nutzdatenquellen (z.B. XML) ist dies entweder das Verzeichnis auf der Festplatte, in dem die Dateien gespeichert sind, oder es wird der Pfad zu einer sogenannten „Master“-Datei angegeben. Eine Master-Datei ist eine ASCII-Datei, in der alle Dateien eingetragen werden, die zu dieser Nutzdatenquelle gehören. Die Spezifikation eines

Verzeichnisses bzw. einer Master-Datei erfolgt über einen Datei-Auswahldialog, der durch das Anklicken des Eingabefeldes *Datenbankbezeichnung* aktiviert wird.

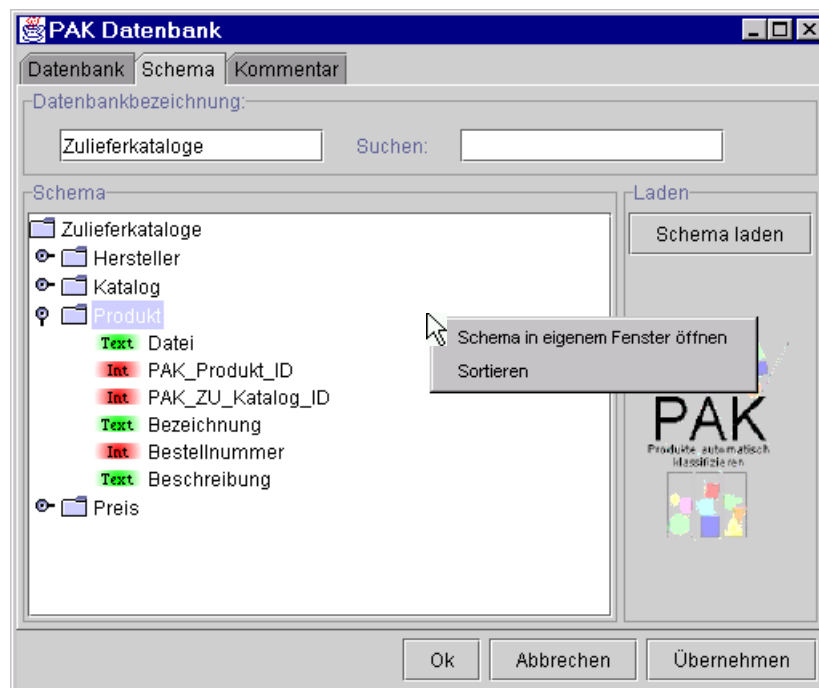


Bild 3-19: Schema einer Nutzdatenquelle

Im Eingabefeld *Suchen* kann eine beliebige Zeichenfolge eingegeben werden. Durch Betätigung der Enter-Taste wird die Baumstruktur nach dieser Zeichenfolge durchsucht. Über ein Kontextmenü kann die Baumstruktur in einem eigenen Fenster dargestellt sowie alphabetisch sortiert werden.

Im Register **Kommentar** kann ein beliebiger, die Nutzdatenquelle beschreibender Text hinterlegt werden.

### 3.3.3.3 Relationen verifizieren/modifizieren

Über Relationen wird festgelegt, wie die Entitäten einer Nutzdatenquelle oder auch Entitäten verschiedener Nutzdatenquellen miteinander in Beziehung stehen. Z.B. werden in einer SQL-Datenbank in zwei Tabellen Personen und Firmen verwaltet, jeweils mit einer eindeutigen ID und verschiedenen Attributen. Eines der Attribute einer Person ist die ID der Firma, in der sie arbeitet. Dann lautet die Relation zwischen den beiden Entitäten „Person“ und „Firma“: `Person.FirmenID ↔ Firma.ID`

Im Bearbeitungsfenster „Relationen“ (zu öffnen über das Menü „Ansicht“) finden Sie ggf. vordefinierte Relationen, die von der KLAUSTER-Software automatisch ermittelt wurden. In diesem Fenster können Sie die Relationen verifizieren bzw. modifizieren/löschen und neue Relationen definieren.



Um die Relationen verifizieren / modifizieren zu können, markieren Sie eine Relation und klicken Sie auf „Bearbeiten“. Im eingeblendeten Bearbeitungsfenster können Sie den Relationsname editieren oder mit Drag & Drop Nutzdatenattribute hinzufügen (in obigem Beispiel die Nutzdatenattribute „Person.FirmenID“ und „Firma.ID“).

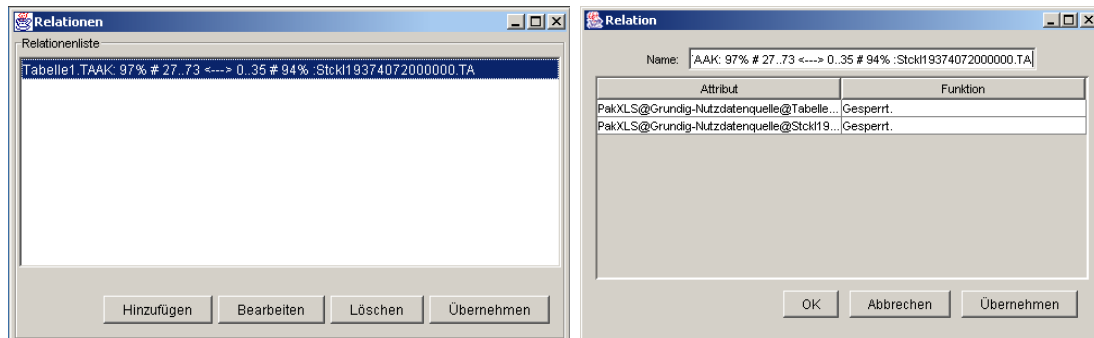



Bild 3-20: Definierte Relationenverifizieren/modifizieren

### 3.3.4 Merkmalsgenerierung

Die zweite Phase beinhaltet die Merkmalsermittlung. Im ersten Schritt werden alle explizit in den Nutzdatenquellen verfügbaren Merkmale ermittelt. Anschließend werden diejenigen Merkmale ermittelt, die einen identifizierenden Charakter haben, d.h. deren Ausprägungen ein zu klassifizierendes Objekt eindeutig identifizieren.

#### 3.3.4.1 Nutzdatenquelle(n) auswählen

Klicken Sie auf das Icon , um die zu analysierende Daten auszuwählen, und anschließend auf „OK“.

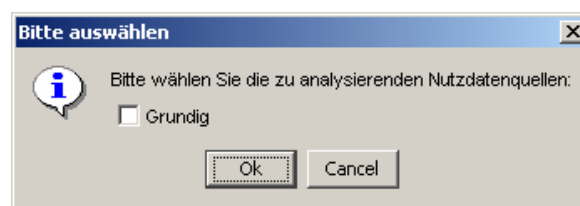


Bild 3-21: Auswahl der Nutzdatenquelle(n)

#### 3.3.4.2 Identifizierende(s) Merkmal(e) verifizieren/modifizieren

Das/die Identifizierende(n) Merkmal(e) wird/werden von der KLAUSTER-Software automatisch identifiziert bzw. vorgeschlagen. Über das Kontextmenü eines identifizierten Merkmals wird das Fenster zum Verifizieren bzw. Modifizieren des Merkmals geöffnet.

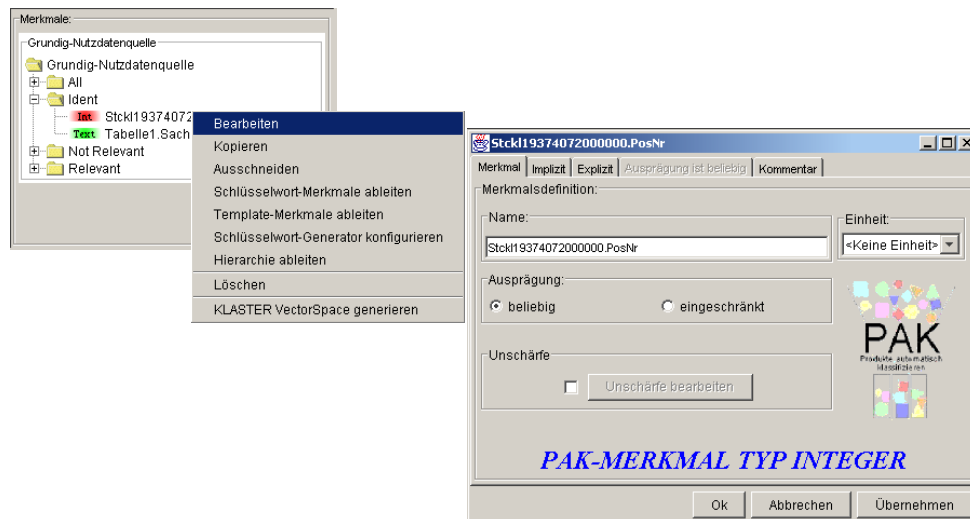


Bild 3-22: Verifizierung/Modifizierung des identifizierenden Merkmals

### 3.3.4.3 Klassifikationsrelevante Merkmale verifizieren/festlegen

Die KLASER-Software identifiziert automatisch klassifikationsrelevante Merkmale. In der Baumstruktur der Merkmale finden Sie relevante Merkmale, die von der KLASER-Software vorgeschlagen werden.

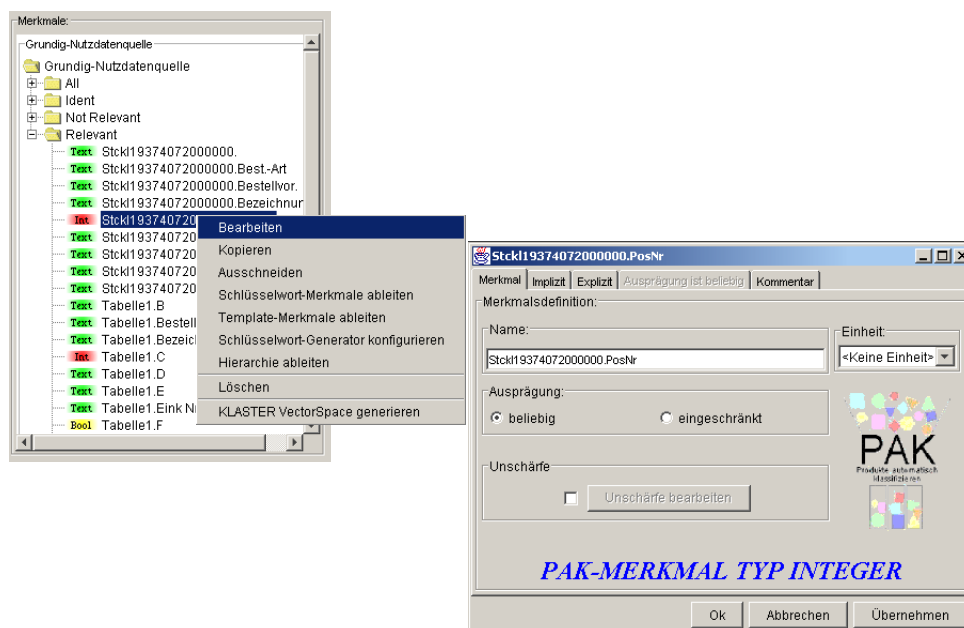


Bild 3-23: Verifizierung der klassifikationsrelevanten Merkmale

Aus diesen gezeigten Merkmalen verifizieren und legen Sie fest, welche Merkmale beim Erstellen des Klassifikationssystems mit einbezogen werden sollen. Hier können Sie auch die Merkmalsdefinitionen bearbeiten.

### 3.3.4.4 Relevante Null/NOT-Null-Merkmale verifizieren/festlegen

Null/NOT-Null-Merkmale sind Merkmale, für die teilweise keine Werte eingetragen worden sind. Die KLAUSTER-Software identifiziert automatisch Null/NOT-Null-Merkmale. Sie können sie verifizieren und bearbeiten wenn nötig. Danach können Sie festlegen, welche Merkmale zu berücksichtigen sind.

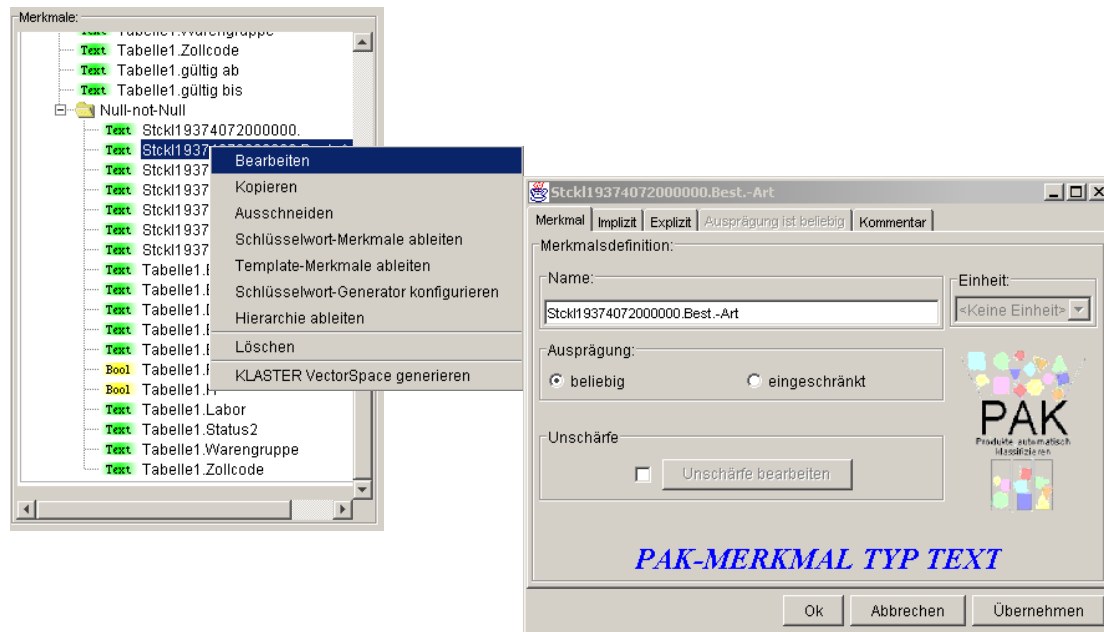


Bild 3-24: Null/NOT-Null-Merkmale verifizieren/festlegen

### 3.3.4.5 Merkmale für Schlüsselwortgenerierung auswählen

In diesem Schritt werden Schlüsselwörter und auf diesen aufbauend boolsche Merkmale generiert. Ein Beispiel hierfür wäre das boolsche Merkmal „Norm enthält DIN“. Dieses Merkmal erhält dann für ein zu klassifizierendes Objekt den Wert „true“, wenn das Feld „Norm“ für dieses Objekt das Schlüsselwort „DIN“ enthält.

### 3.3.4.6 Konfiguration des Schlüsselwortgenerators

Zur Generierung der Schlüsselwörter werden zunächst deren Konfiguration bzw. Bedingungen festgelegt.

Wählen Sie über das Kontextmenü eines Merkmals „Schlüsselwort-Generator konfigurieren“. Alternativ ist diese Funktion auch über das Menü „Extras“ erreichbar. Im geöffneten PopUp-Fenster können Sie die Konfiguration des Schlüsselwortgenerators festlegen.

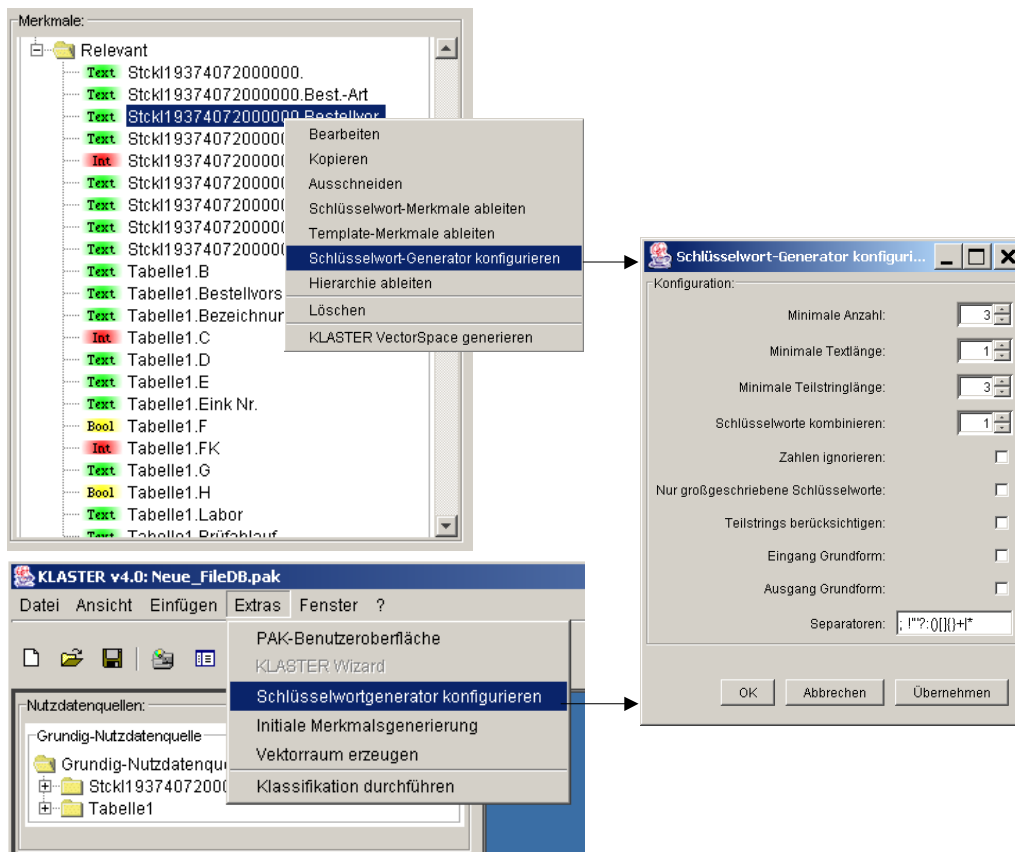


Bild 3-25: Schlüsselwortgenerator konfigurieren

Der Schlüsselwortgenerator bietet mehrere Konfigurationspunkte an. Sie können die minimale Anzahl an Vorkommnissen angeben, damit ein Schlüsselwort berücksichtigt wird. Die minimale Textlänge legt fest, wie viele Zeichen ein Schlüsselwort mindestens haben muss. Wenn die Checkbox „Teilstrings berücksichtigen“ markiert ist, können Sie in „Minimale Teilstringlänge“ eintragen, wie viele Zeichen ein Teilstring mindestens haben muss, um als Schlüsselwort in Betracht zu kommen. Häufig vorkommende Schlüsselworte (z.B. „DIN“ und „2510“) können zu einem booleschen Merkmal kombiniert werden. Dabei werden nur so viele Schlüsselwortkombinationen zusammengefasst, wie maximal angegeben. Weiterhin existieren Optionen, ob Zahlen oder kleingeschriebene Wörter als Schlüsselworte ignoriert werden sollen.

Wie oben erwähnt gibt es die Möglichkeit, dass auch Teilstrings berücksichtigt werden. Dabei wird versucht, auch häufig vorkommende Teilstrings innerhalb identifizierter Schlüsselworte als eigene Schlüsselworte zu erkennen (Bsp.: „Sechskantschraube“, „Zylinderschraube“ wurden als Schlüsselworte erkannt → „schraube“ wird ebenfalls Schlüsselwort).

Für eine bessere Übersicht können die zu analysierenden Strings sowie die generierten Schlüsselworte in ihre Grundform überführt werden. Für Verben gilt der Infinitiv, für Substantive Nominativ Singular. Abschließend teilen die Separatoren einen String in die „Worte“ ein, die als Schlüsselworte in Frage kommen.

### 3.3.4.7 Relevante Schlüsselworte auswählen

Wenn Sie über das Kontextmenü der Merkmale „Schlüsselwort-Merkmale ableiten“ auswählen, werden die abgeleiteten Schlüsselwortmerkmale in einem neuen Unterordner angelegt. Die Konfiguration des Schlüsselwort-Generators legt dabei fest, welche Schlüsselworte abgeleitet werden. Wenn nötig, können Sie Schlüsselworte neu ableiten, dafür wiederholen Sie den Schritt 2.5.1.

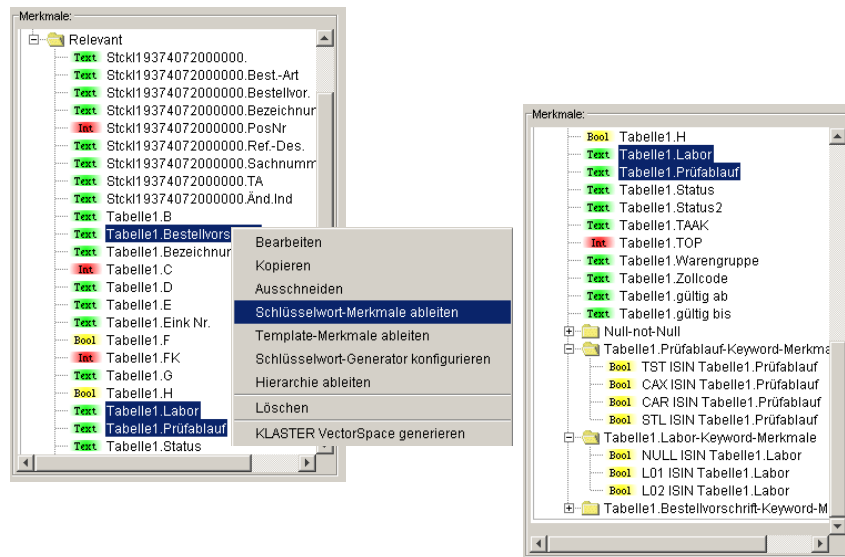


Bild 3-26: Auswahl der relevanten Schlüsselworte

### 3.3.4.8 Merkmale für die Extraktion numerischer Anteile auswählen

Beispiel: Größe/Abmessung M 10 x 46, M 12 x 54, ... → Extraktion von Länge und Durchmesser. Derzeit noch nicht automatisiert, entsprechende Merkmale können manuell angelegt und mit einer impliziten Regel versehen werden.

### 3.3.4.9 Merkmale zur Generierung mathematischer Verknüpfungen auswählen

Beispiel: Länge, Durchmesser → Länge/Durchmesser-Verhältnis. Derzeit noch nicht automatisiert, entsprechende Merkmale können manuell angelegt und mit einer impliziten Regel versehen werden.

### 3.3.4.10 Weitere implizite Merkmale manuell generieren

Für die klassifikationsrelevanten Merkmale werden anschließend weitere Merkmale ermittelt, die nicht explizit in den Datenquellen verfügbar sind. Diese impliziten Merkmale werden durch unterschiedliche Verknüpfungsoperationen aus bereits verfügbaren Merkmalen definiert.

Um weitere implizite Merkmale zu erstellen, legen Sie ein neues Unterverzeichnis, in dem diese gespeichert werden sollen, über das Kontextmenü des obersten Verzeichnisses der Merkmalbaumstruktur an. Im Kontextmenü des neuen Verzeichnisses können Sie wählen, welche Art von Merkmal Sie hinzufügen wollen.

Es wird das Bearbeitungsfenster „Neues Merkmal“ geöffnet, das vier Register (Merkmal, Implizit, Explizit und Kommentar) besitzt.

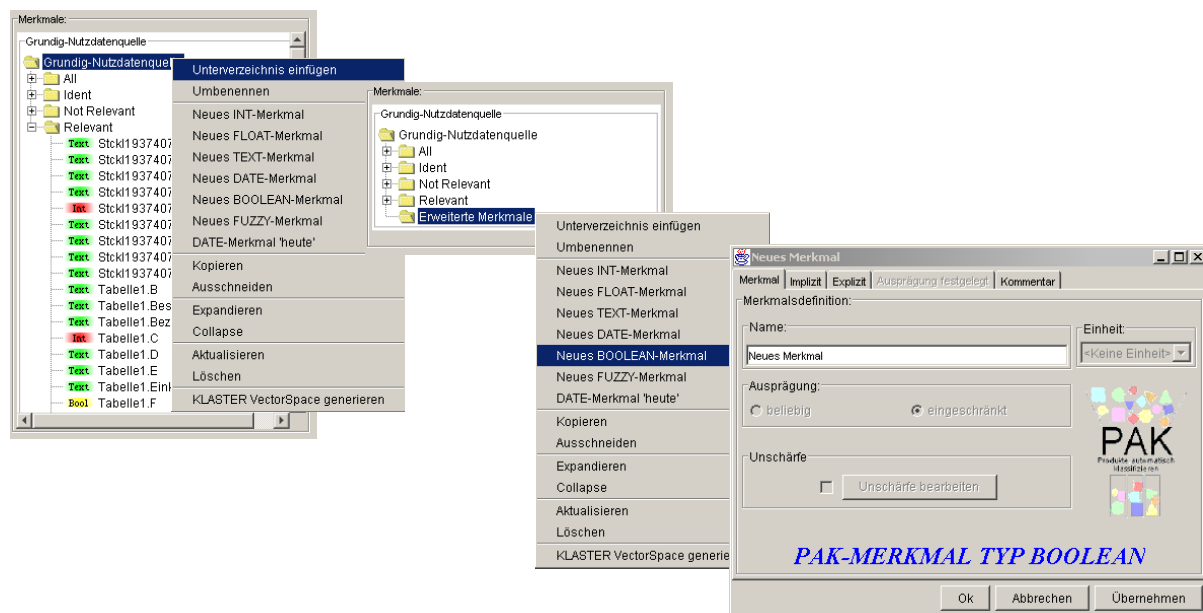


Bild 3-27: Erstellung eines neuen Verzeichnis für weitere implizite Merkmale

Im Register „Implizit“ fügen Sie dem Merkmal eine Berechnungsvorschrift bzw. Regel über den Button „Zufügen“ hinzu. Mit Drag & Drop ziehen Sie Merkmale aus der Merkmalbaumstruktur in das Bearbeitungsfenster der „Impliziten Regel“. In diesem Beispiel soll diese Regel generiert werden: „FK > 110“. Dafür wird zunächst das Merkmal „FK“ zugefügt. Das Kontextmenü des Fensters bietet die Möglichkeit, den richtigen Vergleichsoperator auszuwählen.

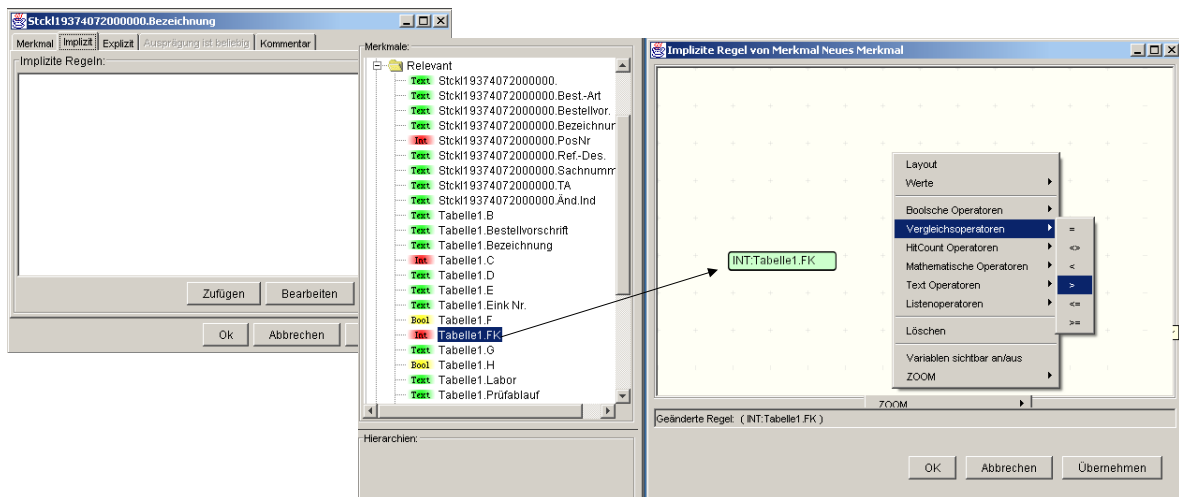


Bild 3-28: Verfügbares Merkmal auswählen

Anschließend definieren Sie den Grenzwert von FK (110) ebenfalls über das Kontextmenü („Werte“ → „Neuer INT Wert“). Nachdem Sie im eingblendeten Textfeld Ihren Wert eingegeben haben, bestätigen Sie die Eingabe mit Enter, und verknüpfen die Merkmale, indem Sie einen Pfeil von jedem Operanden zum Vergleichsoperator ziehen. Abschließend bestätigen Sie die Regel mit „OK“.

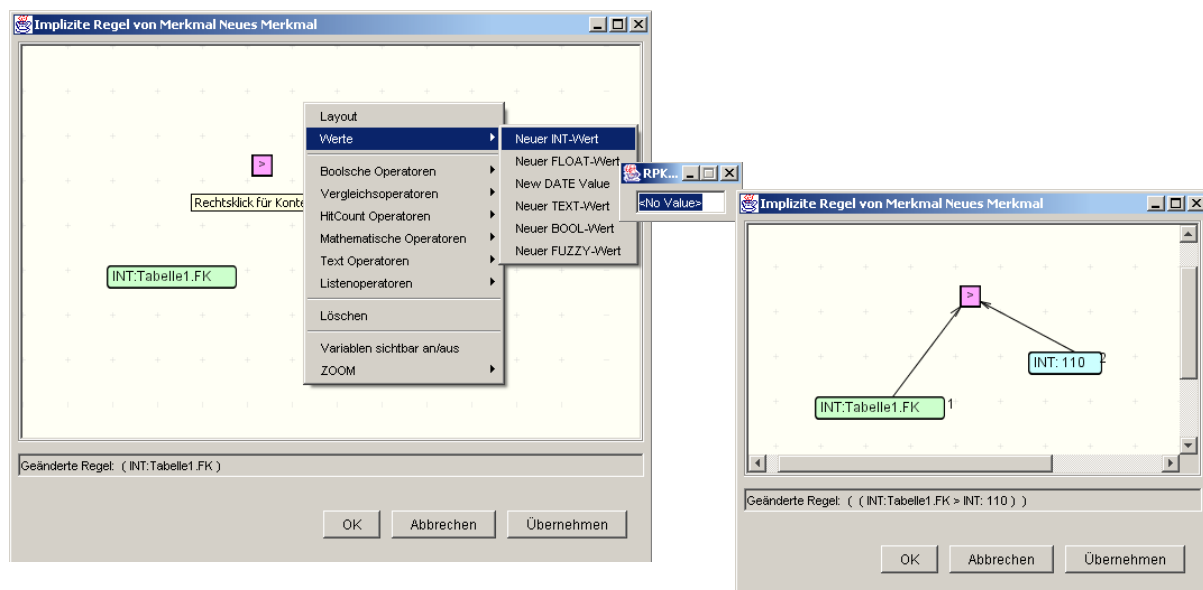


Bild 3-29: Die Regel des impliziten Merkmals generieren

### 3.3.5 Klassensystem generieren und strukturieren

In der dritten Phase werden gemäß der klassischen Cluster-Analyse-Verfahren Vektorräume generiert und Cluster-Analysen durchgeführt. Die Ergebnisse dieser Cluster-Analysen werden aufbereitet, interpretiert und für den Aufbau des Klassifikationssystems (Klassendefinition und hierarchische Strukturierung der Klassen) verwendet.

### 3.3.5.1 Identifizierendes und klassifikationsrelevante Merkmale selektieren

In diesem Schritt werden das identifizierende und die klassifikationsrelevanten Merkmale selektiert und einem VectorSpace hinzugefügt. Die Merkmale im VectorSpace werden danach mit Cluster-Analyse-Algorithmen analysiert. Für die Cluster-Analyse werden nur numerische und boolesche Merkmale berücksichtigt, sonstige Merkmale können dem VectorSpace hinzugefügt werden. Sie werden den generierten Klassen als Sachmerkmale zugefügt.

Über das Kontextmenü eines identifizierenden Merkmals generieren Sie ein „KLASTER VerktorSpace“. Mittels Drag & Drop ziehen Sie alle klassifikationsrelevanten Merkmale in den VectorSpace ein, der in dem eingblendeten Fenster dargestellt ist. Das bereits existierende identifizierende Merkmal können Sie durch ein anderes ersetzen, indem Sie auf dem Textfeld „Identifizierend“ des gewünschten Merkmals klicken. Dieses wird dadurch automatisch an die oberste Position geschoben.

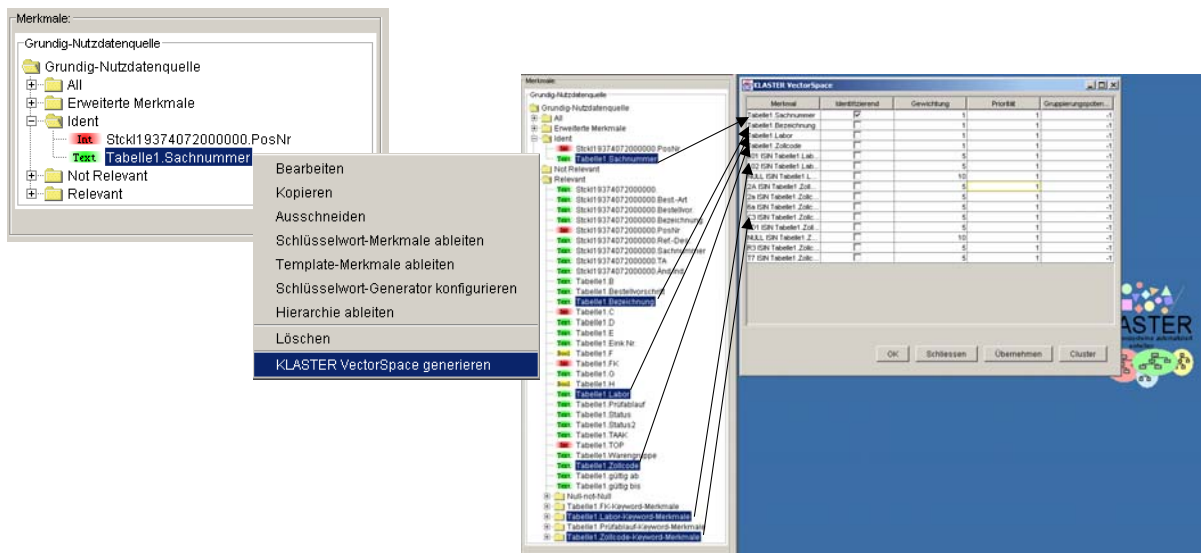


Bild 3-30: Selektion und Einfügen der Merkmale im VectorSpace

### 3.3.5.2 Gewichtungen und Prioritäten verifizieren

Es gibt die Möglichkeit, die Gewichtung sowie die Priorität zu ändern. Die beiden Parameter beeinflussen die zu generierende Klassenstruktur. Je größer die Gewichtung bzw. Priorität desto größer der Einfluss.

Um einen Parameter zu ändern, doppelklicken Sie die entsprechende Zahl, editieren das Feld und bestätigen die Änderung mit Enter.



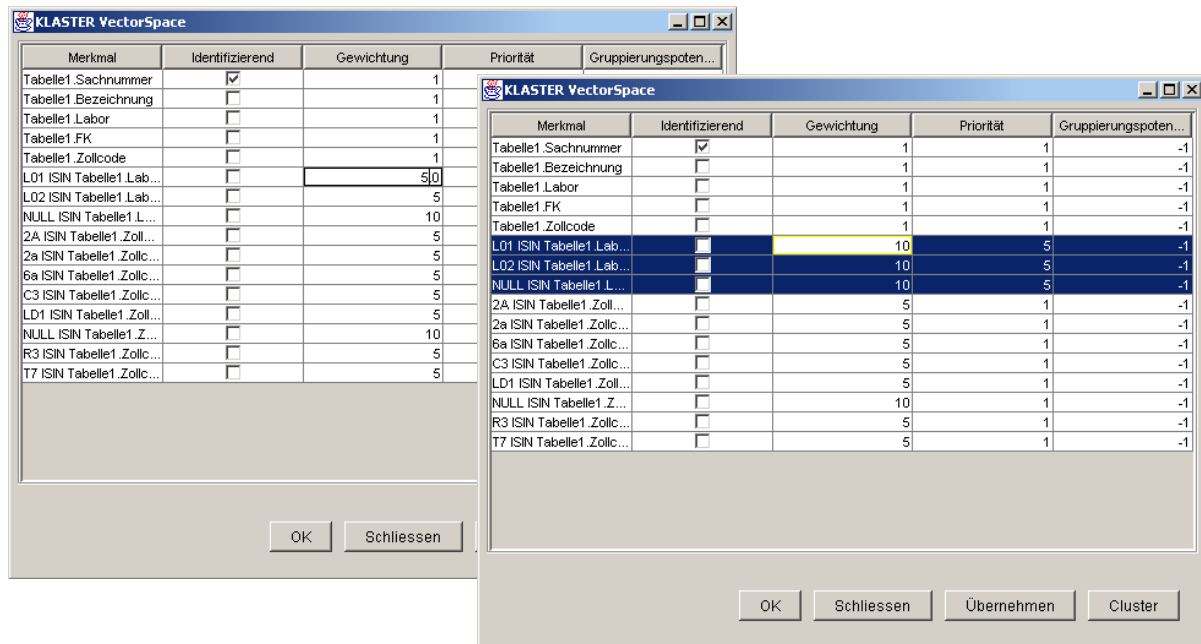


Bild 3-31: Gewichtung und Priorität definieren

### 3.3.5.3 Cluster-Analyse durchführen

Nachdem die Merkmale in den VectorSpace eingefügt wurden, können die Daten mittels einer Cluster-Analyse analysiert werden. Vorher können Sie mittels Kontextmenü (rechte Maustaste) den gesamten Datenbestand oder die Werteverteilung für einzelne Merkmale visualisieren. Wenn Sie im Bearbeitungsfenster KLASTER-VektorSpace den Button „Cluster“ klicken, wird der Analysevorgang gestartet.

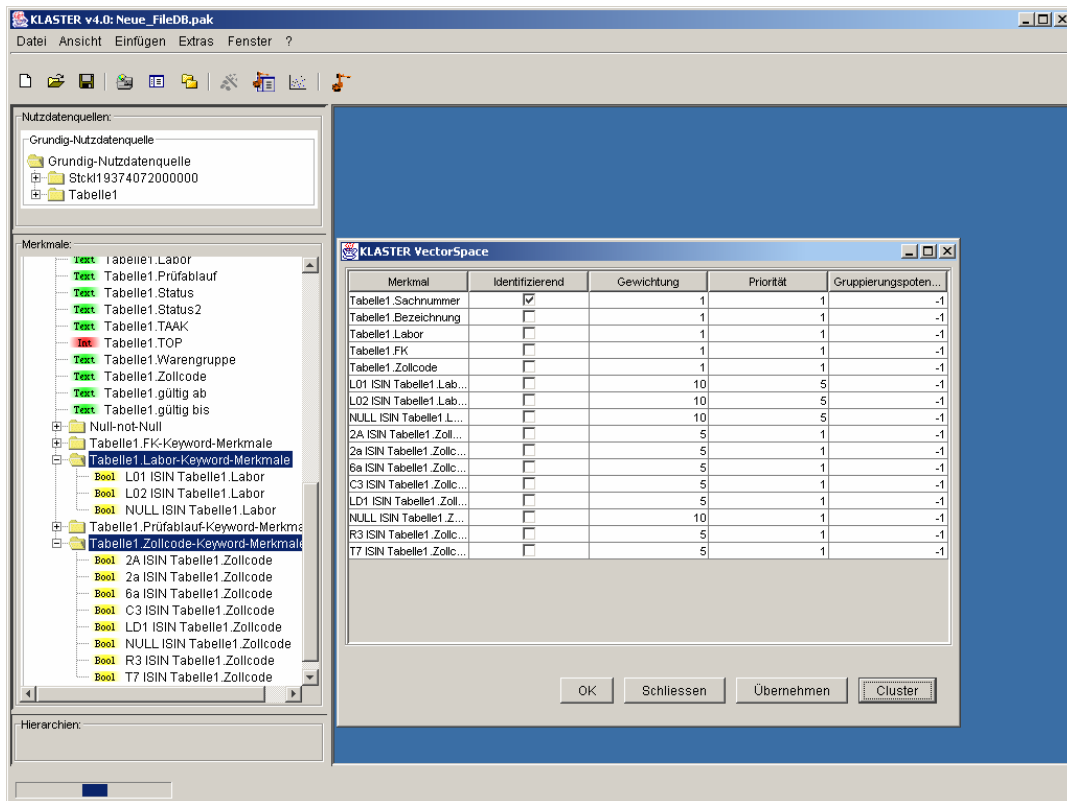


Bild 3-32: Cluster-Analyse durchführen

### 3.3.5.4 MinObj/Level für Klassengenerierung festlegen

Die Ergebnisse der Cluster-Analyse werden in einer Baumstruktur dargestellt (dies ist noch keine Klassenstruktur). Jeder Ordner dieser Baumstruktur repräsentiert einen durch die Analyse erkannten Cluster, die Hierarchie spiegelt die Geschichte der Fusionierung der Cluster wider. Die Darstellungstiefe der Baumstruktur können Sie steuern, indem Sie die Werte für MinObj bzw. Level editieren. Die Zahl des MinObj ist die minimale Anzahl der Objekte in der untersten Position der Baumstruktur. Die Zahl des Levels definiert, wie viele Hierarchieebenen eingeblendet werden.

Sinnvoller Default-Wert für MinObj: 2% der Anzahl der Datensätze, z.B. bei einer Excelta-  
belle mit 500 Zeilen → MinObj = 10.

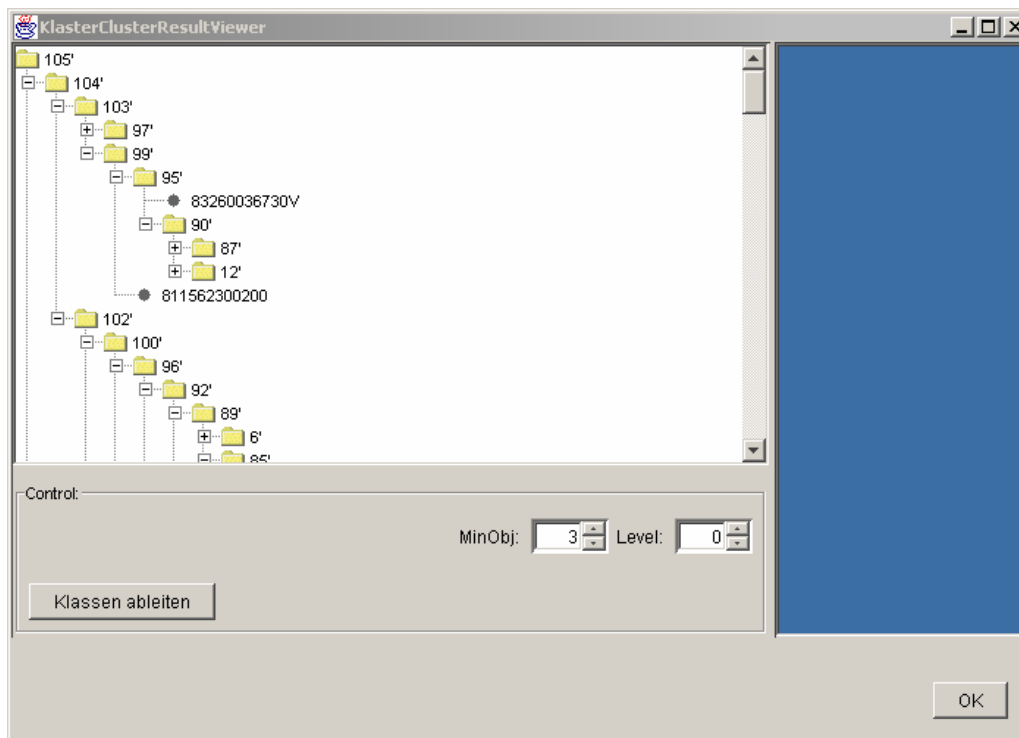


Bild 3-33: Festlegung des MinObj/Level

### 3.3.5.5 Generierte Hierarchie verifizieren

Wenn Sie im Bearbeitungsfenster „KlasterClusterResultViewer“ auf „Klassen ableiten“ klicken, generieren Sie eine Klassenhierarchie, die Sie im anschließend im Hauptfenster (unten links) finden und dort verifizieren bzw. bearbeiten können. Das Kontextmenü einer Klasse bietet verschiedene Bearbeitungsoptionen wie „Unterklasse einfügen“, „Bearbeiten“, etc.

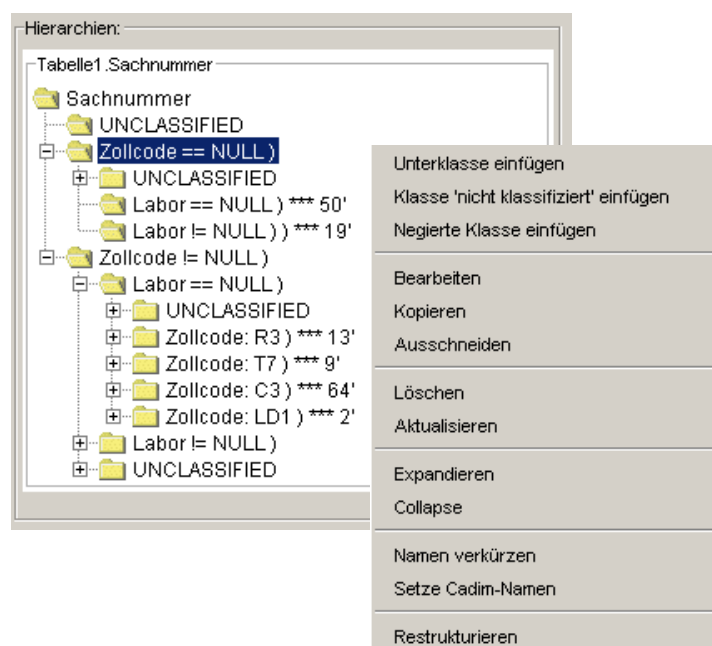



Bild 3-34: Verifizierung der generierten Hierarchie

### 3.3.6 Klassifikation durchführen

#### 3.3.6.1 Nutzdaten und Hierarchien wählen

Ein Klick auf das Icon  öffnet ein PopUp-Fenster, in dem alle angebundenen Nutzdatenquellen und verfügbaren Klassenhierarchien aufgelistet sind. Wählen Sie die entsprechenden Elemente aus und bestätigen Sie mit OK.

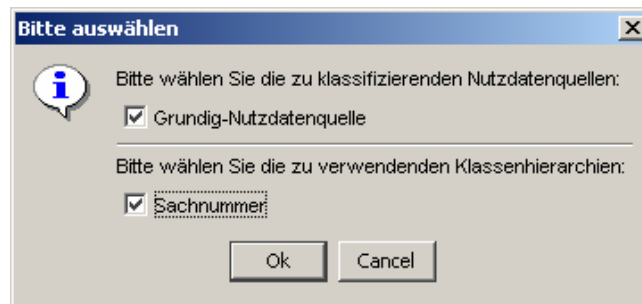


Bild 3-35: Nutzdaten und Hierarchien wählen

#### 3.3.6.2 Klassifikationsergebnis anzeigen

Das Klassifikationsergebnis wird im Fenster „Pak Ergebnisse“ gezeigt. Klicken Sie auf eine Klasse, um die dieser Klasse zugeordneten Objekte anzuzeigen.

Sie können den klassifizierten Datenbestand durchsuchen, indem Sie einen Suchbegriff im entsprechenden Textfeld eingeben anschließend wiederholt auf „Suchen“ drücken.

The screenshot shows the 'PAK Ergebnisse' window. At the top, there is a search bar and buttons for 'Suchen:', 'Klassen', and 'Resultate'. The main area is divided into two sections: 'Hierarchie' and 'Ergebnisse'.

**Hierarchie:** A tree view showing a classification hierarchy. The root is 'Sachnummer', which branches into 'UNCLASSIFIED', 'Zollcode != NULL', 'Labor != NULL', and 'Labor == NULL'. Under 'Labor != NULL', there are sub-nodes for 'Labor: L01' and 'Labor: L02'. 'Labor: L01' further branches into 'UNCLASSIFIED', 'Zollcode: 2a ) \*\*\* 30', 'Zollcode: 6a ) \*\*\* 26', and 'Zollcode: LD1 ) \*\*\* 6'. 'Labor: L02' branches into 'UNCLASSIFIED', 'Zollcode: R3 ) \*\*\* 12', and 'Zollcode: T7 ) \*\*\* 90'. 'Labor == NULL' branches into 'UNCLASSIFIED' and 'Zollcode: LD1 ) \*\*\* 98'. Below 'Labor == NULL', there is another 'UNCLASSIFIED' node, followed by 'Zollcode == NULL', which branches into 'Labor != NULL ) \*\*\* 19', 'Labor == NULL ) \*\*\* 50', and 'UNCLASSIFIED'.

**Ergebnisse:** A table with 11 rows and 6 columns. The columns are: 'Nr.', 'Schwellwert', 'Tabelle1.Sa...', 'Tabelle1.Be...', 'Tabelle1.La...', and 'Tabelle1.Zol...'. The data is as follows:

Nr.	Schwellwert	Tabelle1.Sa...	Tabelle1.Be...	Tabelle1.La...	Tabelle1.Zol...
	Schwellwert...	'Grundig-Nut...	'Grundig-Nut...	'Grundig-Nut...	'Grundig-Nut...
1	1.0	184330210001	ABSCHIRMU...	L01	2a
2	1.0	811562300500	LISHR GWN1...	L01	2A
3	1.0	811562300700	LISHR GWN1...	L01	2A
4	1.0	184380100001	STIRNWAND	L01	2a
5	1.0	184380120001	SEITENTEIL	L01	2a
6	1.0	184380150000	ABSCHIRMU...	L01	2a
7	1.0	184380150001	ABSCHIRMU...	L01	2a
8	1.0	184380160000	ABSCHIRMU...	L01	2a
9	1.0	184380160001	ABSCHIRMU...	L01	2a
10	1.0	184380190001	TRAVERSE ...	L01	2a
11	1.0	184380210001	ABSCHIRMU...	L01	2a

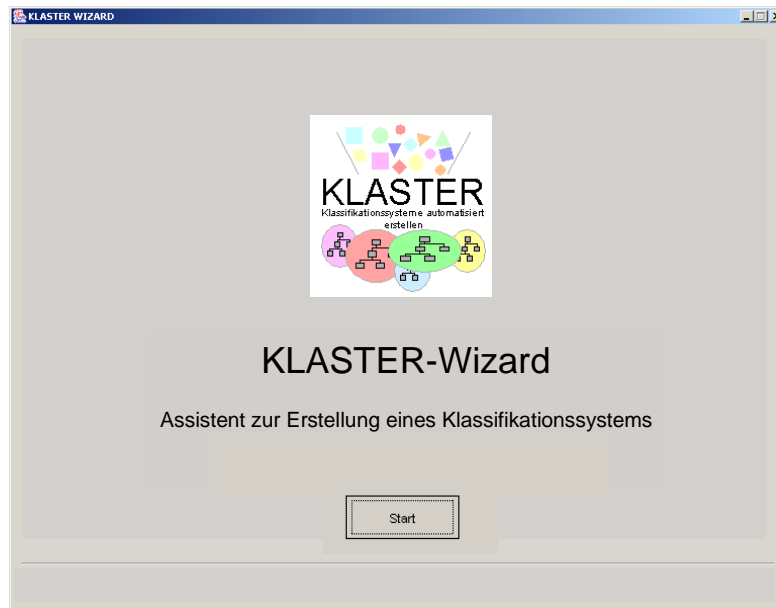
At the bottom of the window, there are three buttons: 'Schließen', 'Klasse als Exceltabelle speichern', and 'In XML-Datei speichern'.

Bild 3-36: Klassifikationsergebnis

Der Export der Ergebnisse ist momentan nach Excel (klassenweise), XML oder SQL möglich.

### 3.3.7 KLAUSTER Wizard

Mit Hilfe vom KLAUSTER Wizard soll die Erstellung eines unternehmensspezifischen Klassifikationssystem dem neuen Anwender erleichtert werden. Klaster Wizard assistiert den Anwender mit schrittweisem Vorgang, einfacher Bedienung und klarer Hinweise für jeden Schritt.



*Bild 3-37: Klaster-Wizard*

Auf dem Klaster-Wizard-GUI findet man 3 Hauptbereiche (vgl. Bild 3-38):

Im Schrittanzeigebereich werden Schritte angefangen von der Anbindung der Nutzdatenquellen bis zur kompletten Merkmalsgenerierung angezeigt. Der Text des gerade bearbeiteten Schritts wird automatisch gelb markiert.

Im Arbeitsbereich finden Sie Schritttitel, Bedienungshinweise zum entsprechenden Schritt und Arbeitsfenster.

Der Kontrollbereich bietet Kontrollknöpfe „Schließen“, „Weiter“ und „Zurück“.

KLAster-Wizard besteht aus 2 Hauptschritten, angefangen mit der Anbindung der Nutzdatenquellen und gefolgt mit der Merkmalsgenerierung. Insgesamt sind 10 Einzelschritte durchzuführen. In den folgenden Abschnitten wird die Vorgehensweise zur Bedienung des Wizards erklärt. Wenn Sie ausführliche Information zu einzelnen Schritten brauchen, können Sie diese im Kapitel 3.3.2 bis 3.3.4. finden.

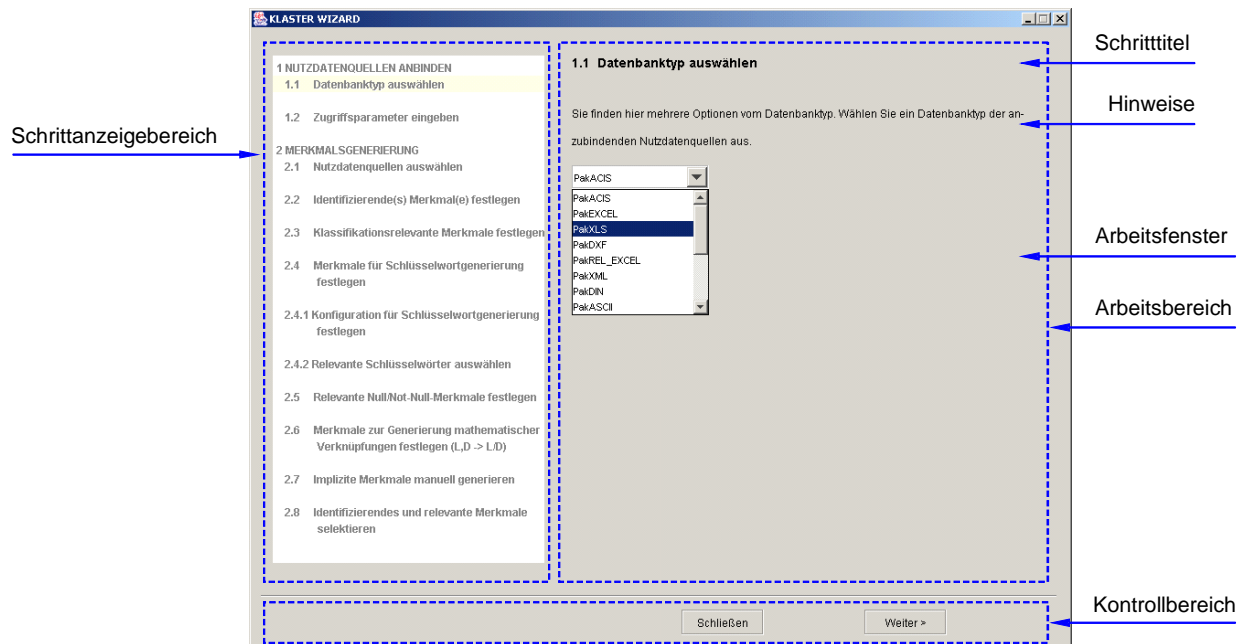


Bild 3-38: Bereiche im KLASSTER-Wizard

## Schritt 1: Nutzdatenquellen anbinden

**Schritt 1.1: Datenbanktyp auswählen:** Auf dem Arbeitsbereich finden Sie mehrere Optionen von Datenbanktypen (z.B.: PakACIS, PakExcel, PakXLS, etc). Wählen Sie einen Datenbanktyp der zu anbindenden Nutzdatenquellen aus. Als Beispiel können Sie den Datenbanktyp PakXLS auswählen.

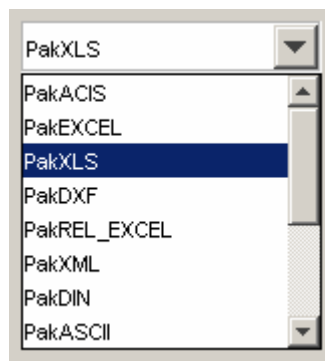


Bild 3-39: Datenbanktyp zur Auswahl

**Schritt 1.2: Zugriffparameter eingeben:** Die Zugriffparameter ist vom Typ der Nutzdatenquelle abhängig. Am Beispiel Excel müssen lediglich die anzubindenden Excel-Dateien angegeben werden. Doppelklicken Sie das Textfeld von der Datenbankbezeichnung und anschließend wählen Sie eine Datei vom FileChooser-Fenster aus, zum Beispiel Teilstamm\_2.excel.

Hostname: localhost Port: 0000

LowLevelHost: LowLevelPort:

Datenbankbezeichnung: C:\KLAUSTER\_R4\Klassifikator\data\userDB\grundig\Te

Login: public Password:

Speichern

Bild 3-40: Zugriffsparameter

## Schritt 2: Merkmalsgenerierung

**Schritt 2.1: Nutzdatenquellen:** Eine Nutzdatenquelle ist ausgewählt worden. Mit einem Klick auf den Checkbox-Knopf können Sie alle Merkmale der Datei anzeigen lassen. Sie bekommen eine Übersicht über vorhandene Merkmale. Die Merkmale sind als identifizierend, relevant und nicht relevant gruppiert.

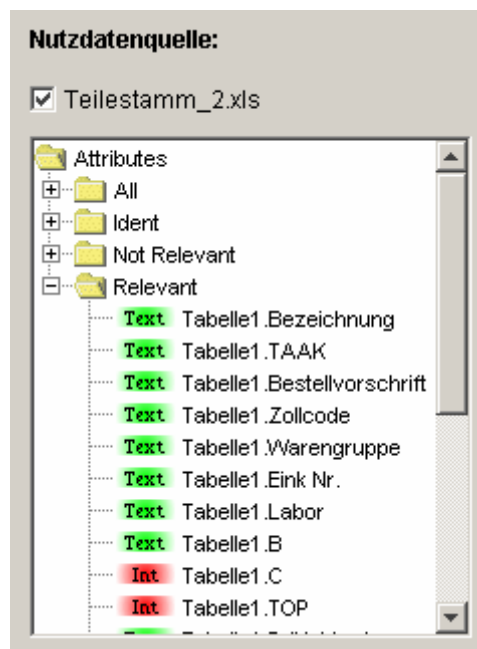


Bild 3-41: Nutzdatenquelle

**Schritt 2.2: Identifizierende(s) Merkmal(e) festlegen:** In diesem Schritt werden identifizierende Merkmal(e) gezeigt. Wählen Sie ein identifizierendes Merkmal, z.B. die Sachnummer.





Bild 3-42: Identifizierendes Merkmal

**Schritt 2.3: Klassifikationsrelevante Merkmale festlegen:** Alle klassifikationsrelevanten Merkmale werden angezeigt. Sie können hier weiter spezifizieren, welche Merkmale zur Erstellung ihres Klassifikationssystems von großer Bedeutung sind. Relevant für unser Beispiel sind die Merkmale Warengruppe, Zollcode, Labor, TOP und FK. Wählen Sie die Merkmale aus.

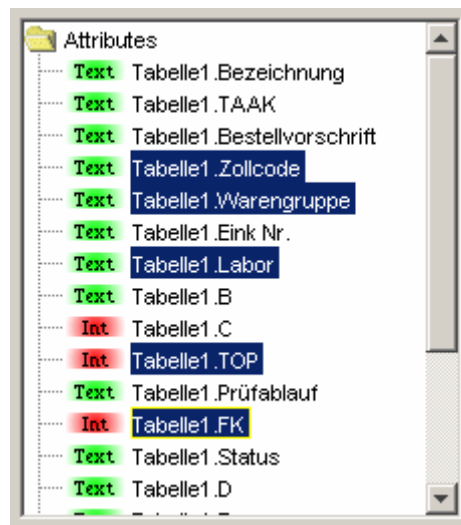


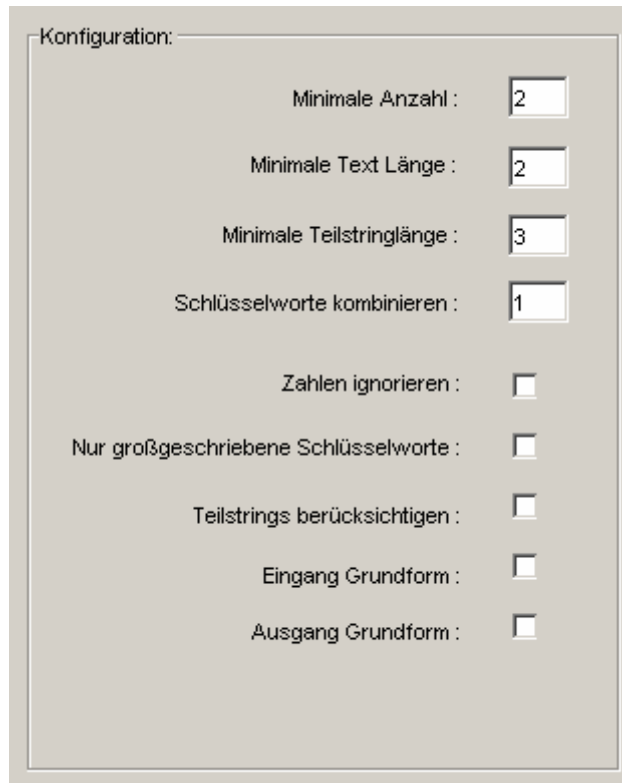
Bild 3-43: klassifikationsrelevante Merkmale

**Schritt 2.4: Merkmale für Schlüsselwortgenerierung festlegen:** In diesem Schritt wählen Sie Merkmale aus, aus denen Schlüsselwörter generiert werden sollen. Wählen Sie zum Beispiel die Merkmale: Warengruppe, Labor, TOP und FK aus.



Bild 3-44: Merkmale für Schlüsselwortgenerierung

**Schritt 2.4.1: Konfiguration:** Sie definieren die Konfigurationsregel zur Generierung der Schlüsselwörter. Das Textfeld jedes Konfigurationselement ist editierbar. In unserem Beispiel können Sie die Konfiguration wie vorgegeben lassen.



Konfiguration:

Minimale Anzahl :	<input type="text" value="2"/>
Minimale Text Länge :	<input type="text" value="2"/>
Minimale Teilstringlänge :	<input type="text" value="3"/>
Schlüsselworte kombinieren :	<input type="text" value="1"/>
Zahlen ignorieren :	<input type="checkbox"/>
Nur großgeschriebene Schlüsselworte :	<input type="checkbox"/>
Teilstrings berücksichtigen :	<input type="checkbox"/>
Eingang Grundform :	<input type="checkbox"/>
Ausgang Grundform :	<input type="checkbox"/>

*Bild 3-45: Konfiguration für Schlüsselwortgenerierung*

**Schritt 2.4.2: Relevante Schlüsselwörter auswählen:** Für jedes im Schritt 2.4 ausgewählte Merkmal sind Schlüsselwörter generiert worden. Sie können hier weiter spezifizieren, welche Schlüsselwörter von großer Bedeutung sind. Relevant für unser Beispiel sind die Schlüsselwörter:

- NULL, KE00 und GO00 vom Merkmal Warengruppe
- NULL, L01 und L02 vom Merkmal Labor
- 0, 1, 2, und 3 vom Merkmal TOP
- 230, 260 und 110 vom Merkmal FK.

Wählen Sie diese Schlüsselwörter aus.

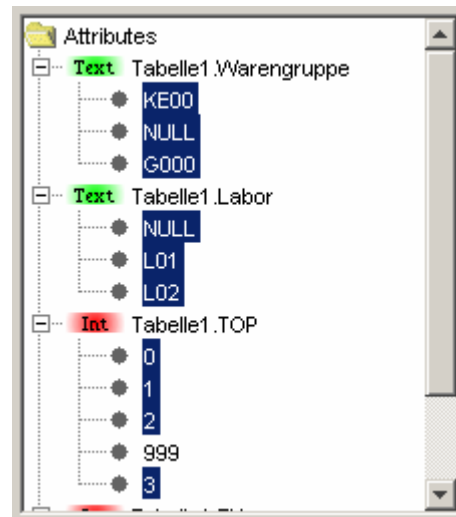


Bild 3-46: Schlüsselwörter auswählen

**Schritt 2.5: Relevante Null/Not-Null-Merkmale festlegen:** Null/Not-Null-Merkmale haben ein großes Gewicht und das Auswählen eines Null/Not-Null-Merkmals führt zur Generierung von zwei Hauptklassen. Wählen Sie zum Beispiel das Merkmal Zollcode.

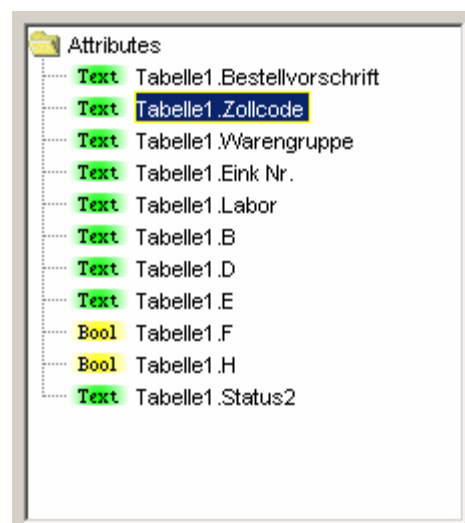


Bild 3-47: Null/Not-Null-Merkmale

**Schritt 2.6: Merkmale zur Generierung mathematischer Verknüpfungen festlegen:** In diesem Schritt haben Sie die Möglichkeit, neue Merkmale zu generieren, indem Sie mathematische Verknüpfungen von jeweils zwei vorhandenen Merkmalen definieren. Geben Sie einen Merkmalname ein. Legen Sie das Datentyp des zu generierenden Merkmals (z.B.: INT) fest. Wählen Sie zwei Merkmale sowie deren mathematischen Verknüpfung aus, z. B. Merkmale FK und TOP mit der mathematischen Verknüpfung: +. Mit einem Klick auf „Generieren“, erstellen Sie das entsprechende Merkmal. Sie können die Werte des Merkmals im Textfeld anzeigen lassen.

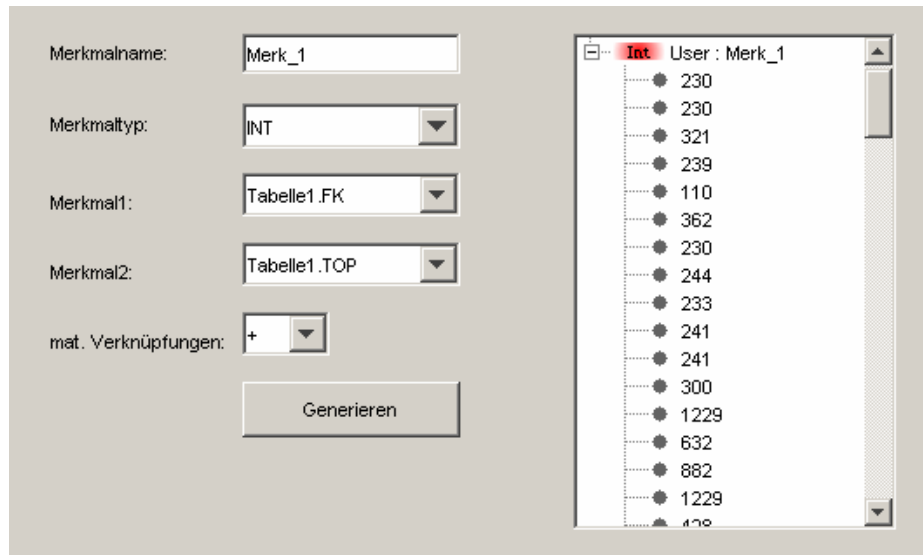


Bild 3-48: Merkmale mit mathematischer Verknüpfungen

**Schritt 2.7: Implizite Merkmale manuell generieren.** Im Kapitel 3.3.4.10 wird die Vorgehensweise zur Erstellung weiterer impliziter Merkmale ausführlich erklärt. Es wird gezeigt, wie ein implizites Merkmal mit dem Verfahren „Drag & Drop“ erzeugt wird. Im Klaster-Wizard soll die Regel für implizite Merkmale mit Text beschrieben werden. Geben Sie einen Merkmalname ein. Legen Sie den Datentyp des zu generierenden Merkmals (z.B.: BOOLEAN) fest. Mit einem Doppelklick auf ein Merkmal, z.B. Tabelle1.TOP fügen Sie das Merkmal im Textfeld für die Regel ein. Geben Sie z.B. folgende Regel ein:  $\text{Merk\_1} > 300$ . Anschließend klicken Sie auf „Generieren“, um das Merkmal zu erstellen. Sie können zurück zum vorherigen Schritt gehen, wenn Sie weitere benutzerdefinierte Merkmale erstellen möchten.

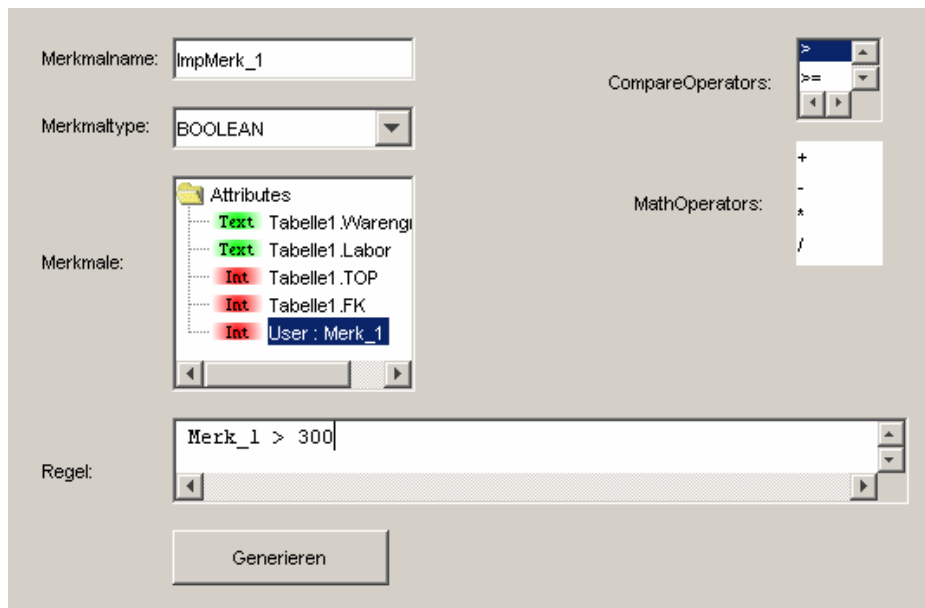


Bild 3-49: Generierung impliziter Merkmale

**Schritt 2.8: Merkmalübersicht:** Sie bekommen eine Übersicht der von Ihnen ausgewählten und generierten Merkmale. Wenn Sie Änderungen durchführen wollen, können Sie mit einem Klick auf den Knopf „zurück“ immer zum vorherigen Schritt zurückkehren. Speichern Sie die Dateien, indem Sie auf „Speichern“ klicken.

Anhand der von Ihnen ausgewählten und generierten Merkmale können Sie das Klassifikationssystem mit KLASSTER automatisch erstellen lassen. Öffnen Sie die vom KLASSTER-Wizard erstellte Datei mit KLASSTER. Im Fenster „Merkmale“ finden Sie den Ordner „ausgewählte Merkmale“. Dort ist die Merkmalübersicht wieder zu finden. Sie brauchen nur die ausgewählten Merkmale und Schlüsselwörter in den Vectorspace zu legen und eine Klassenhierarchie von KLASSTER automatisch erzeugen lassen. Sie können anschließend, wie im Kapitel 3.3.6 beschrieben wird, eine Klassifikation durchführen.

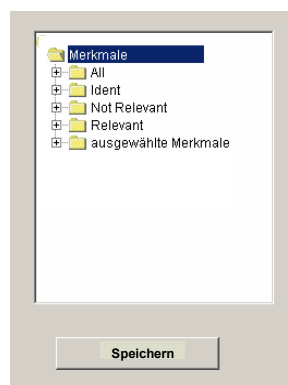


Bild 3-50: Merkmalübersicht

## **4 Praxiseinsatz unternehmensspezifischer Klassifikationssysteme**

Nachdem im letzten Kapitel die theoretischen Konzepte sowie die Funktionalität der innerhalb des Forschungsprojektes KLAUSTER entwickelten Software vorgestellt wurden,

### **4.1 Klassifikation von Projektdaten in der Gebäudeautomation**

#### **4.1.1 Problemstellung und Ziele**

##### **4.1.1.1 Spezifische Problemstellung in der Automatisierungstechnik**

Automatisierungstechnische Anlagen in der Gebäudetechnik sind sehr umfangreich und sehr komplex. Da sie meist hierarchisch strukturiert sind, können identifizierende Merkmale in jedem Planungsdokument gefunden werden. Der Planungsablauf besteht aus dem Basic-Engineering (bzw. einer Projektstudie) und dem Detail-Engineering. Dabei entstehen vielfältige Entwurfsunterlagen, wie R&I-Schemata, Wirk- und Stromlaufpläne für MSR- und Elektro-Technik, Schrankdokumentation inklusive Aufbau-, Klemmenpläne und Kabellisten, Montageanordnungen der MSR- und Elektro-Technik, Materialisten, Lagepläne, Kabeltrassenpläne. Im elektro- bzw. automatisierungstechnischen Bereich sind die technischen Darstellungen vorwiegend zweidimensional und von schematischer Natur. Eine komplexe elektro-technische Anlagendokumentation z.B. für ein Kraftwerk oder eine Chemieanlage kann dabei durchaus mehrere 10.000 Einzeldokumente umfassen, die ein Anwender (z.B. als Anlagen- oder Projekt-Ingenieur) je nach gewünschter Sichtweise nach unterschiedlichen Kriterien (z.B. Einbauort, Funktion, Betriebsmittel oder Dokumentenart) durchsuchen möchte. Meist schon verwendete Schlüsselssysteme bieten dabei Unterstützung. Jedoch ist die Recherche über Schlüsselssysteme in Unterlagen zu Altprojekten nur dann sinnvoll, wenn diese Unterlagen in sich konsistent sind, d.h. wenn bei Änderungen von Dokumente deren Identifikatoren über Schlüsselmodifikationen ständig aktualisiert wurden. Leider gelingt dieses nur unter großem Aufwand, in der Regel meistens nicht. Es ergeben sich folgende Problemsituationen:

- Daten und Dokumente sind zwar über unternehmensspezifische klassische Schlüsselnummernsysteme geordnet aber der inhaltliche Zugriff ist bei großen Dokumentenmengen schwer bzw. nur mit hohem manuellen Aufwand möglich.
- Daten und Dokumente sind objektorientiert in einem CAE-System entsprechend der natürlichen Anlagenstruktur geordnet und eine Recherche ist entsprechend der Anlagenstruktur zwar gut möglich aber bei großen und komplexen Anlagenstrukturen auch bei Benutzung eines Browsers mit hohem manuellen Aufwand verbunden.
- Daten und Dokumente sind im Verlauf vieler Änderungszyklen nicht mehr konsistent.

Eine typische Ausrüstungseinheit der Automatisierungstechnik ist der elektronische Schaltschrank. In diesem sind zur Steuerung von technischen Anlagen die notwendigen Geräte (z.B. Schaltschütze, Relais, Thyristoren, Speicherprogrammierbare Steuerungen – SPS – u.a. ) zu-

sammengefasst. Schaltschränke sind mit den zugehörigen Sensoren und Aktoren der technischen Ausrüstung (z.B. der Klimatechnik) über Elektro- und Kommunikationsleitungen zu einer Funktionsstruktur (z. B. der Klimaregelung eines Gebäudeteils) sowie mit der zentralen Bedieneinheit vernetzt. Der technische Zusammenhang wird in den Planungsdokumenten als Regelschema oder Funktionsplan dargestellt. Diese Dokumente stellen somit eine grundlegende technische Ordnungssystematik der komplexen Anlage dar. Sie sollen mit den vielen anderen Planungs- und Ausführungsdokumenten den Aufbau, die Inbetriebnahme und den Service bei Störung, Wartung und Instandsetzung ermöglichen.

Orientierungshilfe für den Zugriff auf die Projektdokumente und deren Inhalte ist bei Planung, Engineering und Service dringend erforderlich, da die Automatisierungssysteme aufgrund der ständig zunehmenden Möglichkeiten in der Mikrosystem- und IT-Technologie immer umfassender und komplexer werden.

#### **4.1.1.2 Ziele der KLAUSTER-Anwendung**

Mit der Anwendung der KLAUSTER-Methode auf automatisierungstechnische Projektierungsunterlagen wird das Ziel verfolgt, insbesondere aus Funktionsplänen und R&I -Schemas einer Automatisierungsanlage Merkmale zu gewinnen, die die Generierung eines Klassifizierungsschemas für die Projektunterlagen dieser Anlage bzw. einer ganzen Klasse ähnlicher Anlagen ermöglichen.

Zu Beginn eines neuen Projektes soll mit Hilfe einer automatisierten Klassifikation aus vorhandenen archivierten Projektunterlagen ähnliche automatisierungstechnische Anlagen bzw. Komponenten ermittelt werden. Bei positivem Ergebnis können dann größere Teile (im günstigsten Fall alle Teile) der *Stromlaufpläne* und der *Schaltschrankkonstruktion* für das neue Projekt wiederverwendet werden. Auch innerhalb von neuen und insbesondere großen Projekten lassen sich durch die automatische Klassifikation ähnliche Anlagenkomponenten ermitteln, so dass a priori bei der Projektierung der Wiederholungseffekt sofort genutzt werden kann.

Zu einer manuellen Modellierung von Klassifikationsstrukturen ist ein tiefes Fachwissen des Anwendungsbereiches nötig. Darum ist es wünschenswert, wenn durch intelligente Verfahren eine automatisierte Klassierung aus den semantischen Zusammenhängen des vorliegenden Datenmaterials erfolgen kann. Mit dem entwickelten KLAUSTER-Lösungsansatz wird solch eine Methode angeboten.

## 4.1.2 Einsatz der KLAUSTER-Software

### 4.1.2.1 Einsatzkonzept

Die KLAUSTER-Methode soll benutzt werden, um aus einer großen Menge von Daten und Dokumenten unterschiedlichster Projekte wiederverwendbare Teilstrukturen bzw. Baugruppen herausfiltern zu können. Die Anwendung erfolgt in vier Hauptphasen Datenanalyse, Merkmalsgewinnung, Erzeugung des Klassifikationssystems und Anwendung des Klassifikationssystems. Im Folgenden wird die praktische Vorgehensweise zur Nutzung von KLAUSTER für Projektdaten der Gebäudeautomation dargestellt, um die im vorigen Abschnitt dargestellten Probleme zu beherrschen bzw. die genannten Ziele besser zu erreichen.

Mit dem Klaster-Ansatz soll die Recherche von Projektunterlagen aus einem Dokumentenpool erheblich erleichtert werden, indem sie durch Generierung eines Klassifizierungsschlüssels neu geordnet werden können.

Dabei wird die Signifikanz von Merkmalen über die KLAUSTER-Methode ermittelt. Die automatisch zu gewinnenden Merkmale müssen signifikant in Projektdaten und Dokumenten enthalten sein, um als klassifikationsrelevante Merkmale zur Erzeugung eines Klassifikationssystems dienen zu können.

Die Anlagenteile bzw. –strukturen werden klassifiziert entweder

- über vorgegebene Kombinationen von Merkmalen, die aus gespeicherten Katalogen zu Symbolen, Symbolkombinationen und deren Beschriftungen vor der Recherche ausgewählt werden

oder mittels der

- aus Funktionsplänen und R&I- Fließplänen automatisiert gewonnenen klassifikationsrelevanten Merkmalen.

### 4.1.2.2 Datenquellen

Aus der Vielzahl der verschiedenartigsten Darstellungsstrukturen und Dokumentenarten sind zwei Typen zur Merkmalsgewinnung besonders geeignet: die Motorenliste (siehe Bild 4-8) und das R&I-Schema (Regel- und Instrumentationsschema, Beispiel siehe Bild 4-1).

Das Regelschema (*R&I-Schema*) ist eine grundlegende Darstellung einer Funktionseinheit der Gebäudeautomatisierung und als Quelle für Merkmale zur Klassifikation von Anlagenstrukturen und -komponenten besonders gut geeignet, weil es in seiner schematischen Übersicht zu den Anlagenfunktionen alle wesentlichen Funktionsmerkmale enthält.

Generell sind die Datenquellen heterogen, da sie mit unterschiedlichen CAD- bzw. CAE-Werkzeugen erzeugt wurden. Sie liegen als Dateien meist im DXF-, Excel-, Access-, VNS-,



XML-Format vor. Es ist also eine CAE-systemneutrale Klassifikation erforderlich. Zur Merkmalsgewinnung mit anschließender Erzeugung des Klassifikators sind zwei Vorgehensweisen möglich:

- (1) Finden von Suchmustern auf Basis vorgegebener
  - Textfragmente,
  - Symbole,
  - Fließbilder (Signalfluss, Stofffluss, Datenfluss) und
  - Substrukturen aus diesen Elementen, die in Katalogen bzw. Merkmalslexika geführt werden.
- (2) Durchsuchen der Symbole, Fließbilder, Textfragmente in den Schematas, um klassifikationsrelevante Merkmale zu finden.

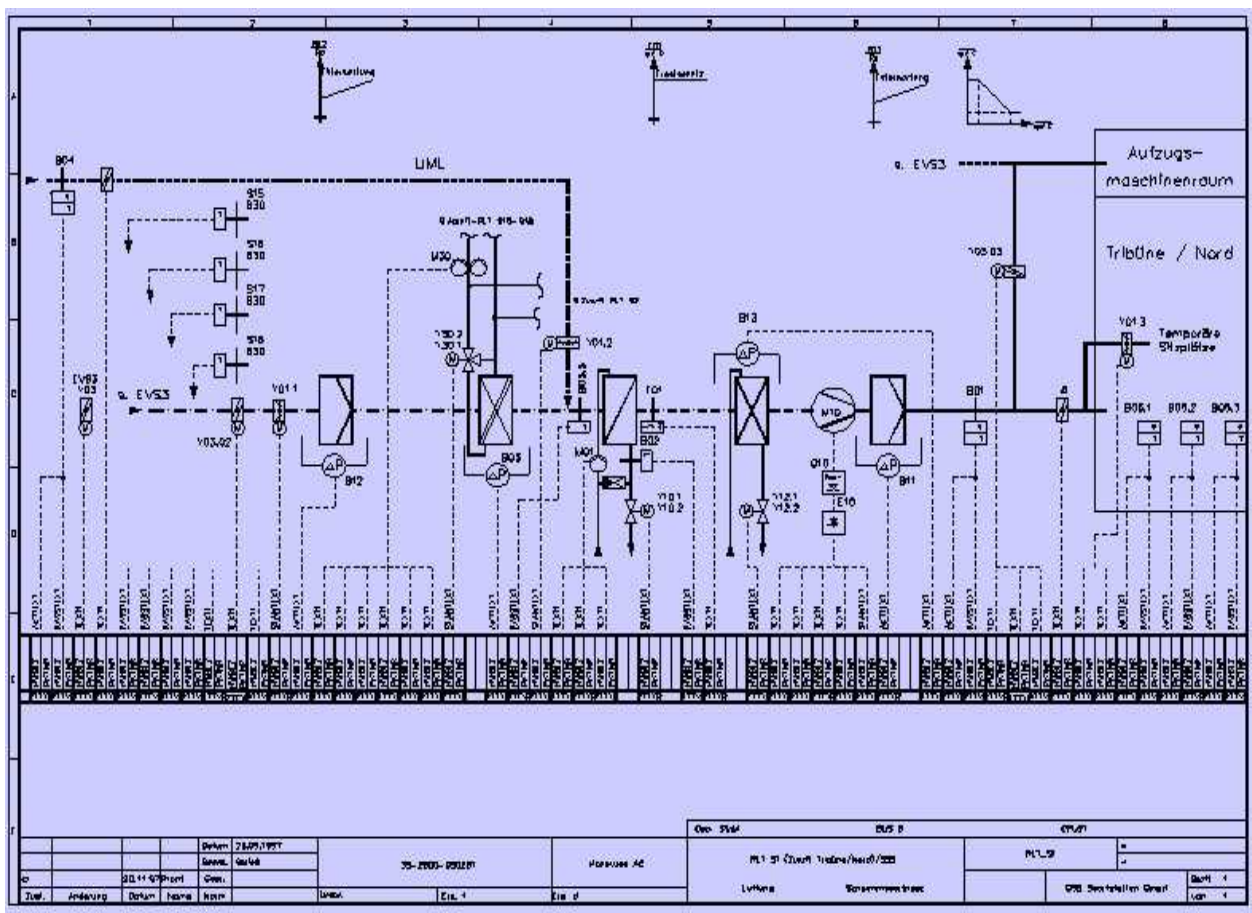


Bild 4-1: Beispiel eines R&I-Schemas

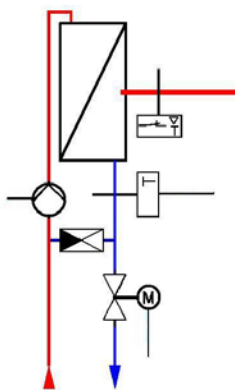
### 4.1.2.3 Merkmalsgewinnung aus DXF<sup>1</sup>-Dateien

Das R&I-Schema ist eine zweidimensionale Grafik meist im DXF-Format. DXF ist primär auf den Austausch von Zeichnungsdaten ausgerichtet. DXF tauscht jedoch nur die grafischen und nicht die logischen Informationen von Schaltplänen aus. Deshalb lassen sich die wichtigsten Informationen in einem Stromlaufplan (z. B. die logische Verknüpfung der Querverweise) über DXF nicht übertragen. Der Austausch von Daten beschränkt sich schwerpunktmäßig auf die reine Grafik.

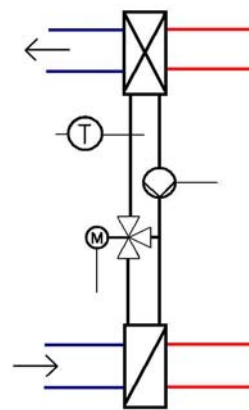
Um klassifizierende Merkmale bzw. Funktionen entnehmen zu können, muss eine intelligente Features- bzw. Zeichenerkennung durchgeführt werden. Die Suchkriterien sind vorgegebenen Symbolkatalogen zu entnehmen.

In folgenden Bildern wird versucht, die Vorgehensweise zur Merkmalsgewinnung verständlich zu machen. Ein Regelschema besteht im Wesentlichen aus mehreren Substrukturen und Einzelgeräten, die zusammen eine Funktionseinheit darstellen. Zum Beispiel sind die abgebildeten Substrukturen (siehe Bild 4-2) in den dargestellten R&I-Schemas (siehe Bild 4-5 und Bild 4-3) erkennbar.

Die Regelschematas werden mit CAE-Systemen (z. B. AUTOCAD, WSCAD, AUCOPLAN u.a.) erzeugt und gepflegt.



Luft-/ Wasser-Wärmetauscher



Wärmerückgewinnung (WRG)

Bild 4-2: Symbole für Luft-/Wassertauscher und Wärmerückgewinnung

<sup>1</sup> DXF = Drawing Exchange Format

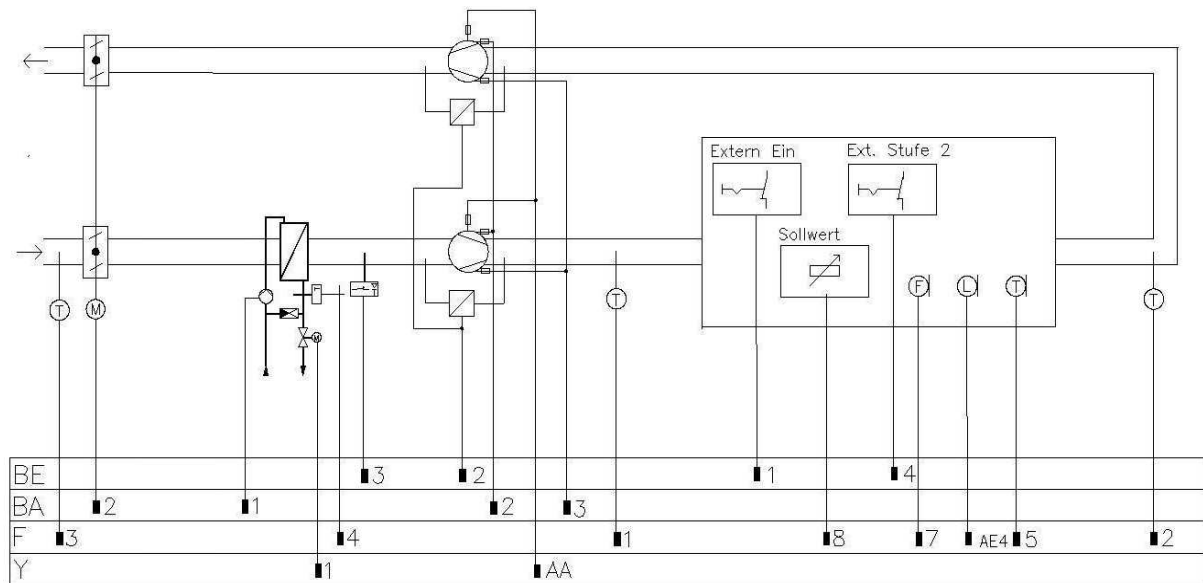


Bild 4-3: Heizregisterregelung mit Substruktur Luft-/ Wasser-Wärmetauscher

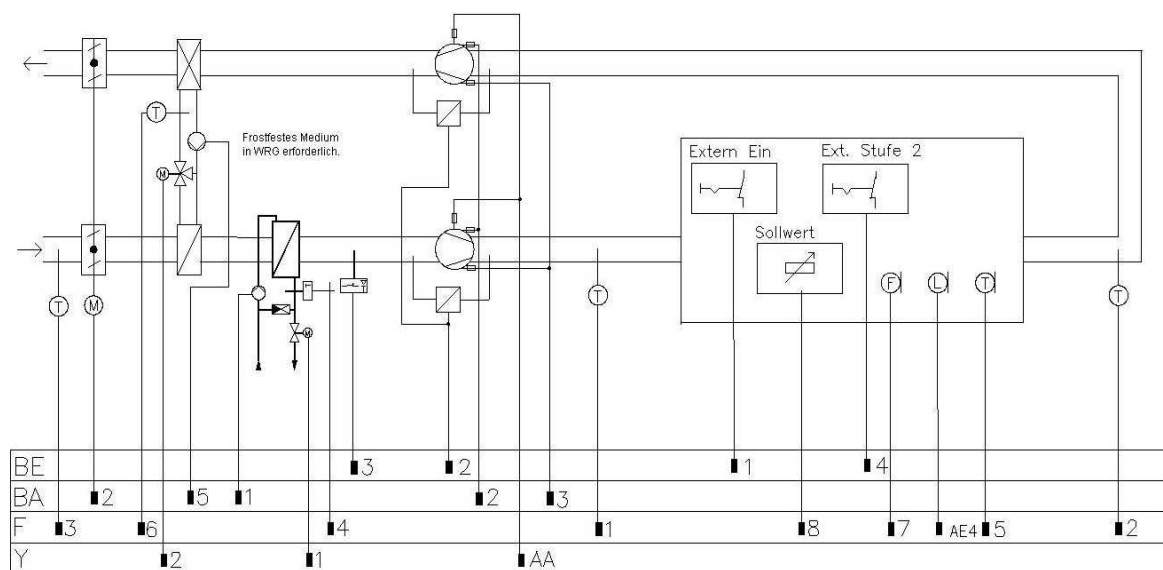


Bild 4-4: Heizregisterregelung mit den Substrukturen Luft-/ Wasser-Wärmetauscher und Wärmerückgewinnung (WRG)

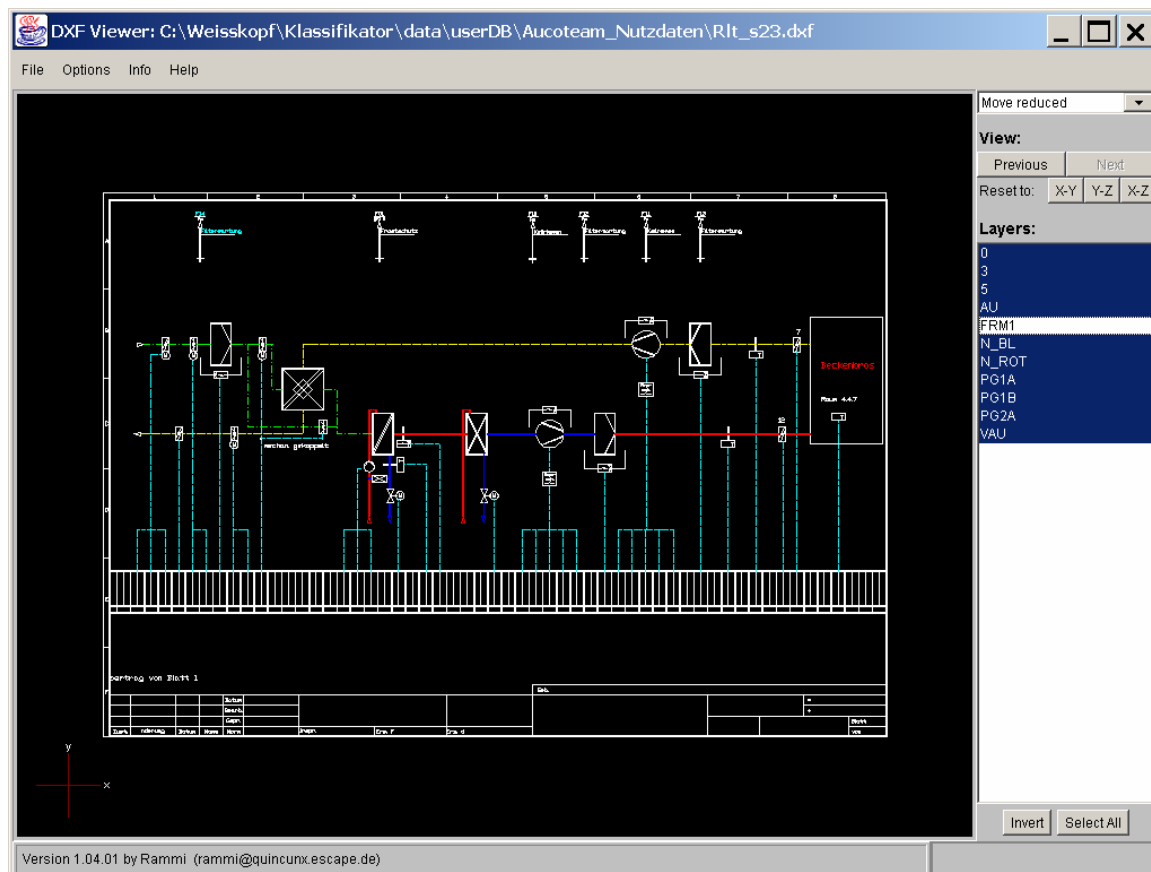
Die R&I-Schemas sind in der Regel DXF-Dateien bzw. lassen sich mit Exportfunktionen der CAE-Systeme in DXF-Dateien konvertieren, so dass diese Dateiart als Ausgangsbasis zur Gewinnung von Merkmalen aus 2D-Darstellungen für den methodischen Ansatz von KLAUSTER dienen kann.

### ***KLASTER-Methode zur Auswertung von DXF-Dateien:***

DXF ist ein Textformat zur geometrischen Beschreibung einer Zeichnung/Szene auf Basis elementarer Entitäten und deren Positionsangaben. Innerhalb des KLASTER-Forschungsprojektes diente der (open-Source) DXF-Viewer der Firma Rammy (siehe Bild 6) zur Visualisierung und Auslesen der DXF-Geometrie-Elemente. Hierzu wurde der Quellcode des Viewers entsprechend angepasst und in die LowLevel-Schnittstelle der KLASTER-Software integriert.

Die LowLevel-Schnittstelle bietet elementare Methoden zum Zugriff auf die DXF-Geometrie Elemente und Attribute. Die ermittelbare Entitäten sind zum Beispiel (auf Englisch): Polyline, Circle, Point, Block, Dimension. Die ermittelbare Attribute sind zum Beispiel (auf Englisch): File ID, File Name, Center, Radius.

Zeichnungen von AUCOTEAM bestehen aus mehreren Symbolen (z.B. Motor, Pumpe, Luftwärmer etc), welche miteinander verbunden sind. Diese Symbole werden in DXF-Dateien durch Blöcke realisiert. Blöcke sind eine Gruppierung von mehreren Basis-Geometrieelementen (z.B. Kreis mit innenliegenden Dreieck).



*Bild 4-5: Open-Source DXF-Viewer*

Ein Block ist eine ermittelbare Entität, deshalb lassen sich auch diese Symbole mit Hilfe der verfügbaren Methode erkennen und visualisieren. Um ein Symbol als solches erkennen zu können, wurde für jedes in AUCOTEAM-Zeichnungen vorkommende Symbol definiert, aus welchen Basis-Geometrie-Elementen der entsprechende Block besteht und wie diese angeordnet sind. Das Symbol eines Motors ist ein Block bestehend aus einem Kreis mit drei Polylines, welche ein M bilden (siehe Bild 4-6, oben rechts), während eine Pumpe durch ein Kreis mit zwei Polylines in Form eines Dachgiebels (in der Mitte des Kreises) beschrieben wird (siehe Bild 4-6, unten rechts). Das Bild 4-6 zeigt mit einem Screenshot der KLAUSTER- und Viewersoftware, wie das Symbol einer Pumpe und das Symbol eines Motors automatisch erkannt werden.

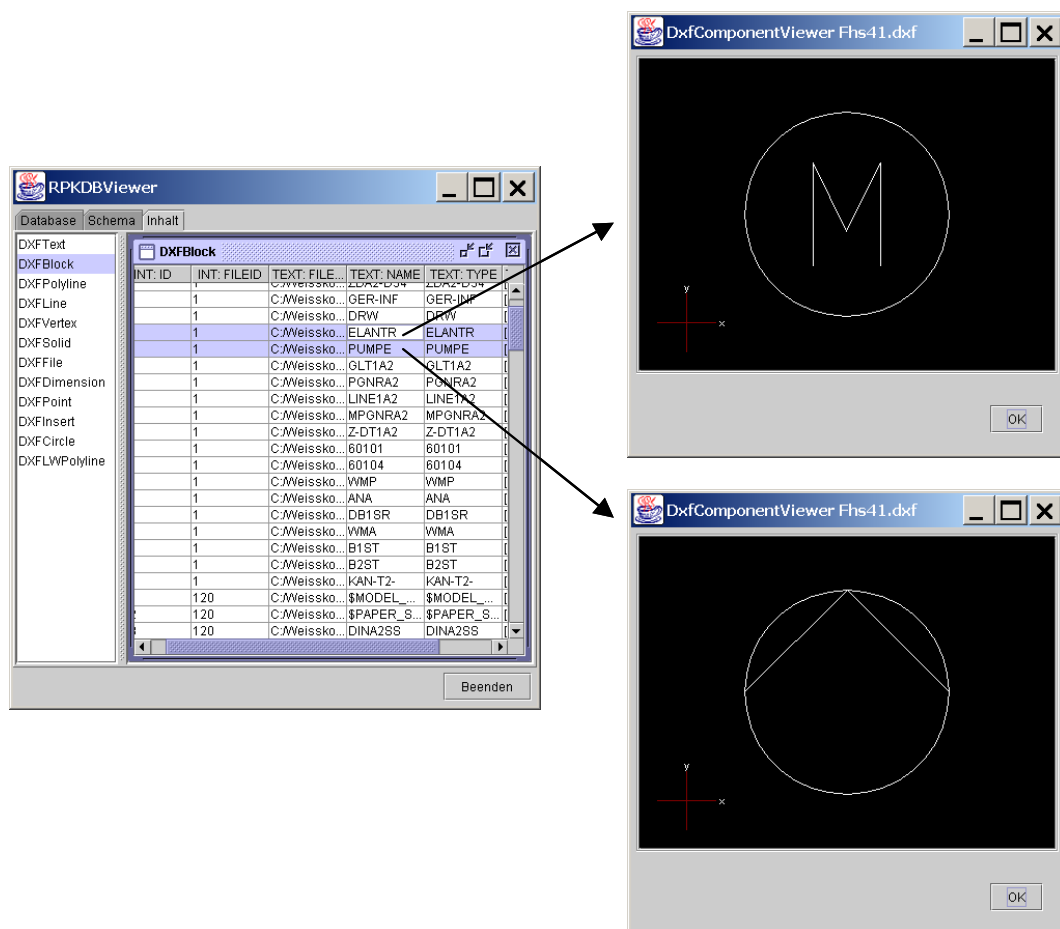
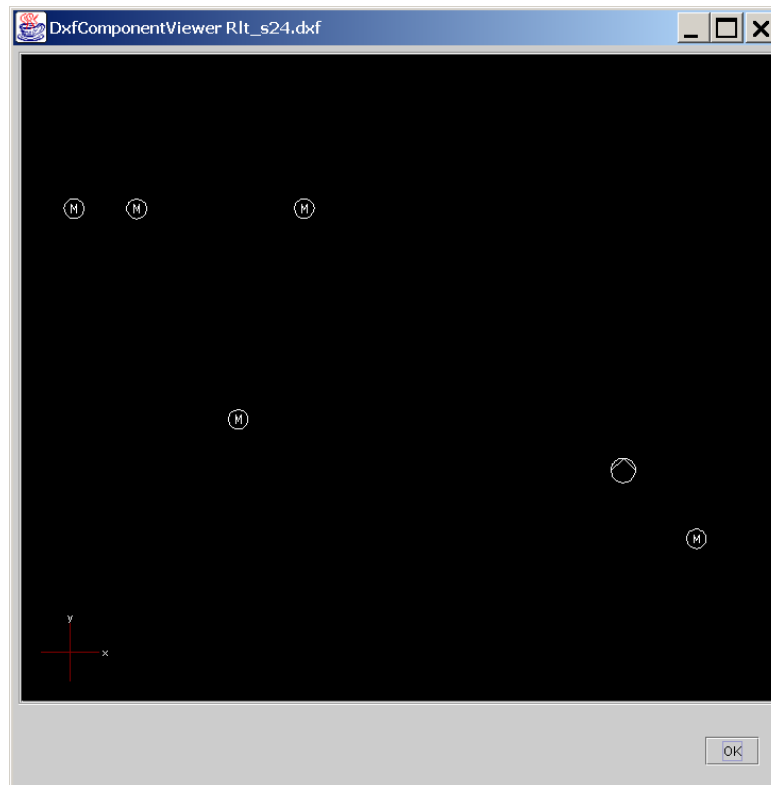


Bild 4-6: Automatische Erkennung von Symbolen in DXF-Dateien

Mit dem oben beschriebenen Verfahren wird es ermöglicht, die Anzahl vorkommender Symbole automatisch zu ermitteln. Die Ermittlung der Symbolanzahl ist für die Auswertung der DXF-Dateien KLAUSTER von großer Bedeutung (siehe Bild 4-7). Damit kann der Anwender implizite Regel zur Erkennung der Subschema definieren. Das Subschema Wärmerückgewinnung (siehe Bild 4-2, rechts) besteht beispielsweise aus 3 Hauptkomponenten, nämlich ein Lufterwärmer, ein Luftkühler und ein Dreiwegventil. So wird dann die implizite Regel des Subschema Wärmerückgewinnung definiert: (Lufterwärmer = 1) und (Dreiwegventil = 1) und

(Dreiwegventil = 1). Danach kann mit Hilfe dieser Regel ermittelt werden, ob ein R&I Schema das Subschema Wärmerückgewinnung enthält oder nicht. Die Regel kann mit vorhandenen Verknüpfungsoptionen (z.B. „und“, „oder“, „exklusives oder“ etc.) erweitert werden. Mit diesem Verfahren wird es ermöglicht, die Struktur von R&I Schema automatisch von KLAUSTER generieren zu lassen.



*Bild 4-7: Anzahl und Position von Symbolen sind nutzbare Informationen zur Klassifikation*

#### **4.1.2.4 Merkmalsgewinnung aus Excel-Dateien**

Eine weitere Quelle zur Gewinnung klassifikationsrelevanter Merkmale sind die sogenannten Motorenlisten. Ventilatoren und Stelleinrichtungen (z. B. Ventile, Klappen, Schieber u.a., siehe Bild 4-18 und Bild 4-20) in HLK- Funktionseinheiten werden durch Motoren bewegt. Die Leistungsklassen dieser Motoren sind ein adäquates Merkmal der Dimensionierung und Konfiguration aller Funktionsbaugruppen. Die Motorenlisten eines HLK-Systems werden meist als Excel-Tabellen geführt (siehe Bild 4-8).

Mit dem Klaster-Verfahren werden solche Excel-Tabellen durchsucht, um die Leistungsklassen der Motoren in Relation zu ihrer Verwendungsfunktion zu ermitteln. Die gefundenen Leistungsklassen (siehe Bild 4-9) der Motoren sind neben den aus den R&I-Schematas gefundenen Funktionsmerkmalen weitere Klassifikationsmerkmale.


			Projekt: Rad und Schwimmsporthalle		Pr.-Nr. 56056																		
Bereich Gebäudeautomatisierung			Gewerk: MSR		Schaltschrank: SS05																		
Bereich Schwimmhalle			Bereich: Schwimmhalle		Aufstellungsort: Technikraum M 05		Änderung:																
Meßprotokoll Motore			Standort:		DDC - Nr.:																		
Bearbeiter: Volkmann				Parameter		Schaltung / Motorschutz																	
Anl.Nr.	BMK	Anlagenteil	Drehzahl (U/min)	Nennleistung (KW)	Nennstrom (A)	gen. Stromaufnahme (A)	8	FU Einstellungen	Versorgungsspannung (V)	Direktanlauf	Stern-Dreieck-Anlauf	Frequenzumrichter	Dahlanderschaltung	Repschalter Hauptstrom	Repschalter Steuerspannung	Keilriemenüberwachung	Thermokontakt innen	Thermokontakt herausgeführt	Kaltleiter	Motorschutzrelais	Hersteller	Typ	Bem.
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
<b>S 1</b>		<b>Zuschauerraum (Nord) Zuluft</b>																					
	M10	ZU Lüfter		37	67				400	x	x							x				225S	
	M01	LH Pumpe		0,37	0,64	0,45			400	x							x					TOP-S 65/7	
	M30.1	WRG Pumpe		1,5		2,56			400	x								x					
	M30.2	WRG Pumpe		1,5		2,63			400	x								x					
<b>S 2</b>		<b>Foyerbereich (Nord) Zuluft</b>																					
	M10	ZU Lüfter		18,5	35				400	x	x							x				180M	
	M01	LH Pumpe		0,2	0,33	0,26			400	x							x					TOP-S 50/4	

Bild 4-8: Beispiel einer Motorentabelle

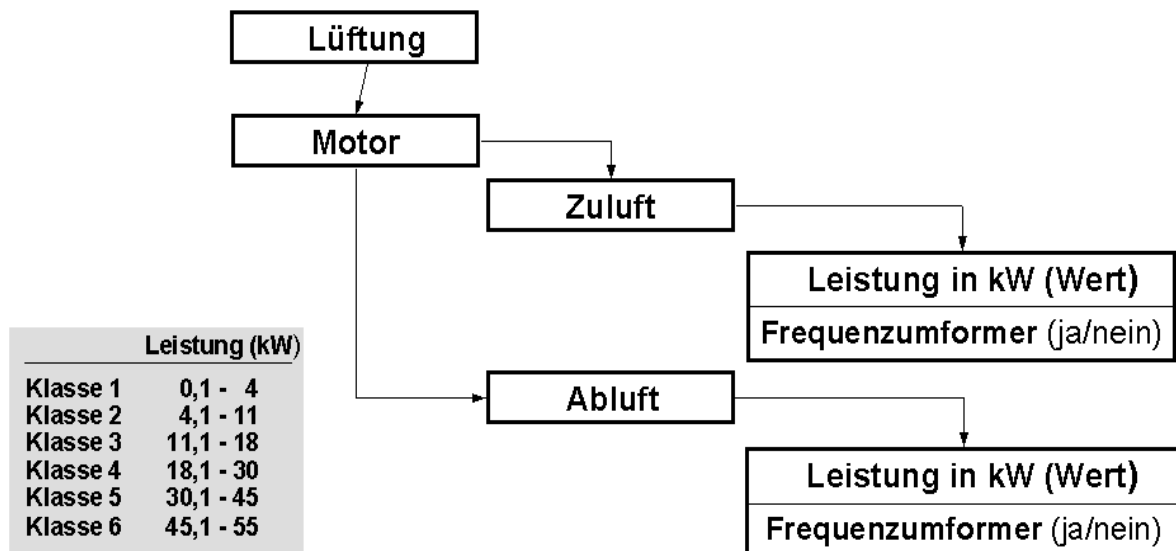


Bild 4-9: Leistungsklassen der Motoren von ausgewerteten Motorenlisten

### 4.1.3 Anwendungstest

#### 4.1.3.1 Organisatorische Vorgehensweise

Das Klaster-Verfahren wurde mit konkreten Daten aus zwei Projekten zur Gebäudeautomatisierung getestet, wobei auch die Daten und Ergebnisse aus dem Vorgängerprojekt PAK einbezogen wurden [Alt-96]. In PAK wurde eine Verfahrensweise entwickelt, um Daten mit einem Klassifizierungsschema zu klassifizieren, das vorher manuell erstellt werden musste. Nunmehr wird diese Verfahrensweise mit der KLAUSTER-Methode kombiniert, indem automatisiert aus dem Datenmaterial klassifikationsrelevante Merkmale gewonnen werden, um mit diesen Merkmalen einen Klassenhierarchie zu generieren und anschließend mit diesem Klassifikationsschema die Klassifizierung durchzuführen.

Die R&I-Schematas und Motorenlisten, die mit allen anderen Projekt-Dokumenten auf CD-ROM's vorliegen, sind der Input des Verfahrens. Die organisatorische Vorgehensweise zur KLAUSTER-Anwendung ist aus dem in Bild 4-10 dargestellten Datenfluss ersichtlich. Im Ergebnis wurden die Projektdateien des Archivs der ausgewählten Großprojekt mit dem mit KLAUSTER generierten Klassifikationssystem neu geordnet. Die Planungs-, technischen Ausführungs- und Servicedokumente wurden über ihre Zeichnungs- und Projektnummern funktionellen Klassenmerkmalen zugeordnet.

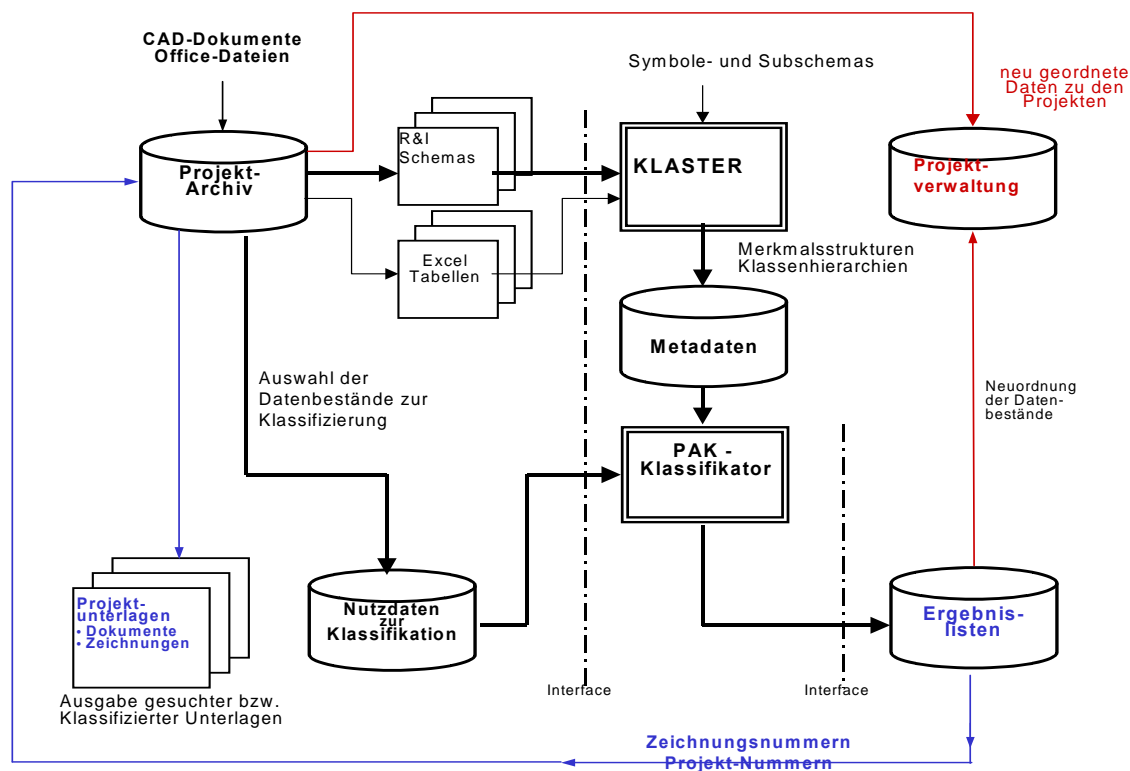


Bild 4-10: Datenfluss zur Datenaufbereitung und Durchführung der Klassifikation



### 4.1.3.2 Arbeitsablauf der KLAUSTER-Anwendung

Im Folgenden werden die Schritte zum Einsatz von KLAUSTER, die im Kapitel 3 ausführlich dargestellt wurden, auf die Anwendung für Projektdaten der Gebäudeautomatisierung projiziert.

#### (1) zu klassifizierende Dateien auswählen

In diesem Schritt ist zunächst der Datenbanktyp PakDXF (siehe Bild 4-11) und anschließend die zu klassifizierende Datei auszuwählen (siehe auch Kapitel 3.3.4.1).

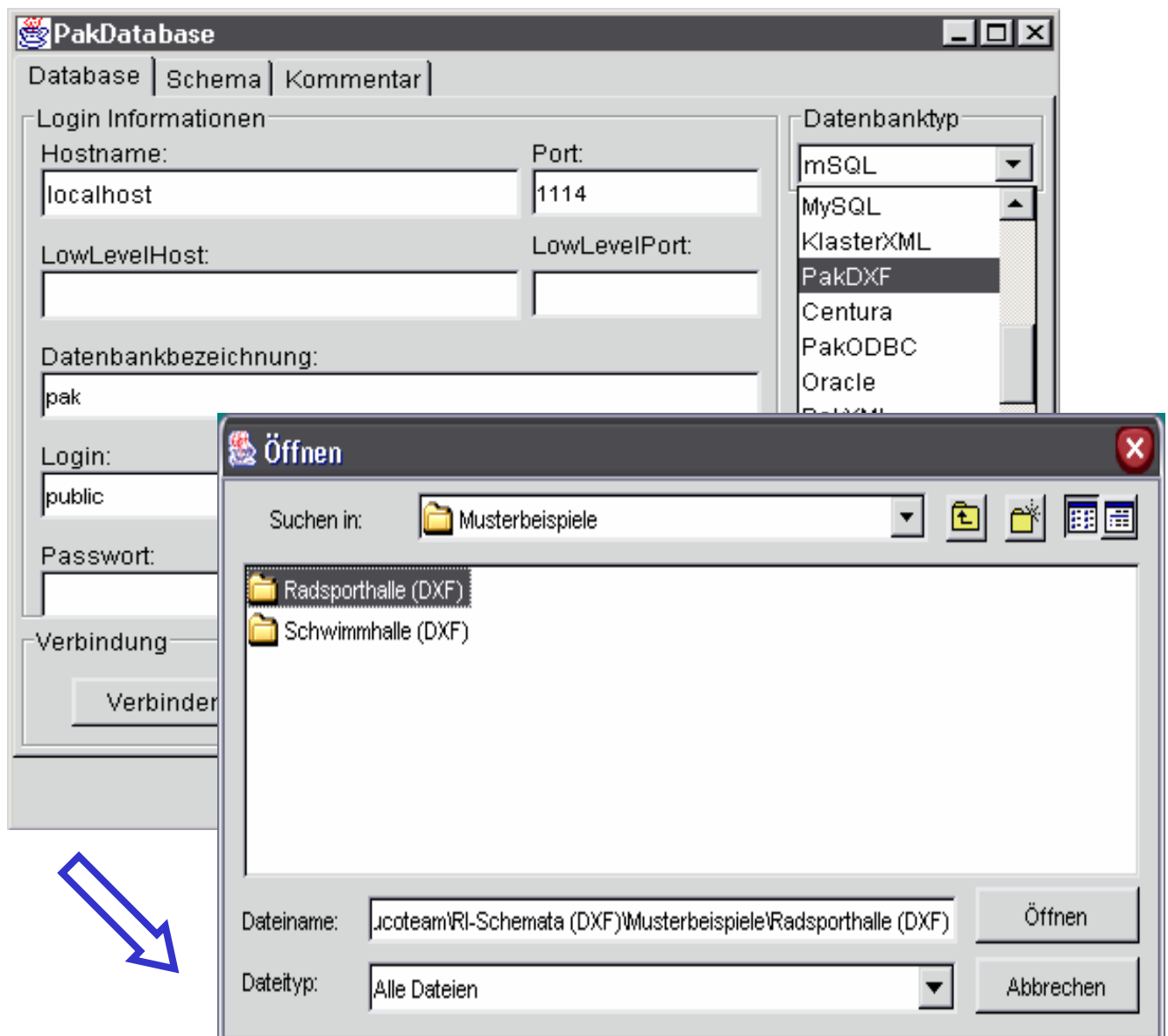


Bild 4-11: Auswählen der zu klassifizierenden Dateien

## (2) Nutzdatenquellen öffnen

Nutzdatenquellen werden geöffnet und angezeigt (siehe Bild 4-12)



Bild 4-12: geöffnete Nutzdatenquellen

## (3) Merkmale automatisch ermitteln

Aufgrund einer Analyse der definierten Nutzdatenquelle(n) werden im nächsten Schritt die klassifikationsrelevante Merkmale automatisch ermittelt (siehe Bild 4-13).

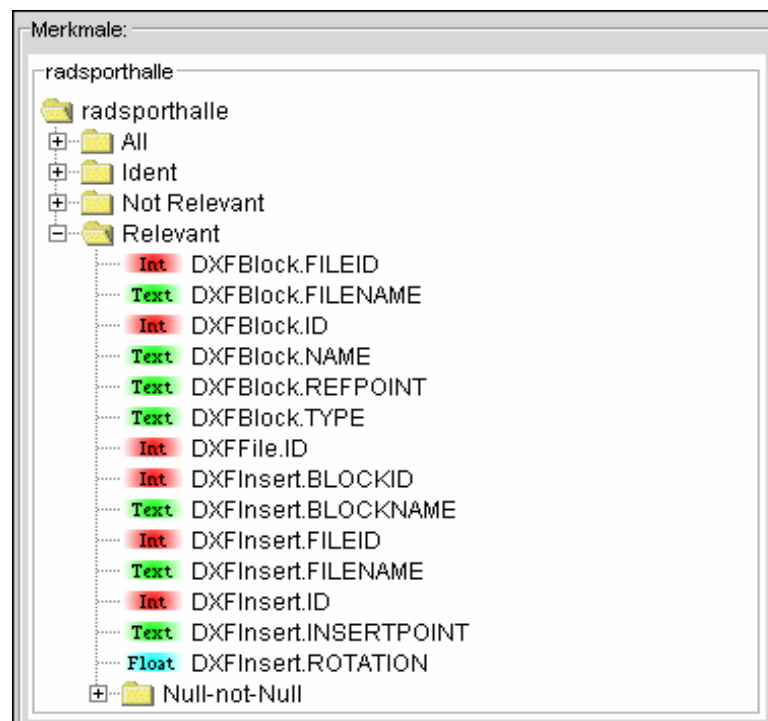


Bild 4-13: Erkennung klassifikationsrelevanter Merkmale

#### (4) Implizite Merkmale definieren

Neben den expliziten (klassifikationsrelevanten) Merkmalen können auch implizite Merkmale über Regeln definiert werden (vgl. Kap. 3.3.4.10). In Bild 4-14 wird die implizite Regel zum Merkmal „enthält Subschema Kältemaschine“ lautet definiert:

Das Subschema beinhaltet folgende Merkmale:

- 2 x Pumpe und
- 2 x Raump2 und
- 2 x Rep-Sch und
- 1 x KMVERDI

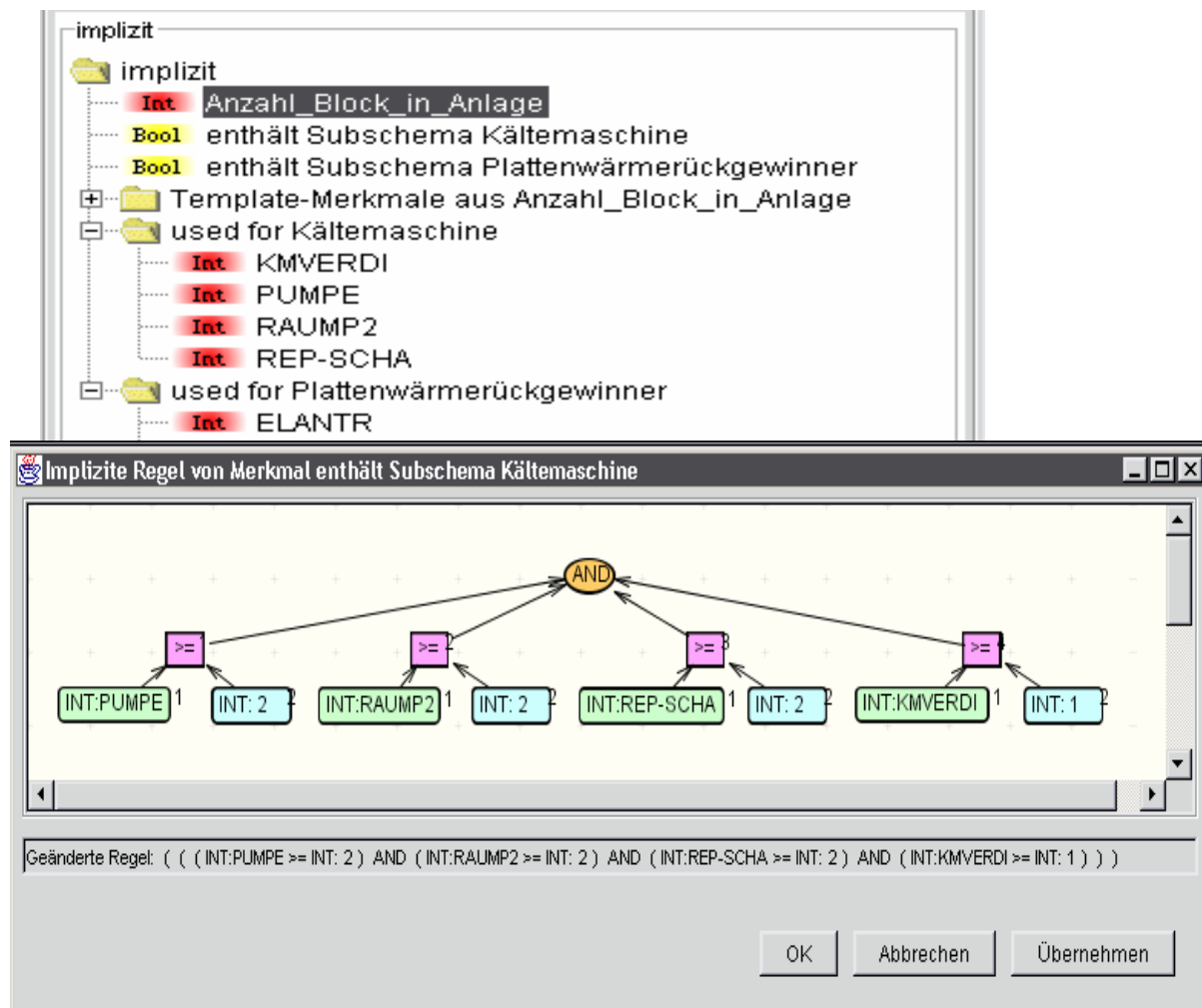


Bild 4-14: KLASSTER-Anzeige der visualisierte Regel für impliziete Merkmale

### Hierarchie generieren mit Cluster-Algorithmus

- Relevante Datei ist in den Klaster-VectorSpace hinzufügen
- Cluster-Algorithmus analysiert die Datei und ermittelt die hierarchische Zuordnung

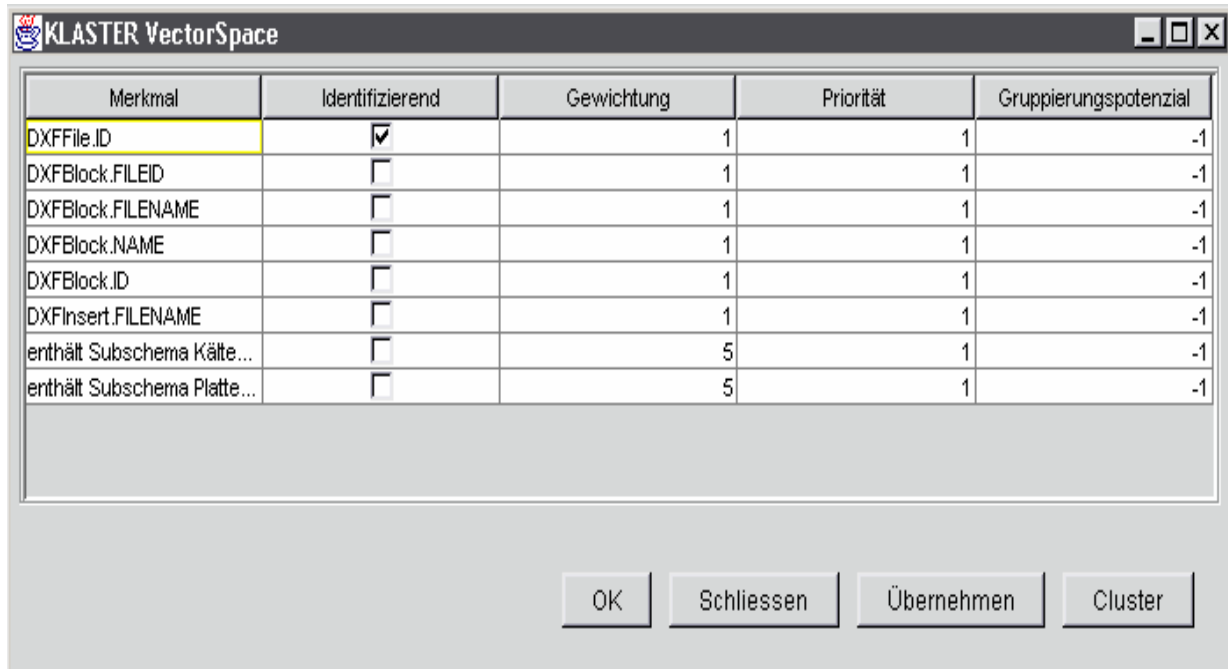


Bild 4-15: Anzeige zum Klaster VectorSpace

### (5) Klassenhierarchie generieren

Hierarchie wird generiert und anschließend angezeigt

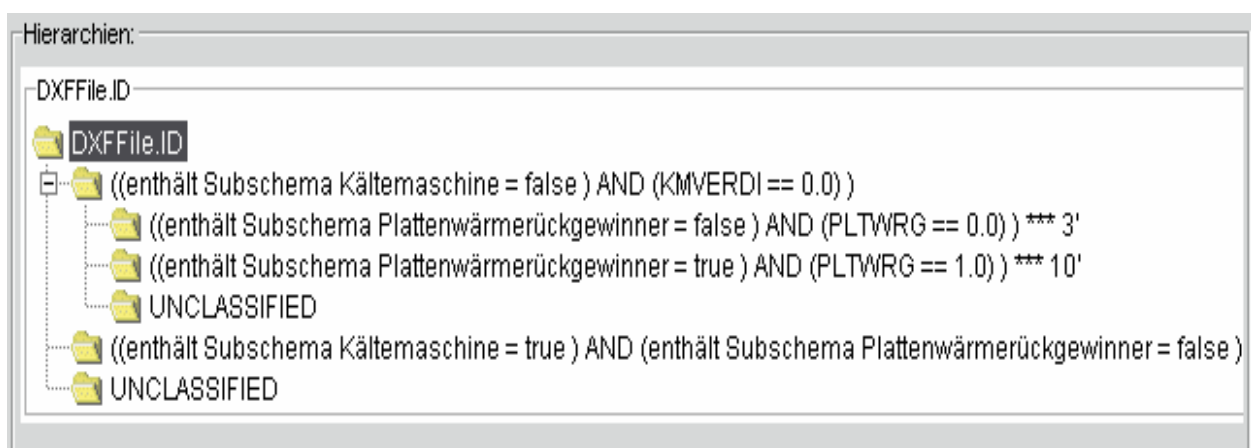
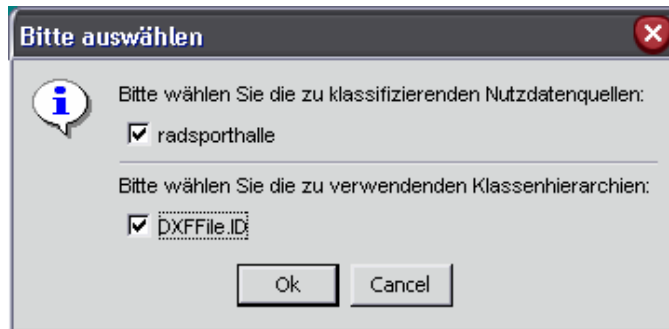


Bild 4-16: KLASTER-Anzeige der generierten Klassenhierarchie

**(6) Klassifizierung durchführen**

Ausgewählte Dateien klassifizieren lassen



PAK Ergebnisse

Suchen:  Klassen  Resultate

Hierarchie

- DXFFile.ID
  - ((enthält Subschema Kältemaschine = false ) AND (KMVERDI == 0.0) )
    - ((enthält Subschema Plattenwärmerückgewinner = false ) AND (PLTWRG == 0.0) ) \*\*\* 3'**
    - ((enthält Subschema Plattenwärmerückgewinner = true ) AND (PLTWRG == 1.0) ) \*\*\* 10'
    - UNCLASSIFIED
  - ((enthält Subschema Kältemaschine = true ) AND (enthält Subschema Plattenwärmerückgewinner = false ) AND (KMVERDI: [1.0..2.0] ) AND (PLTWRG == 0.0) ) \*\*\* 16'
  - UNCLASSIFIED

Ergebnisse

Nr.	Schwellwert	DXFFile.ID	PLTWRG	GLKG	ELANTR	KMVERDI	REP-SCHA	RAUMP2	DXInsert.B...	PUMPE	DXFFile.FILE...
	Schwellwert...	radsporthall...	'implizit' 'use...	'implizit' 'use...	'implizit' 'use...	'implizit' 'use...	'implizit' 'use...	'implizit' 'use...	'radsporthall...	'implizit' 'use...	'radsporthall...
1	1.0	1	0	3	3	0	4	0	DINA2SS,ZD...	0	FVR4_2
2	1.0	1301	0	0	7	0	1	0	DINA2SS,ZD...	1	KT_2
3	1.0	1532	0	0	2	0	2	0	DINA2SS,ZD...	2	KT_3
4	1.0	1789	0	0	2	0	2	0	DINA2SS,ZD...	2	KT_4
5	1.0	2556	0	0	3	0	0	0	DINA2SS,ZD...	0	RLKG46_1
6	1.0	2821	0	3	6	0	2	0	DINA2SS,ZD...	1	RLT_R05
7	1.0	3239	0	3	10	0	1	0	DINA2SS,ZD...	1	RLT_R10
8	1.0	3811	0	1	5	0	2	0	DINA2SS,ZD...	2	RLT_R40
9	1.0	4517	0	0	2	0	0	0	DINA2SS,ZD...	2	RSHEIZ_2
10	1.0	4703	0	0	3	0	0	0	DINA2SS,ZD...	3	RSHEIZ_4
11	1.0	4937	0	0	0	0	0	0	DINA2SS,ZD...	6	RWWB_2
12	1.0	5190	0	0	3	0	2	0	DINA2SS,ZD...	5	RWWM2
13	1.0	5553	0	0	3	0	0	0	MPGNRA2,M...	2	WRG und He...

Schließen Klasse als Exceltabelle speichern In XML-Datei speichern In Datenbank speichern

Bild 4-17: PAK-Anzeige der Klassifikationsergebnisse

**4.1.3.3 Testdaten**

Als Testdaten wurden die R&I-Schematas aus der Gebäudeautomatisierung von zwei Großprojekten (Sporthallen) für die Funktionsbereiche für Heizung, Lüftung und Klima (HLK) zusammengestellt. Die Test-Daten bestanden aus 1300 R&I-Schematas. Diese enthalten 85 HLK – Grundsymbole und 28 HLK – Subschematas.

Zur Zusammenstellung des Datenbestandes aus DXF-Dateien zu RI-Schemas, Subschemas und Grundsymbole wird in den nachfolgenden Abbildungen charakterisiert.

**Grundsymbole:**

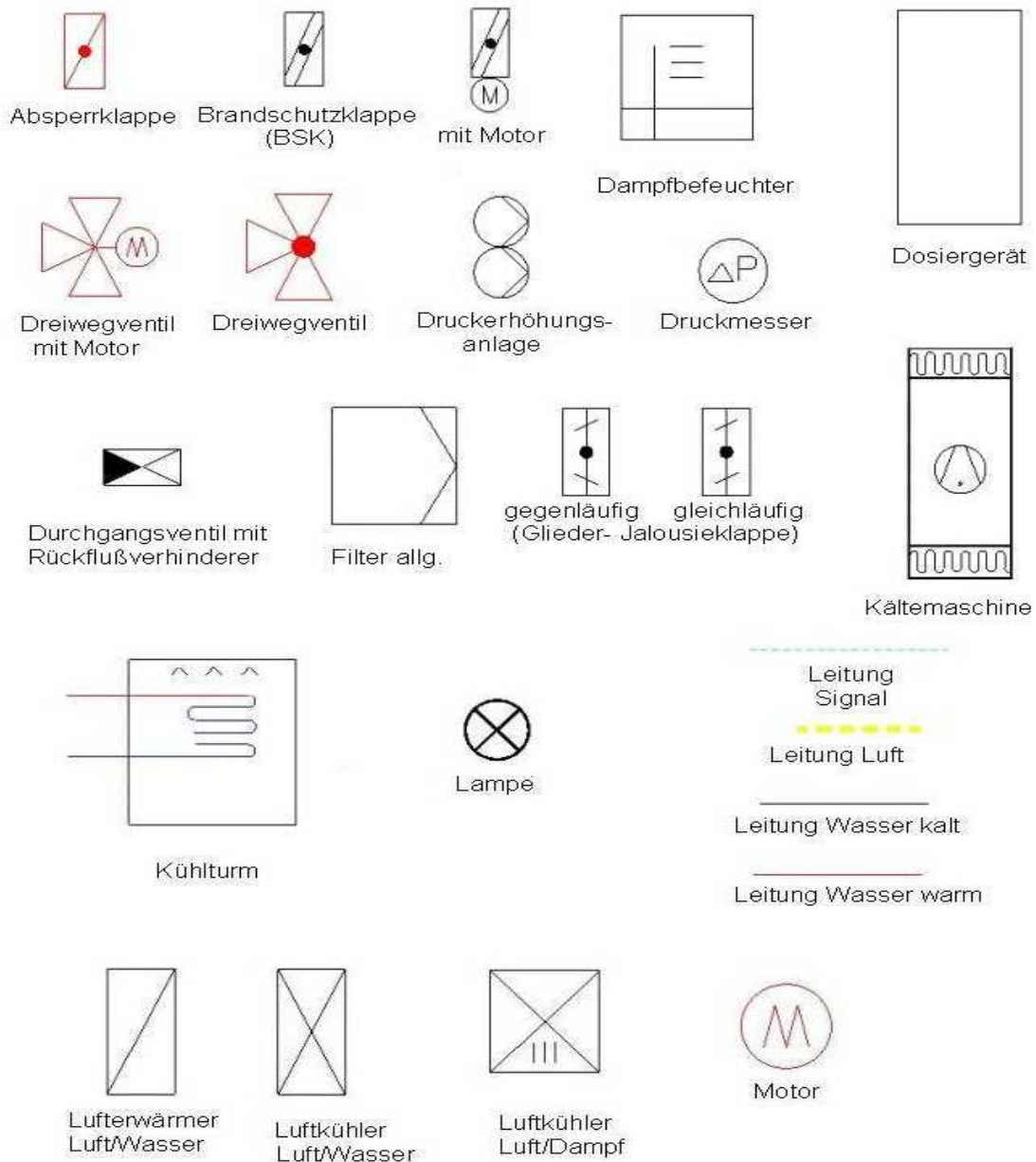


Bild 4-18: Auswahl (1) Grundsymbole

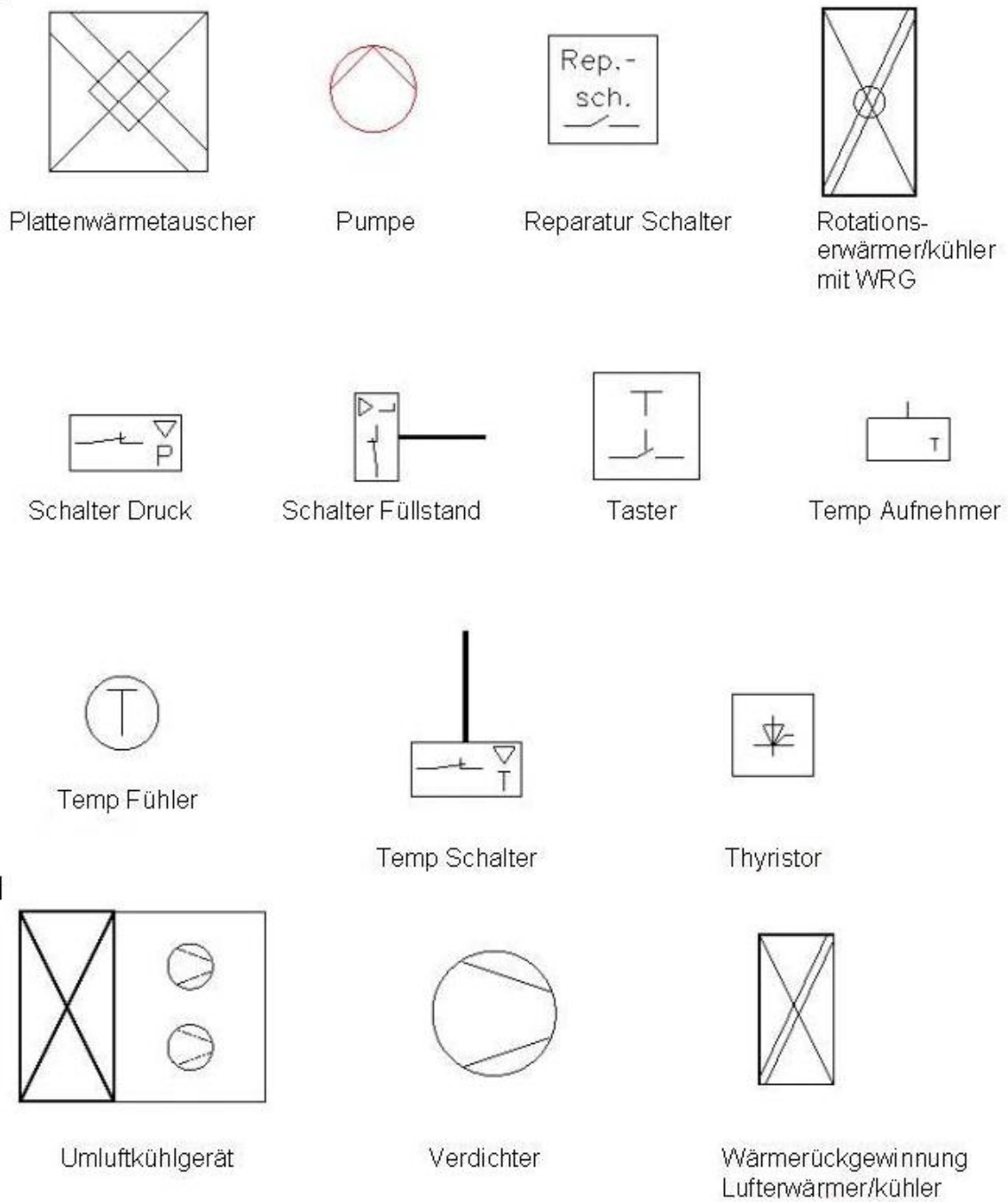


Bild 4-19: Auswahl (2) Grundsymbole



Bild 4-20: Auswahl (3) Grundsymbole

Klassifiziert wurden Funktionseinheiten für die **Funktionskomplexe:**

- Zuluft,
- Abluft,
- Zuluft + Abluft,
- Kälte und Entrauchung.

Nachfolgend (siehe Bild 4-21) wird eine Auswahl von Substrukturen dargestellt, die relativ häufig und meist komplett immer wieder eingesetzt werden. Durch die Klassifizierung der vorhandenen Daten bzw. Projektdokumente können solche Wiederverwendungs- wie auch Wiederholstrukturen herausgefunden werden.



**Substrukturen:**

Subschemata der Heizungs- Klima- und Belüftungstechnik

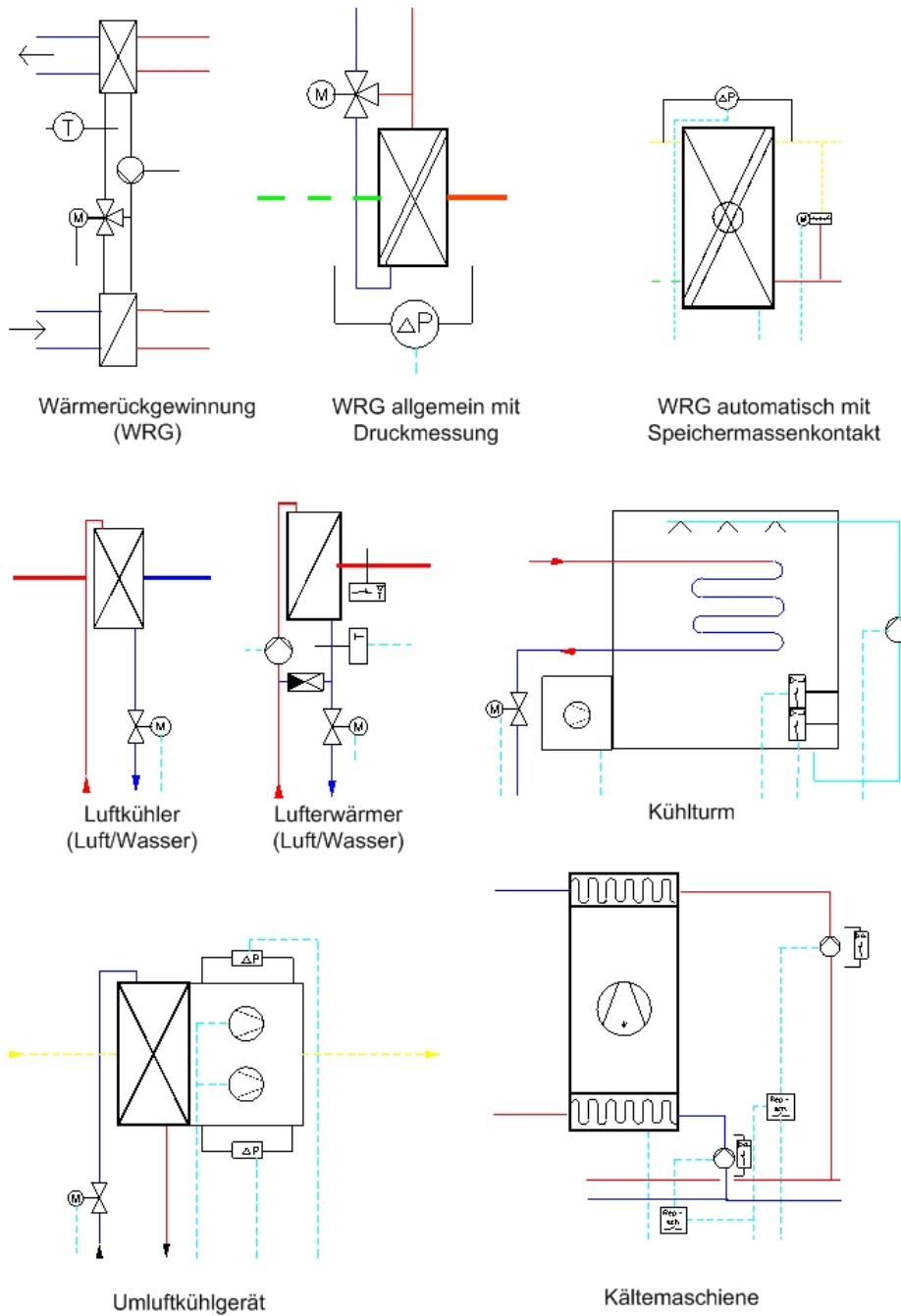


Bild 4-21: Substrukturen zu Heizung, Lüftung, Klima

#### 4.1.4 Ergebnis und zukünftige Nutzung

##### 4.1.4.1 Bewertung der Erprobungsergebnisse

Der Test ergab, dass bei Großprojekten im HLK-Bereich bis zu 90 % der Funktionseinheiten Wiederholstrukturen sein können, die auch in weiteren ähnlichen Projekten wieder eingesetzt werden können.

Wenn es gelingt, zu diesen wiederverwendbaren Strukturen auch alle ERP-Daten zugriffsfähig zu verwalten, kann bei neuen Projekten in Planungs- wie in der Realisierungsphase erheblich an Aufwand eingespart werden. Diese Einsparungen sind in allen Phasen des Life-Cycle eines Automatisierungsprojektes zu erreichen, also nicht nur in Planungs- und Realisierungsphasen sondern insbesondere auch im Betrieb durch die Betreiber, d.h. bei dem Kunden. Aufgrund der hohen Investitionskosten müssen Automatisierungssysteme für eine lange Lebensdauer ausgelegt sein und ständig instand gehalten, an neue Anforderungen angepasst, ertüchtigt und modernisiert werden. Dazu ist ein aufwendiger Service erforderlich, bei dem der Aufwand zum Management der technischen Unterlagen durch die KLAUSTER-Methode reduziert werden kann.

Insgesamt stellt die KLAUSTER-Methode einen universellen Ansatz zum Content-Management dar. KLAUSTER ist ein Forschungsergebnis, das durch weiteren Entwicklungsaufwand in der Praxis wirksam gemacht werden kann. Dabei sind die Verfahren zur Features-Erkennung und Auswertung in Dokumenten und Grafiken weiter zu entwickeln und zu optimieren.

##### 4.1.4.2 Zukünftige Nutzungsszenarien für KASTER

Die Anwendung des KLAUSTER-Lösungsansatzes wird neben der Erzeugung von Klassifizierungssystemen für nicht klassifizierte Datenbestände in der Anlagen- und Prozessautomatisierung vor allem in der Erzeugung von Schlüsselsystemen für projektspezifische Anlagendokumentationen gesehen. Ein solches Schlüsselsystem ist als Bestandteil der Anlagensoftware zur Unterstützung der Inspektion und Wartung sehr interessant, da für den Service-Ingenieur somit der Zugriff auf Dokumentationen von Bauteilen und Funktionsgruppen mit weniger Aufwand verbunden ist.

Desweiteren können über den KLAUSTER-Ansatz nicht mehr konsistente Datenbestände durchsucht werden. KLAUSTER bietet auch für eine automatisierte Querschnittsdurchsuchung eines größeren Datenbestandes wertvolle Hilfe, z. B. wenn auf der Grundlage unterschiedlicher Sichten (z.B. Funktionssicht, Ortssicht, Stammdatensicht u.a.) gesucht wird.

Wichtigstes Ziel zur KLAUSTER-Anwendung besteht somit im Finden wiederverwendungs-fähiger technischer Unterlagen bzw. Produktlösungen. Aus der Evaluierung der KLAUSTER-Lösung wurden folgende **Szenarien zur Anwendung von Klassifikationssystemen** abgelei-

tet, die über eine automatisierte Merkmalsgewinnung aus vorhandenen Datenbeständen zu Projekten der Anlagenautomatisierung generiert werden:

### (1) Recherche nach wiederverwendbaren Projektunterlagen

Input: Suchanfrage gibt Suchmerkmale vor

KLASTER: Finden von Merkmalen aus Regelschematas, Funktionsbeschreibungen, Gerätenamen, Materiallisten → Generierung des Klassifikators → Suche im Archiv

Ergebnis: *wiederverwendbare Unterlagen*

### (2) Materialbestellung

Input: Vorgabe technischer Daten

KLASTER: Finden der Merkmale aus Materialstammdaten → Generierung des Klassifikators → Suche in Materialstammdaten, elektronischen Angebotskatalogen

Ergebnis: *Vorschläge über Einkauf/Bestellisten*

### (3) Ausführungsplanung / Erzeugung der technischen Unterlagen

Input: Anlegen neuer Entwurfsdokumente

KLASTER Finden von Merkmalen der zu klassifizierenden neuen Funktionseinheiten → Generierung des Klassifikators → Vergabe eines Schlüssels für das Entwurfsdokument

Ergebnis: *klassifizierte Entwurfsdokumente für elektrotechnische Anlagen*

### (4) Projektierung von Wartungs- bzw. Instandhaltungsprozessen

Klassifizierung der für die Instandhaltungsprozesse relevanten technischen Unterlagen zwecks einer aufgabenorientierten Verlinkung dieser Dokumente, um im Störfall insbesondere in sehr komplexen Anlagen die Suche der Störungsursache sowie die Wartungsarbeiten schnell durchführen zu können.

Input: alle für die Fehlersuche und Wartung relevanten Dokumente zu einer bestimmten Funktion der Anlage

KLASTER Finden von Merkmalen zu dieser Funktion → Generierung des Klassifikators → Vergabe von Schlüsseln und Linkadressen als XML-Anhang der klassifizierten Dokumente

Ergebnis: *klassifizierte und verlinkte Wartungsunterlagen*

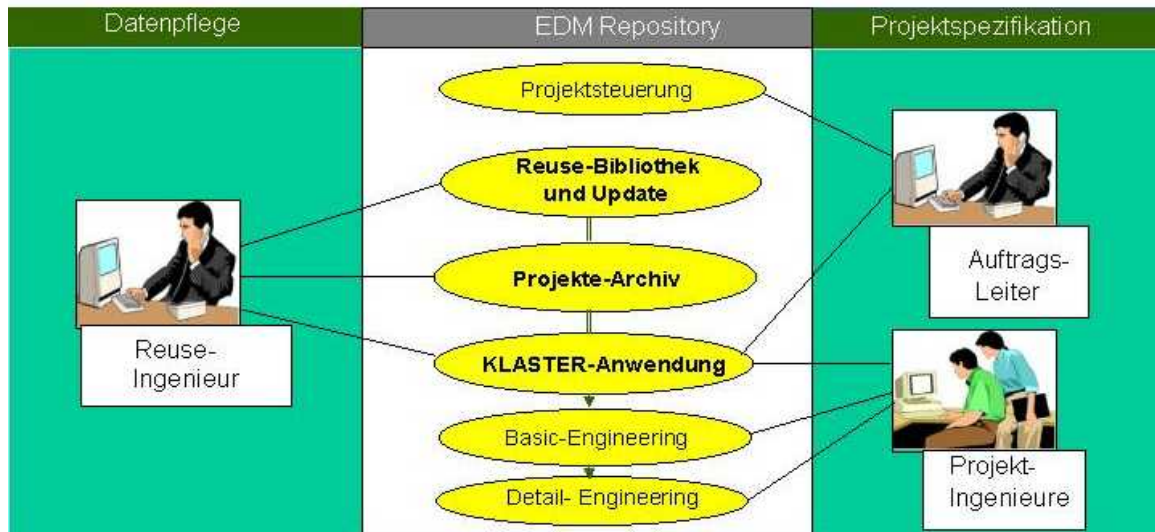


Bild 4-22: Anwendungsszenarium von KLASSTER im Projektierungsprozess

Die Anwendung des KLASSTER-Lösungsansatzes wird vor allem in der Erzeugung von Schlüsselsystemen für spezifische Anlagendokumentationen gesehen. Solche Anlagendokumentationen können z. B. in Web-fähigen Service-Portalen vorgehalten und gepflegt werden. Ein mit KLASSTER generiertes Schlüsselsystem ist als Bestandteil der Anlagensoftware zur Unterstützung der Inspektion und Wartung über Web-Portale für Engineering und Service sehr interessant, da für den Service-Ingenieur der Zugriff auf relevante Dokumentationen von Bauteilen und Funktionsgruppen mit wesentlich weniger Aufwand verbunden ist.

Durch eine zukünftige Realisierung von Web-Portalen für Engineering und Service sind zusätzliche Mehrwert-Dienste sowohl für den Betreiber als auch für den Service-Dienstleister möglich und die so ausgerüsteten technischen Anlagen erfahren einen erheblichen Wertzuwachs. Der zusätzliche wirtschaftliche Nutzen durch ein Engineering- und Service-Portal innerhalb der veranschlagten Lebenszyklus-Zeit einer Automatisierungsanlage kann ca. bis 40% des Investitionsumfanges der Automatisierungsanlage eingeschätzt werden.

Mit solchen add-on-Lösungen zum Web-basierenden Remote-Service, in dem neben den Web-Technologien zur Engineering-Unterstützung die KLASSTER-Methode zur Daten- und Dokumentenrecherche genutzt wird, kann in zwei bis drei Jahren zusätzlicher Umsatz generiert werden und durch die genannte Wertsteigerung von Automatisierungsanlagen zusätzlicher Gewinn und erheblicher Nutzen für den Kunden erwirtschaftet werden.

## 4.2 Anbindung des CSM-Softwaresystems Remarc an die KLASSTER-Software

### 4.2.1 Problemstellung

Viele Unternehmen im Maschinenbau wechseln gegenwärtig auf eines der modernen 3D-CAD-Systeme und möchten die darin erzeugten Produktdaten mit einem PLM-System ver-

walten. In diesem Zusammenhang sind Sortimente von Bauteilen und Formelementen aufzubauen, welche die firmenspezifischen Auswahl- und Vorzugsreihen exakt abbilden, ohne zeitaufwändige, manuelle Modifikationen einsetzbar sind und optimal die Prozessdurchgängigkeit (z.B. automatische Ableitung von Stücklisten) unterstützen.

Die Auswahl eines ganz bestimmten, funktionserfüllenden Bauteils oder Formelements erfolgt durch eine Vielzahl von Mitarbeitern mit z.T. deutlich differierender Problemnähe und Erfahrung. Ein Dialog auf Basis einer Klassenhierarchie soll weitgehende Konsistenz der Entscheidung und effiziente Arbeitsweisen garantieren.

In einem solchen Projekt sind – neben der eigentlichen Implementierung der CAD-PLM-Software - die Kosten einer vollständigen Migration der teilweise bereits in ERP-Systemen existierenden Datenbestände an Norm- und Kaufteilen, die Bereitstellung entsprechender CAD-Geometrien sowie deren beider Verknüpfung zu berücksichtigen.

Das folgende Szenario will aufzeigen, wie eine Lösung der o.g. Problemstellung unter Nutzung marktüblicher Datenbestände mit KLAUSTER-Software unterstützt werden kann.

Als Beispielumgebung wurden das CAD-System Unigraphics R.18 und das PLM-System mySAP PLM gewählt.

#### **4.2.2 Basisdaten**

Für die „Befüllung“ o.g. Systemumgebungen stehen heute eine Vielzahl proprietärer und standardisierter Datenbestände am Markt zur Verfügung, welche die Beschreibung von mechanischen Norm- und Kaufteilen beinhalten.

Für das vorliegende Szenario wurden exemplarisch standardisierte Verwaltungsinformationen mechanischer Bauteile im Format DIN 4000 und ISO 13584-31 genutzt, die direkt dem Liefersortiment der DIN Software GmbH entnommen wurden (vgl. Bild 4-23).

Daneben wurde durch einen Kunden ein Auszug aus seinen Stammdaten in einem tabellierten ASCII-Format sowie ein firmenspezifisches Klassifikationsschema zur Verfügung gestellt.

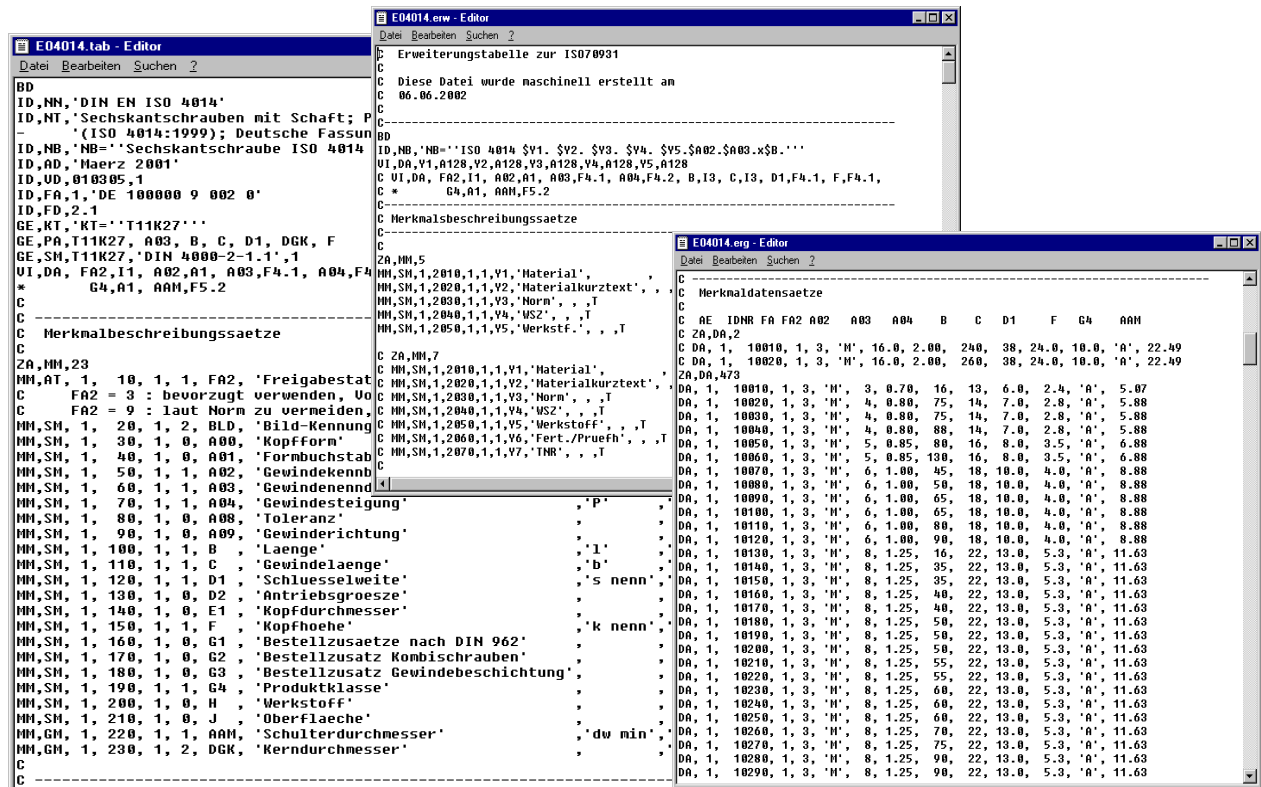


Bild 4-23: Basisdaten DIN 4000

Zur automatischen Generierung der korrespondierenden CAD-Geometrien wurde ein CAD-nativer Partgenerator (Modul cadpartout; Option Unigraphics) aus dem Component Framework REMARC (Hersteller: ARC Solutions GmbH) benutzt.

#### 4.2.3 Aspekte der Realisierung

Der Lösungsansatz geht von einem semantischen Modell aus, das sowohl die standardisierten Merkmale einer speziellen Normreihe (Sachmerkmalsleiste) wie auch deren firmenspezifischen Ausprägungen zusammenführt.

Hierzu werden Merkmale aus Merkmalsalgorithmen und Funktionsmerkmale in ihre diskreten Ausprägungen aufgelöst und damit als Einzelparts verarbeitbar. Auf Basis dieses Modells lassen sich beliebig detaillierte Sichten für die verschiedenen Zielumgebungen ableiten.

Die Realisierung erfolgte in Form eines mehrstufigen, XML-basierten Mediators. Die Steuerung und wiederholte Ausführung der einzelnen Prozessschritte wurde mittels ANT-Skripten realisiert (vgl. <http://ant.apache.org/index.html>).

Die erste Stufe bildet die Normalisierung der Daten im DIN4000- und ASCII-Format auf das Format XML (eXtensible Markup Language). Dabei werden die Ursprungsdaten gewandelt und gleichzeitig werden Merkmalwerte mittels Substringoperationen separiert.

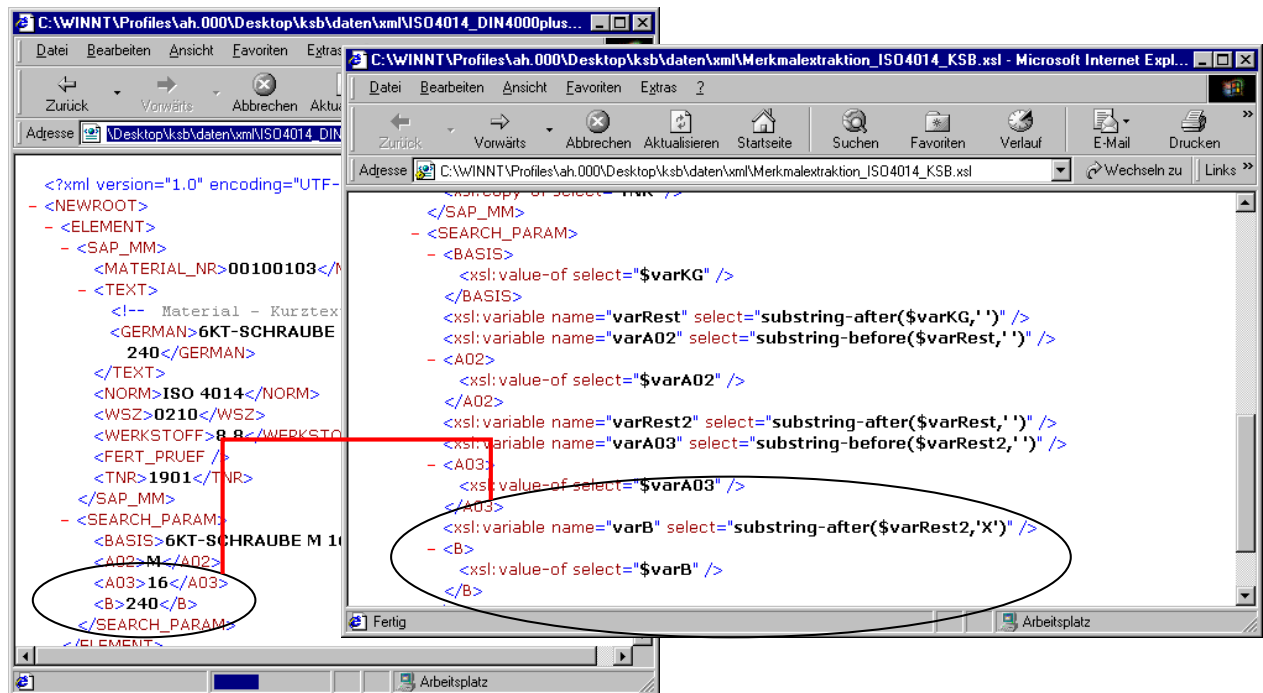


Bild 4-24: XML- Normalisierung

Im zweiten Schritt erfolgt die automatisierte Verknüpfung von DIN- & ERP-Daten anhand der normalisierten Daten im XML-Format (vgl. Bild 4-24).

Als Schlüsselattribute dienen die Bezeichnung der Norm (hier: DIN EN ISO 4014) in Verbindung mit den Nennparametern (hier: Nenngröße M16 UND Nennlänge 240 mm), schematisch dargestellt in Bild 4-25.

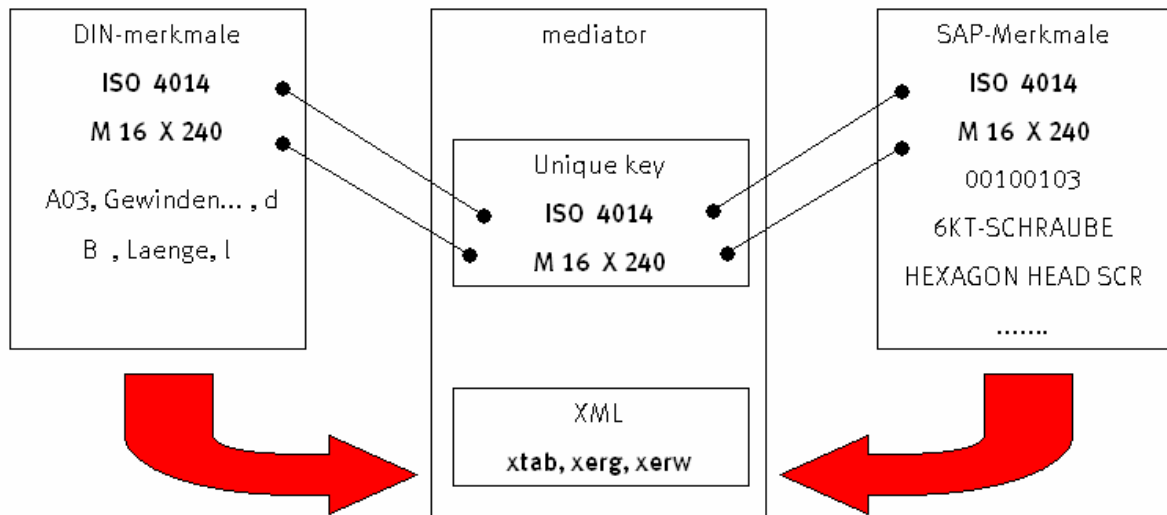


Bild 4-25: Automatisierte Verknüpfung von DIN- & ERP-Daten

Nach der Anpassung der Rohdaten konnte mit der Klassifikation begonnen werden. Zunächst wird jeder Normreihe (hier: DIN EN ISO 4014) anhand des vorgegebenen Schemas (vgl. Bild 4-26) ihre Superklasse (hier: „Schraube“) fest zugeordnet, die wiederum alle zur Weiterverarbeitung benötigten relevanten Merkmale definiert (Superklasse „Schraube“ definiert die Merkmale Nenndurchmesser, Nennlänge und firmenspezifische Artikel.-Nr.).

SAP-Klassenname	Name	SAP-Klassenname	Name
BUSH	Buchse	PARALLEL_KEY	Passfeder
SEAL	Dichtung	RING	Ring
E_CABLE_PART	E_Kabelteil	PIPE	Rohr
FITTING	Fitting	SHACKLE	Schaekel
FLANGE	Flansch	DISC	Scheibe
THREADED_INSERT	Gewindeeinsatz	PIPE_CLAMP	Schlauchbefestigung
GRUB_SCREW	Gewindestift		Schlauchkupplung
	Gleitringdichtung	SCREW	Schraube
SEMI_FINI_PRODUCT	Halbzeug		Schutzkappe
	Kabelbinder	CIRCLIP	Sicherungsring
	Kabelverschraubung	PIN	Stift
	Klemme		Stopfen
	Kupplung		Unterlegblech
NUT	Mutter	PIPE_UNION	Verschraubung
NAIL	Nagel	ANTI_FRICT_BEARING	Waelzlager
	Nippel		

Bild 4-26: vorgegebenes Klassenschema



Genügt die festvorgegebene Segmentierung nicht, kann der KLAUSTER Editor genutzt werden, um zusätzlich dynamisch zu segmentieren.

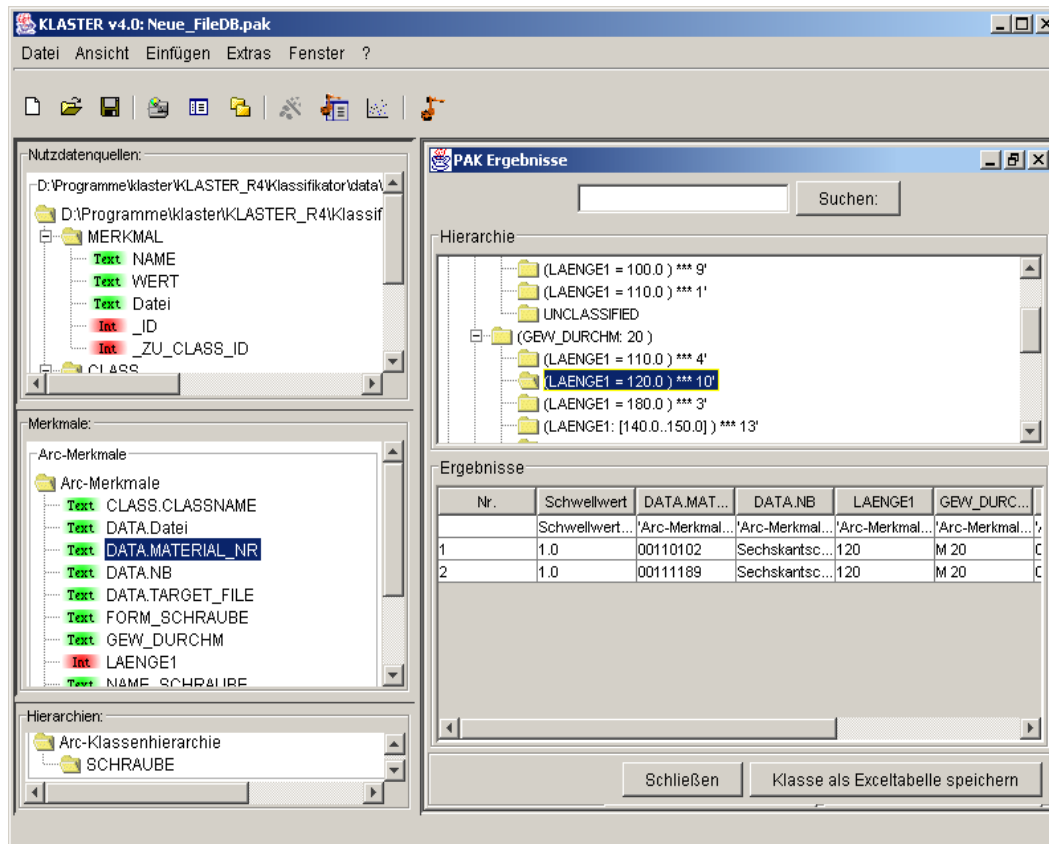


Bild 4-27: Clusterbildung für Schrauben

Zunächst werden aus dieser Menge von Merkmalen wieder die Schlüsselwortmerkmale bestimmt, die entscheidend sind für die Struktur des Klassensystems, und aus diesen Merkmalen Schlüsselworte extrahiert (s. Bild 4-27; weiteres Vorgehen - vgl. Absatz 3).

Für den Import nach CAD und PLM System (Zielumgebung), der Befüllung des Klassensystems und Verknüpfung mit dem Materialstamm wurden mittels des Moduls REMARC UG cadpartout die UGNX-Partgeometrien erzeugt und den parallel aufbereiteten Merkmalfiles zugeordnet.

Im Einzelnen wurden für den Import folgende Daten erzeugt:

- a) UGNX-natives Part mit konfigurierbaren Attributen (vgl. Bild 4-28)

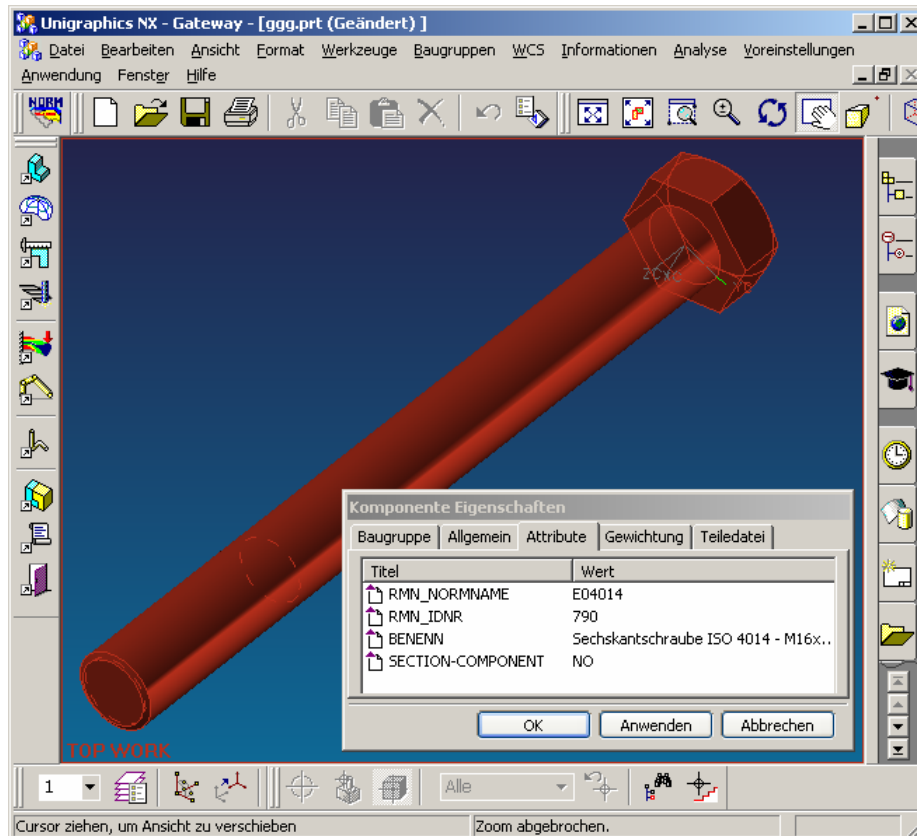


Bild 4-28: Automatisch generiertes, natives CAD-Part

Bei der Generierung der CAD-Parts werden die Parameter der systemneutralen Prozeduren nach ISO 13584-31 mit den aufbereiteten Merkmalswerten initialisiert.

Diese greifen über das API (application programming interface) direkt auf geometrie- und attributerzeugende Routinen sowie auf die Konfigurationen und Nutzereinstellungen von Unigraphics zu und bauen auf diese Weise ein natives Part auf.

Jedes Part erhält einen eigenen entsprechend vorgegebener Regeln generierten Filenamens.

## b) XML-Merkmalfile für Import nach SAP (Textauszug)

```
<NB>Sechskantschraube ISO 4014 - M 16 x 240</NB>
  <TARGET_FILE>100008154711.prt</TARGET_FILE>
  <MATERIAL_NR>100008154711</MATERIAL_NR>
  - <!-- MERKMALSMAPPING -->
  <CLASSNAME>SCHRAUBE</CLASSNAME>
  - <MERKMAL>
    <NAME>NAME_SCHRAUBE</NAME> <WERT>01</WERT>
    </MERKMAL>
  - <MERKMAL>
    <NAME>NORM_SCHRAUBE</NAME> <WERT>DIN EN ISO 4014</WERT>
    </MERKMAL>
  - <MERKMAL>
    <NAME>FORM_SCHRAUBE</NAME> <WERT>mit Schaft</WERT>
    </MERKMAL>
  - <MERKMAL>
    <NAME>GEW_DURCHM</NAME> <WERT>M 16</WERT>
    </MERKMAL>
  - <MERKMAL>
    <NAME>LAENGE1</NAME> <WERT>240</WERT>
    </MERKMAL>
</DATA>
```

Um Metadaten unabhängig von der Attributstruktur des CAD-Parts in das Datenmanagementsystem übertragen zu können, wurde eine parallele Transfermöglichkeit realisiert.

Diese basiert auf einer klassenabhängig vorgegebenen Struktur von Merkmalen, deren Werte aus firmen- und normspezifischen Quellen stammen.

Um später im Datenmanagementsystem eine automatische Zuordnung zu dem jeweiligen CAD-Part zu erreichen, wird der im Prozess nach vorgegebenen Regeln generierte Filename (TARGET\_FILE; hier identisch SAP-Materialnummer) mit übergeben.

#### 4.2.4 Ergebnisse

KLASTER ist ein Ansatz bei der Erstellung und Befüllung von Klassensystemen mit dessen Hilfe die verschiedenen Wege und Möglichkeiten der Klassifizierung zeitnah ausprobiert und ein prototypisches Klassensystem erstellt werden kann.

Entstanden ist eine prototypische Lösung mit hohem Rationalisierungseffekt. Mit dieser wurden bis zum Abschluss des Projekts bei inhouse- und Praxistests (mit Kundendaten) bereits mehr als 13.000 CAD-Bauteile automatisiert erzeugt und in die Zielumgebung importiert.

Mit den nunmehr vorliegenden Erfahrungen wäre im nächsten Schritt die Implementation eines granular konfigurierbaren, performanten Prozesses für die Verarbeitung von Massendaten denkbar.

Die Konfigurationsdaten für diesen Prozess sollten sich dabei mit einem Tool vergleichbar dem KLASTER-Editor explorativ ermitteln und übertragen lassen. Damit würde es möglich, Norm- und Kaufteilbestände, welche regelmäßig analysiert und angepasst werden, zeitnah in den verschiedenen Zielsystemen zu aktualisieren.

### 4.3 Klassifikation von Maschinenelementen in der Produktentwicklung durch Anbindung an Unigraphics

#### 4.3.1 Problemstellung

Viele Unternehmen des Maschinenbaus nutzen seit vielen Jahren CAD-Systeme. In diesem Zusammenhang sind Bauteile in einer großen Anzahl entstanden, die nicht bereits bei ihrer Entstehung einer bestimmten Teileklasse zugeordnet worden sind.

Es wird angenommen, dass auch diese individuell und unabhängig voneinander entwickelten Teile ein erhebliches Wissens- und Wiederverwendungspotenzial repräsentieren.

Allerdings ist die IT-gestützte Recherche nach diesen Teilen bisher nicht möglich, ohne sie vorher aufwändigen Vorbehandlungen (Klassifikation) mit einem hohen Anteil manueller Tätigkeit zu unterziehen.

Das folgende alternative Szenario will aufzeigen, wie klassifikationsrelevante Daten über solche Bauteile aus proprietären Quellen (hier: 3D-CAD) gewonnen und der KLASTER-Software zur Weiterverarbeitung zugeführt werden, um auf diese Weise eine Lösung der o.g. Problemstellung zu realisieren.

Als CAD-Beispielumgebung wurden das System Unigraphics R.18 bzw. 19 (NX) gewählt.

### 4.3.2 Basisdaten

Im vorliegenden Szenario standen aus einem SAP-basierten Archivsystem für die Teileanalyse digitale Zeichnungen im Rasterformat TIFF zur Verfügung.

Zusätzlich waren eine Anzahl Beispielteile als 3D-Volumenmodelle im CAD-System (Benennung <materialnummer.PRT> ) zu erfassen.

### 4.3.3 Vorgehensweise

Innerhalb des Szenarios waren zwei Teilaufgaben zu lösen:

- a) Analyse, Entwurf und Implementierung eines Interfaces und XML-Protokolls zwischen dem CAD-System Unigraphics und KLAUSTER.
- b) Entwurf eines authentischen Szenarios zur Evaluation, Zusammengetragen der Beispieldaten und Testdurchführung.

Innerhalb eines Projektteams wurde ein repräsentatives Spektrum von Einzelteilzeichnungen der Teileklasse „Labyrinthdichtung“, welches über 25 Jahre Entwicklungsarbeit im Hause des Projektpartners MAN TURBO AG und deren Vorgängerunternehmen repräsentiert, visuell gesichtet. Dabei wurden die Einzelteile auf ihre Geometriemerkmale und die damit zusammenhängende klassifizierende Bedeutung untersucht (vgl. Bild 4-29; Beispiel „Bohrung-achspannel“).

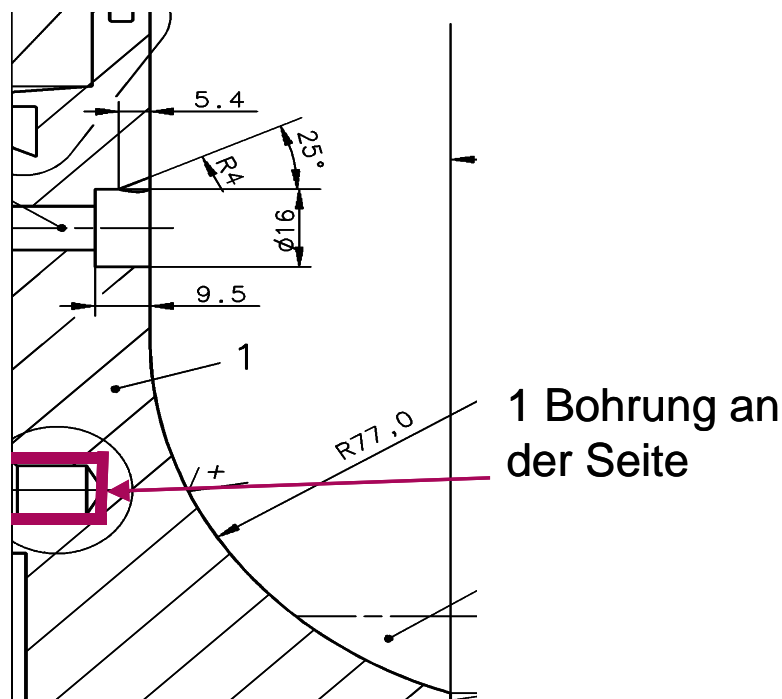


Bild 4-29: geometrisches Merkmal „Bohrung-achsparell“

Im nächsten Schritt wurden entsprechend den gefundenen geometrischen Merkmalen Wertebereiche zugeordnet und auf diese Weise ein „Merkmalsraum“ definiert (vgl. Bild 4-30).

	Vorhanden	Sackloch	Lage zur Rotationsachse	Durchmesser	Verteilung/axial	Verteilung/radial	Gewinde
<b>Bohrung-quer</b>	ja	ja	radial-senkrecht	gestuft	W mal hintereinander	Y mal auf Umfang	ja
	nein	nein	radial-schräg	nicht gestuft			nein

	Vorhanden	Sackloch	Durchmesser	Verteilung/radial	Gewinde
<b>Bohrung-achsparell</b>	ja	ja	gestuft	Y mal auf Umfang	ja
	nein	nein	nicht gestuft		nein

	Element 1	Konturtyp
<b>Kontur-Mantel</b>	Zylinder	innen
	Konus	außen
	Torus	
	Ebene	

Bild 4-30: geometrische Merkmale der Teileklasse „Labyrinthdichtung“ (Auszug)

Auf dieser Basis wurde folgende Arbeitsdefinition getroffen:

*Die Klasse der Labyrinthdichtungen ist demnach eine Menge von Einzelteilen, die der Eigenschaft der Rotationssymmetrie unterliegt und über mindestens eine Querbohrung und eine Anströmkontur verfügt, die sich untereinander in Lage und Anzahl der Längs- und Querbohrungen unterscheiden.*

Um den Test der zu implementierenden Schnittstelle unter praxisnahen Bedingungen durchführen zu können, wurden hierzu insgesamt elf der untersuchten Labyrinthdichtungen im CAD-System Unigraphics als 3D-Volumenmodelle erfasst (vgl. Bild 4-31).

Um die Universalität des Ansatzes zu belegen, wurden dabei bewusst unterschiedliche Modellieretechniken für identische bzw. ähnliche Aufgabenstellungen angewandt.

Beispielsweise wurden Nuten einerseits als separate Formelemente (sogenannte Userdefined Features; UDF) und andererseits als Konturabschnitte innerhalb der Profilkontur eines rotatorischen Sweepkörpers modelliert.

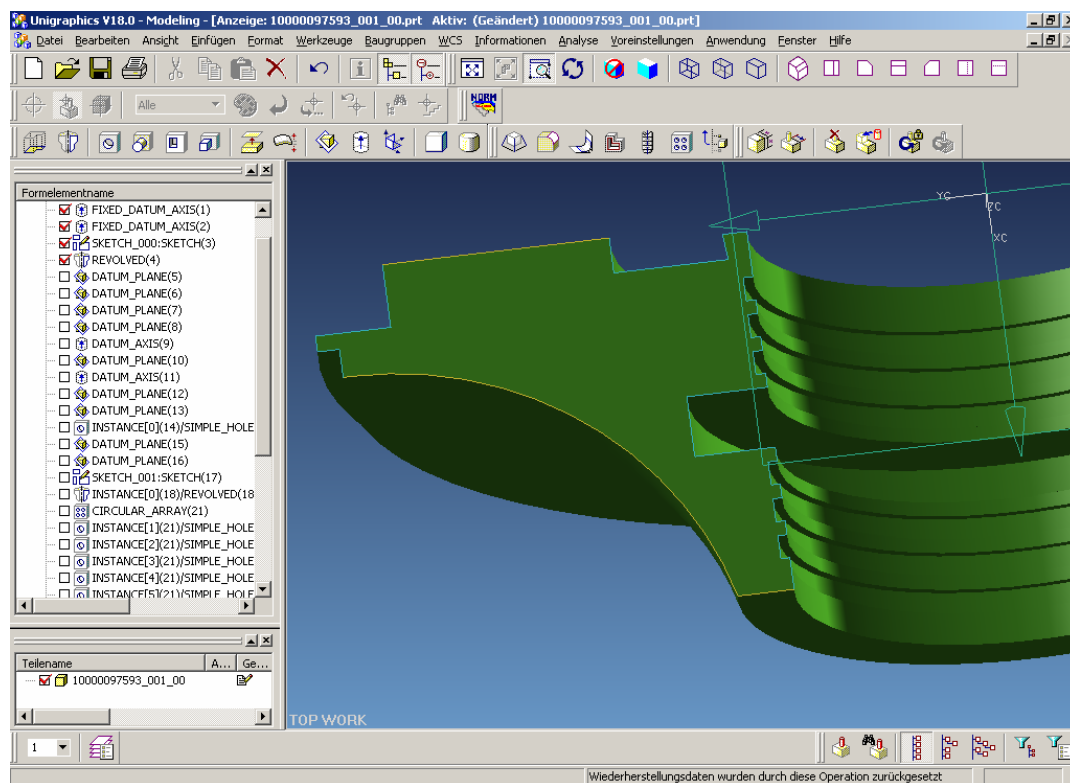


Bild 4-31: Beispielteil der Klasse „Labyrinthdichtung“ in Unigraphics

Die Implementation der Schnittstelle (vgl. Bild 4-32) nutzt die vorhandene API von Unigraphics (UG/Open) sowie den XERCES C++ Parser (<http://xml.apache.org/xerces-c/>) zur Extraktion und Aufbereitung der Flächeninformationen (patches).

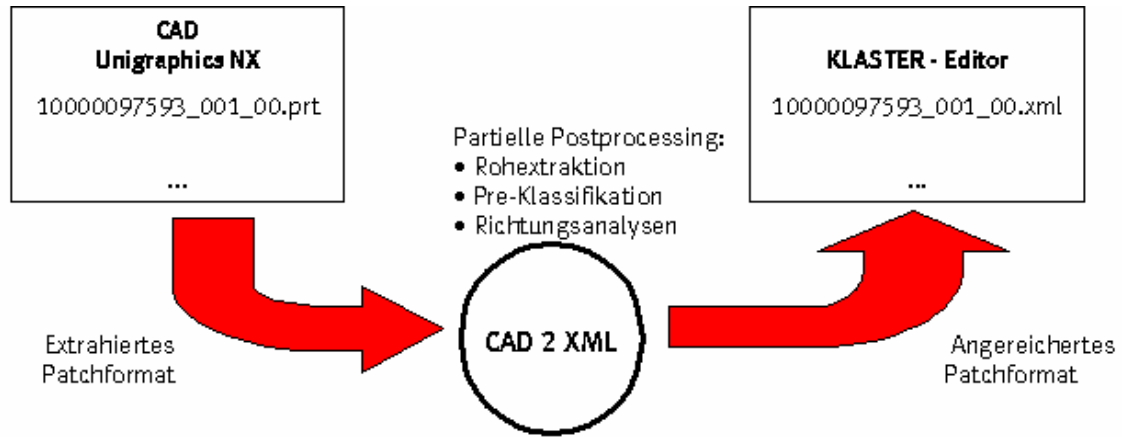


Bild 4-32: Architektur der Schnittstelle

Die im ersten Prozessschritt erzeugten Daten in einem sogenannten Roh- oder Patchformat (XML) entsprechen dem in Bild 4-33 und Bild 4-34 auszugsweise dargestellten Schema.

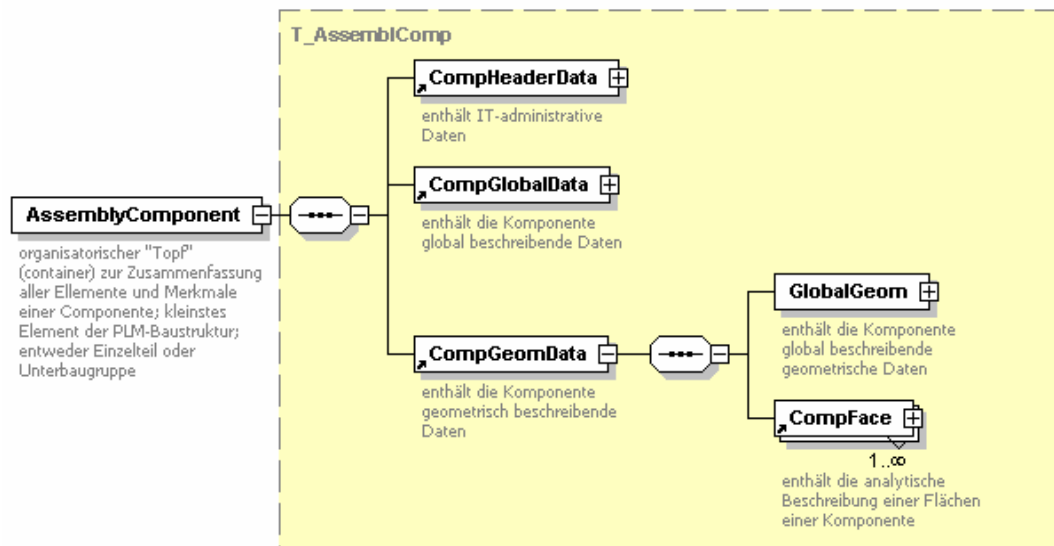


Bild 4-33: XML-Schema „AssemblComp“

Im Einzelnen ist die Schnittstelle in der Lage, folgende Informationen zu extrahieren:

- **Allgemeine Teilinformationen** wie Benennung, Name des Erstellers, Erstellungsdatum/-zeit des CAD-Modells



- **globale geometrische Informationen** wie Länge, Breite, Höhe, Volumen, Fläche, umgebender Quader.

- **Flächeninformationen:**

Gestalt: Typ (Plane, Cylinder, Cone, Revolved); Konvex/Konkav; Radius, Länge, Breite, Höhe, Flächeninhalt

Position: Koordinaten der Fläche; Lage zur Hauptachse

Berandung: Typ der Elemente (Kurve oder Linie); Geometrie der Elemente (Mittelpunkt, Radius, Start- und Endpunkt)

Achse: Richtung; Startpunkt; Eindeutige ID der Achse; Hauptachse ja/nein

Bounding Box: Koordinaten des umgebenden Quaders (des Flächenstücks)

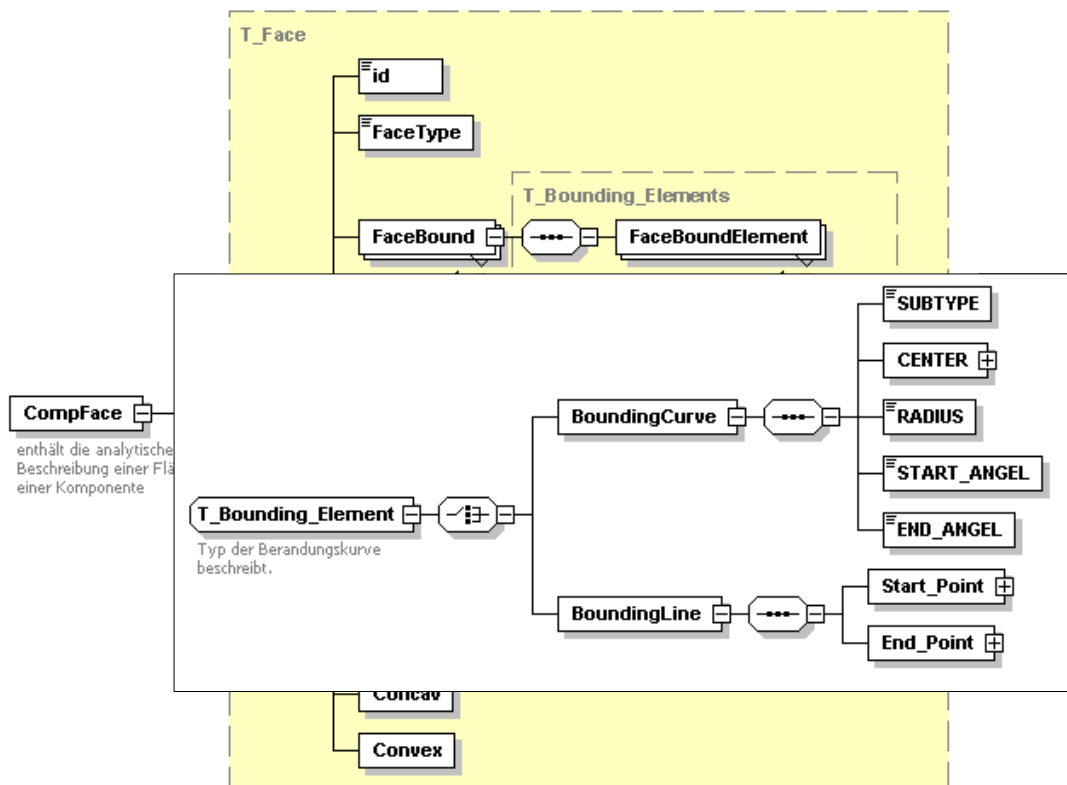


Bild 4-34: XML-Schema „CompFace“

An die Extraktion der Rohdaten und deren Ausgabe in eine sogenannte Basis-XML-Datei schließt sich ein weiterer XML-Postprozess an, der das Ergebnis für die Verarbeitung in der KLAUSTER-Software partiell aufbereitet und dabei zusätzliche Information einfügt.

Dabei ermittelt der Postprozess Merkmale wie Rotationssymmetrie, analysiert die Richtung von Flächen (radial, koaxial) und stellt zusammengehörige Flächen, sogenannte Flächenver-

bände – hier am Beispiel der "Bohrung" diskutiert – fest, die wiederum Konstruktionselemente konstituieren.

Eine Methode zur Ermittlung einer Bohrung geht bspw. davon aus, dass eine Bohrung aus einem oder mehreren Flächenstücken (faces) vom Typ Zylinder (cylinder), Kegel (cone), Drehfläche (revolved) oder bogenberandeter ebener Fläche (plane (arc)) besteht.

Darüber hinaus gilt für jedes Flächenstück einer Bohrung, dass mindestens eine ihrer Berandungskurven (facebound) mit der Berandungskurve des folgenden Flächenstückes zusammenfällt, dass ihre Achsen ebenfalls zusammenfallen (wird gekennzeichnet durch identische Axis-ID, vgl. Bild 4-35). Um 180 Grad gegenüber stehende Bohrungen sind hingegen zu separieren.

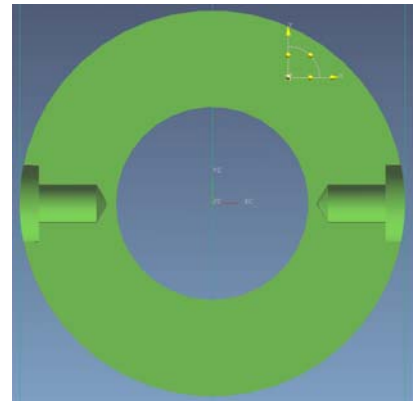
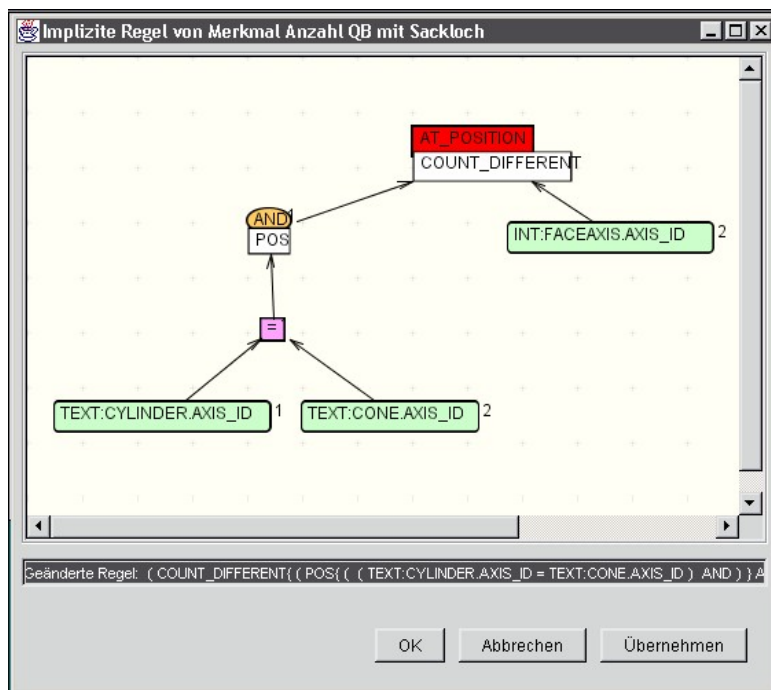


Bild 4-35: implizite KLASSTER-Regel „Anzahl Querbohrungen mit Sackloch“.

Im Ergebnis dieser Behandlung steht eine angereicherte XML-Übergabedatei zum Import in die KLAUSTER-Software zu Verfügung, welche zum gegenwärtigen Entwicklungsstand die Klassierung nach den folgenden Merkmalen erlaubt:

- enthält gestufte Parallelbohrung
- enthält Parallelbohrung
- enthält gestufte Querbohrung
- enthält Querbohrung
- enthält Querbohrung mit Sackloch
- enthält schräge Querbohrung

Hierarchie

- OBJECT.DOCUMENTNAME
  - (Enthält gestufte Parallelbohrung = false )
    - (Enthält Querbohrung = false ) \*\*\* 3'
    - (Enthält Querbohrung = true ) \*\*\* 4'
    - UNCLASSIFIED
    - (Enthält gestufte Parallelbohrung = true ) \*\*\* 5'**
    - UNCLASSIFIED

Ergebnisse

Nr.	Schwellwert	OBJECT.DOCUMENTNAME	Anzahl gestufter Parallelbohrung	PB Anz. FID	FACE ID	FACE.
	Schwellwert...	'8XML'. 'Not Relevant'. 'OBJEC...	'8XML'. 'erweitert Parallelbohrung'!	'8XML'. 'erwe...	'8XML'. 'erwe...	'8XML'.
1	1.0	1-11865394	8	16	20,21,22,23,...	20,21,2
2	1.0	1-25591667_001_03	7	22	13,15,20,21,...	13,15,2
3	1.0	10000097593_001_00	8	17	43,44,45,46,...	43,44,4

Schließen    Klasse als Exceltabelle speichern    In XML-Datei speichern    In Datenbank speichern

Bild 4-36: Art.-Nr.: 1-11865394 (besitzt 8 gestufte Parallelbohrungen)

#### 4.3.4 Ergebnis / Nutzen

Mit der Prototyplösung werden etwa 80 % der geometrischen Anforderungen abgedeckt. Es konnte gezeigt werden, dass sich der verfolgte Ansatz neutral gegenüber den verschiedenen Modelliermethoden verhält. Nur einige wenige Spezifika des verwendeten Modelers (hier: Unigraphics/Parasolids) waren programmtechnisch zu behandeln.

Die Extraktion von klassifikationsrelevanten (Flächen-)Daten aus CAD-Geometriemodellen ist ein gangbarer Weg um Teilespektren - auch unterschiedlicher Herkunft – maschinell vergleichbar zu machen.

In gegenwärtigen Entwicklungsstand können keine explizit im CAD-Modell vorhandene Information über komplexere Formelemente (z.B. Gewinde) extrahiert und in den Klassifikationsprozess einbezogen werden. Bei Weiterentwicklung des Interfaces sollten sich derartige komplexere Informationen auch direkt (d.h. ohne Rekonstruktion aus der Flächenstruktur) übernehmen lassen.

Generell hat die rein geometrische Klassifizierung relativ enge Grenzen. Aus diesem Grund muss in Zukunft weiteres Verbesserungspotential erschlossen werden, insbesondere durch Erkennen funktionaler Kriterien.

In der Kopplung mit Lösungskatalogen lassen sich sogenannte Wirkpaare/Wirkkomplexe, das sind teil- und baugruppenübergreifende Relationen zwischen Flächen (faces) und/oder Kanten (edges), in bestehenden Konstruktionen ermitteln und zur Wiederverwendung bereitstellen.

Damit ließen sich zusätzliche Effekte entlang der Prozesskette in der Entwicklung und Konstruktion (Anpassung kann sich auf die Gestaltung der „Restkonturflächen“ beschränken) wie auch in der Fertigung (automatische Zuordnung von Werkzeugen und Bearbeitungszyklen) erschließen.

### 4.4 Verwendung und Pflege von unternehmensspezifischen Klassifikationssystemen in SAP/R3

#### 4.4.1 Problemstellung

Bei MAN TURBO AG sind verschiedene strategische Systeme im Einsatz: SAP R/3 zur betriebswirtschaftlichen Auftragsabwicklung mit den Modulen: Projektssystem, Materialwirtschaft, Produktionssteuerung, Personalwirtschaft, Vertrieb, Versand sowie zentralen Funktionen wie Dokumentenverwaltung, Klassifikation, Änderungsdienst und Workflow.

Diverse technische EDV-Systeme wie die CAD-Systeme I-DEAS, Pro/Engineer und UG in der Maschinenkonstruktion, PDMS in der Anlagenplanung, AUCOPLAN in der MSR-

Technik, das NC-Programmiersystem EXAPT, Programme zur Werkzeug- und Messmittelverwaltung und das System ANSYS zur FE-Analyse

In allen Systemen ist eine Vielzahl von Daten zu Produkten enthalten. Um optimale Abwicklungsprozesse umsetzen zu können, muss die Möglichkeit bestehen, diese Informationen allen Mitarbeitern zur Verfügung stellen zu können.

#### 4.4.2 Sichten auf Stammdaten

Bei der Auftragsbearbeitung hat jeder Mitarbeiter eine, seiner Aufgabenstellung entsprechende, spezifische Sicht auf Stammdaten. In modernen ERP-Systemen sind daher den Daten in der Datenbank auch entsprechende Sichten zugeordnet. Dadurch wird erreicht, dass der gesamte Datenbestand in übersichtliche Verarbeitungseinheiten aufgeteilt ist. Jeder Mitarbeiter sieht die Daten, die für sein Arbeitsgebiet relevant sind.

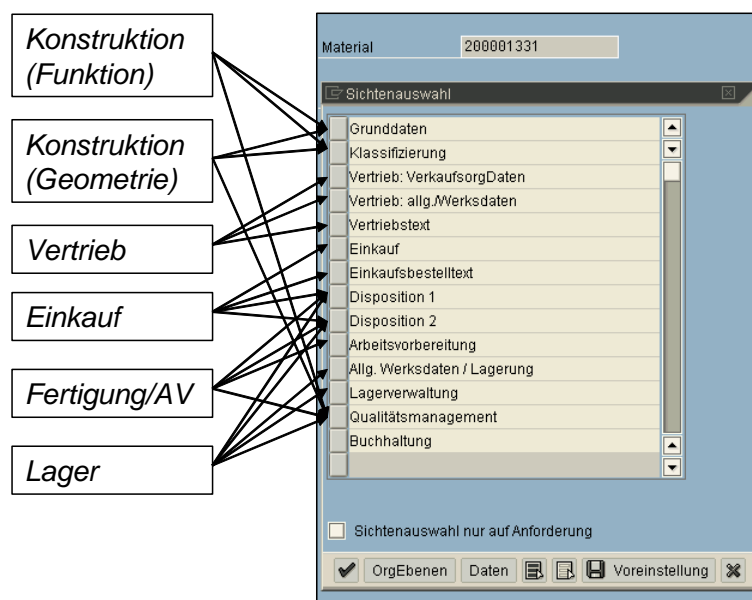


Bild 4-37: Sichtenauswahl

Im Umgang mit Stammdaten ist es wichtig, in dem existierenden Stammdatenbestand schnell und flexibel suchen zu können. Eine bewährte Methode dazu ist die Klassifizierung. Dabei werden bewertbare Merkmale definiert, die in hierarchischen Klassifikationsschemata strukturiert werden. Zur Unterstützung der Mitarbeiter ist es wünschenswert, Klassifikationsschemata zu erstellen, die die Sicht des Mitarbeiters auf die von ihm benutzten Stammdaten abbilden. Dies bedeutet, dass abteilungsspezifische Klassifikationsschemata zu definieren sind, was manuell einen enormen Aufwand für die Erstellung und Pflege verursacht und daher bisher nicht praktiziert wurde. Ein Konstrukteur wird z.B. Materialstämme im ERP-System SAP R/3 entweder nach geometrischen oder funktionalen Kriterien suchen. Bei der geometrisch orientierten Sicht auf die Teile wird er unterscheiden nach Rotationsteil oder Extrusionsteil.

Weitere Kriterien zur Eingrenzung können das Verhältnis von Durchmesser zur Länge des Bauteils sein (langes, dünnen Bauteil). Geometrieorientierte Merkmale werden vorzugsweise in den Daten der CAD-Systeme zu finden sein.

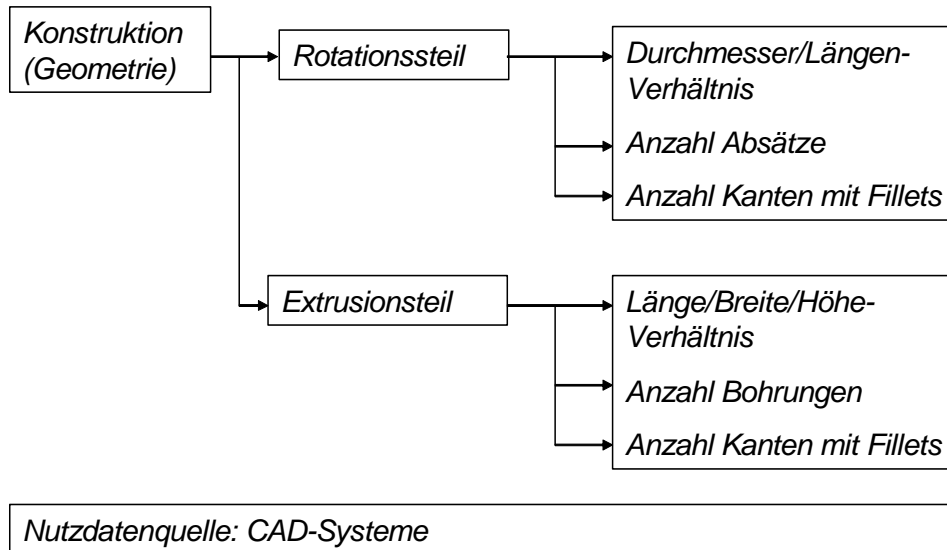


Bild 4-38: Nutzdatenquelle: CAD-Systeme

Bei der funktionsorientierten Sicht auf Materialstämme stehen beispielsweise die Verwendungsmöglichkeit eines Teils in einer Baugruppe oder der Anlage im Vordergrund.

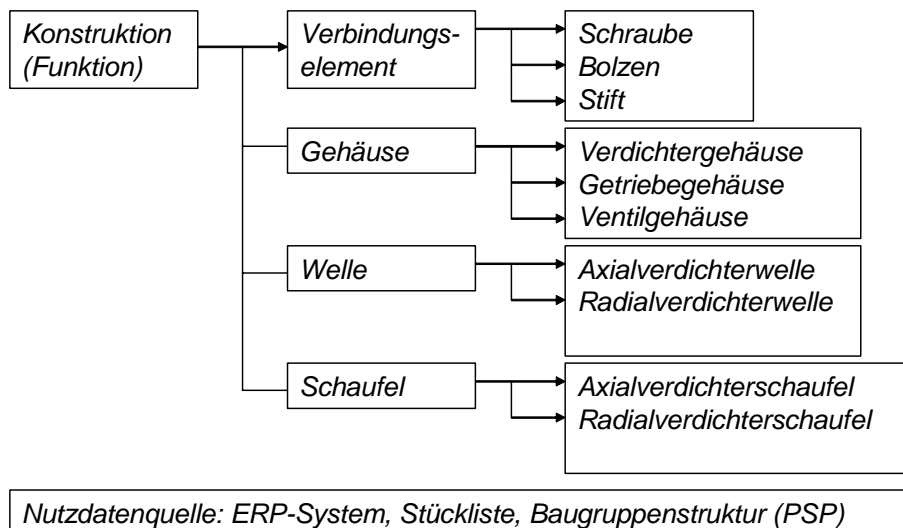


Bild 4-39: Nutzdatenquelle: ERP-System, Stückliste, Baugruppenstruktur (PSP)

Funktionale Ausprägungen können beispielsweise enthalten sein in: Verbindungselement, Gehäuse, Welle oder Schaufel. Die Datenquellen für diese Merkmale sind primär die ERP-Systeme, in denen die Stücklisten und Baugruppenstrukturen (PSP-Strukturen) verwaltet werden.

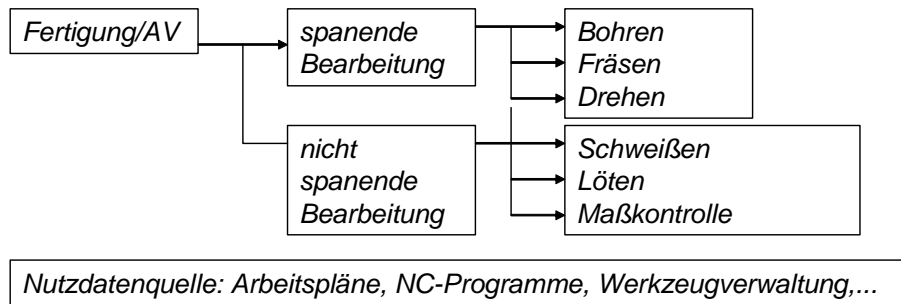


Bild 4-40: Nutdatenquelle: Arbeitspläne, NC-Programme, Werkzeugverwaltung

Für einen Mitarbeiter in der Arbeitsvorbereitung oder Fertigung wird die Sicht auf die Art der Bearbeitung des Teils im Vordergrund stehen. Ein NC-Programmieren wird ein NC-Programm z.B. mit den Merkmalsausprägungen „spanende Bearbeitung“ oder „nicht spanende Bearbeitung“ beschreiben.

Analog werden auch alle anderen Mitarbeiter ihre spezifischen Merkmalsausprägungen der Bearbeitungsobjekte definieren.

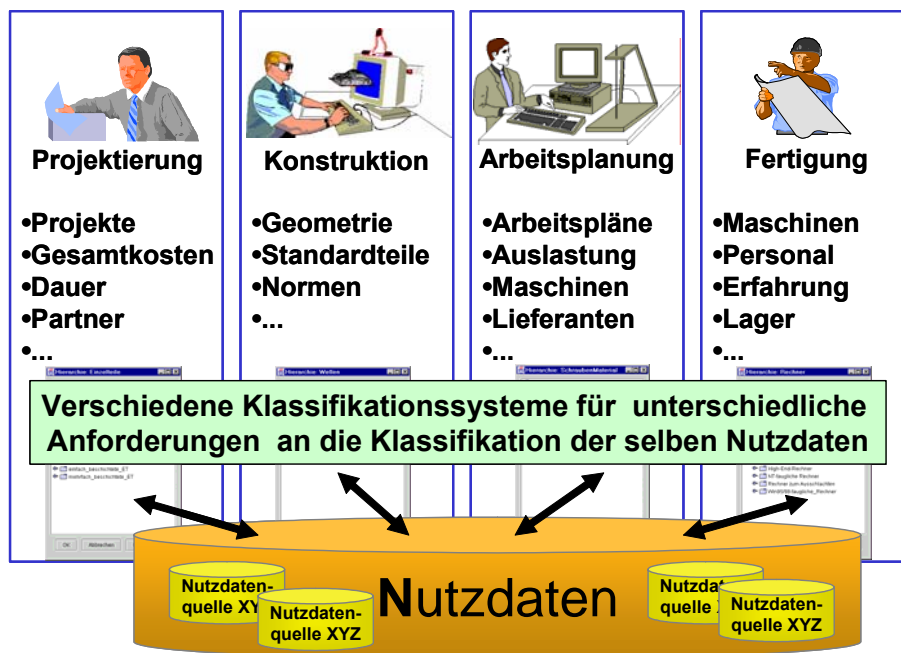


Bild 4-41: Spezifische Klassifikationssysteme entsprechend den Anforderungen

Sollte es gelingen, Klassifikationsschemata aus den verfügbaren Nutdaten automatisiert zu generieren und die Merkmalsbewertung automatisch durchzuführen, könnten in einem Unternehmen auch mehrere Klassifikationsschemata parallel verwendet werden.

Aus der Vielzahl der verschiedenen Sichten auf ein Maschinenteil war für die MAN TURBO im Projekt KLASSTER die Klassifikation von Produkten bzw. -teilen aus der konstruktiven Sicht relevant. Basis waren die im SAP R/3 verfügbaren Produktdaten (Materialstämme, Do-

kumente, Projektdaten). Die mit KLAUSTER erstellten Schemata für Materialstämme und Dokumente sollten im SAP verfügbar sein. Basis war das zurzeit noch manuell gepflegte Klassifikationsschema der MAN TURBO AG am Standort Zürich. Dieses Schema sollte modifiziert werden, damit es den erweiterten Anforderungen an allen Unternehmensstandorten genügt.

#### **4.4.3 Szenarienbeschreibung**

Im Rahmen des KLAUSTER Projektes wurden von MAN TURBO AG zwei spezifizierte Szenarien bearbeitet.

##### **Szenario 1:**

Generierung von Klassifikationsschemata auf der Basis von nicht klassifizierten SAP Stammdaten für Materialien und Dokumente.

Mit KLAUSTER sollte ein Vorschlag für ein Klassifikationsschema für SAP Stammdaten (Materialien / Dokumente) erzeugt werden.

##### **Vorgehensmodell:**

- Generierung des Klassifikationsschemas
- Ermittlung der klassifikationsrelevanten Datenfelder aus den SAP Stammdaten für Materialien und Dokumente
- Generierung der Merkmale
- Generierung der Bewertungsregeln
- Generierung von Klassen
- Zuordnung der Merkmale zu den Klassen
- Generierung von Klassenhierarchien
- Strukturierte Zuordnung der Klassen in die Klassenhierarchie
- Anwendung des generierten Klassifikationsschemas auf einen (eingeschränkten) SAP-Datenbestand im SAP-Produktionssystem
- Klassifikation von Materialstammdaten mit der PAK-Software
- Klassifikation von Dokumentdaten mit der PAK-Software
- Übertragung des Klassifikationsschemas in das SAP-Klassifikationssystem (SAP-Testsystem)



**Szenario 2:**

Modifikation eines manuell geführten Klassifikationssystems für SAP Materialstammdaten.

Mit KLAUSTER sollte eine manuell geführte Klassifikation automatisiert werden. Das Klassifikationsschema sollte auf einen nicht klassifizierten Datenbestand angewendet werden.

**Vorgehensmodell:**

- Modifikation des vorhandenen Klassifikationsschemas
- Übernahme des Klassifikationsschemas in KLAUSTER / PAK (Klassen und Merkmale)
- Erzeugung der Bewertungsregeln für die Merkmale aus dem klassifizierten SAP-Datenbestand Zürich
- Überprüfung der Klassifikationsrelevanz der übernommenen Merkmale für den SAP-Datenbestand Oberhausen / Berlin
- Korrektur/Modifikation des übernommenen Klassifikationsschemas
- Anwendung des modifizierten Klassifikationsschemas auf einen (eingeschränkten) SAP-Datenbestand
- Anwendung des modifizierten Klassifikationsschemas auf den SAP-Datenbestand Oberhausen / Berlin
- Anwendung des modifizierten Klassifikationsschemas auf den SAP-Datenbestand Zürich
- Übertragung des Klassifikationsschemas in das SAP-Klassifikationssystem (SAP-Testsystem)

**Voraussetzungen:**

- Es existiert ein Online-Zugriff auf die SAP-Produktionssysteme.
- Die ABAPS zur Extraktion der Datenfelder und Dateninhalte sind installiert.
- Es existieren im System repräsentative Datenbestände für Materialien und Dokumente
- Es existieren in den Systemen Objektverknüpfungen zwischen Stammdaten für Materialien und Dokumente
- Es existieren im System Zürich repräsentative, klassifizierte Datenbestände für Materialien
- Customizing im SAP: Klassifikationsrelevanz von Materialstammsätzen und Dokumenteninfosätzen

#### 4.4.4 KLAUSTER-Einsatz bei MAN TURBO AG

Bereits im Vorgängerprojekt PAK (Produkte automatisch klassifizieren) wurde der Zugriff auf SAP R/3 als Nutzdatenquelle realisiert.

Einzelheiten zur EDV-technischen Umsetzung des Zugriffs auf SAP R/3 Tabellen und Datenfelder aus der KLAUSTER- und PAK-Software werden im Beitrag des Projektpartners Eigner agile behandelt.

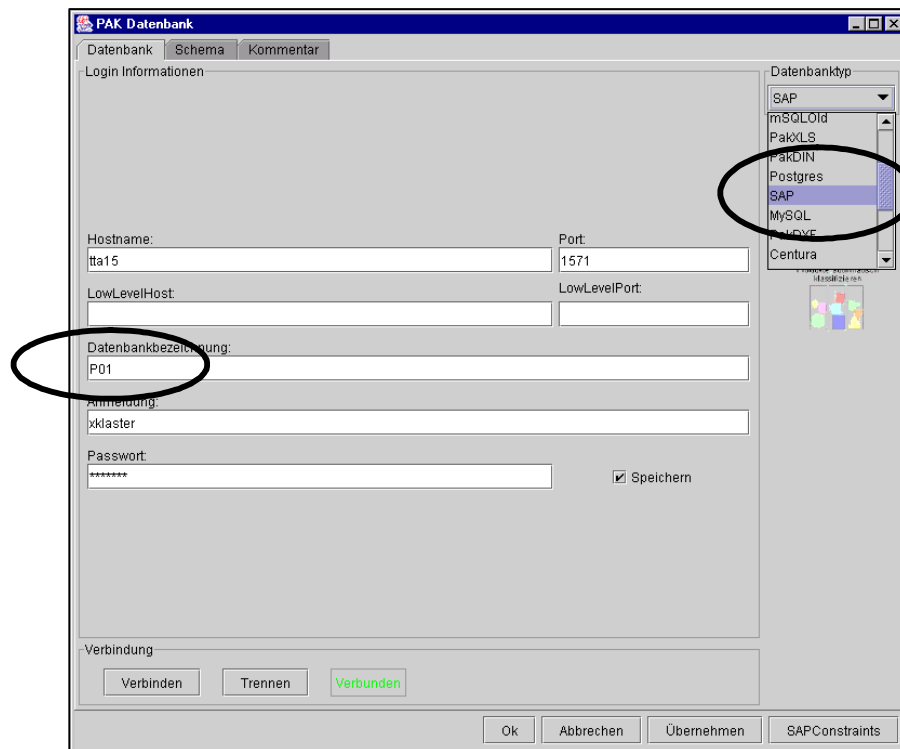


Bild 4-42: Auswahl von SAP als Nutzdatenquelle

Nach der erfolgreichen Anmeldung im SAP R/3 stehen der KLAUSTER Software zurzeit die SAP-R/3 -Tabellen für Materialstämme und Dokumente zur Verfügung.

Dies sind die Tabellen MARA, MAKT, DRAW, DRAD und DRAT. Für die Daten in den Feldern können für die Clusteranalyse Bedingungen formuliert werden, um das Datenvolumen auf eine repräsentative Menge einzugrenzen.

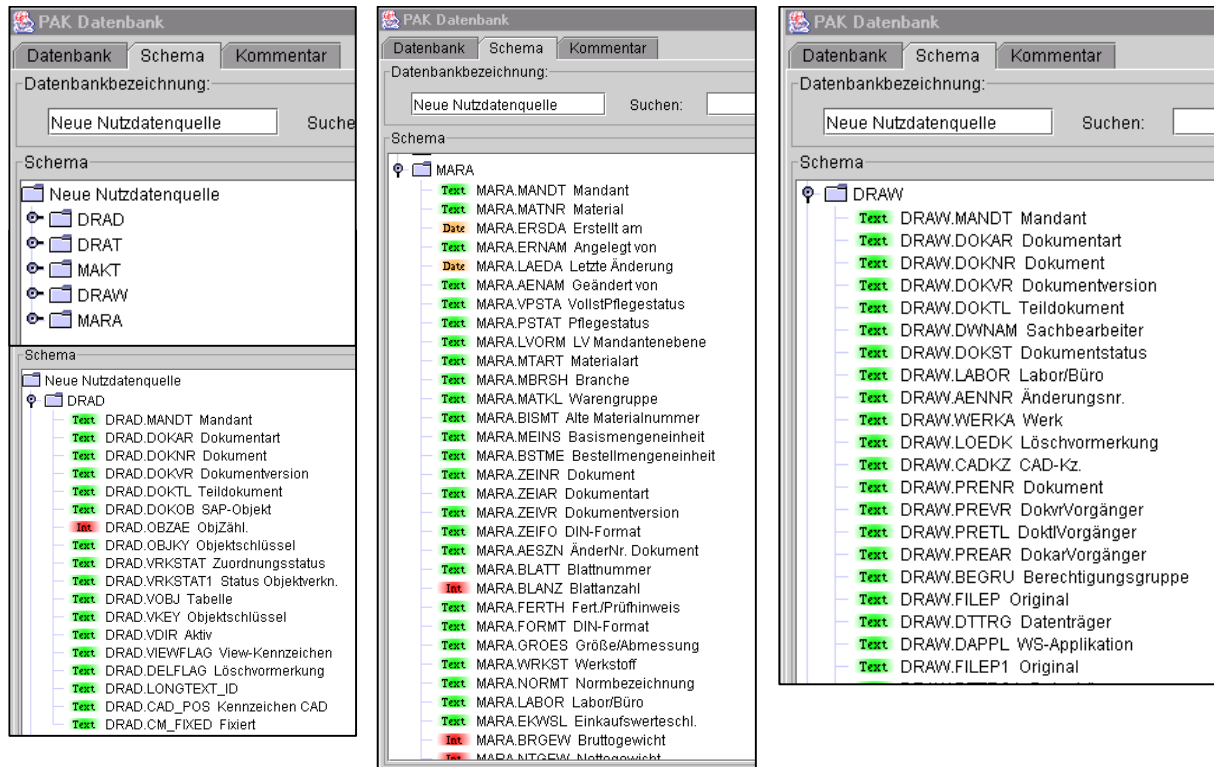


Bild 4-43: Ergebnis der Analyse der drei SAP Tabellen DRAD, MARA und DRAW

Die KLAUSTER Software analysiert nun die Einträge in den Datenfeldern und ordnet die Merkmale in vier Kategorien.

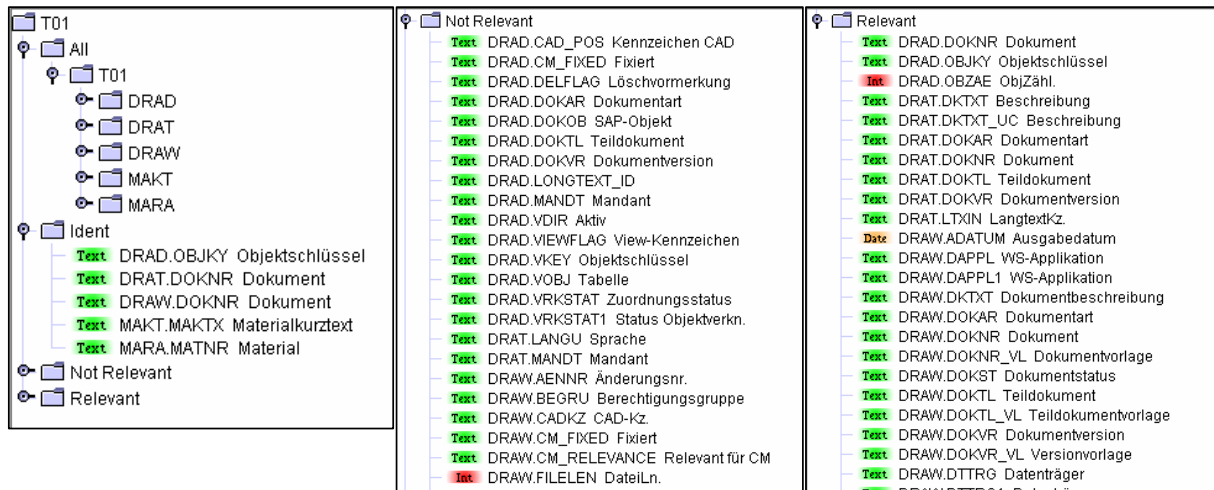


Bild 4-44: Ergebnis der Merkmalanalyse der SAP-Datenquelle

- **Kategorie ALL:** fasst alle betrachteten Merkmale als Protokollkategorie zusammen
- **Kategorie Ident:** fasst Merkmale zusammen, die identifizierenden Charakter haben. Identifizierend ist ein Merkmal, wenn es für jeden Datensatz einen unterschiedlichen Wert besitzt, wie z.B. die Materialnummer.
- **Kategorie Not Relevant:** fasst Merkmale zusammen, die nicht bewertet sind oder die für alle Datensätze den gleichen Wert besitzen, wie z.B. Mandant.
- **Kategorie Relevant:** fasst alle Merkmale zusammen, aus deren Ausprägungen sich Wertegruppen bilden lassen, wie z.B. Norm.

Aus den Kategorien lassen sich nun beliebige Merkmalfelder in einen Vektorraum übertragen, mit dem der Datenbestand und die Werteverteilung analysiert werden können.

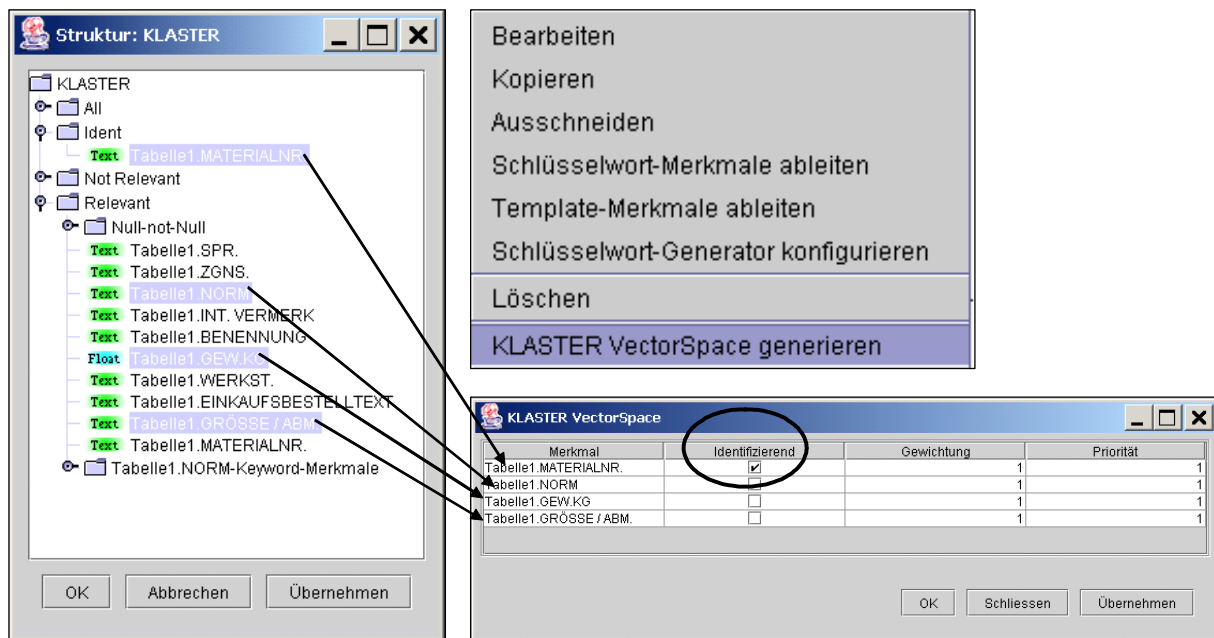


Bild 4-45: Definition der klassifikationsrelevanten Merkmale für die Klassifikation der SAP-Daten

Zusätzlich hat die KLASSTER Software bereits die klassifikations-relevanten Merkmale zu Klassen zusammengefasst und die Klassen hierarchisch zugeordnet. Die Generierung dieser Klassenhierarchie lässt sich durch eine Gewichtung der Merkmale beeinflussen (siehe Bild 4-46).

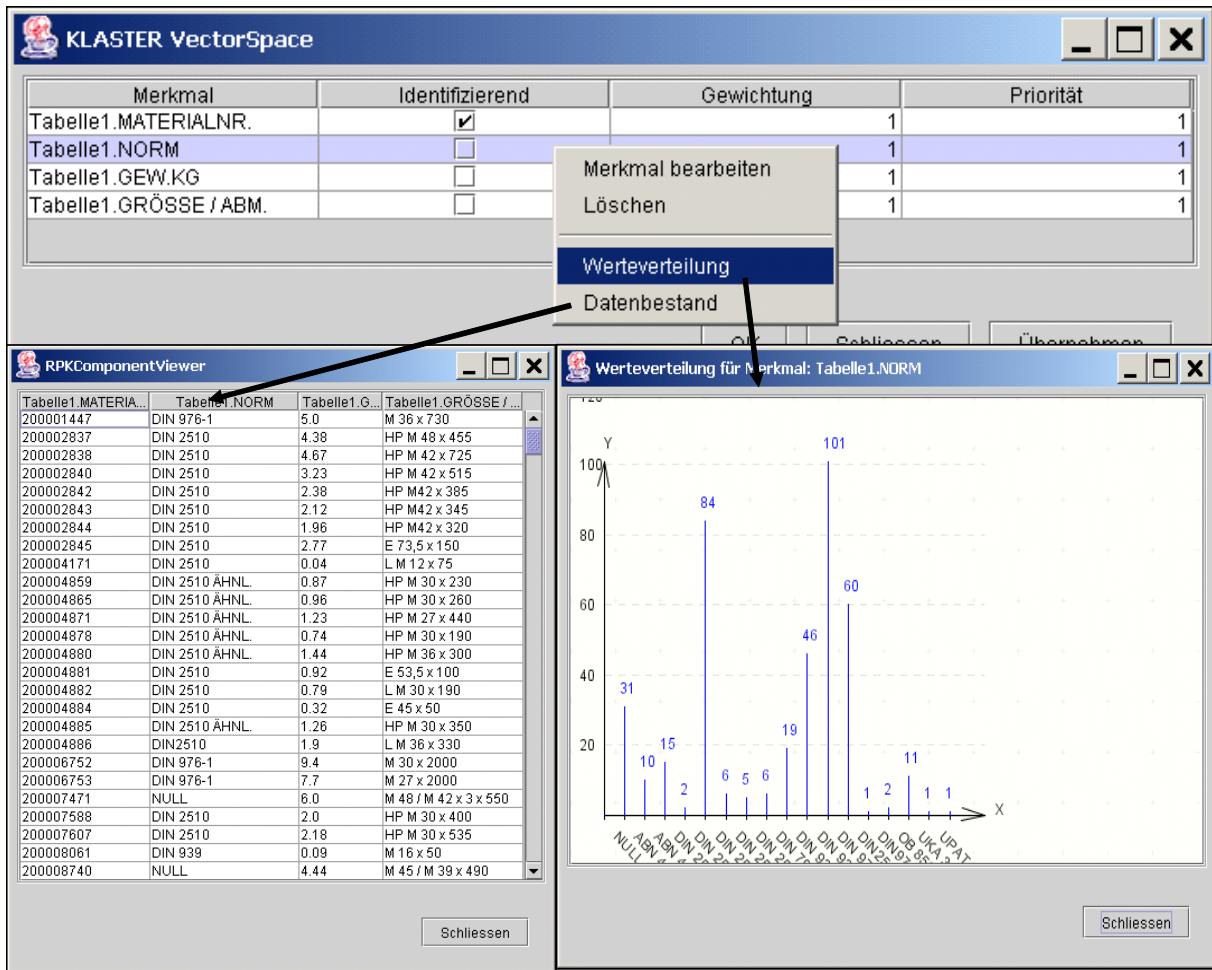


Bild 4-46: Werteverteilung der Merkmale anzeigen und Gewichtung definieren

Die Regeln, nach denen die Stammdaten den einzelnen Klassen zugeordnet werden, sind in einer lesbaren und nachvollziehbaren Form aufgebaut und können nach Bedarf angezeigt werden.

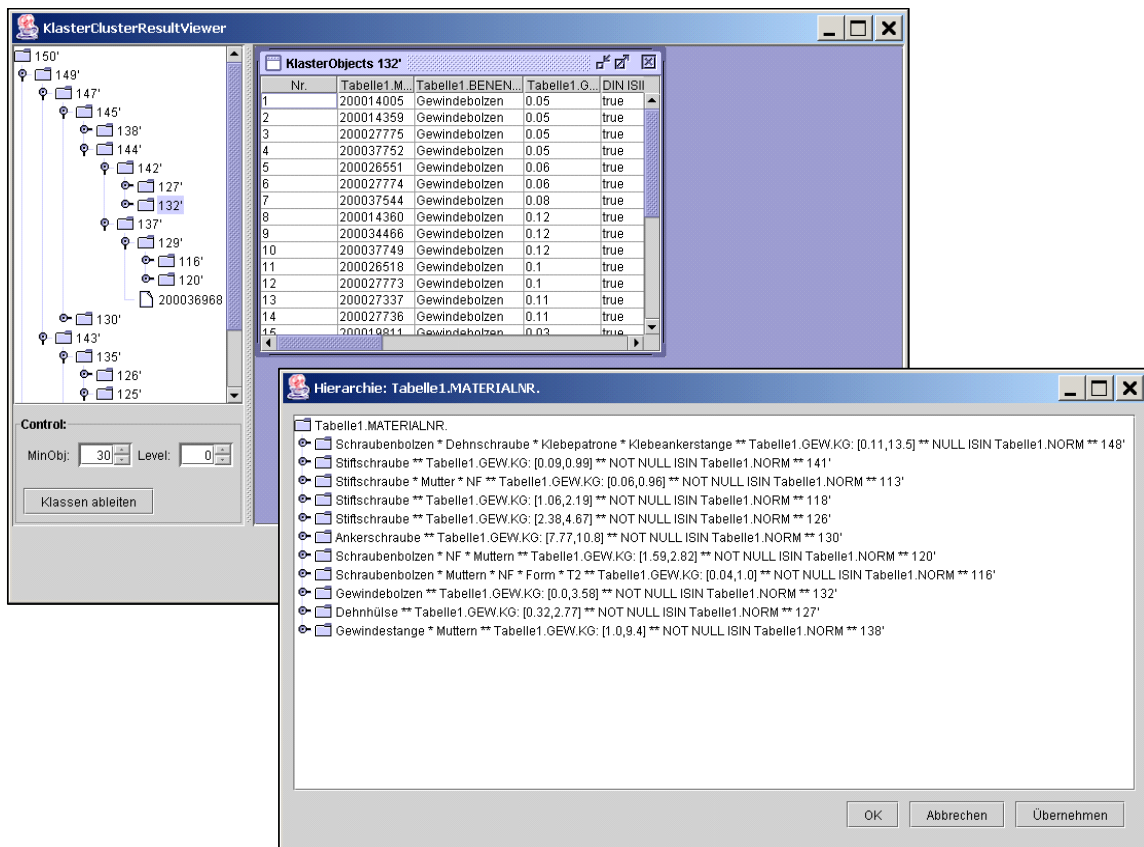


Bild 4-47: Automatisch erzeugtes Klassifikationssystem für SAP-Daten

#### 4.4.5 Technische Umsetzung der SAP Schnittstelle

Eine der Hauptaufgaben der Firma Eigner im Projekt KLAUSTER war die Entwicklung und Evaluierung der SAP-Anbindung an das KLAUSTER Tool, um die von MAN TURBO AG gestellten Anforderungen abdecken zu können. Die Firma Eigner ist ein renommiertes Unternehmen im Bereich Softwareentwicklung für Product Lifecycle Management (PLM). Eine der wichtigen Schnittstellen, die Eigner bereits für das eigene System Eigner PLM anbietet, ist die SAP-Schnittstelle. Die Erfahrungen mit der Entwicklung dieser PLM Schnittstelle waren sehr hilfreich bei der Implementierung der SAP-Schnittstelle für KLAUSTER.

Aufgrund der Architektur des KLAUSTER Tools (Java Anwendung, stand-alone) stellte sich zuerst die Frage, mit welchen Schnittstellentools (APIs) man sich am besten mit dem SAP R/3 System verbindet, um dort die gewünschten Aktionen durchzuführen. Folgende Lese- und Schreiboperationen in R/3 wurden identifiziert:

- Ermitteln der klassifizierbaren Entitäten und deren Relationen zu anderen Entitäten
- Auslesen von Entitätsdaten (z.B. Materialstamm, Tabellenname: MARA) und deren Merkmalsnamen
- Auslesen von Nutzdaten (z.B. Materialstammsätze, Dokumentinfosätze, Dateien) zur Bewertung und Klassifizierung im KLAUSTER Tool

- Zurückschreiben von Klassen, Klassenstrukturen und Merkmalen, die durch das KLAUSTER Tool ermittelt wurden
- Zurückschreiben von Klassifizierungsdaten d.h. Klassifizierung der Nutzdaten in R/3

Um diese Operationen durchzuführen, wurden die verschiedenen Schnittstellentechnologien analysiert, die SAP für ihr R/3 System anbietet. Neben asynchronen Schnittstellen wie ALE (Application Link Enabling) wurden synchrone Schnittstellen wie Java-RFC (Remote Function Call) und Jco (Java Connector) evaluiert, um eine möglichst mächtige und performante Low-Level Schnittstelle entwickeln zu können.

Zum Anfang des Projektes wurde die Low-Level Schnittstelle auf Basis von Java-RFC entwickelt. Java-RFC hatte aber einen gewissen Overhead bezüglich der Installationsanforderungen, da zusätzliche Middleware wie Corba bzw. SAP JNI benötigt wurde. Nachforschungen bei SAP haben ergeben, dass eine schlankere Kommunikationsplattform namens JCo existiert. Es hat sich herausgestellt, dass die SAP Schnittstelle JCo besser geeignet ist, da sie einfacher in der Handhabung ist und keine zusätzliche Middleware benötigt wird und man trotzdem aus einer Java Anwendung heraus die Schnittstellenfunktionen im SAP-System ansprechen kann. Desweiteren wird die Java-RFC mit dem Hinweis auf das Nachfolgeprodukt JCo von SAP nicht mehr unterstützt. Aus diesen Gründen wurde die komplette Low-Level Schnittstelle auf JCo umgestellt. Somit ergab sich die in Bild 4-48 gezeigte Architektur für die KLAUSTER-SAP R/3 Schnittstelle:

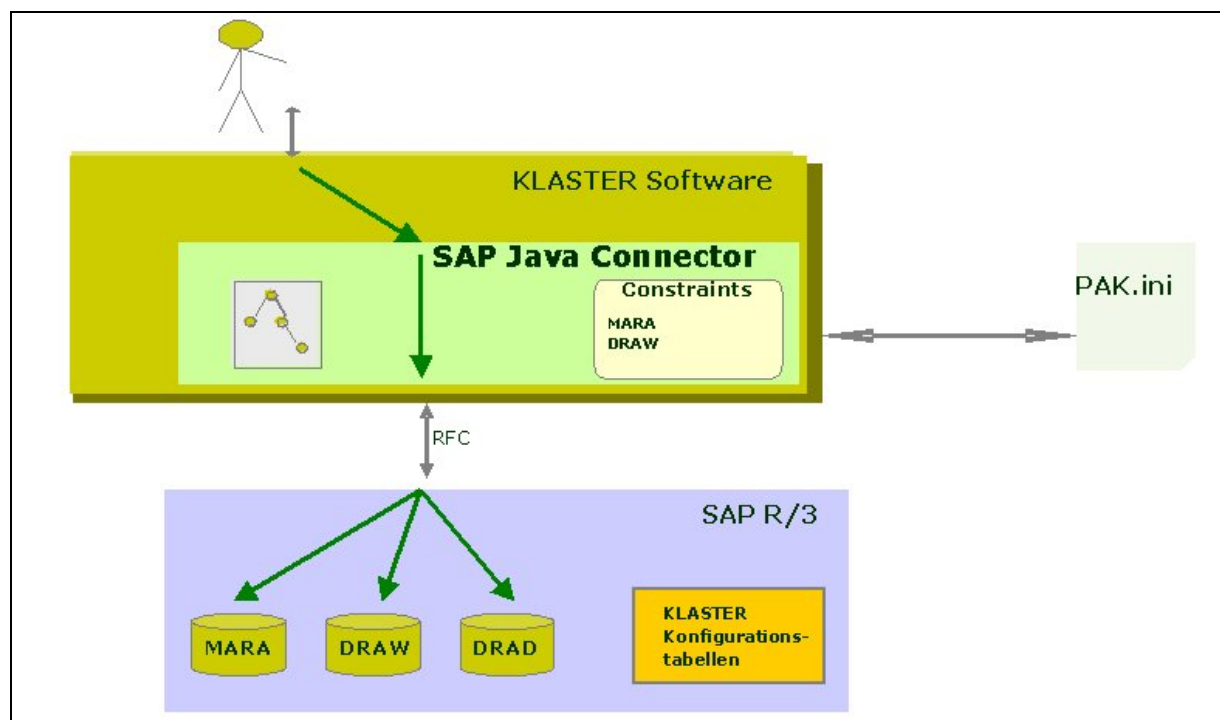


Bild 4-48: Architektur der KLAUSTER - SAP R/3 Schnittstelle

Wie in obigem Bild bereits dargestellt, besteht die hier vorgestellte Lösung aus Java Klassen die Methoden zur Verfügung stellen, die von der KLAUSTER Software aufgerufen werden, um die oben beschriebenen Aktionen im R/3 durchzuführen. Weitere Konfigurationsparameter wurden zum einen in der pak.ini Datei (zentrale KLAUSTER Konfigurationsdatei) abgelegt, zum anderen in speziellen Konfigurationstabellen für ergänzende Informationen zu den klassifizierbaren Objekten in R/3 gespeichert.

Die Konfigurationstabellen und die Ermittlungs- bzw. Suchfunktionen sind Bestandteil der Entwicklungsklasse /EIGNER/KLAUSTER und werden über einen Transportauftrag in das SAP-System eingespielt. Die Entwicklungsklasse /EIGNER/KLAUSTER befindet sich in einem eignen Namensraum, d.h. das SAP-System wird durch diese Entwicklungsklasse erweitert./ergänzt.

Die Konfigurationstabelle /EIGNER/ZPAK1 enthält die Relation der zusätzlichen Entitäten zu den klassifizierbaren Objekten. In /EIGNER/ZPAK2 werden die zusätzlichen bzw. ergänzende Felder einer Entität gespeichert und /EIGNER/ZPAK3 beinhaltet die zusätzlichen Entitäten für den Klassifikator.

Ablauf der LowLevel-Schnittstelle SAP :

1. Die Methode connect() baut die Verbindung mit dem SAP-System auf.
2. Die Methode getEntities() ermittelt die klassifizierbaren Objekte des SAP-System und die zusätzlichen Entitäten aus der Tabelle /EIGNER/ZPAK3.
3. Die Methode getRelation() ermittelt die abhängigen, zusätzlichen Entitäten einer klassifizierbaren Entität anhand der Tabelle /EIGNER/ZPAK1.
4. Die Methode getAttributes() ermittelt die Felder, die ergänzenden Felder aus der Tabelle /EIGNER/ZAPK2 und die Klassen mit den Merkmalen einer klassifizierbaren Entität.
5. Die Methode execQuery() sucht Datensätze einer klassifizierbaren Entität anhand der in der Pak.ini konfigurierten Funktion und Constraints.
6. Die Methode close() schließt die Verbindung zum SAP-System.

Bei der Evaluierung der SAP Standardschnittstellenfunktionen (BAPIs) wurde festgestellt, dass nicht alle Lese- und Schreiboperationen vollständig durch BAPI Funktionalität abgedeckt werden konnte. Deshalb war es erforderlich, zusätzlich eigene RFCs (Remote Function Calls) in SAP R/3 zu entwickeln.

Während der Entwicklung der Low-Level Schnittstelle stellte sich auch heraus, dass ein Test-tool sehr hilfreich wäre, um direkt die Funktionen zu testen. Aus diesem Grund wurde ein Tool entwickelt, das erlaubt, die entwickelten Funktionen im R/3 als eigenständige Einheiten zu testen (Unit-Tests). Damit können dann verschiedene Entitäten, Relationen und Attribute ausgelesen werden. Auf Basis dieser Daten kann dann letztendlich die Suchfunktion aufgeru-



fen werden, was die R/3 Daten im SWING basierten Testtool anzeigt (siehe Screenshot in Bild Bild 4-49). Hinter den Buttons im Screenshot verbergen sich die oben erwähnten Methoden, wobei der disconnect für die Methode close() steht.

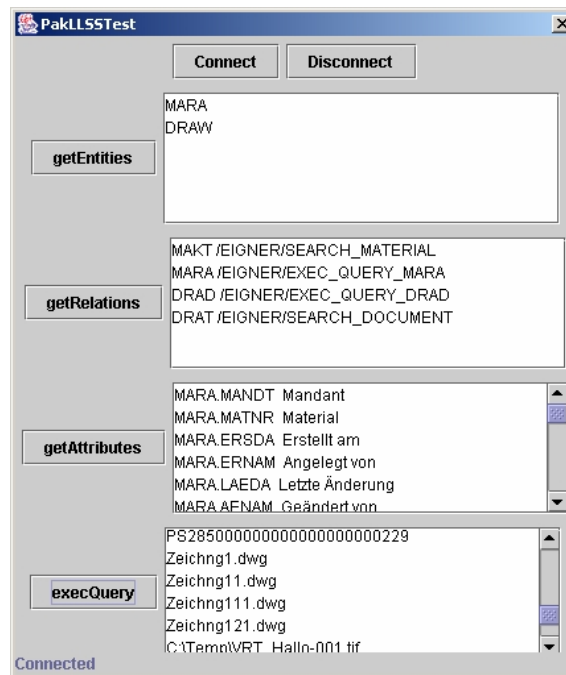


Bild 4-49: Testtool für Low-Level Schnittstelle

Nachdem die Funktionen zum Auslesen der Entitätendaten und Nutzdaten entwickelt waren und schon erfolgreich bei MAN TURBO AG getestet werden konnten, ging es darum, die ermittelten Klassifikationsinformationen nach R/3 zurückzuschreiben. In SAP R/3 wurde per BAPI Explorer eruiert, welche BAPIs zum Thema Klassifizierung (Merkmale, Klassen und Objektklassifizierung) vorhanden sind. Dies wurde anhand der existierenden Dokumentation und der Parameterbeschreibung der BAPIs im BAPI Explorer durchgeführt.

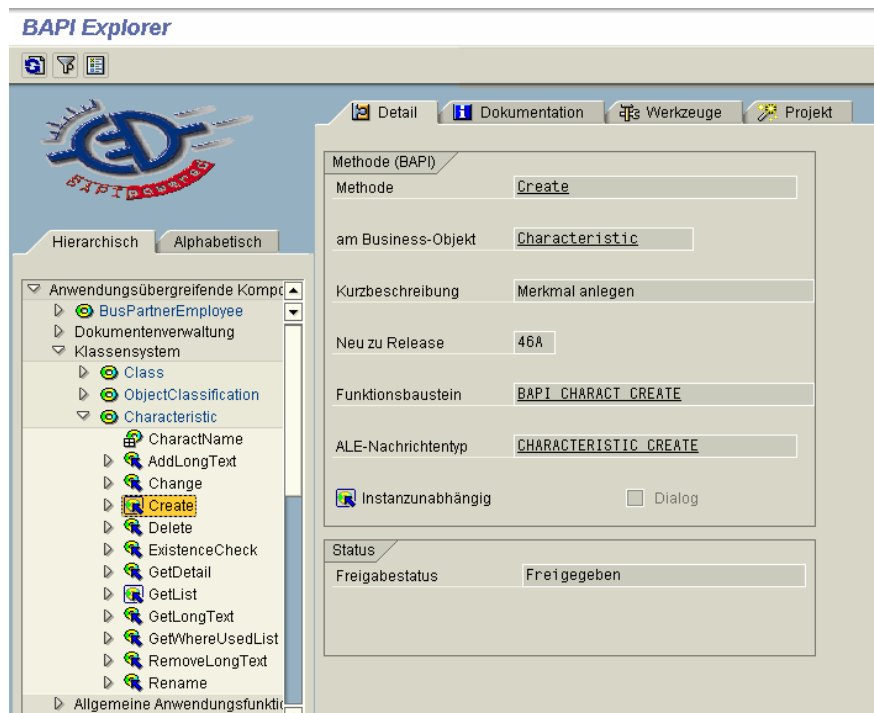


Bild 4-50: Anzeige des BAPI zur Merkmalspflege im BAPI Explorer in SAP R/3

Die BAPIs für die Klassifikation wurden in der KLAUSTER Java-Schnittstelle entsprechend gekapselt und konnten somit vom KLAUSTER Tool aus aufgerufen werden. Folgende BAPIs wurden in R/3 zur Anlage der Klassifizierung benutzt :

- Im ersten Schritt wurden die Merkmale mit dem BAPI BAPI\_CHARACT\_CREATE in R/3 angelegt.
- Danach wurde das BAPI BAPI\_CLASS\_CREATE zum Anlegen von Klassen und den zugeordneten Merkmalen in R/3 verwendet.
- Anschließend wurde mit dem BAPI BAPI\_HIERA\_ADDSUBCLASS die Klassenhierarchie / Klassenstruktur aufgebaut.
- Zum Abschluss wurde die Klassifizierung der R/3 Objekte anhand des BAPIs BAPI\_OBJCL\_CREATE durchgeführt.

Somit konnte man automatisiert zu einer Klassenstruktur und einer Klassifizierung in R/3 gelangen, die ansonsten manuell hätten eingegeben werden müssen, was zeitaufwendig und fehleranfällig gewesen wäre (siehe Bild 4-51).

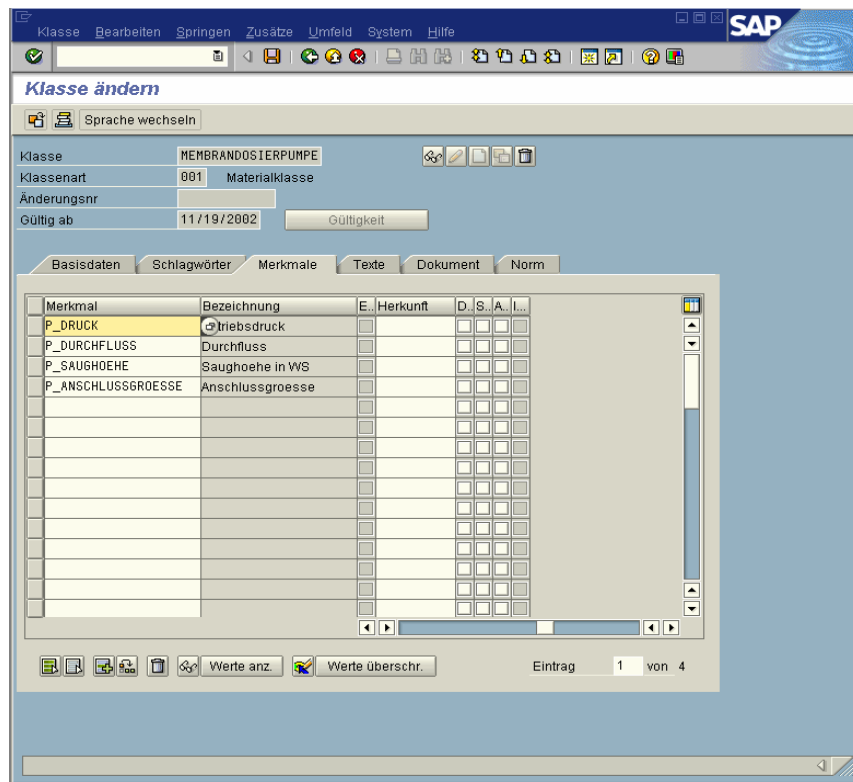


Bild 4-51: Ergebnis der Zuordnung Merkmal zu Klasse in SAP R/3

Zusammenfassend lässt sich anmerken, dass SAP sicherlich ein umfangreiches Portfolio von Schnittstellentechnologien anbietet, die für den externen Zugriff auf R/3 benutzt werden können. Die Entwicklung der Schnittstellenfunktionen für die Anbindung von KLAUSTER an SAP R/3 gestaltete sich recht problemlos, wobei auch hier der Teufel im Detail steckt, wenn man besondere Anforderungen hat.

Im Rahmen der Schnittstellenentwicklung war es trotz allem notwendig, zusätzliche Konfigurationstabellen und RFC-Funktionen in R/3 anzulegen, um die gewünschte Funktionalität anzubieten.

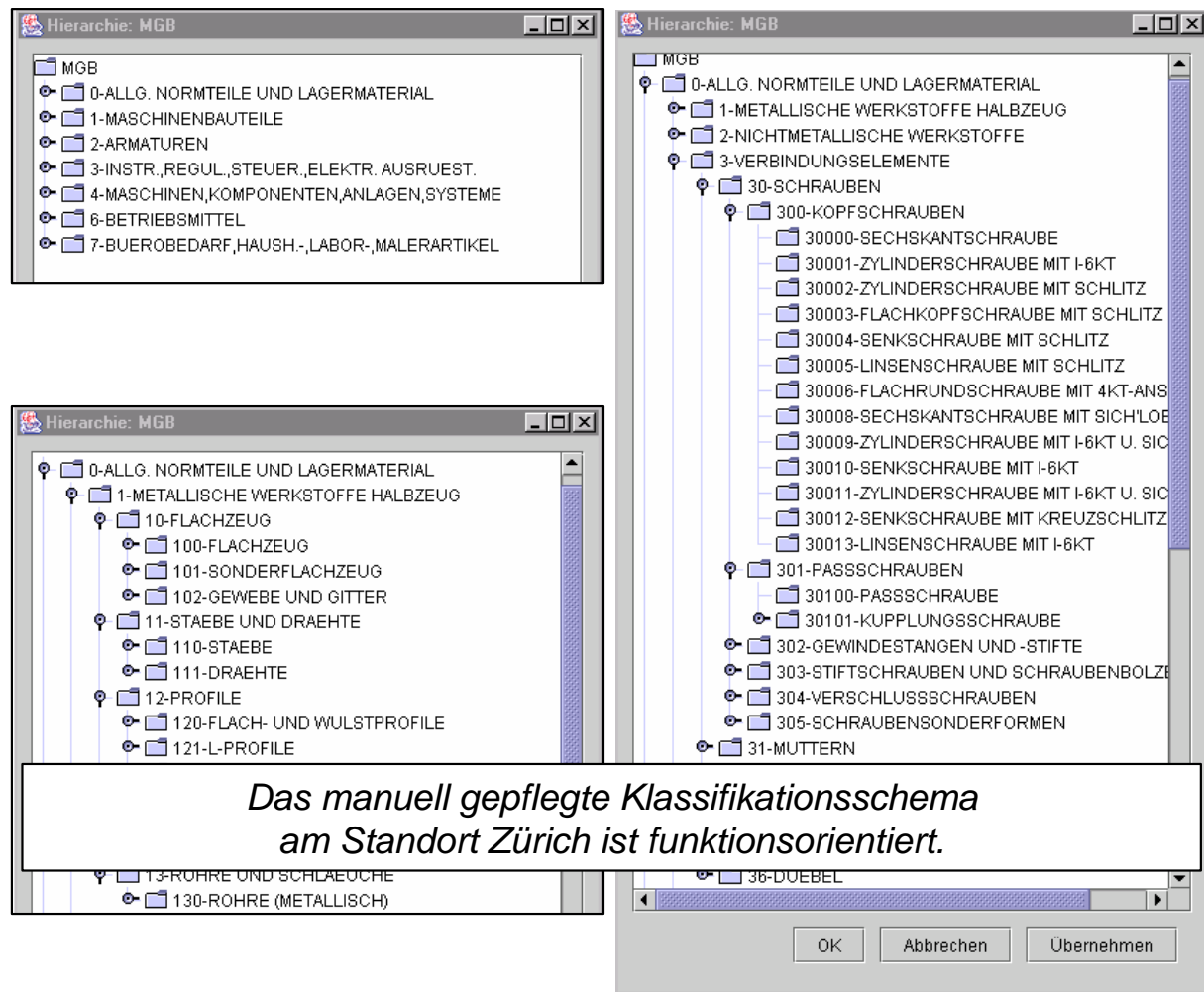
Mit Hilfe der KLAUSTER - SAP R/3 Schnittstelle konnte das "Beste" aus den 2 angeschlossenen Systemen genutzt werden:

- KLAUSTER für die entitätenübergreifende, automatisierte Erstellung von Klassen und Merkmalen inklusive Durchführung der Klassifizierung
- SAP R/3 für die Verwaltung der Nutzdaten als auch zur Ablage und Verfügungstellung der Ergebnisse der KLAUSTER Software im Rahmen des Klassifikationssystems

#### 4.4.6 Ergebnis/Nutzen

Die Bearbeitung des Anwendungsszenario 1 „Generierung von Klassifikationsschemas auf der Basis von nicht klassifizierten SAP Stammdaten für Materialien und Dokumente“ konnte im Rahmen des Projektes abgeschlossen werden. Der Prototyp der KLAUSTER-Software und die PAK-Software sind ein durchgängiges EDV-System mit einer ergonomisch ansprechenden Benutzeroberfläche. Die Ergebnisse der Klaster-Programmläufe unter Verwendung des produktiven SAP R/3-Datenbestandes der MAN TURBO AG sind als gut zu bewerten. Das KLAUSTER-System hat aus dem verfügbaren Bestand von Materialstammdaten sinnvolle Merkmale abgeleitet und die Merkmale zu nachvollziehbaren Klassengruppen zusammengeführt. Die anschließende Klassifikation mit der PAK-Software kam zu brauchbaren Ergebnissen. Nicht bearbeitet werden konnte die Anforderung, die mit KLAUSTER erzeugten Merkmale und Klassenhierarchien ins Klassensystem SAP R/3 zurück zu schreiben.

Das Anwendungsszenario 2 „Generierung von Klassifikationsschemata aus einem manuell geführten Klassifikationssystem für SAP Materialstammdaten“ wurde im Rahmen des KLAUSTER-Projekt nicht weiter verfolgt. Der Grund lag im Wesentlichen darin, dass das Klassifikationssystem am Standort Zürich vorwiegend funktionsorientiert aufgebaut ist. Eine automatisierte Klassifikation durch Erzeugung einer funktionsorientierte Klassenhierarchie lässt sich mit dem bisher verfolgten KLAUSTER-Konzept nicht realisieren. Eine Lösung hierzu könnte in einem Folgeprojekt erarbeitet werden.



*Bild 4-52: Das manuell gepflegte Klassifikationsschema am Standort Zürich ist funktionsorientiert*

Die automatisierte Generierung und Anpassung von Klassifikationssystemen in Verbindung mit der KLAUSTER- und PAK-Software kann zur Verringerung der Teilevielfalt genutzt werden, so dass das Anwachsen der heute über 1 Mio Materialstämme durch Wiederverwendung eingedämmt werden kann. Dies kann auch durch eine Unterstützung bei der standortübergreifenden Standardisierung erfolgen. Die Wiederverwendung von Norm- und Standardteilen führt zu größeren Beschaffungsmengen pro Teil und damit zu günstigeren Einstandspreisen. Außerdem kann durch die Verwendung von Standards, die in Klassensystemen festgeschrieben sind, eine Verringerung der „individuellen Konstruktionsnote“ und dadurch die Reduzierung der Komplexität der Produkte erfolgen. Dadurch kann sich auch eine Verringerung des Konstruktionsstundenaufwands und des Aufwands zur Stammdatenpflege entlang der gesamten Prozesskette Beschaffung, Arbeitsvorbereitung und Fertigung ergeben.

## **4.5 Klassifikation von Teiledaten im Bereich Unterhaltungselektronik**

### **4.5.1 Problemstellung**

Um die flexible Einsetzbarkeit der KLAUSTER-Software in den unterschiedlichsten Unternehmensbereichen zu demonstrieren wurde ihr erfolgreicher Einsatz im Maschinenbau, der Gebäudeautomation und der Unterhaltungselektronik demonstriert.

Gerade im Bereich der Unterhaltungselektronik herrscht unter den Herstellern ein enormer Preisdruck in Bezug auf die gelieferten Endprodukte. Um konkurrenzfähig zu bleiben ist es in dieser Branche daher unerlässlich, immer neue Wege zu finden um Produkte noch günstiger produzieren zu können. Ein Ansatzpunkt, der ein großes Einsparpotenzial bietet, ist die Optimierung des Bereichs der Produktentwicklung. Gerade im Bereich der Elektrotechnik bietet sich die Verwendung von Wiederholteilen an, da in diesem Bereich, anders als beim Maschinenbau, die Geometrie eines Bauteils eine kleinere Rolle spielt und es eine eindeutige Beziehung zwischen der Funktion und einem Lösungselement gibt.

### **4.5.2 Testdaten**

Im vorliegenden Testszenario stellte der Kunde Verwaltungsinformationen elektronischer Bauteile zur Verfügung. Die Daten wurden aus SAP in Excel-Tabellen exportiert, mit einem Datenvolumen von mehr als 100 Datensätzen pro Tabelle. Da die KLAUSTER-Software mit dem Excelformat gut arbeiten kann, musste das Datenformat nicht geändert werden.

### **4.5.3 Testdurchführung**

Bevor mit der Erstellung eines Klassensystems basierend auf diesen Daten begonnen werden konnte, mussten die „Rohdaten“ in den Excel-Tabellen bereinigt bzw. aufbereitet werden (Bild 4-53):

- Spalten mit Inhalt aber ohne Spaltenbezeichnung mussten eine Benennung erhalten.
- Leere Spalten mussten gelöscht oder benannt werden.
- Spalten mit identischer Bezeichnung aber unterschiedlichen Inhalten mussten über eine geeignete Benennung eindeutig gemacht werden.

The screenshot shows a Microsoft Excel spreadsheet titled 'Teilestamm\_2.xls'. The active cell is O25, containing the formula '= CAX-TST'. The spreadsheet displays a table with columns labeled with letters P through M. The data is as follows:

	P	Q	R	S	T	U	V	W	X	Y	Z	A	AE	A	AD
1	FK	Status								gültig ab	gültig bis	K	L	M	Status2
2	230	Freigegeben	26.07.1999 13:23:14	Y	ERBER_E					09.04.1999 13:42:13	01.01.3000 00:00:00	u	0		
3	230	Freigegeben			KIEFNER_T					03.11.1997 15:40:32	01.01.3000 00:00:00	u	0		
4	230	Freigegeben			KIEFNER_T					21.12.1996 00:00:00	01.01.3000 00:00:00	u	0		
5	230	Freigegeben			ERBER_E					04.12.1997 14:09:40	01.01.3000 00:00:00	u	0		
6	110	In Arbeit		Y	KIEFNER_T					28.05.2002 13:38:12	01.01.1900 00:00:00	u	0	M	
7	230	Freigegeben			KIEFNER_T					19.04.1997 00:00:00	01.01.3000 00:00:00	u	0	M	
8	230	Freigegeben			KIEFNER_T					20.10.1995 00:00:00	01.01.3000 00:00:00	u	0		
9	230	Freigegeben	09.07.1998 13:10:31	Y	ERBER_E					09.07.1998 13:09:44	01.01.3000 00:00:00	u	0		
10	230	Freigegeben	11.10.2000 13:02:16	Y	ERBER_E					11.10.2000 13:02:06	01.01.3000 00:00:00	u	0	M	
11	230	Freigegeben	14.07.2000 08:30:52	Y	ERBER_E					14.07.2000 08:29:56	01.01.3000 00:00:00	u	0		
12	230	Freigegeben	30.05.2000 14:29:43	Y	ERBER_E					30.05.2000 14:27:06	01.01.3000 00:00:00	u	0	M	
13	230	Freigegeben			KIEFNER_T					10.11.1999 12:49:52	01.01.3000 00:00:00	u	0		

Bild 4-53: Benennung nicht bezeichneter Spalten (Bustaben A bis M)

Nach der Anpassung der Rohdaten konnte mit der Klassifikation begonnen werden. Zunächst wurden die Nutzdaten an die Software angebunden und geladen. Anschließend wurde die Erzeugung von Merkmalsstrukturen angestoßen.

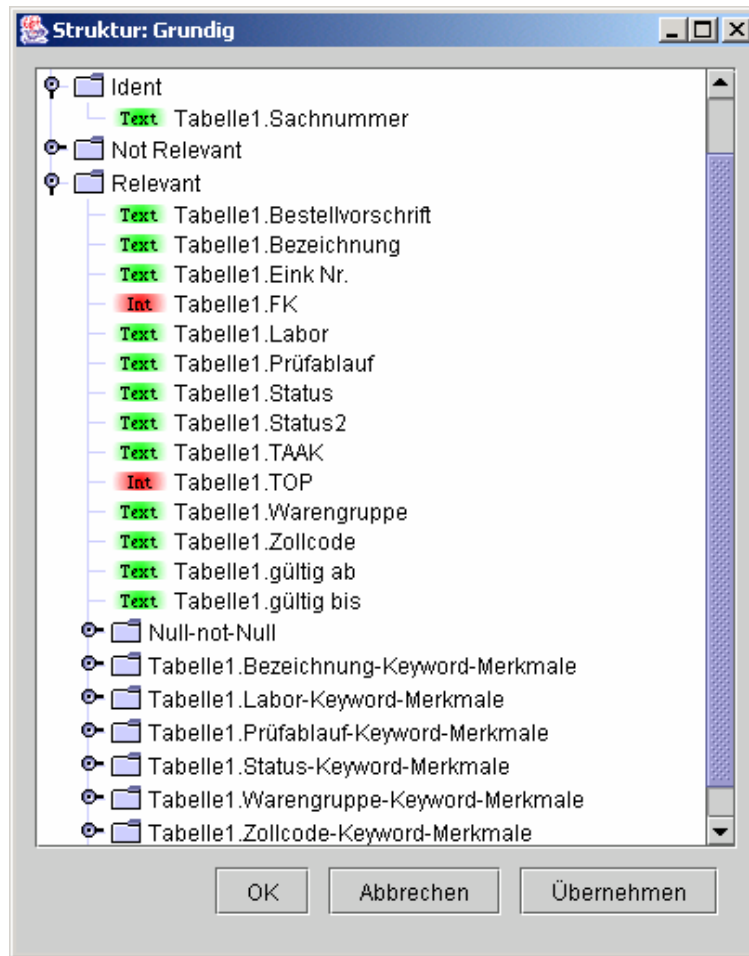


Bild 4-54: Resultierende Merkmalsstruktur

Wie in Bild 4-54 erkennbar, werden die Merkmale sortiert in relevant, nicht relevant und identifizierend. Das identifizierende Merkmal ist hier die Sachnummer. Die nicht relevanten Merkmale werden nicht zur Weiterverarbeitung benötigt. Die relevanten Merkmale stellen somit die Grundlage zur Weiterbearbeitung dar.

Zunächst werden aus dieser Menge von Merkmalen die Schlüsselwortmerkmale bestimmt, die entscheidend sind für die Struktur des Klassensystems, und aus diesen Merkmalen Schlüsselworte extrahiert. Ein einfaches Beispiel wäre hier das Schlüsselwortmerkmal „Norm“, das z.B. die Schlüsselworte „DIN“, „EN“ und „ISO“ mit verschiedenen Nummern darstellen könnte. In dem vorliegenden Fall sind die gewählten Schlüsselwortmerkmale „Zollcode“ und „Warengruppe“. Da an diesem Punkt über Form und Aussehen des späteren Klassensystems entschieden wird, sollte die Wahl der Schlüsselwortmerkmale wohl überlegt sein. Je nach Verwendungszweck des Klassensystems werden Schlüsselwortmerkmale funktionaler oder geometrischer Art gewählt.

Neben den Schlüsselwortmerkmalen werden weitere Merkmale selektiert, die jedoch nicht mehr zur Klassifizierung, sondern nur noch für die Lesbarkeit des Klassensystems ausgewählt werden. Im vorliegenden Fall waren das „Bezeichnung“, „FK“ und „TOP“.



Aus der Summe dieser Merkmale und Schlüsselwortmerkmale wurde ein Vektorraum generiert, der mit Hilfe von Custer-Algorithmen analysiert und bearbeitet und aus dessen Ergebnis anschließend eine Klassenhierarchie erzeugt wird, entsprechend den Vorgaben des Benutzers (Bild 4-55).

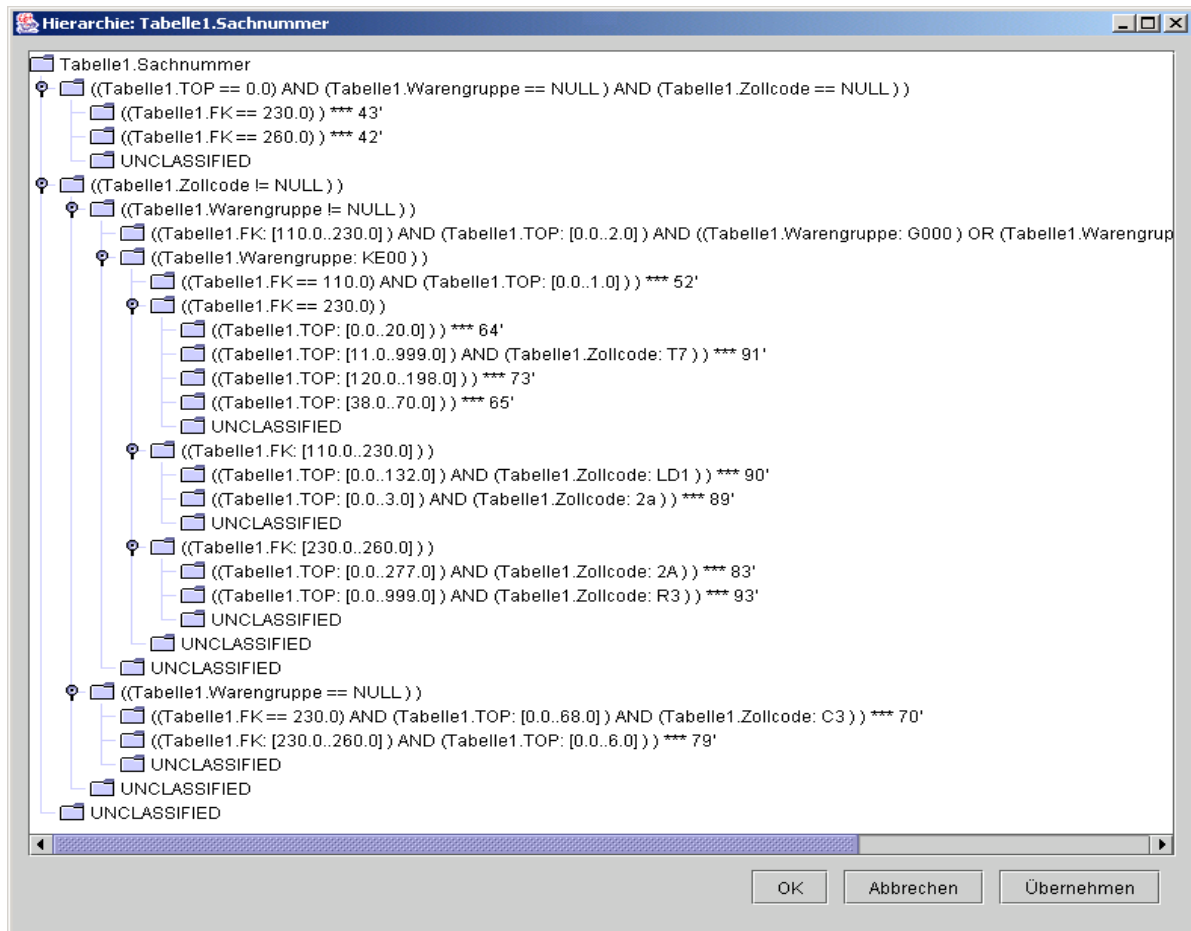


Bild 4-55: Erzeugte Klassenhierarchie

Die anhand der selektierten Merkmale generierte Klassenhierarchie teilt die Menge der Objekte auf in Objekte mit und ohne Zollcode. Die Menge der Objekte die einen Zollcode besitzen wird wiederum aufgeteilt in Objekte mit und ohne Warengruppen, etc. Es wird also deutlich, dass das generierte Klassensystem die Objekte anhand eines Diskriminators in Klassen einsortiert und somit die Anforderungen eines wohlstrukturierten Klassensystems erfüllt.

Ob das System für den Anwender von Nutzen ist liegt nur an der Wahl der Merkmale. Für eine Menge von Datensätzen sind unterschiedlichste Klassensysteme denkbar, je nach Verwendungszweck.

#### **4.5.4 Ergebnisse**

KLASTER ist ein geeignetes Hilfsmittel zur Erstellung von Klassensystemen. Soll für eine große Datenmenge ein Klassensystem entwickelt werden, so können mit Hilfe von KLASTER die verschiedenen Wege und Möglichkeiten der Klassifizierung in Hinblick auf die Wahl der Merkmale schnell ausprobiert und ein prototypisches Klassensystem erstellt werden. Eine vollständig automatische Klassifikation ermöglicht KLASTER jedoch nicht und eine Nachbearbeitung des entstandenen Klassensystems ist sicherlich notwendig.

Da es sich bei KLASTER um einen Prototypen handelt, ist die benötigte Rechenzeit die für eine bestimmte Datenmenge nötig ist relativ hoch. Hier findet sich noch Verbesserungspotenzial.

### **4.6 Kommerzielle Umsetzung der Ergebnisse aus KLASTER**

#### **4.6.1 Übersicht**

Das Softwarehaus simus systems GmbH wurde im Juni 2002 als Spinoff der Universität Karlsruhe von zwei ehemaligen Mitarbeitern des RPK gegründet und vertreibt mit Ihrem Software-Produkt simus classmate eine kommerzielle Umsetzung der Konzepte der Verbundprojekte PAK und KLASTER. Bereits zwei Jahre nach der Gründung konnten sie sich im Markt etablieren und haben mittlerweile zehn Mitarbeiter, was den steilen Aufstieg des jungen Unternehmens dokumentiert. Zu den Referenzkunden zählen unter anderem der Anlagenbauer Voith Paper GmbH & Co. KG mit Sitz in Heidenheim, Trumpf Werkzeugmaschinen GmbH & Co. KG in Ditzingen sowie die LIEBHERR Aerospace GmbH an den Standorten Lindenberg und Toulouse (Frankreich).

#### **4.6.2 Ein Referenzprojekt der SIMUS-Systems GmbH, Karlsruhe**

In diesem Abschnitt soll exemplarisch ein Referenzprojekt der simus systems GmbH dargestellt werden: Die Jäger GmbH aus Münster und die Schlatter AG aus Schlieren/Schweiz sind Maschinenbauer für Drahtwebmaschinen und Schweißmaschinen. Innerhalb des Projektes mussten zwei „Klassifikationswelten“ zusammengefügt werden, wobei bei Jäger keine Klassifikation der Stammdaten in Klassen und Merkmale existierte. Die zukünftige Klassenstruktur wurde von den beiden Firmen vorgegeben und in simus classmate importiert. Folgend wurden die Stammdaten von Jäger klassifiziert, mit den Schlatter Daten zusammengefügt und in ein gemeinschaftliches Klassifikationssystem migriert. Im Zuge der Migration wurden eine Dublettenanalyse und eine Datenbereinigung durchgeführt.

## 5 Zusammenfassung und Ausblick

Innerhalb der Produktentwicklung und -fertigung nimmt der Informationsbedarf und das Datenvolumen immer weiter zu, weshalb dem effizienten Zugriff auf bereits gespeicherte Daten eine strategisch wichtige Bedeutung zukommt. Die Klassifikation von Produkten ist in diesem Zusammenhang ein geeignetes Werkzeug, um durch Informationsverdichtung und somit der Bereitstellung semantisch höherwertiger Informationen über die gespeicherten Daten, diese schneller zugreifbar zu machen. Im Bereich der produzierenden Unternehmen des Maschinenbau Sektors ist der Anteil der Neukonstruktionen eher gering. Meist kann auf bestehende Lösungen zurückgegriffen werden, die dann angepasst werden. Grundlage hierfür ist das schnelle Finden von Teilen, die ähnliche Eigenschaften besitzen und somit eine Mehrfachverwendung erlauben. Viele Unternehmen verweigern heute dennoch den Einsatz der Klassifikation, um ihre Datenbestände zu strukturieren. Die Gründe hierfür sind zum Einen, dass verfügbare Standard-Klassifikationssysteme wie z.B. [eCl@ss](#) oder UN/SPC den unternehmensspezifischen Anforderungen nicht genügen, weil sie beispielsweise nicht auf das Produktspektrum des Unternehmens abgestimmt sind. Zum Anderen ist die manuelle Pflege einer Klassifikation zeitaufwendig und fehleranfällig [GaKn-86].

Das Ziel des Forschungsvorhabens KLAUSTER war die Konzeption und Realisierung eines Softwaresystems zur automatischen Klassifikation von Produkten. Das in diesem Projekt verfolgte Konzept ermöglicht, die Tätigkeiten bei der Erstellung und Anpassung unternehmensspezifischer Klassifikationssysteme - schwerpunktmäßig produzierender Unternehmen im Bereich Maschinenbau - zu automatisieren. Hierdurch können Fehlerquellen und großer Zeitaufwand der manuellen Erstellung von Klassifikationssystemen vermieden werden. Darüber hinaus soll die Entwicklung unternehmensspezifischer, effizient einsetzbarer Klassifikationssysteme gewährleistet werden.

Der im Rahmen des Projektes entwickelte Software-Prototyp wurde innerhalb des Konsortiums im realen, produktiven Umfeld validiert. Hierbei zeigte sich die Anwendbarkeit des Konzepts in der Praxis sowie das enorme Einsparpotential. Im ersten Anwendungsfall wurden bei der Firma AUCOTEAM GmbH Projektdateien der Anlageautomatisierung in der Gebäudetechnik klassifiziert. Ein weiterer Anwendungsfall zeigt die Anbindung an das CSM-System Remarc von ARC Solutions. Innerhalb des dritten Anwendungsfalles wurde die von ARC-Solutions realisierte Anbindung der KLAUSTER-Software an Unigraphics beschrieben. Auf diese Weise war es möglich, Maschinenelemente bei der MAN TURBO AG zu analysieren und zu klassifizieren. Aufbauend auf vorangegangenen Anwendungsfall wurde im nächsten Fall die Pflege des Klassifikationssystems bei der Firma MAN TURBO AG beschrieben. Die Materialstammdaten werden bei MAN TURBO AG in einem SAP R/3-System verwaltet. Aufgrund mangelnder Strukturierung der Daten in der Vergangenheit hatten sich über eine Million Materialstämme angehäuften, wodurch das System nahezu unhandhabbar wurde. Durch die automatische Klassifikation der Materialstammdaten, die zunächst im Bereich der Standard- und Normteile durchgeführt wurde, konnten viele Dubletten und veraltete Materialstammsätze herausgefiltert werden. Laut MAN TURBO AG lag der Anteil der veralteten oder mehrfach angelegten Materialstämme im Bereich der Standard- und Normteile bei 80-90%.

Somit konnte die Teilevielfalt deutlich verringert werden, was eine Verminderung des Konstruktionsaufwandes und somit eine spürbare Kostenreduzierung zur Folge hatte. In einem letzten Anwendungsfall wurde eine Klassifikation von Teiledaten im Bereich Unterhaltungselektronik durchgeführt. Wie bereits in den Anwendungsfällen zuvor, konnten zufriedenstellende Ergebnisse erzielt werden.

Die Tragfähigkeit des im KLAUSTER-Projekt verfolgten Konzepts in der Praxis wird nicht zuletzt durch die erfolgreiche kommerzielle Umsetzung der Ergebnisse durch den Spinn-off *simus systems GmbH* der Universität Karlsruhe eindrucksvoll belegt. Das Unternehmen begann im Juni 2002 mit zwei Mitarbeitern und hat sich innerhalb von knapp zwei Jahren im Markt etabliert und beschäftigt bereits zehn Mitarbeiter.

Basierend auf der innerhalb des KLAUSTER-Projektes erarbeiteten und innerhalb dieses Berichtes vorgestellten Vorgehensweise, sind eine Reihe von wissenschaftlichen Arbeiten und Forschungsaktivitäten vorstellbar, welche auf den Ergebnissen dieses Projektes aufsetzen können.

Eine sinnvolle Erweiterung des erarbeiteten Konzeptes wäre die Automatisierung bislang manuell gepflegter Klassifikationssysteme. In vielen Unternehmen werden Klassifikationssysteme seit Jahren manuell gepflegt. Eine Anpassung solcher Klassifikationssysteme, beispielsweise an eine geänderte oder erweiterte Produktpalette, ist so gut wie ausgeschlossen, da dies bedeuten würde, den gesamten, bereits klassifizierten Datenbestand manuell nachklassifizieren zu müssen. Künftige Forschungsarbeiten können sich mit der Thematik befassen, aus der Analyse von bislang manuell gepflegten Klassifikationssystemen automatisch die Klassifikationsregeln für eine künftige automatische Klassifikation abzuleiten. Ein derart automatisiertes Klassifikationssystem kann den aktuellen Bedürfnissen des Unternehmens jederzeit angepasst werden, da die Nachklassifizierung der Altdaten in kürzester Zeit automatisch durchgeführt werden kann.

Es gibt Ansatzpunkte, um das Klassifikationssystem zu erweitern und die Suche nach existierenden Lösungen entscheidend zu verbessern. Durch Erweiterung der Klassifikation um funktionale Merkmale würde der Konstrukteur in die Lage versetzt, nach einer Baugruppe mit gewissen Funktionen und Abmessungen zu suchen. Auf dem Markt existieren noch keine zufriedenstellende softwaretechnischen Lösungen zur automatischen funktionsorientierten Klassifikation von Baugruppen (bzw. Wiederholteilen). Die Bestimmung funktionaler Merkmale einer Baugruppe sowie die geeignete Abbildung auf eine Merkmalstruktur sind nichttriviale Probleme. Ein System zur automatischen Erstellung von funktionsorientierten Klassifikationssystemen wäre in der Lage die kostenintensive und fehleranfällige manuelle Erstellung von Klassifikationssystemen zu ersetzen.

## **6 Anhang**

### **6.1 Projektpartner**

In diesem Abschnitt werden die Partner des Verbundprojektes KLAUSTER vorgestellt.

#### **6.1.1 Atos Origin**

##### **6.1.1.1 Überblick**

Atos Origin ist ein international führender Anbieter von IT-Dienstleistungen. Das Unternehmen bietet das gesamte Spektrum an Beratung und Dienstleistungen der Informationstechnologie. Die Kompetenzbereiche umfassen Consulting, Systemintegration und Outsourcing einschließlich eServices und Business-Solutions. Atos Origin erzielt einen Jahresumsatz von mehr als 5 Milliarden Euro und beschäftigt 45.000 Mitarbeiter in 50 Ländern.

Atos Origin ist der weltweite IT-Partner der Olympischen Spiele. Zu den Kunden des Unternehmens gehören unter anderem ABN AMRO, Airbus, Akzo-Nobel, BMW, Bosch, British Petroleum, E-Plus, Euronext, Fiat, KarstadtQuelle, KPN, Neckermann, Philips, Porsche, Renault, Schlumberger, Shell, Siemens, T-Mobile, Unilever, Vivendi Universal und Vodafone.

In der Region Germany and Central Europe, zu der Deutschland, Schweiz, Österreich und Polen gehören, beschäftigt das Unternehmen rund 3.500 Mitarbeiter.

Atos Origin ist an der Euronext Premier Marché in Paris notiert und firmiert als Atos Origin, AtosEuronext, Atos Worldline und Atos Consulting.

Weitere Informationen finden Sie auf unserer Website unter [www.atosorigin.de](http://www.atosorigin.de) oder [www.atosorigin.com](http://www.atosorigin.com).

##### **6.1.1.2 Product Lifecycle Management bei Atos Origin**

Atos Origin bietet im Bereich Product Lifecycle Management ein vollständiges Service-Portfolio, angefangen bei der Analyse des Produktlebenszyklus mit allen relevanten Geschäftsprozessen, Daten und Softwaresystemen, über die Systemauswahl bis hin zur Systemanpassung und -einführung, der Schulung von Mitarbeitern und dem Systembetrieb (Administration und Helpdesk). Diese Dienstleistungen werden von über 250 hochqualifizierten Beratern erbracht, die auf mehrere europäische Standorte verteilt sind und damit einerseits die räumliche Nähe zum Kunden und andererseits auch die erfolgreiche Durchführung von großen, internationalen PLM Projekten sicherstellen. Im Bereich der Implementierung verfügt Atos Origin über umfangreiche Expertise im Bereich der Produkte SAP PLM, Teamcenter Enterprise und Engineering, Windchill und eMatrix.

Aufgrund langjähriger Erfahrung, der erfolgreichen Zusammenarbeit mit Kunden und Partnern auf internationaler Ebene und der Anwendung eines erprobten Analyse- und Implementierungsverfahrens ist Atos Origin in der Lage, hochwertige PLM Lösungen in kurzer Zeit zu realisieren.

Zu den Kunden von Atos Origin im Bereich PLM gehören sowohl mittelständische und große Unternehmen aus Deutschland, als auch internationale Konzerne aus den Branchen Luft- und Raumfahrttechnik, Maschinenbau, Automobil und Automobil-Zulieferindustrie, Elektrotechnik, Finanzen, Telekommunikation und Informationstechnologie.

Für diese Unternehmen wurden in zahlreichen Projekten unter anderem PLM-Lösungen für

- Workflow-Management,
- Change-Management,
- Teileverwaltung und Produktstrukturierung,
- Dokumenten-Management,
- PDM-ERP Integration und
- ERP-CAD Integration

realisiert.

### **6.1.1.3 Zielsetzung zur Nutzung von KLAUSTER**

Eine effiziente Klassifikationsstruktur zur Teileverwaltung im Unternehmen ist eine Grundvoraussetzung für die vermehrte Wiederholteilverwendung. Diese wiederum führt zu kürzeren Entwicklungszeiten und Einsparungen entlang der gesamten Prozesskette der Produkterstellung (Entwicklung bis Montage), da Zeichnungen, Stücklisten, Arbeits- und Prüfpläne, Montageanleitungen, usw. bei Wiederholteilen bereits vorliegen und nicht neu erstellt werden müssen. Die Verwendung von Wiederholteilen führt zu einer Verringerung der Teilevielfalt im Unternehmen, wodurch erhebliche Kosten gespart werden können. Wenn mehr gleiche Teile verwendet werden, so erhöhen sich die Losgrößen und Bestellmengen, was wiederum zu Einsparungen führt.

Die breite Kundenbasis von Atos Origin aus verschiedensten Branchen in Deutschland zieht daraus einen erheblichen wirtschaftlichen Nutzen. Auf diese Weise stellt Atos Origin einen Multiplikator dar, der dafür sorgt, dass die Projektergebnisse über die Projektlaufzeit hinaus weiter verfolgt und einer großen Anzahl von Unternehmen zur Verfügung gestellt werden.

Die entstandene Prototypsoftware soll bei Atos Origin über die Projektlaufzeit hinaus im produktiven Geschäftsumfeld bei Kunden zur Datenbereinigung und -migration beim PLM-

Systemwechsel eingesetzt werden. Insbesondere erwartet Atos Origin den folgenden kurzfristigen Nutzen:

- Zeit- und Kostenersparnis bei der Erstellung unternehmensspezifischer Klassifikationssysteme im Rahmen von Kundenprojekten
- Vermeidung subjektiver Entscheidungen bzgl. der Klassifikationsrelevanz von Produktmerkmalen
- Automatische Ermittlung zusätzlicher, klassifikationsrelevanter Merkmale, dadurch Aufbau effizienterer Klassifikationssysteme
- Automatische Regelgenerierung für die automatische Klassifikation, z.B. für bislang manuell geführte Klassifikationssysteme
- Automatisierung der Erweiterung bestehender Klassifikationssysteme, z.B. bei Erweiterung des Produktspektrums bei einem Kunden

## **6.1.2 Institut für Rechneranwendung in Planung und Konstruktion (RPK)**

### **6.1.2.1 Übersicht**

Das *Institut für Rechneranwendung in Planung und Konstruktion (RPK)* der Universität Karlsruhe (TH) ist Teil der Fakultät für Maschinenbau und betreibt Forschung und Entwicklung im Bereich der CAD/CAM-Technologien mit dem Ziel, die Unterstützungspotenziale der Informations- und Kommunikationstechnologien zu erarbeiten und für den gesamten Produktentwicklungsprozess bereitzustellen. Die Forschungsgebiete umfassen Arbeiten in den Bereichen Grundlagenforschung, industrienaher Forschung und Technologietransfer zu Klein- und mittelständischen Unternehmen. Die Forschungsergebnisse finden Eingang in die Ausbildung von Studenten, insbesondere in Vorlesungen, Seminaren und Praktika, die an der Fakultät für Maschinenbau in Karlsruhe gehalten werden. Alle Ausbildungsaktivitäten werden für Studenten des Maschinenbaus, der Informatik und der Wirtschaftswissenschaften angeboten.

Das RPK besteht seit 1977 und arbeitet an verschiedenen Themengebieten im CAD/CAM-Umfeld wie z.B. Produktmodellierung, Fertigungs- und Montageplanung, Informations- und Expertensystemen sowie CAD/CAM-Planung und Organisation. Das RPK ist in Standardisierungsaktivitäten innerhalb des DIN und der ISO beteiligt, wo Beiträge zur Produktmodellentwicklung, der Standardisierung von Schnittstellen (STEP) und der Entwicklung von Application Protocols geleistet werden. Das Institut ist Gründungsmitglied des ProSTEP-Vereins, welcher im Herbst 1993 mit dem Ziel gegründet wurde, Beiträge zur STEP-Normung zu leisten, STEP-Werkzeuge zu entwickeln und den Standard in die industrielle Praxis einzuführen.

Das Institut ist von 1990 bis 2002 im Sonderforschungsbereich SFB 346 "Rechnerintegrierte Konstruktion und Fertigung von Bauteilen" der Universität Karlsruhe federführend tätig. Schwerpunkte sind hierbei die Entwicklung eines integrierten Produkt- und Produktions-

modells (PPM) und von Anwendungen innerhalb der frühen Phasen der Konstruktion. Der tatsächliche Hintergrund des RPK im Umfeld der Rechneranwendung ist weiterhin gegründet auf die Mitarbeit in zahlreichen ESPRIT-Projekten wie z.B. CAD\*I, IMPACT, VIMP, DYNAMO, CACID, PISA, KACTUS, LITE, FODATEC, DOCSTEP, IMS Testcase 6, Gremien wie zahlreichen VDI- und DIN-Arbeitskreisen (CEFE, AIF, CAM\*I, KOMFORCE, etc.) sowie in zahlreichen öffentlich und industriell geförderten Projekten, z.B. CAD-Referenzmodell, Forschungsschwerpunkt Informationslogistik, Verbundprojekt „Automatische Klassifikation von Produkten“ (PAK), etc.

### **6.1.2.2 Zielsetzung zur Nutzung von KLAUSTER**

Die Zielsetzung für das Institut für Rechneranwendung in Planung und Konstruktion (RPK) war unter anderem das Know-How im Bereich der Produktklassifikation mit dem unternehmensspezifischen Klassifikationssystem zu erweitern. Dieses Know-How konnte damit unter anderem in verschiedene Lehrveranstaltungen wie Vorlesungen, Übungen und Praktika ein und wird so an die Studenten weitergegeben werden.

Durch die enge Zusammenarbeit mit Industrieunternehmen bekam das RPK Einblick in die dringenden Probleme in der industriellen Praxis. Diese Erfahrungen werden für eine ergebnisorientierte Ausrichtung künftiger Forschungsaktivitäten genutzt.

Das während der Projektlaufzeit erarbeitete Know-How im Bereich der automatischen Klassifikation ist darüber hinaus eine Grundlage für die Durchführung künftiger Forschungsvorhaben im Themenfeld der funktionalen Klassifikation.

### **6.1.3 Arc Solutions GmbH**

#### **6.1.3.1 Übersicht**

ARC Solutions GmbH ist ein am Erfolg seiner Kunden orientiertes Team, bestehend aus 6 Mitarbeitern, Maschinenbauingenieuren, Informatikern und Softwareentwicklern. Seit 2002 konnte sich das Unternehmen im Bereich Product Lifecycle Management (PLM) erfolgreich am Markt etablieren und setzt ca. 0,5 Mio. € jährlich um. Hauptstandort ist Chemnitz/Sachsen. Vertretungen existiert in Deutschland und der Schweiz.

#### **Leistungsangebot**

ARC Solutions bietet Ingenieur-Dienstleistungen, Software, Content und Services an. PLM- und CAx-Standardprodukte führender Systemanbieter und eigene anwenderorientierte Tools der REMARC Serie werden zu integrierten Lösungen für Fertigungsunternehmen kombiniert. Ein Schwerpunkt liegt dabei im Bereich Klassifikation sowie Norm- und Wiederhol- und Ähnlichteilmanagement.



Langjährige Erfahrungen bestehen mit marktgängigen PLM- und CAD-Systemen, wie Teamcenter, Unigraphics, Pro/Engineer, SolidEdge, Medusa, Bravo/BravoFrame u.a.

Ein weiterer Schwerpunkt könnte in der kundenspezifischen Softwareentwicklung, wie der Anpassung (customizing) Installation und Inbetriebnahme (rollout) komplexer PLM-Lösungen, der Systemintegration sowie der Schulung etabliert werden. Das Leistungsangebot schließt auch die Erweiterung, Ertüchtigung (software refactoring) und Migration bestehender Lösungen ein.

### **Zielkunden**

Unternehmen des Maschinen-, Anlagen- und Fahrzeugbaus; CAD-PLM-Anbieter und Systemintegratoren.

#### **6.1.3.2 Zielsetzung zur Nutzung von KLAUSTER**

ARC Solutions will die automatisierte Klassifikation auf Datenbestände bei ihren Kunden in den Bereichen der Konstruktion und Planung einsetzen. Bei der Realisierung und Betrieb von PLM-Lösungen entstehen erhebliche Datenbestände, die durch eine automatisierte Klassifikation besser auswert- und handhabbar gemacht werden sollen, indem die Erstellung und Anpassung unternehmensspezifischer bzw. prozessspezifischer Klassifikationssysteme mit der KLAUSTER-Methode unterstützt wird. Dabei erwartet ARC Solutions durch die Mitarbeit eine Verbesserung des eigenen Produkt- und Serviceportfolios. Durch ihr Produkt REMARC besitzt ARC Solutions Erfahrungen im Bereich Norm- und Wiederholteilmanagement. Diese und weitere Vorhaben dienen der zunehmenden Differenzierung der Produktfamilie REMARC für das Component Supply Management (CSM) am Markt.

#### **6.1.4 AUCOTEAM GmbH**

##### **6.1.4.1 Übersicht**

Die AUCOTEAM- Ingenieurgesellschaft für Automatisierungs- und Computertechnik GmbH ist ein Systemhaus für Automatisierungstechnische Lösungen und ist 1991 durch Management by Out unter Mitarbeiterbeteiligung von 25% der Belegschaft aus einem vormaligen Forschungszentrum für Automatisierungsgeräte des abzuwickelnden ehemaligen großen DDR-Industriekombinates „Elektro-Apparate-Werke Berlin“ (KEAW) hervorgegangen. Das Unternehmen konnte sich erfolgreich auf dem Markt etablieren und seinen Umsatz, der heute bei 19 Mio. € liegt, kontinuierlich Jahr für Jahr steigern.

AUCOTEAM verfügt an seinem derzeitigen Standort in 10407 Berlin, Storkower Straße 115a gegenwärtig über 165 Mitarbeiter, meistens Ingenieure und einige Facharbeiter für Softwareengineering, für Planung und Projektierung im Automatisierungsanlagenbau und für dazugehörige Fertigung, Ausführung und Service. Der Fertigungsbereich wurde 2001 als 100%-

ige Tochter der AUCOTEAM GmbH mit Standort in der Nähe ausgegründet. Weiterhin wird seit 2003 durch AUCOTEAM eine Berufsfachschule für Automatisierungs- und Computertechnik betrieben.

### **Unternehmenstätigkeit**

Die Geschäftstätigkeit erstreckt sich über die Bereiche *Automatisierungsanlagen* für Industrie, Umwelt und Gebäude (MSR- und Schaltanlagentechnik, Prozessleittechnik und Fernwirktechnik), *Datenkommunikationsanlagen* für Behörden, Verwaltungen und Betriebe, *Softwareengineering und Systemintegration* für Automatisierungs- und Informationssysteme mit marktgängigen Hard- und Softwareprodukten sowie *Auftragsentwicklung* und andere *innovative Dienstleistungen* (F&E-Dienstleistungen, Umweltprüfungen, Informationsdienstleistungen, Schulung sowie Innovationsberatung und -förderung).

### **Leistungsangebot**

AUCOTEAM bietet Ingenieur-Dienstleistungen für den gesamten Lebenszyklus einer Automatisierungslösung an, wie Planung, Realisierung (Aufbau / Installation), Inbetriebnahme und Service z. B. durch Fernwartung. Das Leistungsangebot schließt auch die Erweiterung und Ertüchtigung (Softwareengineering) bestehender Automatisierungsanlagen. Große Erfahrungen bestehen mit marktgängigen Prozessleitsoftware-Systemen, wie WinnCC, Factory Link, Aprot, InTouch u.a. Ein weiterer Schwerpunkt der Leistungen liegt in der kundenspezifischen Softwareentwicklung, wie der Datenbankprojektierung und -installation (Oracle, SQL-Server u.a.), der Softwareentwicklung für eingebettete und für Web-fähige Lösungen, der Systemintegration sowie der Schulung für Informations- und Verwaltungssysteme.

### **Zielkunden**

Stahl- und Walzwerke, Energiewirtschaft, Stadtwerke, Entsorgungswirtschaft, Wasser-Abwasserwerke, Gebäudewirtschaft, Logistik, Behörden und Verwaltungen.

#### **6.1.4.2 Zielsetzung zur Nutzung von KLAUSTER**

Die AUCOTEAM GmbH wird die automatisierte Klassifikation auf Datenbestände in den Bereichen der Planung, Realisierung und des Betriebens von Automatisierungsanlagen und Prozessleitsystemen einsetzen. Bei der Realisierung von Automatisierungsanlagen und im praktischen Betrieb prozessleittechnischer Lösungen entstehen erhebliche Datenbestände, die durch eine automatisierte Klassifikation besser auswert- und handhabbar gemacht werden sollen, indem die Erstellung und Anpassung unternehmensspezifischer bzw. prozessspezifischer Klassifikationssysteme mit der KLAUSTER-Methode unterstützt werden kann. Hierdurch ist folgender Nutzen möglich:

- Finden wiederverwendungsfähiger Strukturen in technischen Unterlagen bzw. Produktlösungen

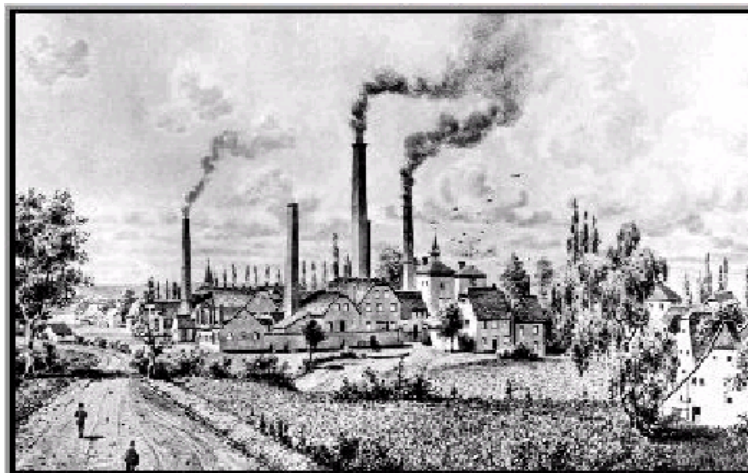
- Zeit- und Kostenersparnis bei der Erstellung von technischen Dokumentationen und bei der Auswahl der zu beschaffenden Bauteile
- Vermeidung subjektiver Entscheidungen bzgl. der Klassifikationsrelevanz von Produkt- bzw. Anlagenmerkmalen
- Ermittlung zusätzlicher klassifikationsrelevanter Merkmale für eine effiziente Klassifikation
- Regelgenerierung für eine automatisierte Klassifikation, z.B. für bislang manuell geführte Klassifikationssysteme

Dazu ist die entstandene Prototypsoftware weiter zu erproben und weiterzuentwickeln, indem mit neu zu gewinnenden quantitativen Aussagen zur Leistungsfähigkeit einige algorithmische Verfahren von KLAUSTER verbessert werden, um den Rechenzeitaufwand zu reduzieren.

## 6.1.5 Das Unternehmen MAN TURBO AG

### 6.1.5.1 Überblick

Die MAN TURBO AG ist ein Unternehmen des Maschinen- und Anlagenbaus. Es blickt auf eine über 200-jährige Geschichte zurück. Im Jahr 1758 gründete Franz Ferdinand Domherr von Wengen das erste schwerindustrielle Unternehmen im Ruhrgebiet: die St. Antony Hütte bei Osterfeld, auch „Wiege der Ruhrindustrie“ genannt. 1782 wurde in Sterkrade die Hütte „Gute Hoffnung“ in Betrieb genommen. „Gute Hoffnung“ in Sterkrade wurde zum ersten Schienenhersteller Deutschlands. 1808 schlossen sich die drei Hütten St. Antony, Gute Hoffnung und Neu Essen zur Hüttengesellschaft und Handlung Jacobi, Haniel & Huysen zusammen.



*Bild 6-1: Wiege der Ruhrindustrie - Stammhütte „Gute Hoffnung“, Oberhausen-Sterkrade 1834*

1873 wurde das Unternehmen in Gutehoffnungshütte Actienverein für Bergbau und Hüttenbetrieb (GHH) umbenannt. Die Gutehoffnungshütte strebte eine Ausweitung der Geschäftstätigkeit auf die stahlverarbeitende Industrie an. 1921 wurde die MAN durch die GHH unter der Leitung von Paul Reusch übernommen. Die GHH und die MAN erlitten im zweiten Weltkrieg schwere Kriegsschäden und verloren alle Auslandsniederlassungen. Nach der Besetzung übernahmen die Alliierten die Kontrolle über die Unternehmen der Gutehoffnungshütte. 1996 übernahm die MAN Gutehoffnungshütte AG Oberhausen die Turboverdichteraktivitäten der Deutschen Babcock AG. Das so entstandene Unternehmen MAN Turbomaschinen GmbH GHH BORSIG vereinigte sich im Jahr 2001 mit der Sulzer Turbo zur MAN TURBO AG. Die MAN TURBO AG hat mit Oberhausen als Firmensitz, Berlin, Zürich und Schio vier produzierende Standorte und ein weltweit verzweigtes Servicenetz (siehe Bild 6-2).



**Standort Oberhausen**  
 MAN TURBO AG  
 Steinbrinkstr. 1  
 46145 Oberhausen  
 Deutschland

Tel.: +49-208-692-01  
 Fax: +49-208-692-2019  
[www.manturbo.com](http://www.manturbo.com)



**Standort Zürich**  
 MAN TURBO AG  
 Schweiz  
 Hardstr. 319  
 8005 Zürich  
 Schweiz

Tel.: +41-1-278-2211  
 Fax: +41-1-278-2989



**Standort Berlin**  
 MAN TURBO AG  
 Egellsstr. 21  
 13507 Berlin  
 Deutschland

Tel.: +49-30-4301-03  
 Fax: +49-30-4301-2841

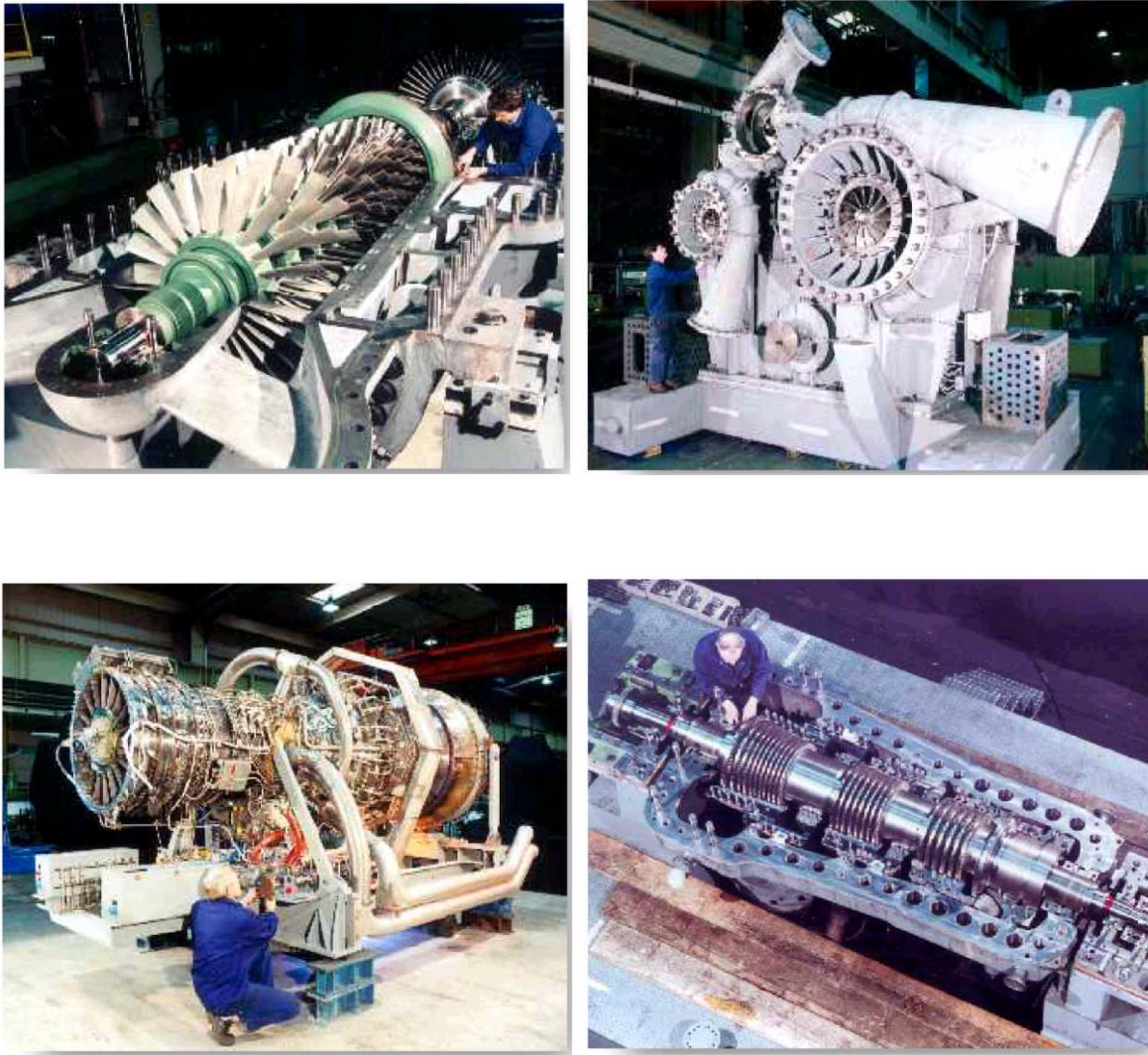


**Standort Schio**  
 MAN TURBO S.r.l.  
 De Pretto  
 Via Daniele Manin 16/18  
 36015 Schio (VI)  
 Italien

Tel.: +39-0445-691-511  
 Fax: +39-0445-511-138

*Bild 6-2: Firmensitz und produzierende Standorte der MAN TURBO AG*

Das Unternehmen beschäftigt weltweit ca. 2.500 Mitarbeiter und machte 2003 einen Umsatz von 567 Mio. Euro. Es bietet seinen Kunden die weltweit umfassendste Produktpalette für Kompressoren und Turbinen an für nahezu jeden technischen Prozess - von Einzelmaschinen bis zu kompletten Maschinensträngen. Hierzu gehören u. a. Axialkompressoren, Radialkompressoren, Getriebekompressoren, Prozessgas-Schraubenkompressoren, Prozessgasturbinen axialer und radialer Bauart, Industriedampfturbinen und Industriegasturbinen als mechanischer Antrieb und Generatorantrieb bis 26 MW. Mit einem eigenen Maschinenleitsystem „turbolog“ werden die Maschinen kontrolliert. Außerdem unterhält das Unternehmen einen weltweiten After Sales Service für die oben aufgeführten Produkte, der eine schnelle Ersatzteilversorgung (OEM), fachkundige Reparaturen und Umbauten und eine umfassende technische Beratung garantiert. Der Service umfasst ebenso Automatisierungssysteme, wie Maschinen-, Regel- und Überwachungssysteme, Maschinendiagnosen und Teleservice.



*Bild 6-3: Fertigung bei der MAN TURBO AG*

### **6.1.5.2 Motivation und Ausgangslage für die Teilnahme am KLAUSTER -Projekt**

MAN TURBO AG weist in dem Prozess der Auftragsabwicklung einige Besonderheiten auf. Als Einzelmaschinenbauer fallen sehr viele ähnliche, aber nicht identische Baugruppen und Komponenten an. Es muss eine schnelle und sichere Analyse der Auslegungs- und Konstruktionsparameter von bereits entwickelten Maschinen erfolgen, um die technologische Erfahrung für neue Maschinen nutzen zu können. Dazu gehört u.a. auch die kritische Überprüfung und Optimierung des Engineeringprozesses. Zur Beantwortung von Fragen in diesem Umfeld war es notwendig zu klären, ob Klassifikationsmöglichkeiten von Baugruppen und Komponenten bei MAN TURBO AG umsetzbar sind.

Die MAN TURBO AG hat sich in der Vergangenheit mehrfach intensiv mit dem Thema „Klassifikation von Produkten“ befasst. Es wurden verschiedene Verfahren zur Klassifikation analysiert und teilweise firmenspezifisch entwickelt. In allen Fällen hat sich aber gezeigt, dass der Aufwand einer manuellen Definition und Pflege der Klassensysteme (Klassen, Merkmale) auf Grund der komplexen Produktstruktur wirtschaftlich nicht zu rechtfertigen war. Auf die

Klassifikation von Stammdaten für Materialien wurde daher an den Standorten Oberhausen und Berlin verzichtet, da auftragspezifische Teile in der Einzelfertigung geringe Wiederverwendungen haben. Durch die fehlende Klassifikation der Bauteile ist das Suchen im Bestand der Materialstammsätze im SAP R/3 nicht möglich oder sehr zeitaufwändig.

### **6.1.5.3 Zielsetzung zur Nutzung von KLAUSTER**

Bei Sulzer Turbo ist die Klassifikation der Materialstämme im Einsatz. Durch den Zusammenschluss 2001 musste die MAN TURBO AG die Frage der Klassifikation neu überdenken. Hierzu sollten durch die Mitarbeit im Verbundprojekt KLAUSTER einerseits fundierte Lösungsansätze erarbeitet werden, andererseits war die MAN TURBO AG bereit, die eigenen problematischen Erfahrungen mit dem Thema Klassifikation in das Projekt einzubringen.

Die Zielsetzung für die MAN TURBO AG im Projekt KLAUSTER war die automatisierte Generierung von Merkmalsgruppen und Klassenstrukturen zur Verwendung im SAP R/3 Klassifikationssystem. Die Klassifikationssystematik sollte speziell angepasst sein für Materialstämme aus den Bereichen Anlagenplanung, Regelungstechnik, Konstruktion und Service. Sie sollte bestehende Objekthierarchien und Objektverknüpfungen z.B. zu Zeichnungen, Datenblättern und Zeugnissen berücksichtigen. Die MAN TURBO AG hat sich schwerpunktmäßig mit der Analyse der eigenen Datenbestände zur Ableitung von bewertbaren Merkmalen, Merkmalgruppen und Klassen befasst. Die Evaluierung der erarbeiteten Konzepte wurde im realen, produktiven Umfeld durchgeführt.

MAN TURBO AG plante den Einsatz der KLAUSTER- und PAK-Software nicht nur für die Klassifikation von Materialstammsätzen sondern auch für die Klassifikation von Dokumenten im Zusammenhang mit dem Aufbau eines elektronischen Archivs für auftragsbezogene Dokumente. Die Klassifikation sollte so flexibel gestaltet sein, dass pro Dokumentenart und Dokumententyp unterschiedliche Merkmale und Klassenhierarchien definiert werden können. Durch die hohe Flexibilität sollten beliebige Merkmale von beliebigen SAP-Objekten direkt ansprechbar sein. Dies sollte durch eine tabellengesteuerte, konfigurierbare Merkmalzuordnung erfolgen.

Die Ergebnisse der KLAUSTER- und PAK- Systeme sollten ins SAP R/3-System automatisiert übertragen werden.

## **6.1.6 Firma EIGNER Germany GmbH**

### **6.1.6.1 Übersicht**

Eigner Germany GmbH ist ein weltweit führender Anbieter von Precision Product Lifecycle Management (PLM)-Lösungen, die es ermöglichen, Produktdaten im gesamten Unternehmen und darüber hinaus in der Supply Chain zu nutzen. Damit werden unterschiedlichste Kundenanforderungen umfassend abgedeckt, die globale Entwicklung mit Fokus auf eine innovative

Produktentwicklung unterstützt und eine gleichbleibend hohe Produktqualität sowie Wartung und Service gewährleistet. Das heißt, die Effizienz innerhalb des gesamten Produktlebenszyklus wird maßgeblich verbessert. Eigner Germany GmbH wurde 1985 gegründet.

Die Firma Eigner Germany GmbH bietet Produkte und Dienstleistungen im Bereich Produkt Lifecycle Management an. Eigner PLM verwaltet entscheidende Engineering-Informationen wie Projekte, Produktspezifikationen, CAD-Modelle, Zeichnungen, Stücklisten, Arbeitsaufträge, Änderungsanträge/-aufträge, Dokumente und ermöglicht gleichzeitig ein umfangreiches Konfigurationsmanagement. Produktentwicklungsteams, die sich über mehrere organisatorische, geographische oder firmeninterne Grenzen erstrecken, haben 24 Stunden an 7 Tagen in der Woche zu jeder Zeit Zugriff auf korrekte und aktuelle Projektinformationen, die in Eigner PLM verwaltet werden.

Um die Engineering-Daten effektiv im gesamten Unternehmen zu verteilen, bietet Eigner PLM Schnittstellen zu den führenden Business Applikationen wie CAD/CAE/CAM-Systeme, Enterprise Resource Planning (ERP)-, Supply Chain Management (SCM)- und Customer Relationship Management (CRM). Informationen werden nahtlos in diese Systeme übernommen, womit sich die kostenaufwändige Neueingabe von Daten erübrigt. Damit stehen die Engineering-Daten durchgängig zur Verfügung und bilden die Basis für intelligente und fundierte Entscheidungsprozesse. Eigner PLM-Lösungen können den speziellen Anforderungen angepasst und derart erweitert werden, dass die relevanten Unternehmensprozesse des Kunden abgebildet werden können.

Eigner kann weltweit auf mehr als 800 zufriedene Kunden aus unterschiedlichsten Branchen wie Aerospace und Defense, Automobil und Transport, High Tech sowie aus dem Maschinen- und Anlagenbau und dem Medizinsektor verweisen. Ausgewählte Kunden sind: Siemens, Philips, Lockheed Martin, Varian Semiconductor Equipment Associates, Volkswagen, DaimlerChrysler Aerospace, Hübner, Endress+Hauser und Boysen.

#### **6.1.6.2 Zielsetzung zur Nutzung von KLAUSTER**

Die Eigner wird in vielen Projekten mit dem Thema Klassifizierung konfrontiert. So wird beispielsweise verlangt, Klassifizierungsinformationen mittels Schnittstellen zu übertragen. Die Klassifizierungsinformationen werden aus den verschiedenen angebundenen Systeme gesammelt und im PLM-System zur Klassifizierung genutzt. Bisher werden solche Probleme noch projektbezogen gelöst. An dieser Stelle verspricht sich die Firma Eigner, die im Forschungsprojekt gemachten Erfahrungen in bestehende und ergänzende Produkte einfließen zu lassen, um den Automatisierungsgrad in Bezug auf Klassifizierung zu erhöhen.

Die Firma Eigner war bereits an dem Verbundprojekt PAK beteiligt und hat in diesem Rahmen die Anbindung von CADIM an die in PAK entstandene Klassifikationssoftware realisiert.



## **6.1.7 Arc Solutions GmbH**

### **6.1.7.1 Übersicht**

ARC Solutions GmbH ist ein am Erfolg seiner Kunden orientiertes Team, bestehend aus 6 Mitarbeitern, Maschinenbauingenieuren, Informatikern und Softwareentwicklern. Seit 2002 konnte sich das Unternehmen im Bereich Product Lifecycle Management (PLM) erfolgreich am Markt etablieren und setzt ca. 0,5 Mio. € jährlich um. Hauptstandort ist Chemnitz/Sachsen. Vertretungen existiert in Deutschland und der Schweiz.

#### **Leistungsangebot**

ARC Solutions bietet Ingenieur-Dienstleistungen, Software, Content und Services an. PLM- und CAx-Standardprodukte führender Systemanbieter und eigene anwenderorientierte Tools der REMARC Serie werden zu integrierten Lösungen für Fertigungsunternehmen kombiniert. Ein Schwerpunkt liegt dabei im Bereich Klassifikation sowie Norm-, Wiederhol- und Ähnlichteilmanagement.

Langjährige Erfahrungen bestehen mit marktgängigen PLM- und CAD-Systemen, wie Teamcenter, Unigraphics, Pro/Engineer, SolidEdge, Medusa, Bravo/BravoFrame u.a.

Als weiterer Schwerpunkt konnte in der kundenspezifischen Softwareentwicklung, wie der Anpassung (customizing) Installation und Inbetriebnahme (rollout) komplexer PLM-Lösungen, der Systemintegration sowie der Schulung etabliert werden. Das Leistungsangebot schließt auch die Erweiterung, Ertüchtigung (software refactoring) und Migration bestehender Lösungen ein.

#### **Zielkunden**

Unternehmen des Maschinen-, Anlagen- und Fahrzeugbaus; CAD-PLM-Anbieter und Systemintegratoren.

### **6.1.7.2 Zielsetzung zur Nutzung von KLAUSTER**

ARC Solutions will die automatisierte Klassifikation auf Datenbestände bei ihren Kunden in den Bereichen der Konstruktion und Planung einsetzen. Bei der Realisierung und Betrieb von PLM-Lösungen entstehen erhebliche Datenbestände, die durch eine automatisierte Klassifikation besser auswert- und handhabbar gemacht werden sollen, indem die Erstellung und Anpassung unternehmensspezifischer bzw. prozessspezifischer Klassifikationssysteme mit der KLAUSTER-Methode unterstützt wird. Dabei erwartet ARC Solutions durch die Mitarbeit eine Verbesserung des eigenen Produkt- und Serviceportfolios. Durch ihr Produkt REMARC besitzt ARC Solutions Erfahrungen im Bereich Norm- und Wiederholteilmanagement. Diese und weitere Vorhaben dienen der zunehmenden Differenzierung der Produktfamilie REMARC für das Component Supply Management (CSM) am Markt.

## 7 Literatur

- [Alt-96] Altshuller, G.: And Suddenly the Inventor Appeared. Technical Innovation Center, Inc., Worcester, 1996.
- [Bei-81] Beitz, W.; Grote, K.-H. (Hrsg.): Taschenbuch für den Maschinenbau. Springer-Verlag, Berlin, Heidelberg, New York, Barcelona, Budapest, Hong Kong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, 1997.
- [DIN-81] DIN (Hrsg.): DIN 4000: Sachmerkmalleisten: Begriffe und Grundsätze. Beuth Verlag, Berlin, 1981.
- [DIN-87] DIN - Deutsches Institut für Normung e.V.: DIN 1463: Erstellung und Weiterentwicklung von Thesauri. Teil 1: Einsprachige Thesauri. Teil 2: Mehrsprachige Thesauri, Beuth-Verlag, Berlin, 1987.
- [Drei-75] Dreibholz, D.: Ordnungsschemata bei der Suche von Lösungen. Konstruktion 27 (1975) S. 233-240.
- [eClass-02] <http://www.eclass.de> : Online-Dokumentation des eClass-Klassifikationssystems, Stand Mai 2002
- [Ehr-95] Ehrlenspiel, K.: Integrierte Produktentwicklung: Methoden für Prozessorganisation, Produkterstellung und Konstruktion. Carl Hanser Verlag, München, Wien, 1995.
- [GaKn-86] Gallagher, C. C.; Knight, W. A.: Group Technology Production Methods in Manufacture. Chichester, Ellis Horwood, 1986.
- [Gra-01] Grabowski, H.; Lossack, R.; Weißkopf, J.: Datenmanagement in der Produktentwicklung. Hanser-Verlag, München, Wien, 2001
- [ISO-95] ISO (Hrsg.): ISO/CD 13584-10: Parts library: Conceptual Model of Parts Library. International Standardization Organization, Genf, 1995.
- [ISO-97] ISO (Hrsg.): ISO/DIS 13584-1: Parts library: Overview and Fundamental Principles. International Standardization Organization, Genf, 1997.
- [ISOa-97] ISO (Hrsg.): ISO/IS 13584-42: Parts library: Description methodology: Methodology for structuring parts families. International Standardization Organization, Genf, 1997.
- [Kälb-80] Kälberer, G.: Automatische Klassifizierung von Arbeitsplätzen mit Hilfe der Clusteranalyse. Zwf 75, Nr.3, S.109-112, 1980.
- [KaPa-84] Kaufmann, H.; Pape, H.: Clusteranalyse. In Fahrmeir, L.; Hamerle, A. (Hrsg.): Multivariate statistische Verfahren; de Gruyter, Berlin, 1984
- [Mee-99] Meerkamm, H.; Heynen, C.; Sander, S.: Integriertes Komplexitätsmanagement auf Basis Intelligenter Lösungsbibliotheken. Zwf Zeitschrift für wirtschaftlichen Fabrikbetrieb 94 (1999) S. 498-502.

- [Ond-98] Ondracek, N.: Überführung bestehender Sachmerkmale in ein Merkmal-Lexikon. In: DIN (Hrsg.), Referatensammlung der DIN-Tagung Merkmallexikon in der Anwendung. Beuth Verlag, Berlin, Wien, Zürich, 1998.
- [Opi-71] Opitz, H.: Werkstückbeschreibendes Klassifizierungssystem. Giradet Verlag, Essen, 1971.
- [Osh-02] O'Shea, M. A.: Produktkonzeption und -gestaltung - Ein Ansatz zur Überbrückung der methodischen Lücke. Zwf Zeitschrift für wirtschaftlichen Fabrikbetrieb 97 (2002).
- [Ott-00] Otto, K.; Wood, K.: Product Design: Techniques in Reverse Engineering and New Produkt Development. Prentice Hall, Upper Saddle River, NJ, 2000.
- [Pah-97] Pahl, G.; Beitz, W.: Konstruktionslehre: Methoden und Anwendung. Springer Verlag, Berlin, Heidelberg, New York, Barcelona, Budapest, Hong Kong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, 1997.
- [Rot-94] Roth, K.: Konstruieren mit Konstruktionskatalogen, Band 2 Konstruktionskataloge. Springer Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona, Budapest, 1994.
- [San-01] Sander, S.: Konzept einer digitalen Lösungsbibliothek für die integrierte Produktentwicklung. VDI Verlag, Düsseldorf, 2001.
- [Schö-99] Schöttner J.: Produktdatenmanagement in der Fertigungsindustrie. München, Carl-Hanser-Verlag, 1999.
- [Sto-97] Stone, R. B.: Towards a Theory of Modular Design. Dissertation, The University Of Texas At Austin, Texas, 1997.
- [VDI-2210] Verein Deutscher Ingenieure: VDI-Richtlinie 2210: Datenverarbeitung in der Konstruktion; Analyse des Konstruktionsprozesses im Hinblick auf den EDV-Einsatz, VDI-Verlag Düsseldorf, November 1975
- [Wei-02] Weißkopf, Jörg.: Automatische Produktdatenklassifikation in heterogenen Datenbeständen, Shaker Verlag, 2002
- [Wien-71] Wiendahl, H.-P.: Funktionsbetrachtungen technischer Gebilde – Ein Hilfsmittel zur Auftragsabwicklung in der Maschinenbauindustrie. Dissertation, TH Aachen, 1971.
- [Tho-90] Thoben, K.-D.: CAD Sparen durch Wiederholkonstruktionen. VDI-Verlag, Düsseldorf, 1990.