

Komponentendokumentationen – Eine systematische Bewertung von Ordnungssystemen aus formaler Sicht

Peter Fettke, Peter Loos

Technische Universität Chemnitz, Professur Wirtschaftsinformatik II, D-09107 Chemnitz, Germany, Tel.: +49/371/531-4375, Fax: -4376, E-Mail: {peter.fettke|p.loos}@wirtschaft.tu-chemnitz.de, WWW: <http://www.tu-chemnitz.de/wirtschaft/wi2/>

Zusammenfassung. Das Zusammensetzen von einzelnen, interagierenden Software-Komponenten zu einem Gesamtsystem setzt voraus, dass die zur Realisierung des Gesamtsystems benötigten Komponenten zuvor in einer Komponentendokumentation aufgefunden worden sind. Eine zielgerichtete und systematische Suche ermöglicht ein Ordnungssystem, das Komponenten strukturiert verwaltet. Ordnungssysteme können unterschiedlich ausgelegt werden. Dieser Beitrag stellt zunächst unterschiedliche Ordnungssysteme vor. Anschließend werden Kriterien zur Bewertung von Ordnungssystemen abgeleitet, anhand derer die zuvor dargestellten Ordnungssysteme kritisch diskutiert werden.

Schlüsselworte: Komponentenklassifikation, Komponententaxonomie, Komponentensuche, Komponenten Retrieval, Qualität von Ordnungssystemen

1 Ausgangssituation und Problemstellung

Um die Vision eines effizienten unternehmensinternen oder -externen Komponentenmarktplatzes zu verwirklichen, müssen Konsumenten von Komponenten in der Lage sein, die für sie geeigneten Komponenten aufzufinden. Eine zielgerichtete Suche erfordert ein Ordnungssystem, das Komponenten strukturiert verwaltet. Ordnungssysteme können unterschiedlich konzipiert werden, so dass sich die Frage stellt, welches Ordnungssystem vorzuziehen ist.

In dieser Arbeit sollen Ordnungssysteme aus formaler Sicht systematisch untersucht werden: *aus formaler Sicht* bedeutet, dass Form, Struktur und Aufbau von Ordnungssystemen interessieren und Fragen zu ihrer konkreten inhaltlichen Ausgestaltung nur exemplarisch und nicht umfassend diskutiert werden; *systematisch* bedeutet, dass explizite Kriterien zur Bewertung der Ordnungssysteme verwendet werden. Damit unterscheidet sich die vorliegende Untersuchung zum einen von dem allgemeinen Teil der Literatur zur Dokumentations- und Ordnungslehre (GAUS 1995; MANECKE 1997; SALTON, MCGILL 1983) und zum anderen von dem Teil der Literatur, der zwar explizit Ordnungssysteme für Komponenten betrachtet, allerdings die vorgestellten Systeme eher cursorisch untersucht (ARNOLD, STEPOWAY 1988; BURTON, ARAGON, BAILEY, KOEHLER, MAYES 1988, S. 32; CONVENT, WERNECKE 1994; FRAKES 1989; FRAKES, NEJMEH 1988, S. 144; FRANK 1999, S. 13; HARS, HEIB, KRUSE, MICHELY, SCHEER 1992; HENNINGER 1997; HOFFMANN, SCHEER, BACKER 1992; LUNG, URBAN 1995; MEIER, RÖPERT 1994; MERTENS 1965, S. 67–80; ONUEGBE 1988; PODGURSKI, PIERCE 1993; PRIETO-DÍAZ 1991; PRIETO-DÍAZ, FREEMAN 1987; PRIETO-DÍAZ, JONES 1988; SOMMERVILLE, WOOD 1986). Indes muss eingeräumt werden, dass in dieser Untersuchung keine eigenen Ordnungssysteme hervorgebracht werden, sondern dass auf die in den genannten Quellen zurückgegriffen wird.

2 Terminologische Vorbemerkung

Komponente Eine Komponente ist ein „wiederverwendbarer, abgeschlossener und vermarktbare Softwarebaustein, der Dienste über eine wohldefinierte Schnittstelle zur Verfügung stellt und in zur Zeit der Entwicklung unvorhersehbaren Kombinationen mit anderen Komponenten einsetzbar ist“ (FELLNER, RAUTENSTRAUCH, TUROWSKI 1999, S. 34).

Komponentendokumentation Eine Komponentendokumentation ist ein Speicher für Komponenten, auf den unter inhaltlichen Suchanfragen gezielt zugegriffen werden kann (In Anlehnung an eine allgemeinere Definition von GAUS 1995, S. 21f.). Er speichert den binären Programmcode sowie evtl. vorhandene textuelle Beschreibungen der zu archivierenden Komponenten.

Ordnungssystem Zentraler Bestandteil einer Komponentendokumentation ist ein Ordnungssystem, das definiert, wie Komponenten in der Dokumentation ablegt und wieder aufgefunden werden können.

Deskriptor Komponentendokumentationen verwenden zum Erschließen von Komponenten Deskriptoren. Ein Deskriptor ist eine Zeichen- und/oder Zahlenfolge, die zur inhaltlichen Kennzeichnung einer Komponente verwendet wird. Im einfachsten Fall ist ein Deskriptor ein Schlagwort, das eine Komponente inhaltlich beschreibt. Deskriptoren können sprachtheoretisch als Prädikatoren interpretiert werden, wodurch es möglich wird, zwischen Intension und Extension eines Deskriptors zu unterscheiden: Die Intension eines Deskriptors ist seine Bedeutung. Dagegen ist seine Extension die Gesamtheit aller Komponenten, denen der Deskriptor zugeordnet wurde (Zur sprachtheoretischen Unterscheidung der Intension und Extension eines Prädikators siehe bspw. SEIFFERT 1991, S. 58–62.).

Indexieren Das Zuordnen eines Deskriptors zu einer Komponente heißt indexieren. Wird die Menge zur Verfügung stehender Deskriptoren vor dem Indexieren verbindlich festgelegt, wird gebunden indexiert, andernfalls ungebunden. („Anstatt *indexieren* müßte es im Deutschen eigentlich *deskribieren* oder *indizieren* heißen, jedoch hat sich in Anlehnung an den englischen Fachausdruck *indexing* die Bezeichnung indexieren . . . durchgesetzt.“ (GAUS 1995, S. 3))

Terminologische Kontrolle Unter einer Terminologischen Kontrolle versteht man „alle Maßnahmen, die direkt oder indirekt der Definition und Abgrenzung der Begriffe und der Zuordnung von Benennungen und Begriffen dienen“ (GAUS 1995, S. 59). Eine Terminologische Kontrolle ist zwingend notwendig, um Homonyme und Teil- bzw. Vollsynonyme offen zu legen. Die Terminologische Kontrolle wird bspw. dadurch unterstützt, dass einem Deskriptor eine Reihe Nicht-Vorzugsbenennungen zugeordnet werden. Nicht-Vorzugsbenennungen sind Wörter, die teil- bzw. vollen synonym zum zugeordneten Deskriptor einzustufen sind, allerdings nicht zur Kennzeichnung einer Komponente verwendet werden sollen. Ein weiteres Beispiel für eine Terminologische Kontrolle ist die Definition von über- bzw. untergeordneten Deskriptoren zwecks klarer Abgrenzung ihrer Intensionen.

3 Beschreibung verschiedener Ordnungssysteme

Klassifikation ohne Überlagerung

Die Klassifikation ohne Überlagerung ist ein verhältnismäßig einfaches Ordnungssystem. Grundgedanke dieses Systems ist es, einer Komponente jeweils genau einen Deskriptor zu indexieren. Es wird gebunden indexiert. Die verwendeten Deskriptoren sollen sich gegenseitig ausschließen. Die Extension eines Deskriptors kann als eine Äquivalenzklasse interpretiert werden. Jede Komponente ist Element genau einer der gebildeten Äquivalenzklassen.

Ein Ordnungssystem, das auf dem Prinzip der Klassifikation beruht, muss so aufgebaut sein, dass jede zu archivierende Komponente durch einen Deskriptor beschrieben werden kann. Diese Anforderung lässt sich formal sicherstellen, indem ein Deskriptor *sonstiges* eingeführt wird.

Je nach Zielsetzung des Ordnungssystems können die gewählten Deskriptoren unterschiedlich große Sachverhalte in einer Äquivalenzklasse zusammenfassen. Es ist sinnvoll, Anwendungsbereiche, die im Zentrum der Dokumentation liegen und für die eine große Anzahl verschiedener Komponenten vorliegen, durch möglichst spezielle Deskriptoren abzudecken, um so in diesen Bereichen eine höhere Indexierungsgenauigkeit zu erreichen. Anwendungsbereiche, die eher Randgebiete der Dokumentation tangieren, können durch breiter fokussierte Deskriptoren abgedeckt werden, weil dort keine so hohe Indexierungsgenauigkeit benötigt wird.

Die Klassifikation ohne Überlagerung wird an einem Beispiel näher erläutert: Es soll ein Ordnungssystem aufgebaut werden, das Komponenten dokumentiert, die Funktionen zur Realisierung von Systemen zur Produktionsplanung und -steuerung (PPS) implementieren. In diesem Szenario bietet es sich an, typische PPS-Aufgaben als Deskriptoren zu verwenden: es können bspw. die Deskriptoren *Primärbedarfsplanung*, *Materialbedarfsplanung*, *Durchlaufterminierung*, *Kapazitätsabgleich*, *Auftragsfreigabe*, *Feinterminplanung* sowie *Betriebsdatenerfassung* verwendet werden. Einer zu archivierenden Komponente wird bei einer Klassifikation ohne Überlagerung genau ein Deskriptor zugeordnet: Bspw. wird einer Komponente, die einen Algorithmus zur Maschinenbelegungsplanung implementiert, der Deskriptor *Feinterminplanung* indexiert.

Klassifikation mit Überlagerung

Eine Klassifikation ohne Überlagerung geht von der Annahme aus, dass einer Komponente genau ein Deskriptor indexiert wird. Diese Annahme ist oft zu restriktiv: Werden Deskriptoren wie im obigen Beispiel so gewählt, dass sie betriebliche Aufgaben beschreiben, ist es durchaus denkbar, dass eine Komponente, die eine Funktionalität zur Feinterminplanung realisiert, ebenfalls Funktionen zur Materialbedarfsplanung anbietet. Dieses Beispiel zeigt, dass es sinnvoll sein kann, einer Komponente mehrere Deskriptoren zu indexieren, obwohl die verwendeten Deskriptoren intensional verschieden sind.

Eine Klassifikation mit Überlagerung gestattet es, einer Komponente mehrere Deskriptoren zu indexieren. Bei diesem Ordnungssystem wird gebunden indexiert und die verwendeten Deskriptoren schließen sich intensional aus.

Monohierarchische Klassifikation

Eine Klassifikation mit oder ohne Überlagerung trifft die Annahme, dass die verwendeten Deskriptoren intensional verschieden sind. Diese Annahme wird bei einer Monohierarchischen Klassifikation dahingehend relaxiert, dass bei diesem Ordnungssystem hierarchische Beziehungen zwischen Deskriptoren explizit zugelassen werden. Eine hierarchische Beziehung zwischen einem übergeordneten Deskriptor O und einem untergeordneten Deskriptor U wird dahingehend verstanden, dass der Deskriptor O den Deskriptor U intensional vollständig umfasst. Dies bedeutet, dass die Indexierung eines Deskriptors U dazu führt, dass zwingend ein evtl. übergeordneter Deskriptor O indexiert wird. Sprachtheoretisch kann der Deskriptor O als Oberbegriff und der Deskriptor U als Unterbegriff interpretiert werden. Werden die Extensionen der Deskriptoren O und U mengentheoretisch betrachtet, bedeutet dies, dass die Extension des Deskriptors U eine Teilmenge der Extension des Deskriptors O ist. Alle einem Deskriptor unmittelbar untergeordnete Deskriptoren müssen sich gegenseitig ausschließen. Ferner ist es bei einer Monohierarchischen Klassifikation nicht zulässig, einem untergeordneten Deskriptor mehrere Deskriptoren überzuordnen: Einem untergeordneten Deskriptor darf maximal ein Deskriptor übergeordnet werden.

Eine Monohierarchische Klassifikation wird von BOOCH (BOOCH 1987, S. 36-40) vorgeschlagen: Auf oberster Hierarchieebene wählt er die Deskriptoren *Structures*, *Tools* und *Subsystems*. Diese werden in weiteren Hierarchieebenen verfeinert (siehe Bild 1).

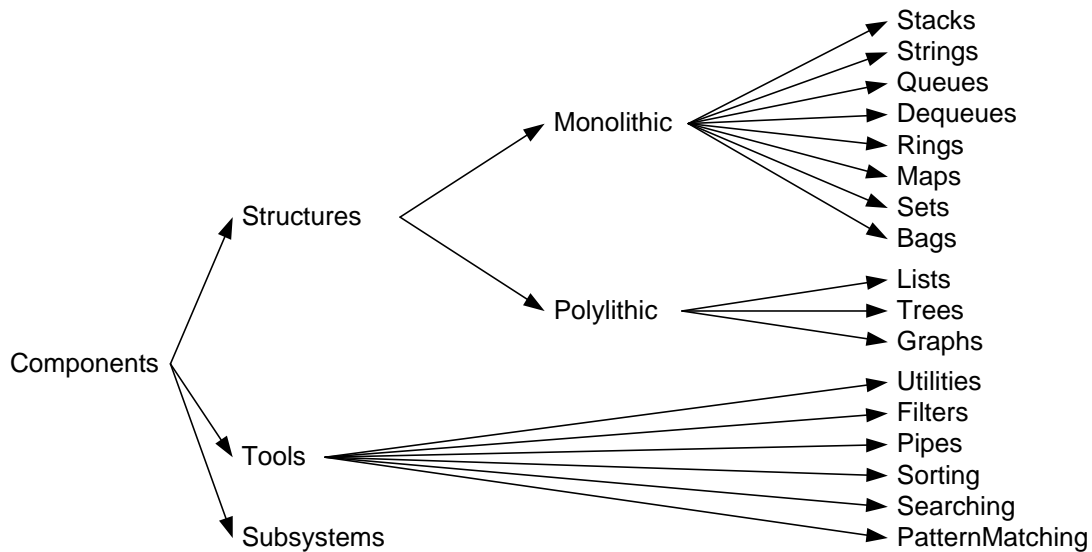


Bild 1: Monohierarchische Klassifikation nach BOOCH

Polyhierarchische Klassifikation

Eine Monohierarchische Klassifikation unterstellt die Annahme, dass einem untergeordneten Deskriptor maximal ein Deskriptor übergeordnet werden darf. Von dieser Annahme wird bei einer Polyhierarchischen Klassifikation abgesehen: Es ist möglich, einem Deskriptor sowohl mehrere Deskriptoren unter- als auch überzuordnen. Abgesehen von diesem Unterschied gleichen sich Polyhierarchische und Monohierarchische Klassifikation.

Register

Das Inhaltsverzeichnis am Anfang eines Fachbuches oder das Stichwortverzeichnis am Ende sind allgemein bekannte Beispiele für Register. Abstrakt gesprochen, besteht ein Register aus einer Liste von Deskriptoren, die alphabetisch, chronologisch oder systematisch angeordnet sind. Einer Komponente können mehrere Deskriptoren indiziert werden, d. h. diese Komponente ist dann über mehrere Deskriptoren innerhalb eines Registers zugreifbar. Beim Einsatz von Registern kann sowohl gebunden als auch ungebunden indiziert werden.

Darüber hinaus können problemlos mehrere Register in einer Dokumentation gleichzeitig eingesetzt werden. Jedes der gepflegten Register ist eigenständig nutzbar und erfüllt einen speziellen Recherchezweck: Bspw. kann jeweils ein eigenes Register zur Dokumentation von Komponenten innerhalb einer Branche eingesetzt werden. Ist bei der Archivierung einer Komponente ersichtlich, dass diese in unterschiedlichen Branchen Verwendung finden könnte, werden die entsprechenden Deskriptoren in den entsprechenden Registern indiziert.

Ferner können in einem Register die verzeichneten Deskriptoren hierarchisch angeordnet werden. Bspw. kann in einem Register, dessen Deskriptoren die Funktion von Komponenten beschreiben, eine Funktion in einer zweiten Hierarchisierungsstufe hinsichtlich der jeweiligen Branche weiter differenziert werden.

Facettenklassifikation

Die Kernidee bei diesem Ordnungssystem ist es, eine Komponente hinsichtlich unterschiedlicher Gesichtspunkte, sog. Facetten, zu klassifizieren. Jede Facette besteht aus unterschiedlich vielen Deskriptoren. Bei der Erschließung einer Komponente wird aus jeder Facette ein geeigneter Deskriptor indiziert.

Facette	Deskriptoren	
Scope	Specification	Implementation
Purpose	Domain	Technology
Granularity	Fine	Coarse
Abstraction	General	Specific

Tabelle 1: Facettenklassifikation nach KAIN

Bspw. kann eine Komponente hinsichtlich der Facetten Funktion (Deskriptoren: Beschaffung, Produktion, Absatz), Branche (Deskriptoren: Industrie, Handel, Dienstleistung) und Komponentenarchitektur (Deskriptoren: ActiveX, JavaBeans, CORBA) klassifiziert werden. Eine Facettenklassifikation hat Gemeinsamkeiten mit anderen Ordnungssystemen: Die Klassifikation hinsichtlich einer Facette kann als Klassifikation ohne Überlagerung verstanden werden. Ferner gleicht eine Klassifikation mit n Facetten einer Klassifikation mit n systematisch geordneten Registern, wobei die zusätzliche Bedingung formuliert wird, dass jeder Komponente genau ein Deskriptor aus jedem Register zwingend indexiert werden muss.

Bei der Implementierung einer Facettenklassifikation ist darauf zu achten, dass die einzelnen Facetten möglichst orthogonal zueinander stehen, also unabhängig voneinander indexiert werden können. Ferner kann es sinnvoll sein, in jeder Facette einen Deskriptor nicht sinnvoll aufzunehmen, der dann indexiert wird, wenn eine Klassifizierung hinsichtlich dieser Facette keinen Sinn ergibt.

Eine Facettenklassifikation wird von KAIN (KAIN 1996) vorgeschlagen. Er definiert vier Facetten mit jeweils zwei Deskriptoren (siehe Tabelle 1).

Begriffskombination

Die Grundidee der Begriffskombination ist es, die Deskriptoren so zu wählen, dass jeder Deskriptor für sich alleine nur einen sehr begrenzten, einzelnen Sachverhalt einer Komponente erschließen kann, allerdings eine Menge gleichzeitig indexierter Deskriptoren eine Komponente sehr detailliert und differenziert beschreibt. Die Deskriptoren müssen also intensional nicht verschieden sein, sondern sollen sich sogar vielfältig überlagern. Dadurch wird es möglich, mehrere Deskriptoren gleichzeitig zur Erschließung einer Komponente zu verwenden, also eine Kombinationsfähigkeit der Deskriptoren zu erreichen. Die Anzahl einer Komponente indexierten Deskriptoren ist von Komponente zu Komponente variabel. Bei der Begriffskombination wird gebunden indexiert.

Schlagwortvergabe

Die Schlagwortvergabe ist eine Vorstufe der Begriffskombination. Auch dabei werden einer Komponente eine unterschiedliche Anzahl von Deskriptoren zugeordnet. Allerdings wird nicht gebunden, sondern ungebunden indexiert, wodurch im Allgemeinen keine hohe Kombinationsfähigkeit der Deskriptoren erreicht wird.

Freitextsuche

Die Freitextsuche setzt zwingend voraus, dass neben dem binären Programmcode einer Komponente ebenfalls eine textuelle Beschreibung der Komponente zur Verfügung steht. Diese Komponentenbeschreibung kann bspw. ein Spezifikationsdokument der Komponente oder eine Beschreibung ihrer Funktion, ihrer Aufgabe, ihres Einsatzgebietes o. ä. sein. Alle Wörter der Komponentenbeschreibung werden als Deskriptoren betrachtet, soweit sie nicht in einer Liste der sog. Nicht-Stichwörter stehen. Die Liste der Nicht-Stichwörter vermeidet die Indexierung von Wörtern, die so häufig in Texten auftreten, dass sie keinen Beitrag zur inhaltlichen Kennzeichnung einer Komponente leisten können. Dies sind bspw.

Artikel und Konjunktionen. Durch dieses Vorgehen erfordert das Indexieren einer Komponente keine intellektuelle Leistung und kann vollständig maschinell durchgeführt werden.

Untersuchungen haben gezeigt, dass die Verwendung des Quelltextes einer Komponente als textuelle Beschreibung problematisch ist, da u. a. die Bedeutung von Schlüsselwörtern im Quelltext oft per Konvention festgelegt wird oder willkürlich ist und somit nicht zur Erschließung der Komponente geeignet ist (PRIETO-DÍAZ 1991, S. 92).

Vektorraummodell

Der Kerngedanke des Vektorraummodells ist es, eine Komponente durch einen Vektor aus einem n -dimensionalen Raum zu repräsentieren (Für eine allgemeine Erläuterung des Vektorraummodells siehe FERBER 1999, S. 919-922; für einen speziellen Ansatz zur Komponentendokumentation OSTER-TAG, PRIETO-DÍAZ, BRAUN 1992; für eine Übersicht über Proximitätsmaße BACKHAUS, ERICHSON, PLINKE, WEIBER 1996, S. 264f.). Jede Dimension des Raumes dient zur Beschreibung eines Komponentenmerkmals. Weiterhin wird ein sog. Proximitätsmaß für den aufgespannten Vektorraum definiert. Ein solches Maß quantifiziert den Abstand oder die Ähnlichkeit zwischen zwei Vektoren des Raumes. Das Vektorraummodell kann als ein Ordnungssystem für Komponenten verstanden werden, wobei ein Beschreibungsvektor als Deskriptor interpretiert wird. Dabei ist es denkbar, Komponentenmerkmale sowohl auf metrischen als auch auf nicht-metrischen Skalen zu erfassen. Typische Merkmale sind bspw. Preis und Laufzeitkomplexität (metrisches Skalenniveau) oder Branche und Funktion (nicht-metrisches Skalenniveau).

Ein Ablauf zum Auffinden einer Komponente gestaltet sich folgendermaßen: Ein Rechercheur formuliert eine Anfrage im Form eines Merkmalsvektors. Anschließend wird auf Basis des definierten Proximitätsmaßes für jede archivierte Komponente der Abstands- bzw. Ähnlichkeitswert im Hinblick auf die gestellte Anfrage berechnet. Auf der Grundlage dieser Berechnung können die Komponenten in der Dokumentation in eine Rangfolge gebracht werden, die die Treffermenge der Anfrage darstellt.

Komponentenkatalog

Innerhalb der Ingenieurwissenschaften sind Konstruktionskataloge ein bekanntes methodisches Hilfsmittel im Konstruktionsprozess (ROTH, FRANKE, SIMONEK 1972; ROTH 1994, s. insb. S. 1-2). Das Konzept der Konstruktionskataloge wird von einigen Autoren auf die Entwicklung betrieblicher Informationssysteme übertragen (LANG 1998, S. 12-13; ORTNER, LANG, KALKMANN o. J., S. 4-6; ORTNER, LANG, KALKMANN 1999, S. 36f.). Bei diesem Ansatz können Konstruktionskataloge als ein Ordnungssystem für Komponenten interpretiert werden. Zur besseren begrifflichen Unterscheidung sollen derartige Kataloge im Folgenden Komponentenkataloge genannt werden.

Ein Komponentenkatalog ist eine in Tabellenform vorliegende Übersicht von Komponenten, die nach methodischen Gesichtspunkten erstellt wird, innerhalb eines gegebenen Rahmens weitestgehend vollständig sowie systematisch gliedert ist. Bild 2 zeigt eine schematische Darstellung eines Komponentenkataloges, aus der die typische Unterteilung in Haupt-, Gliederungs- und Zugriffsteil und einen evtl. ergänzenden Anhang hervorgeht.

Im Hauptteil des Kataloges werden die zu archivierenden Komponenten jeweils zeilenweise aufgezählt. Eine Komponente kann bspw. durch Text, Beschreibungen ihrer Schnittstellen und/oder einem grafischen Modell dokumentiert werden. Der Gliederungsteil bringt die Elemente des Hauptteils in eine einheitliche Systematik und ist vergleichbar mit einer Monohierarchischen Klassifikation. Eine Anforderung an den Gliederungsteil ist es, die im Hauptteil beschriebenen Komponenten widerspruchsfrei und eindeutig zu systematisieren. Der Zugriffsteil beschreibt verschiedene Merkmale einer Komponente und ermöglicht ein gezieltes Auffinden von Komponenten mit spezifischen Eigenschaften. Je nach angestrebter Zielsetzung des Kataloges kann die Anzahl vorhandener Zugriffsmerkmale stark variieren. Der Zugriffsteil entspricht formal dem Vektorraummodell.

Gliederungsteil			Hauptteil			Zugriffsteil			Anhang	
1	2	3	1	2	Nr.	1	2	3	1	2
					1					
					2					
					3					
					4					
					5					
					6					
					7					
					8					

Bild 2: Schematische Darstellung eines Komponentenkataloges

Die wesentlichen Eigenschaften der elf Ordnungssysteme werden in Tabelle 2 gegenübergestellt.

4 Kriterien zur Beurteilung von Ordnungssystemen

Die folgenden Kriterien zur Beurteilung von Ordnungssystemen entstanden durch unterschiedliche Vorgehensweisen: Zunächst wurden Kriterien in der allgemeinen Literatur zur Dokumentations- und Ordnungslehre gesichtet (GAUS 1995, siehe insb. S. 148f.). Des Weiteren wurden die Ausführungen in Arbeiten ausgewertet und konsolidiert, die speziell Bewertungen von Ordnungssystemen für Komponentendokumentationen thematisieren (CONVENT, WERNECKE 1994, S. 61–65; FRAKES 1989, S. 251f.; FRANK 1999, S. 13; HENNINGER 1997, S. 113–115; LUNG, URBAN 1995, S. 171; MERTENS 1965, S. 67-75; MEIER, RÖPERT 1994, S. 49–51; PODGURSKI, PIERCE 1993, S. 288f.; PRIETO-DÍAZ 1991, S. 89; PRIETO-DÍAZ, FREEMAN 1987, S. 8f.; SOMMERVILLE, WOOD 1986, S. 17f.). Ergänzt wurden die Kriterien durch eigene Überlegungen, die darauf abzielten, Eigenschaften von Ordnungssystemen zu bestimmen, die im Hinblick auf die Implementierung und Benutzung von Ordnungssystemen als idealtypisch erscheinen.

Vollständigkeit

Das Kriterium Vollständigkeit ist ein Maß zur Beurteilung, wie gut ein Ordnungssystem in der Lage ist, das zu ordnende Sachgebiet vollständig abzudecken. Eine gute Vollständigkeit ist dann gegeben, wenn Komponenten unabhängig ihrer speziellen Eigenschaften und Merkmale erschlossen werden können.

Präzision

Das Kriterium Präzision ist ein Maß zur Beurteilung, wie präzise ein Ordnungssystem in der Lage ist, eine Komponente zu erschließen. Eine gute Präzision ist dann gegeben, wenn die Indexierungsgenauigkeit von Komponenten leicht gesteigert werden kann und wenn Komponenten gruppierbar oder auffindbar sind, die hinsichtlich bestimmter Merkmale ähnlich sind (Der Nutzen von Analogien bei der Recherche wird ausführlich in MAIDEN, SUTCLIFFE 1992 herausgestellt.).

Konsistenz

Das Kriterium Konsistenz ist ein Maß zur Beurteilung, ob Widersprüche oder Unstimmigkeiten innerhalb eines Ordnungssystems vorhanden sind. Eine gute Konsistenz ist dann gegeben, wenn die Annahmen des Ordnungssystems nicht bzw. nicht leicht verletzt werden können, wenn eine Terminologische Kontrolle

Ordnungssystem	Wesentliche Eigenschaften
Klassifikation ohne Überlagerung	Jeder Komponente wird genau ein Deskriptor indexiert. Die Deskriptoren sind intensional verschieden. Es wird gebunden indexiert.
Klassifikation mit Überlagerung	Einer Komponente können mehrere Deskriptoren indexiert werden. Die Deskriptoren sind intensional verschieden. Es wird gebunden indexiert.
Monohierarchische Klassifikation	Jeder Komponente wird genau ein Deskriptor indexiert. Einem Deskriptor können mehrere Deskriptoren untergeordnet werden. Alle einem Deskriptor direkt untergeordneten Deskriptoren müssen sich gegenseitig ausschließen. Jedem Deskriptor darf maximal ein Deskriptor übergeordnet werden. Es wird gebunden indexiert.
Polyhierarchische Klassifikation	Einem Deskriptor können mehrere Deskriptoren unter- und übergeordnet werden. Alle einem Deskriptor direkt untergeordneten Deskriptoren müssen sich gegenseitig ausschließen. Es wird gebunden indexiert.
Register	Ein Register ist ein alphabetisches, chronologisches oder systematisches Verzeichnis von Deskriptoren, wobei eine hierarchische Anordnung möglich ist. Einer Komponente können mehrere Deskriptoren indexiert werden. Es können mehrere Register gleichzeitig eingesetzt werden. Es wird gebundenen oder ungebundenen indexiert.
Facettenklassifikation	Es wird hinsichtlich unterschiedlicher Gesichtspunkte, sog. Facetten, klassifiziert. Eine Facette besteht aus mehreren Deskriptoren. Jeder Komponente wird genau ein Deskriptor aus jeder Facette indexiert. Es wird gebunden indexiert.
Begriffskombination	Die Deskriptoren werden so gewählt, dass sie jeweils möglichst nur einen einzelnen Sachverhalt abbilden und möglichst gut kombinierbar sind. Einer Komponente können beliebig viele Deskriptoren indexiert werden. Es wird gebunden indexiert.
Schlagwortvergabe	Die Schlagwortvergabe ist eine Vorstufe der Begriffskombination. Einer Komponente können beliebig viele Deskriptoren indexiert werden. Es wird ungebunden indexiert.
Freitextsuche	Es wird zwingend eine textuelle Komponentenbeschreibung benötigt. Alle Wörter einer Komponentenbeschreibung sind Deskriptoren, soweit sie nicht in einer Liste von Nicht-Stichwörtern stehen.
Vektorraummodell	Eine Komponente wird durch einen n -dimensionalen Vektor beschrieben. Die Ähnlichkeit bzw. der Abstand zweier Komponenten wird über ein Proximitätsmaß berechnet.
Komponentenkatalog	Das Ordnungssystem ist eine tabellarische Übersicht von Komponenten, die aus Haupt-, Gliederungs- und Zugriffsteil besteht. Der Zugriffsteil entspricht formal dem Vektorraummodell.

Tabelle 2: Wesentliche Eigenschaften der Ordnungssysteme

gut unterstützt wird und wenn die verwendeten Deskriptoren nicht zu Widersprüchen hinsichtlich der Klassifikation führen.

Erweiterbarkeit

Das Kriterium Erweiterbarkeit ist ein Maß zur Beurteilung, wie gut ein Ordnungssystem erweitert werden kann. Eine gute Erweiterbarkeit ist dann gegeben, wenn neue Deskriptoren im Ordnungssystem problemlos aufgenommen werden können und wenn das Ordnungssystem bei der Dokumentation sehr vieler Komponenten noch praktikabel ist.

Benutzbarkeit

Das Kriterium Benutzbarkeit ist ein Maß zur Beurteilung, wie klar und verständlich ein Ordnungssystem aufgebaut und benutzt werden kann. Eine gute Benutzbarkeit ist dann gegeben, wenn der Grundgedanke des Ordnungssystems nachvollzogen werden kann, wenn die Intensionen der Deskriptoren jeweils verständlich sind und wenn das Ordnungssystem nicht von der Verwendung einer bestimmten natürlichen Sprache abhängig ist.

Wirtschaftlichkeit

Das Kriterium Wirtschaftlichkeit ist ein Maß zur Beurteilung, wie wirtschaftlich ein Ordnungssystem implementiert und benutzt werden kann. Die Wirtschaftlichkeit wird bestimmt durch die fixen Kosten zur organisatorischen und technischen Implementierung des Ordnungssystems (Implementierungskosten), durch die variablen Kosten der Archivierung einer Komponente (Archivierungskosten), durch die variablen Kosten der Recherche (Recherchekosten) und durch die Opportunitätskosten, die durch die Recherche in der Komponentendokumentation eingespart werden können. Derartige Opportunitätskosten sind bspw. Kosten, die zur Eigenentwicklung einer Komponente aufgebracht werden müssten.

Die vorgestellten Bewertungskriterien und -aspekte sind in Tabelle 3 zusammengefasst.

5 Bewertung der Ordnungssysteme

Die Bewertung der Ordnungssysteme kann anhand der vorgestellten Kriterien methodisch grundsätzlich auf zwei Arten erfolgen: Zum einen können die Ordnungssysteme jeweils hinsichtlich aller Kriterien isoliert bewertet werden (isolierte Bewertung). Zum anderen können alle Ordnungssysteme jeweils hinsichtlich eines Kriteriums vergleichend bewertet werden (vergleichende Bewertung). Die isolierte Bewertung wurde zwecks besserer Übersicht in Form von Tabellen vorgenommen. Siehe dazu die Tabellen 4-14 am Ende des Artikels. Die vergleichende Bewertung folgt.

Vollständigkeit

Die Vollständigkeit der Ordnungssysteme ist jeweils entweder gegeben oder kann formal durch Definition eines Deskriptors *sonstiges* sichergestellt werden. Eine Ausnahme bildet die Freitextsuche: Hier ist Vollständigkeit an die Bedingung geknüpft, dass eine textuelle Komponentenbeschreibung vorliegt.

Präzision

Bei der Klassifikation ohne oder mit Überlagerung sowie der Mono- und Polyhierarchischen Klassifikation ist es prinzipiell möglich durch Einführung verfeinerter Deskriptoren die Indexierungsgenauigkeit zu erhöhen. Allerdings ist ab einer kritischen Anzahl die Menge der Deskriptoren nicht mehr überschaubar,

Kriterium	Bewertungsaspekte
Vollständigkeit	Ist das Ordnungssystem in der Lage, alle Komponenten inhaltlich zu erschließen? Gibt es Komponenten, die nicht in das Ordnungssystem eingeordnet werden können?
Präzision	Wie viele Klassen gibt es im Ordnungssystem? Können Suchanfrage nicht nur scharf, sondern auch vage formuliert werden? Wie hoch ist die Indexierungsgenauigkeit des Ordnungssystems?
Konsistenz	Ist das Ordnungssystem in sich widersprüchlich? Kann es bei unsachgemäßer Benutzung leicht zu Widersprüchen kommen? Wie gut wird eine Terminologische Kontrolle unterstützt?
Erweiterbarkeit	Wie gut können neue Deskriptoren aufgenommen bzw. vorhandene spezialisiert werden? Wie viele Komponenten können mit dem Ordnungssystem noch praktikabel dokumentiert werden?
Benutzbarkeit	Ist der Grundgedanke des Ordnungssystems nachvollziehbar? Sind die Intensionen der Deskriptoren jeweils verständlich? Ist das Ordnungssystem abhängig von der Benutzung einer bestimmten natürlichen Sprache?
Wirtschaftlichkeit	In welchem Verhältnis stehen die Implementierungs-, Archivierungs- sowie Recherchekosten zu den eingesparten Opportunitätskosten?

Tabelle 3: Bewertungsaspekte zur Beurteilung von Ordnungssystemen

und die Klassifikation damit nicht mehr praktikabel. Dieses Problem ist bei der Mono- und Polyhierarchischen Klassifikation weniger stark ausgeprägt, da hierbei die Deskriptoren in hierarchischen Beziehungen angeordnet werden können. Mit Registern ist eine ähnliche Präzision wie mit der Klassifikation möglich. Schlagwortvergabe sowie Freitextsuche bieten nur dann eine gute Präzision, wenn eine normierte oder standardisierte Fachsprache vorhanden ist, andernfalls entstehen erhebliche terminologische Probleme. Die Facettenklassifikation bietet durch die Kombinationsfähigkeit der Deskriptoren verschiedener Facetten im Vergleich zur Klassifikation bei gleicher Anzahl Deskriptoren eine erhebliche größere Indexierungsgenauigkeit. Die Indexierungsgenauigkeit kann bei einer Begriffskombination aufgrund der höheren Kombinationsfähigkeit der Deskriptoren weiter gesteigert werden. Das Vektorraummodell sowie Komponentenkataloge ermöglichen aufgrund der Beschreibung von Komponenten mit metrischen Merkmalen ebenfalls eine hohe Indexierungsgenauigkeit. Eine Suche nach ähnlichen Komponenten wird vom Vektorraummodell und einem Komponentenkatalog sehr gut unterstützt, von der Facettenklassifikation und von der Begriffskombination gut und von den restlichen Ordnungssystemen gar nicht.

Konsistenz

Mono- und Polyhierarchische Klassifikation, hierarchische Register sowie Komponentenkataloge unterstützen aufgrund ihrer hierarchischen Struktur eine Terminologische Kontrolle. Bei einer Facettenklassifikation wird eine Terminologische Kontrolle durch das Gruppieren mehrerer Deskriptoren zu einer Facette gefördert. Die anderen Ordnungssysteme kennen keine Konzepte zur Terminologischen Kontrolle, so dass hier zusätzliche Maßnahmen eingeleitet werden müssen. Dies ist allerdings bei der Freitextsuche sowie der Schlagwortvergabe aufgrund ihrer Konzeption nicht möglich. Liegt für ein Sachgebiet noch keine eindeutige Systematik vor, was bei einer Dokumentation von Komponenten die Regel sein dürfte, führt dies bei der Klassifikation ohne Überlagerung, der Monohierarchischen Klassifikation sowie Komponentenkatalogen zu Problemen, da diese Ordnungssysteme eine eindeutige Systematisierung voraussetzen.

Erweiterbarkeit

Die Erweiterbarkeit der Klassifikation ohne und mit Überlagerung, der Mono- sowie Polyhierarchischen Klassifikation und der systematischen Register ist bei einer großen Deskriptorenzahl problematisch, da zwar formal leicht ein neuer Deskriptor definiert werden kann, allerdings seine Intension eindeutig von schon vorhandenen Deskriptoren abgegrenzt werden muss. Des Weiteren kann bei einer Monohierarchischen Klassifikation die hierarchische Struktur der Deskriptoren nur schwer, bei einer Polyhierarchischen Klassifikation weniger schwer verändert werden. Bei einer Facettenklassifikation ist es relativ leicht möglich neue Facetten aufzunehmen, wobei allerdings bei steigender Facettenzahl darauf geachtet werden muss, die Unabhängigkeit der Facetten zu bewahren. Eine Freitextsuche bzw. eine Schlagwortvergabe kann leicht erweitert werden, solange die Präzision des Ordnungssystems noch angemessen bleibt. Die Aufnahme eines neuen Beschreibungsmerkmals erfordert bei dem Vektorraummodell sowie dem Komponentenkatalog einen hohen Aufwand, da sämtliche schon archivierte Komponenten hinsichtlich des neuen Merkmals bewertet werden müssen. Zudem muss bei dem Vektorraummodell das Proximitätsmaß bei der Aufnahme weiterer Merkmale neu definiert werden.

Benutzbarkeit

Die Klassifikation ohne und mit Überlagerung, die Mono- und Polyhierarchische Klassifikation sowie Register sind klare Ordnungssysteme, die gut nachvollzogen werden können. Indes muss eingeräumt werden, dass tiefe Hierarchien mit vielen Deskriptoren schwer verständlich sind, insb. wenn diese dem Benutzer nicht vertraut sind. Ebenso erfordert die Facettenklassifikation eine gewisse Einarbeitungszeit. Bei normierten oder standardisierten Sprachen erreichen Freitextsuche und Schlagwortvergabe eine gute Benutzbarkeit. Allerdings sind bei Komponentenbeschreibungen diese Voraussetzungen in der Regel nicht erfüllt, so dass dann eine Recherche bei diesen Ordnungssystemen nur wenig komfortabel und nicht systematisch durchgeführt werden kann. Das Vektorraummodell ist leicht benutzbar, allerdings kann es zu Schwierigkeiten kommen, die Arbeitsweise des Proximitätsmaßes nachzuvollziehen. Ebenso leicht benutzbar sind Komponentenkataloge.

Die Sprachabhängigkeit der Freitextsuche sowie der Schlagwortvergabe ist sehr hoch und die der Klassifikation ohne und mit Überlagerung hoch. Die anderen Ordnungssysteme sind wenig sprachabhängig.

Wirtschaftlichkeit

Die Implementierungskosten für ein Ordnungssystem, bei dem gebunden indexiert wird, sind erheblich höher als die Implementierungskosten für ein Ordnungssystem, bei dem ungebunden indexiert wird. Demnach sind die Implementierungskosten der Freitextsuche und der Schlagwortvergabe niedrig, ebenso die Implementierungskosten eines Registers, bei dem ungebunden indexiert wird. Die anderen Ordnungssysteme erfordern, bevor das Ordnungssystem genutzt werden kann, eine verbindliche Definition einer Menge von Deskriptoren, die den Komponenten indexiert werden können. Folglich sind die Kosten bei diesen Systemen deutlich höher. Besonders hoch sind die Implementierungskosten für einen Komponentenkatalog, da die Erstellung eines Kataloges umfangreiche Untersuchungen zur Verfügbarkeit von Komponenten und ihrer Strukturierbarkeit voraussetzt.

Die Archivierungskosten der Freitextsuche tendieren gegen Null, die der Schlagwortvergabe liegen ebenfalls sehr niedrig. Bei der Klassifikation mit oder ohne Überlagerung, der Mono- sowie Polyhierarchischen Klassifikation sowie den Registern sind die Archivierungskosten als moderat einzustufen. Facettenklassifikation und Begriffskombination erfordern eine Indexierung vieler Deskriptoren und sind dementsprechend aufwendiger. Aufwendig sind auch das Vektorraummodell sowie der Komponentenkatalog, da eine Komponente jeweils hinsichtlich vieler Merkmale erschlossen werden muss.

Die Benutzung eines Ordnungssystems verursacht hohe Recherchekosten, wenn die Auswahl geeigneter Deskriptoren zur Formulierung der Suchanfrage nicht ausreichend unterstützt wird. Dement-

sprechend sind die Recherchekosten der Freitextsuche bzw. der Schlagwortvergabe im Vergleich zu den anderen Ordnungssystemen besonders hoch.

Eine Komponentendokumentation ist wirtschaftlich dann sinnvoll, wenn die Summe der Kosten zur Implementierung, Archivierung sowie Recherche geringer ausfällt als die Summe der Opportunitätskosten der einzelnen Recherchen. Da die Archivierungs- sowie die Opportunitätskosten von der Anzahl der archivierten Komponenten, der Häufigkeit gestellter Recherchen, der Anzahl der Verwendungen einer archivierten und aufgefundenen Komponente sowie weiteren Faktoren abhängig sind, ist an dieser Stelle keine allgemeine Aussage bezüglich der Wirtschaftlichkeit eines speziellen Ordnungssystems möglich.

6 Schlussfolgerungen und Forschungsbedarf

Vor dem Hintergrund der vorgenommenen Bewertung der Ordnungssysteme ergeben sich weitere Fragestellungen und Probleme, die im Folgenden angesprochen werden.

- Erweiterung der Bewertungskriterien und Explikation von Randbedingungen: Die durchgeführte Untersuchung hat gezeigt, dass Ordnungssysteme nicht einwertig beurteilt werden können, sondern dass Trade-Offs bestehen: Bspw. hat die Steigerung der Präzision eines Ordnungssystems Auswirkung auf seine Wirtschaftlichkeit. Dies bedeutet, dass die Ordnungssysteme nicht auf einer Präferenzskala eindeutig angeordnet werden können, ohne ihren Verwendungskontext näher zu spezifizieren. Folglich wäre es zum einen angebracht, die vorgestellten Kriterien zu verfeinern und zu erweitern. Zum anderen sollten die Randbedingungen expliziert werden, die eine Bewertung beeinflussen.
- Konkretisierung der Ordnungssysteme: In dieser Arbeit wurden Ordnungssysteme ausschließlich aus formaler Sicht vorgestellt. Um die vorgestellten Ordnungssysteme verwenden zu können, müssen diese inhaltlich konkretisiert werden, indem Vorschläge für konkrete Deskriptoren erarbeitet werden (Ansätze für die inhaltliche Konkretisierung finden sich bspw. bei BOOCH 1987, S. 36–46; FHG 1998; HAGEMEYER, ROLLES 1997; KRAMPE 1999, S. 77-82; REMME 1997, S. 92; SAMETINGER 1997, S. 117–128; SCHEER, ZIMMERMANN, KLABUNDE 1996).
- Methodische Unterstützung: Die notwendigen Aktivitäten zur Implementierung eines Ordnungssystems und zur Archivierung sowie Recherche einer Komponente können weiter im Detail untersucht werden. Es ist zu klären, wie Komponentendokumentationen erstellt werden können und wie und wann Komponentendokumentationen im Softwareentwicklungsprozess zum Einsatz kommen.
- Empirische Untersuchungen: In diesem Beitrag wurden die Ordnungssysteme ausschließlich deduktiv bewertet. Die dabei gewonnenen Aussagen sollten durch empirische Untersuchungen ergänzt werden.
- Bewertung weiterer Ordnungssysteme: In der Literatur werden weitere Ordnungssysteme diskutiert (CHEN, HENNICKER, JARKE 1992; DAMIANI, FUGINI, BELLETTINE 1999; SOMMERVILLE, WOOD 1986). Auch ist es denkbar, Ordnungssysteme, die in anderen Wissenschaftsdisziplinen entwickelt worden sind, auf die Dokumentation von Komponenten zu übertragen. Hierbei könnten bspw. die aus den Ingenieurwissenschaften bekannten Sachmerkmalsleisten (MEINL 1990) auf die Dokumentation von Komponenten transferiert werden.

Zusammenfassend muss festgestellt werden, dass die Frage, welches Ordnungssystem zum Aufbau einer Komponentendokumentation vorzuziehen ist, nicht abschließend beantwortet werden kann. Indes wurden in dieser Arbeit Kriterien zur Beurteilung von Ordnungssystemen für Komponentendokumentationen herausgearbeitet, die bei der Beantwortung der aufgeworfenen Frage eine Orientierung geben und

die eine Grundlage für weitere systematische Untersuchungen bilden können. Ferner ist darauf hinzuweisen, dass sich die vorgestellten Ordnungssysteme nicht gegenseitig ausschließen, sondern komplementär angewendet werden können. Bspw. kann eine Hierarchische Klassifikation mit einer Schlagwortvergabe kombiniert werden.

Literatur

- ARNOLD, S. P.; STEPOWAY, S. L.: The Reuse System: Cataloging and Retrieval of Reusable Software, S. 138–141, in: TRACZ 1988, 1988.
- BACKHAUS, K.; ERICHSON, B.; PLINKE, W.; WEIBER, R.: Multivariate Analysemethoden – Eine anwendungsorientierte Einführung, Berlin et al.: Springer, 8. Aufl., 1996.
- BOOCH, G.: Software Components with Ada, Menlo Park, CA et al.: Benjamin/Cummings, 1987.
- BUDER, M.; REHFELD, W.; SEEGER, T.; STRAUCH, D. (Hrsg.): Grundlagen der praktischen Information und Dokumentation, München et al.: K. G. Saur, 1997.
- BURTON, B. A.; ARAGON, R. W.; BAILEY, S. A.; KOEHLER, K. D.; MAYES, L. A.: The Reusable Software Library, S. 129–137, in: TRACZ 1988, 1988.
- CHEN, P. S.; HENNICKER, R.; JARKE, M.: On the Retrieval of Reusable Software Components, Aachener Informatik-Berichte 92–42, RWTH Aachen – Fachgruppe Informatik, Aachen, 1992.
- CONVENT, B.; WERNECKE, W.: Bausteinverwaltung und Suchunterstützung – Basis für die Software-Wiederverwendung, in: HMD, Bd. 31, Nr. 180, S. 59–70, 1994.
- DAMIANI, E.; FUGINI, M. G.; BELLETTINE, C.: A Hierarchy-Aware Approach to Faceted Classification of Object-Oriented Components, in: ACM Transactions on Software Engineering and Methodology, Bd. 8, Nr. 3, S. 215–262, 1999.
- DAVIES, R. (Hrsg.): Intelligent Information Systems: Progress and Prospects, New York et al.: Ellis Horwood, 1986.
- FELLNER, K. J.; RAUTENSTRAUCH, C.; TUROWSKI, K.: Fachkomponenten zur Gestaltung betrieblicher Anwendungssysteme, in: Information Management & Consulting, Bd. 14, Nr. 2, S. 25–34, 1999.
- FERBER, R.: Dokumentsuche und Dokumenterschließung, in: RECHENBERG, POMBERGER 1999, 2. Aufl., 1999.
- FHG (Hrsg.): Das GiPP Typisierungsmodell für Geschäftsprozesse, 1998, URL <http://www.ipa.fhg.de/100/projekte/gipp-ak1/Inhalte/Thesenpapiere/Ergeb.inhalt.html> (Abruf: 2000-07-03).
- FRAKES, W. B.: Panel Session: Information Retrieval and Software Reuse, in: Proceedings of the twelfth annual international ACM SIGIR conference on Research and development in Information Retrieval, S. 251–256, Cambridge, MA USA, 1989.
- FRAKES, W. B.; NEJMEH, B. A.: An Information System for Software Reuse, S. 142–151, in: TRACZ 1988, 1988.
- FRANK, U.: Componentware – Software-technische Konzepte und Perspektiven für die Gestaltung betrieblicher Informationssysteme, in: Information Management & Consulting, Bd. 14, Nr. 2, S. 11–18, 1999.
- GAUS, W.: Dokumentations- und Ordnungslehre – Theorie und Praxis des Information Retrieval, Berlin et al.: Springer, 2. Aufl., 1995.
- HAGEMEYER, J.; ROLLES, R.: Aus Informationsmodellen weltweit verfügbares Wissen machen: Ein Modell-Thesaurus zur Erhöhung von Verständlichkeit und Wiederverwendbarkeit, in: IM Information Management & Consulting, Sonderausgabe Business Engineering, S. 56–59, 1997.

- HARS, A.; HEIB, R.; KRUSE, C.; MICHELY, J.; SCHEER, A.-W.: Approach to Classification for Information Engineering - Methodology and Tool Specification, Heft 92, Institut für Wirtschaftsinformatik im Institut für empirische Wirtschaftsforschung an der Universität des Saarlandes, Saarbrücken, 1992.
- HENNINGER, S.: An Evolutionary Approach to Constructing Effective Software Reuse Repositories, in: ACM Transactions on Software Engineering and Methodology, Bd. 6, Nr. 2, S. 111–140, 1997.
- HOFFMANN, W.; SCHEER, A.-W.; BACKER, R.: Konzeption eines Ereignisklassifikationssystems in Prozeßketten, Heft 95, Institut für Wirtschaftsinformatik im Institut für empirische Wirtschaftsforschung an der Universität des Saarlandes, Saarbrücken, 1992.
- KAIN, J. B.: Components: The basics: Enabling an application or system to be the sum of its parts, in: Object Magazine, Bd. 6, Nr. 2, S. 64–69, 1996.
- KRAMPE, D.: Wiederverwendung von Informationssystementwürfen – Ein fallbasiertes werkzeuggestütztes Ablaufmodell, Deutscher Universitäts-Verlag, 1999.
- LANG, K.-P.: Variantenkonstruktion betriebswirtschaftlicher Anwendungssoftware – Teil 1: Methode der semantischen Komposition, Arbeitsbericht 98/02, FG Wirtschaftsinformatik I der Technischen Universität Darmstadt, Darmstadt, 1998.
- LUNG, C.-H.; URBAN, J. E.: An Approach to the Classification of Domain Models in Support of Analogical Reuse, in: Proceedings of the 17th international conference on software engineering on Symposium on software reusability, S. 169–178, Seattle, WA USA, 1995.
- MAIDEN, N. A.; SUTCLIFFE, A. G.: Exploiting Reusable Specifications Through Analogy, in: Communications for the ACM, Bd. 35, Nr. 4, S. 55–64, 1992.
- MANECKE, H.-J.: Klassifikation, S. 141–159, in: BUDER et al. 1997, 1997.
- MEIER, J.; RÖPERT, A.: Systematische Software-Wiederverwendung – Konzepte und Verfahren bei Siemens Nixdorf Informationssysteme AG, in: HMD, Bd. 31, Nr. 180, S. 46–58, 1994.
- MEINL, F.: Sachmerkmale – Schlüssel zur technischen Gestaltung, Beschreibung und Information, Ehningen bei Böblingen: expert, 1990.
- MERTENS, P.: Betriebliche Dokumentation und Information, Meisenheim am Glan: Anton Hain, 1965.
- ONUEGBE, E. O.: Software Classification, S. 161–167, in: TRACZ 1988, 1988.
- ORTNER, E.; LANG, K.-P.; KALKMANN, J.: Anwendungssystementwicklung mit Komponenten, in: Information Management & Consulting, Bd. 14, Nr. 2, S. 35–45, 1999.
- Ein Szenario für die Anwendungssystementwicklung mit Komponenten, Arbeitsbericht, FG Wirtschaftsinformatik I der Technischen Universität Darmstadt, Darmstadt, o. J.
- OSTERTAG, E.; PRIETO-DÍAZ, R.; BRAUN, C.: Computing Similarity in a Reuse Library System: An AI-Based Approach, in: ACM Transactions on Software Engineering and Methodology, Bd. 1, Nr. 3, S. 205–228, 1992.
- PODGURSKI, A.; PIERCE, L.: Retrieving Reusable Software by Sampling Behavior, in: ACM Transactions on Software Engineering and Methodology, Bd. 2, Nr. 3, S. 286–303, 1993.
- PRIETO-DÍAZ, R.: Implementing Faceted Classification for Software Reuse, in: Communications of the ACM, Bd. 34, Nr. 5, S. 88–97, 1991.
- PRIETO-DÍAZ, R.; FREEMAN, P.: Classifying Software for Reusability, in: IEEE Software, S. 6–16, 1987.
- PRIETO-DÍAZ, R.; JONES, G. A.: Breathing New Life into Old Software, S. 152–160, in: TRACZ 1988, 1988.
- RECHENBERG, P.; POMBERGER, G. (Hrsg.): Informatik-Handbuch, München, Wien: Carl Hanser, 2. Aufl., 1999.

- REMME, M.: Konstruktion von Geschäftsprozessen – Ein modellgestützter Ansatz durch Montage generischer Prozeßpartikel, Wiesbaden: Gabler, 1997.
- ROTH, K.: Konstruieren mit Konstruktionskatalogen – Band 2: Kataloge, Berlin et al.: Springer, 2. Aufl., 1994.
- ROTH, K.; FRANKE, H.-J.; SIMONEK, R.: Aufbau und Verwendung von Katalogen für das methodische Konstruieren, in: Konstruktion, Bd. 24, S. 449–458, 1972.
- SALTON, G.; MCGILL, M. J.: Introduction to Modern Information Retrieval, Auckland et al.: McGraw-Hill, 1983.
- SAMETINGER, J.: Software Engineering with Reusable Components, Berlin et al.: Springer, 1997.
- SCHEER, A.-W.; ZIMMERMANN, V.; KLABUNDE, S.: Referenzmodellbibliothek: Engineering von Geschäftsprozessen auf Basis standardisierter Teilprozesse, in: Tagungsband zum 1. GiPP-Fachforum, 24.6.1996, FIZ d. BMW AG, 1996.
- SEIFFERT, H.: Einführung in die Wissenschaftstheorie – Sprachanalyse – Deduktion – Induktion in der Natur- und Sozialwissenschaft, Bd. 1, München: Beck, 11. Aufl., 1991.
- SOMMERVILLE, I.; WOOD, M.: A software components catalogue, S. 13–32, in: DAVIES 1986, 1986.
- TRACZ, W. (Hrsg.): Tutorial: Software Reuse: Emerging Technology, Washington, DC: IEEE Computing Society, 1988.

Anhang: Isolierte Bewertung der Ordnungssysteme

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges formal sichergestellt werden.
Präzision	Das Ordnungssystem besitzt eine hohe Trennschärfe, da jede Komponente in genau eine Klasse eingeteilt wird. Die Eindeutigkeit der Klassifikation kann zu Problemen führen, falls mehrere Klassen für eine Komponente sachlich treffend sind. Eine hohe Indexierungsgenauigkeit erfordert viele Deskriptoren.
Konsistenz	Das Auffinden ähnlicher Komponenten wird nicht unterstützt. Die Konsistenz des Ordnungssystems wird problematisch, falls eine Komponente aus sachlichen Gründen nicht in genau eine Klasse eingeordnet werden kann. Eine ergänzende Terminologische Kontrolle ist notwendig.
Erweiterbarkeit	Die Erweiterbarkeit ist problematisch, weil mit steigender Deskriptorenzahl die Trennschärfe der Deskriptoren schwierig sicherzustellen ist.
Benutzbarkeit	Das Ordnungssystem ist klar, verständlich und einfach: Jede Komponente ist eindeutig in eine Klasse eingeordnet. Bei großer Deskriptorenzahl geht die Übersicht über das Ordnungssystem verloren.
Wirtschaftlichkeit	Die Sprachabhängigkeit des Ordnungssystems ist sehr hoch. Die Implementierungskosten sind moderat, da zwar gebunden indexiert wird, allerdings nur ein relativ einfaches Ordnungssystem erstellt wird. Die Archivierungskosten sind gering, da jeweils nur eine passende Klasse für eine Komponente gefunden werden muss. Die Recherchekosten sind relativ gering.

Tabelle 4: Bewertung der Klassifikation ohne Überlagerung

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges formal sichergestellt werden.
Präzision	Eine Komponente kann durch mehrere Deskriptoren sachlich oft angemessener indexiert werden als dies bei einer Klassifikation ohne Überlagerung möglich wäre. Falls einer Komponente zu viele Deskriptoren indexiert werden, geht die Trennschärfe der Deskriptoren verloren.
Konsistenz	Eine hohe Indexierungsgenauigkeit erfordert viele Deskriptoren. Das Auffinden ähnlicher Komponenten wird nicht unterstützt.
Erweiterbarkeit	Eine ergänzende Terminologische Kontrolle ist notwendig. Die Erweiterbarkeit ist problematisch, weil mit steigender Deskriptorenzahl die Trennschärfe der Deskriptoren schwierig sicherzustellen ist.
Benutzbarkeit	Das Ordnungssystem ist klar, verständlich und einfach: Jede Komponente wird in wenige Klassen eingeordnet. Bei großer Deskriptorenzahl geht die Übersicht über das Ordnungssystem verloren. Die Sprachabhängigkeit des Ordnungssystems ist sehr hoch.
Wirtschaftlichkeit	Die Implementierungskosten sind moderat, da zwar gebunden indexiert wird, allerdings nur ein relativ einfaches Ordnungssystem erstellt wird. Die Archivierungskosten sind gering, da jeweils nur wenige geeignete Deskriptoren indexiert werden müssen. Die Recherchekosten sind relativ gering.

Tabelle 5: Bewertung der Klassifikation mit Überlagerung

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges formal sichergestellt werden.
Präzision	Die Präzision des Ordnungssystems kann durch die Einführung verfeinerter Deskriptoren erhöht werden. Das Auffinden ähnlicher Komponenten wird nicht unterstützt.
Konsistenz	Die hierarchische Anordnung der Deskriptoren bereitet Schwierigkeiten, wenn Sachgebiete nicht eindeutig systematisiert werden können oder wenn die Systematisierung sich im Zeitablauf verändert. Durch die Definition über- und untergeordneter Deskriptoren wird eine Terminologische Kontrolle unterstützt.
Erweiterbarkeit	Grundlegende Veränderungen in der Hierarchisierung können bei tiefen Hierarchien Probleme bereiten. Sehr große Sachgebiete können gut erschlossen werden.
Benutzbarkeit	Flache Hierarchien sind klar und verständlich. Große Hierarchien sind schwer zu durchschauen und zu verstehen. Die Sprachabhängigkeit des Ordnungssystems ist gering.
Wirtschaftlichkeit	Die Implementierungskosten sind bei tiefen Hierarchien hoch, da eine saubere Abgrenzung der Deskriptoren erreicht werden muss und gebunden indexiert wird. Die Archivierungskosten sind gering, wenn der Archivar mit dem Ordnungssystem vertraut ist. Die Recherchekosten sind relativ gering.

Tabelle 6: Bewertung der Monohierarchischen Klassifikation

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges formal sichergestellt werden.
Präzision	Die Präzision des Ordnungssystems kann durch die Einführung verfeinerter Deskriptoren erhöht werden.
Konsistenz	Das Auffinden ähnlicher Komponenten wird nicht unterstützt. Nicht eindeutig systematisierte Sachgebiete können gut abgebildet werden. Durch die Definition über- und untergeordneter Deskriptoren wird eine Terminologische Kontrolle unterstützt.
Erweiterbarkeit	Veränderungen an der Hierarchie können leichter durchgeführt werden als bei einer Monohierarchischen Klassifikation, obgleich sie dennoch Probleme bereiten können.
Benutzbarkeit	Sehr große Sachgebiete können gut erschlossen werden. Große Polyhierarchien sind schwer zu durchschauen und zu verstehen. Die Sprachabhängigkeit des Ordnungssystems ist gering.
Wirtschaftlichkeit	Die Implementierungskosten sind bei tiefen Hierarchien hoch, da eine saubere Abgrenzung der Deskriptoren erreicht werden muss und gebunden indexiert wird. Die Archivierungskosten sind gering, wenn der Archivar mit dem Ordnungssystem vertraut ist.

Tabelle 7: Bewertung der Polyhierarchischen Klassifikation

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges formal sichergestellt werden.
Präzision	Die Präzision des Ordnungssystems kann durch die Einführung verfeinerter Deskriptoren erhöht werden. Systematische und nicht-systematische Ordnungen werden unterstützt. Es sind Ordnungen hinsichtlich verschiedener Aspekte möglich. Das Auffinden ähnlicher Komponenten wird nicht unterstützt.
Konsistenz	Mehrdeutig systematisierte Sachgebiete können gut abgebildet werden. Eine ergänzende Terminologische Kontrolle ist notwendig.
Erweiterbarkeit	Die Erweiterung hierarchischer Register ist im Gegensatz zu nicht-hierarchischen Registern problematisch.
Benutzbarkeit	Register sind klar und verständlich aufgebaut. Register sind gut geeignet, wenn das Sachgebiet nicht systematisch gegliedert werden kann und/oder wenn verschiedene, klar trennbare Fragestellungen auftreten, nach denen Komponenten gesucht werden können. Die Sprachabhängigkeit des Ordnungssystems ist groß.
Wirtschaftlichkeit	Wenn ungebunden indexiert wird, sind die Implementierungskosten gering. Die Archivierungskosten sind gering, da jede Komponente jeweils nur in die entsprechenden Register aufgenommen werden muss. Die Recherchekosten sind relativ gering.

Tabelle 8: Bewertung des Registers

Kriterium	Bewertung
Vollständigkeit	Die Vollständigkeit des Ordnungssystems kann durch einen Deskriptor sonstiges innerhalb jeder Facette formal sichergestellt werden.
Präzision	Im Vergleich zur Klassifikation ist durch die Kombinationsfähigkeit der Deskriptoren eine wesentlich geringere Anzahl an Deskriptoren notwendig, um eine gleiche Indexierungsgenauigkeit zu erreichen. Das Klassifizieren in Facetten ermöglicht implizit eine Suche nach ähnlichen Komponenten.
Konsistenz	Falls eine Komponente hinsichtlich einer Facette nicht sinnvoll klassifiziert werden kann, kommt es zu Konsistenzproblemen. Das Gruppieren der Deskriptoren zu Facetten fördert die Terminologische Kontrolle.
Erweiterbarkeit	Das Aufrechterhalten einer hohen Kombinationsfähigkeit der Deskriptoren kann bei steigender Facettenzahl erhebliche Probleme verursachen.
Benutzbarkeit	Eine Facettenklassifikation ist gut geeignet für Sachgebiete mit klar abgegrenzten Teilgebieten. Es ist eine relativ lange Einarbeitung notwendig, um eine Facettenklassifikation zu verstehen. Die Sprachabhängigkeit des Ordnungssystems ist gering.
Wirtschaftlichkeit	Die Implementierungskosten sind hoch, da hohe Anforderungen an die Bestimmung der Deskriptoren gestellt werden. Archivierungskosten sind moderat, da jede Komponente systematisch hinsichtlich jeder Facette klassifiziert werden muss. Die Recherchekosten sind relativ gering.

Tabelle 9: Bewertung der Facettenklassifikation

Kriterium	Bewertung
Vollständigkeit	Das Kriterium ist problematisch, falls keine textuelle Komponentenbeschreibung vorliegt.
Präzision	Es kann nicht zwischen relevanten und weniger relevanten Deskriptoren bei der Recherche unterschieden werden. Trotzdem kann eine gute Präzision bei standardisierten bzw. normierten Fachsprachen erreicht werden. Ähnliche Komponenten können nur unzureichend aufgefunden werden.
Konsistenz	Das Ordnungssystem bietet keine Unterstützung zur Konsistenzsicherung, so dass erhebliche Probleme durch Homonyme und Teil- bzw. Vollsynonyme entstehen können, die nicht behoben werden können.
Erweiterbarkeit	Die Erweiterbarkeit ist solange gegeben, bis aufgrund fehlender Präzision das Ordnungssystem unbrauchbar wird.
Benutzbarkeit	Eine Recherche ist unkomfortabel und wenig zielgerichtet möglich, da keine Terminologische Kontrolle erfolgt und der Recherchleur auf seine Phantasie bei der Auswahl geeigneter Suchwörter angewiesen ist. Die Sprachabhängigkeit des Ordnungssystems ist sehr hoch.
Wirtschaftlichkeit	Die Implementierungskosten sind gering, da kein Deskriptorensystem aufgebaut werden muss. Die Archivierungskosten sind quasi gleich Null, falls die Komponentenbeschreibungen schon in digitaler Form vorliegen. Die Recherchekosten sind sehr hoch, da häufig zu große, zu kleine oder nicht relevante Treffermengen ermittelt werden.

Tabelle 10: Bewertung der Freitextsuche

Kriterium	Bewertung
Vollständigkeit	Das Kriterium ist unproblematisch.
Präzision	Eine sehr detaillierte Erschließung der Komponenten ist auch in großen Dokumentationen möglich. Komponenten, denen ein gleicher Deskriptor indexiert wurde, können als ähnlich interpretiert werden, bzw. ähnliche Komponenten können durch die Suche nach einem gemeinsamen Deskriptor gefunden werden.
Konsistenz	Eine ergänzende Terminologische Kontrolle ist notwendig.
Erweiterbarkeit	Bei der Aufnahme neuer Deskriptoren muss strenggenommen jede schon archivierte Komponente überprüft werden, ob ihr der neue Deskriptor indexiert werden muss.
Benutzbarkeit	Wenn die Deskriptoren einem Rechercheur geläufig sind, ist das Ordnungssystem klar und verständlich. Andernfalls kann es zu Problemen führen. Die Sprachabhängigkeit des Ordnungssystems ist gering.
Wirtschaftlichkeit	Die Implementierungskosten sind hoch, da hohe Anforderungen an die Bestimmung der Deskriptoren zu stellen sind. Die Archivierungskosten sind hoch, da strenggenommen eine Indexierung erst dann abgeschlossen ist, wenn alle Deskriptoren hinsichtlich ihrer Eignung überprüft worden sind. Die Recherchekosten sind relativ gering.

Tabelle 11: Bewertung der Begriffskombination

Kriterium	Bewertung
Vollständigkeit	Das Kriterium ist unproblematisch.
Präzision	Theoretisch ist eine hohe Präzision bei der Erschließung der Komponenten möglich. Diese kann allerdings durch das ungebundene Indexieren und die damit verbundene fehlende Terminologische Kontrolle erheblich verschlechtert werden. Ähnliche Komponenten können gefunden werden, falls ein Deskriptor übereinstimmt.
Konsistenz	Das Ordnungssystem bietet keine Unterstützung zur Konsistenzsicherung. Es können erhebliche Probleme durch Homonyme und Teil- bzw. Vollsynonyme entstehen, die nicht durch eine Terminologische Kontrolle behoben werden können.
Erweiterbarkeit	Die Erweiterbarkeit ist solange gegeben, bis aufgrund fehlender Präzision das Ordnungssystem unbrauchbar wird.
Benutzbarkeit	Das Ordnungssystem ist klar und verständlich. Die Sprachabhängigkeit des Ordnungssystems ist sehr hoch.
Wirtschaftlichkeit	Die Implementierungskosten sind gering, da kein Deskriptorensystem aufgebaut werden muss. Die Archivierungskosten sind gering, da die Vergabe geeigneter Schlagwörter relativ leicht möglich ist. Die Recherchekosten sind hoch, da häufig zu große, zu kleine oder nicht relevante Treffermengen ermittelt werden.

Tabelle 12: Bewertung der Schlagwortvergabe

Kriterium	Bewertung
Vollständigkeit	Das Kriterium ist unproblematisch.
Präzision	Es kann eine sehr hohe Präzision erreicht werden, da Komponenten hinsichtlich unterschiedlicher Dimensionen qualifiziert werden. Über die Definition eines Proximitätsmaßes ist eine gute Unterstützung für die Suche nach ähnlichen Komponenten möglich.
Konsistenz	Die Bestimmung eines geeigneten Proximitätsmaßes hat erhebliche Auswirkungen auf die Konsistenz. Eine ergänzende Terminologische Kontrolle ist notwendig.
Erweiterbarkeit	Die Aufnahme weiterer Bewertungsdimensionen ist mit erheblichen Aufwand verbunden, da sämtliche bereits archivierte Komponenten hinsichtlich des neuen Merkmals bewertet werden müssen. Zudem muss das Proximitätsmaß geeignet angepasst werden.
Benutzbarkeit	Das Ordnungssystem ist leicht benutzbar, allerdings kann es zu Schwierigkeiten kommen, die Arbeitsweise des Proximitätsmaßes nachzuvollziehen.
Wirtschaftlichkeit	Die Implementierungskosten sind sehr hoch, da geeignete Beschreibungsdimensionen ermittelt werden müssen und ein angemessenes Proximitätsmaß gefunden werden muss. Die Archivierungskosten sind hoch, da eine Komponente hinsichtlich aller Beschreibungsdimensionen bewertet werden muss. Die Recherchekosten sind relativ gering.

Tabelle 13: Bewertung des Vektorraummodells

Kriterium	Bewertungsaspekte
Vollständigkeit	Das Kriterium ist unproblematisch.
Präzision	Es kann eine sehr hohe Präzision erreicht werden, da Komponenten hinsichtlich unterschiedlicher Dimensionen qualifiziert werden. Ähnliche Komponenten können sehr gut aufgefunden werden, da diese im selben Katalog verzeichnet sind.
Konsistenz	Das Kriterium ist unproblematisch.
Erweiterbarkeit	Die Aufnahme weniger neuer Komponenten ist unproblematisch. Werden allerdings viele Komponenten neu aufgenommen, kann dies umfangreiche Änderungen im Gliederungs- und Zugriffsteil erfordern.
Benutzbarkeit	Komponentenkataloge sind klar, verständlich und gut benutzbar.
Wirtschaftlichkeit	Die Implementierungskosten sind sehr hoch, da die Erstellung des Gliederungsteils und die Wahl geeigneter Zugriffsmerkmale einen erheblichen Aufwand erfordern. Die Archivierungskosten sind hoch, da eine Komponente hinsichtlich aller Beschreibungsdimensionen bewertet werden muss. Die Recherchekosten sind relativ gering.

Tabelle 14: Bewertung des Komponentenkatalogs