

Materialien

Heft 33

Christof Wolters

Wie muß man seine Daten
formulieren bzw. strukturieren,
damit ein Computer
etwas Vernünftiges damit anfangen kann ?

Mit einem Glossar von Carlos Saro

aus dem
Institut für
Museums-
kunde

Staatliche Museen
zu Berlin –
Preußischer
Kulturbesitz

**Staatliche Museen Preußischer Kulturbesitz
Institut für Museumskunde Berlin**

Heft 33

Christof Wolters

**Wie muß man seine Daten
formulieren bzw. strukturieren,
damit ein Computer
etwas Vernünftiges damit anfangen kann ?**

Mit einem Glossar von Carlos Saro

ISSN 0931-7961 Heft 33

Berlin 1991

Das Institut für Museumskunde stellt in regelmäßigen Abständen Materialien aus der laufenden Arbeit interessierten Fachleuten zur Verfügung. Diese Hefte gelangen nicht in den Buchhandel und werden nur auf begründete Anfrage abgegeben. Eine Liste der bisher erschienenen "Materialien-Hefte" (ISSN 0931-7961) befindet sich am Ende dieses Heftes.

Vorbemerkung

Die Dokumentationsabteilung im Institut für Museumskunde der Staatlichen Museen Preussischer Kulturbesitz (IfM) hat die Aufgabe, bundesweit Museen und museale Einrichtungen bei der Einführung der Informationstechnik mit Rat und Tat zu unterstützen. Hierbei arbeitet sie mit dem Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB) zusammen.

Diese Aktivitäten liegen im Bereich einer professionell durchgeführten Rationalisierung; Computerisierung ist nicht Selbstzweck, sondern eine Möglichkeit, ohnehin durchzuführende Arbeiten in puncto Zeit und Kosten effizienter zu gestalten bzw. Dinge durchzuführen, für die man beim Einsatz konventioneller Methoden weder die Zeit noch das Geld hätte.

Mit der Vermittlung der hierfür notwendigen Kenntnisse und Fertigkeiten ist ein kleines Institut wie das IfM wegen des rapide ansteigenden Beratungsbedarfs der Museen heute noch weit überfordert. Wir konzentrieren daher unsere Aktivitäten auf eine Zusammenarbeit mit den für die Museumsbetreuung zuständigen Einrichtungen in den Bundesländern. Wir haben die Hoffnung, daß mehr und mehr Bundesländer für diese Aufgabe eigene Dienstleistungsangebote entwickeln werden. Bevor das erreicht ist, versucht das IfM interessierten Museen zumindest eine „Hilfe zur Selbsthilfe“ anzubieten; auch wenn dieses oder jenes Bundesland noch nicht in der Lage ist, seine Museen auf diesem Gebiet professionell zu beraten, soll es einzelnen Museen zumindest erleichtert werden, sich hier selber schlau zu machen.

Zum Inhalt: Zur Zeit der Großrechner waren sich noch alle Beteiligten darüber einig, daß man für den Einsatz der Informationstechnik professioneller Hilfe bedarf. Man war bereit, sich den Anforderungen der Maschine anzupassen, man versuchte, sich „computergerecht“ zu verhalten. Die Einführung leicht zu bedienender und leistungsstarker Computer im Bürobereich hat diese Arbeitsbedingungen aber grundlegend verändert. Auch Leute, die von Computer noch nichts verstehen, können heute in wenigen Tagen lernen, mit Programmen für „Textverarbeitung“ ganz selbstverständlich umzugehen. Sie erwarten daher, daß das bei einem Einsatz des Computers für die Inventarisierung genauso problemlos sei.

Von einer solchen Benutzerfreundlichkeit der Programme sind wir im deutschen Museumswesen aber noch weit entfernt. Das hat einen einfachen Grund: In die eben erwähnten einfach zu handhabenden Programme wurde inzwischen hunderte oder gar tausende von „Mannjahren“ investiert; ein erheblicher Teil dieser Mittel diente dazu, den Computer den spezifischen Bedürfnissen bestimmter Arbeitsplätze anzupassen, bzw. die daran arbeitenden Mitarbeiter auszubilden.

Bis das auch für das Museum gilt, wird wohl noch einige Zeit vergehen; dieser Markt ist zu klein, als daß sich solche Investitionen auf rein kommerzieller Basis kurzfristig auszahlen könnten. Das Institut für Museumskunde versucht hier Hilfestellung zu geben.

Das vorliegende Heft 33 der "Materialien" ist aus Beratungen und Kursen hervorgegangen. Es versucht, die für die Einführung der Informationstechnik im Museum unabdingbaren grundlegenden Computerkenntnisse für Museumsleute in Form eines Lern- und Lesebuchs zu vermitteln. Es schließt damit an Heft 30 (Jane Sunderland und Leonore Sarasan, Was muß man alles tun, um den Computer im Museum erfolgreich einzusetzen?) direkt an und soll zusammen mit ihm benutzt werden.

Andreas Grote

Berlin, im September 1991

Inhaltsverzeichnis

Einleitung: Zweck des Handbuchs, Gliederung	7
Das Handbuch als Lern- und Lesebuch - Hilfe zur Selbsthilfe	9
Worum geht es bei diesem Handbuch ?	10
1. Hauptteil: Formale Grundlagen für die Definition von Datenfeldern	13
Grundbegriffe: Saubere und schmutzige Daten	14
Die Formulierung von Textdaten (Schlagwörter, Namen, freie Texte)	18
Welche Arten von Textdaten gibt es ?	18
Was kann ein "normales" Computerprogramm mit Textdaten machen ?	20
Wiedergabe von Textdaten (Druck von Karteikarten, Katalogen usw.)	21
Das Sortieren von Textdaten	22
Wie sortiert ein Computer alphabetisch ?	22
Was ist eigentlich eine richtige alphabetische Ordnung ?	24
Alphabetische und systematische Register - wozu dienen sie eigentlich ?	26
Das Suchen von Textdaten (Retrieval)	30
Volltextrecherche	30
Suchen "im Dialog"	31
Welche Informationen gehören zum Objekt ? Das Arbeiten mit Hilfsdateien	33
Statt langer Erläuterungen ein Beispiel: Erzeugung systematischer Register	33
Grundbegriffe: Vermeidung von "Redundanz" durch Hilfsdatenbanken	38
Präferenz- und Stopplisten	38
Abkürzungsdateien	39
Lexika	39
Thesauri	41
Zusammenfassung: Wie benutzt der Computer Hilfsdateien ?	42
Stand der Forschung zu Fragen der Terminologie im Museum	43
Zusammenfassung: Stufenmodell für die Formulierung von Textdaten	45
Anhang zum 1. Hauptteil: Komplizierte Formen der Syntax eines Datenfeldes	48
Inventarnummern und andere Numerierungssysteme	48
Was sind Inventarnummern ?	49
Was kann ein "normales" Computerprogramm mit Inventarnummern machen ?	50
Einfache arabische Zahlen mit Zusätzen	50
Buchstaben mit Zahlen	51
Kombination aus mehreren Zahlengruppen	51
Römische Zahlen	53
Weitere Fehlerquellen, Gegenmittel	53
Kombination mehrerer Arten von Inventarnummern in einem Museum	54
Zusätzliche Probleme: Sammel- und Bereichsangaben	55
Tricks zur Umgehung solcher Probleme: "Die Dokumenten-Nummer"	55
Was muß ein Programm können, um Inventarnummern richtig zu sortieren ?	56
Sortieren für Druckausgaben	56
Zusatzprogramme für Sammel- und Bereichsangaben	58
Beispiel für das richtige Sortieren von Bildschirmausgaben	59

2. Hauptteil: Inhaltliche Grundlagen für die Definition von Datenfeldern und -strukturen	60
Was kann ich eigentlich alles in ein einzelnes Datenfeld stecken ?	64
Was tut man mit ungenauen Daten ? (Rembrandt oder niederländisch 17. Jh. ?)	65
Was macht man mit "historischen" Daten ? (Chemnitz oder Karl-Marx-Stadt ?)	67
Listenartige Datenkataloge mit einfacher ("flacher") Datenstruktur	67
Zusammenfassung: Was sind die Grenzen fest vordefinierter Datenkataloge ?	73
Was ist ein Museumsobjekt ?	75
Komplexe Datenstrukturen (Mehrfach- und Gruppenfelder, Verweise)	76
Einfach strukturierte Objekte mit je einer einzigen Information pro Datenfeld	77
Einfach strukturierte Objekte mit mehreren Informationen pro Datenfeld	78
Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern	80
Konvolute aus einfach strukturierten Objekten	82
Darstellung von Zusammenhängen durch Verweise	84
Zusammenfassung: Die Suche nach einfacheren und flexibleren Datenstrukturen	87
Anhang: Arbeitsblätter für die produktorientierte Analyse von Textdaten	93
Arbeitsblatt zur Struktur von Dateien und Datensätzen	95
Zur Wiederholung: IST-Analyse von Datenfeldern nach Sarasan	97
Arbeitsblätter zur Beschreibung von einzelnen Feldern	99
Arbeitsblätter zur Beschreibung von Numerierungssystemen	106
Arbeitsblätter zum Zeichensatz	114
Glossar	123
Register	127
Veröffentlichungen aus dem Institut für Museumskunde	129

Einleitung: Zweck des Handbuchs, Gliederung

Dieses Handbuch ist aus vielen, zusammen mit Carlos Saro vom Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB) zusammen durchgeführten Kursen und Einzelberatungen entstanden. Es ergänzt und vertieft die 1990 als Heft 30 der Materialien aus dem Institut für Museumskunde¹ publizierten und schnell zur "Pflichtlektüre" avancierten Aufsätze von Jane Sunderland und Lenore Sarasan.

Heft 30 konzentrierte sich auf die Planung und gab daher nur sehr knappe Erläuterungen zum Funktionieren des Computers. Heft 33 versucht diese Lücke zu stopfen und das nötige "Computerwissen" in einer für Museumsleute verständlichen Form zu vermitteln.

Bevor Sie sich mit diesem neuen Handbuch auseinandersetzen, sollten Sie sich die Einleitung durchlesen. Sie enthält eine Reihe nützlicher Informationen zum Zweck des Handbuchs und zu seiner Gliederung.

Ein paar Worte zum Zweck dieses Handbuchs

Nur wenige von Ihnen werden sich wirklich für den Computer interessieren. Für die meisten Museumsleute gehört zu den vielen Arbeitsmitteln, die sie in ihrem Beruf benutzen, ohne dafür eine entsprechende Ausbildung bekommen zu haben.

Dies wäre nicht so schlimm, wenn uns wie auf manchen anderen Gebieten (z.B. im Bürobereich) bereits heute eine wirklich "intelligente" und nicht nur oberflächlich auf die Arbeit im Museum abgestimmte Software zur Verfügung stehen würde. Eine solche Software könnte das Ausbildungsproblem zwar nie vollständig lösen, könnte es aber erheblich erleichtern.

Daraus ergeben sich ein paar sehr berechtigte Fragen: "Warum gibt es diese Software noch nicht?" und "Was soll ich nun machen?" Ich glaube, daß Ihnen dieses Handbuch ein paar Antworten geben wird und möchte hier in der Einleitung nur einige wenige, aber wichtige Hintergrundinformationen geben.

- **Intelligente Software:** Vielleicht haben Sie bereits Museen oder Programmanbieter kennengelernt, die mehr oder weniger behaupten, schon jetzt über solche Programme zu verfügen. Eine Aussage wie "Wenn Sie unser System benutzen, brauchen Sie sich keine Sorgen mehr zu machen" ist sehr verführerisch, selbst wenn uns die täglich auf uns niederprasselnde Werbung gegen solche Sprüche unempfindlich gemacht haben sollte. Wie schützt man sich gegen solche Dinge? Vernünftige Gegenmittel wie Verbraucherorganisationen, Testzeitschriften usw. gibt es in diesem Bereich noch nicht und Museumsverbände und Museumsämter beginnen erst, sich mit solchen Problemen auseinanderzusetzen. Bis Ihnen irgendjemand diese Arbeit abnimmt, müssen Sie sich also weitgehend selber helfen. Dieses neue Handbuch ist ein Beitrag dazu.
- **Was soll man machen?** Hier gibt es grundsätzlich zwei Antworten. Die erste heißt natürlich "Lernen" und davon handelt dieses Handbuch. Die zweite und langfristig nicht weniger wichtige Antwort besteht darin, daß Sie sich mit dieser Situation nicht zufriedengeben und hierfür von den für Sie zuständigen Stellen ganz massiv Unterstützung anfordern. Was da im einzelnen zu tun wäre, ist bekannt oder zumindest ausreichend publiziert². Es reicht also nicht aus, das Fehlen solcher Hilfen zu beklagen, man muß schon auch etwas dafür tun.

1) Jane Sunderland und Lenore Sarasan, Was muß man alles tun, um den Computer im Museum erfolgreich einzusetzen?

2) Vgl. hierzu Christof Wolters, Vorschläge zur Planung zentraler Dienstleistungen für Museen in der Bundesrepublik Deutschland und Berlin (West), in: Museumsblatt 2, 1990, Hrsg. Landesstelle für Museumsbetreuung in Zusammenarbeit mit dem Museumsverband Baden-Württemberg e.V. Neckarhalde 30 A, D 7400 Tübingen, S. 6-26 (s. Publikationen auf Seite 133).

Bei allen diesen Dingen gilt natürlich das alte Sprichwort "Wer nicht hören will muß fühlen". Niemand wird Sie ernsthaft mit dem Rohrstock bedrohen, aber wenn Sie nicht über unbeschränkte Mittel an Zeit und Geld verfügen, sollten Sie sich nicht nur selber schlau machen, sondern auch ernsthaft daran arbeiten, die in der Bundesrepublik Deutschland noch völlig unterentwickelte Infrastruktur für Systementwicklung, Schulung und Betreuung beim Einsatz der Informationstechnik im Museum zu verbessern.

Der Zweck dieses Handbuchs ist also letztlich eine Hilfe bei der für Museen immer dringlicher werdenden Aufgabe, die Betriebskosten zu senken.

Warum sind Handbücher so schwer zu lesen (und zu schreiben) ?

Ein völlig logisch und systematisch aufgebautes Handbuch, in dem jede Sache an der richtigen Stelle (und nur an dieser) steht, ist für noch nicht Eingeweihte nur sehr schwer zu benutzen. Ständig muß man hin- und herblättern, Verweisen auf Dinge nachgehen, von denen man nicht weiß, ob man sie wirklich braucht.

Wenn Sie schon einmal versucht haben sollten, ein Computersystem mit Hilfe des (hoffentlich) mitgelieferten Handbuchs zu lernen und das Gelernte für Ihre eigenen Zwecke anzuwenden, so werden Sie sicher einiges zur Frage der Verständlichkeit solcher Handbücher beizutragen haben. Auch Fachleute geben zu, daß die meisten Handbücher meist erst dann nützlich sind, wenn man das darin beschriebene System bereits weitgehend verstanden hat.

Für den Autor eines solchen Buches sieht die Sache gar nicht so verschieden aus. Wenn er - wie in diesem Falle - nicht auf bereits vielleicht in anderen Sprachen erarbeitete Bücher zurückgreifen kann, wird er sehr schnell an seine Grenzen stoßen. Obwohl er im Grunde schon alles weiß (oder vielleicht auch nur so tut), wird er beim schreiben sehr schnell feststellen, daß ihm für viele Dinge die überzeugenden Beispiele fehlen, daß die ursprünglich klare Gliederung zu starr ist und daß man im Grunde nicht erwarten kann, daß die Leser das so verstehen, wie er es gerne hätte. Und das ist der Punkt, an dem diese Arbeit auch für den Autor fruchtbar wird. Letztlich wird er feststellen, daß er die Sache erst dann wirklich kapiert hat, wenn er das Manuskript abgeschlossen hat.

Alle diese Schwierigkeiten haben mit dem Computer zunächst gar nichts zu tun. Ein Statistiker wird sich in einem statistischen Programmpaket relativ schnell zurechtfinden, dem normalen Menschen muß man zunächst die Grundlagen der Statistik (und ein bißchen Mathematik) beibringen. Sie sollten deswegen nicht erstaunt sein, wenn Sie mit einem modernen Programm für Textverarbeitung schon nach wenigen Tagen ganz vernünftig umgehen können; Sie wissen, wie ein Brief oder ein Manuskript aussehen sollten, Briefköpfe, Adressen, Überschriften, Inhaltsverzeichnisse und Anmerkungen sind Ihnen vertraut, entsprechende Hilfsprogramme sind deswegen meist leicht zu kاپieren und anzuwenden.

Etwas anders stellt sich die Sache bei der wissenschaftlichen Dokumentation mit Hilfe von Datenbanksystemen dar. Wenn Sie keine Ausbildung als wissenschaftlicher Dokumentar oder Bibliothekar hinter sich haben, werden Sie - trotz etwaiger Erfahrungen mit der Inventarisierung - erhebliche Schwierigkeiten haben, und zwar auf zwei Gebieten:

- **Regelwerke:** Wenn Sie in einer Bibliothek ein Buch finden, so liegt das daran, daß dort Bibliothekare arbeiten, die mit Hilfe eines Regelwerks (wie z.B. der RAK³) brauchbare Kataloge hergestellt haben. In einem solchen Regelwerk können Sie nachlesen, wie Sie die für einen Katalog erforderlichen Informationen "ansetzen" müssen, damit ein anderer sie wiederfinden kann.

3) Regeln für die alphabetische Katalogisierung, Band 1: Regeln für wissenschaftliche Bibliotheken. Wiesbaden: Reichert, 1983.

Schauen Sie sich z.B. das Kapitel "Ansetzung der Namen von Personen" (§ 300-342) an. Wahrscheinlich werden Ihnen solch starre Regeln nicht gefallen (sie sind in dieser Form wohl auch schon ein bißchen veraltet), sie werden aber bei der Lektüre feststellen, daß man eine Menge über historische und andere Namen wissen muß, um ein brauchbares Regelwerk zu entwickeln und dann zuverlässig anzuwenden.

- **Computer:** Schauen Sie in das vom IfM publizierte "Pflichtenheft zum Vergleich von Computersystemen für Museumssammlungen"⁴; die meisten der dort verwendeten Begriffe werden Ihnen wahrscheinlich neu oder "nur so ungefähr klar" sein.

Wenn Ihnen das alles als viel zu kompliziert und aufwendig erscheint und Sie daraufhin beschließen, in Ihrem Museum lieber Dinge zu tun, für die Sie schon jetzt das nötige Hintergrundwissen haben, so ist das gut zu verstehen. Wenn Sie es trotzdem wissen wollen, kann Ihnen das hier vorgelegte Heft 33 der Materialien aus dem Institut für Museumskunde dabei helfen, einen annehmbaren Kompromiß zwischen dem Denkbaren und dem Machbaren zu finden.

Langer Rede kurzer Sinn: wenn Sie meinen, daß es völlig ausreicht, Ihre Inventare "so wie sie sind" in den Computer zu tippen, um dann eine vernünftige Computer-Inventarisierung zu haben, dann müssen Sie diese Meinung ändern⁵. Das Schlaraffenland (zusammengesetzt aus mhd. slûr = Faulenzerei und Affe), das Land in welchem dem Menschen ohne jede geistige oder körperliche Anstrengung alle materiellen Güter und Genüsse zuteil werden, ist ein Märchen (und selbst dafür mußte man sich durch eine Mauer von Hirsebrei hindurchfressen).

Das Datenbanksystem, das man im Museum ohne Vorkenntnisse erfolgreich benutzen kann, das Handbuch, das einem sagt, was man tun soll, ist zur Zeit nicht auf dem Markt.

Das Handbuch als Lern- und Lesebuch - Hilfe zur Selbsthilfe

Wahrscheinlich werden Sie kaum die Zeit haben, um sich jetzt auch noch eine gründliche Zusatzausbildung in wissenschaftlicher Dokumentation zu besorgen. Sie müssen hier also - wie wahrscheinlich bei anderen Aufgaben auch - dafür sorgen, daß Sie sich trotzdem die nötigen Kenntnisse aneignen. Erforderlich wäre ein ganzes Bündel von Maßnahmen, zu dem das IfM gerne beitragen will, das es aber nicht alleine schaffen kann. Wie stellen wir uns unseren Beitrag vor ?

Die Antwort lautet **Hilfe zur Selbsthilfe**. Das Institut für Museumskunde kann und will Sie nicht zwingen, einen bestimmten Weg einzuschlagen, es möchte Ihnen aber dabei helfen, Probleme zu erkennen und zu lösen. Wenn Sie damit einverstanden sind, müßten Sie sich überlegen, welche der nach unserer Ansicht und Erfahrung unabdingbar notwendigen Kenntnisse in Ihrem Hause bereits vorhanden sind und welche ggf. noch erworben bzw. aufgefrischt werden müßten. Die Entscheidung liegt bei Ihnen.

Um Ihnen den Überblick zu erleichtern, schlage ich eine grobe Gliederung der für einen effizienten Einsatz der Informationstechnik bei der Inventarisierung notwendigen Kenntnisse und Fertigkeiten vor:

- **Im Prinzip bereits vorhanden: Museumswissen.** Dieses reicht vom Fachwissen über die gesammelten Objekte über das dokumentarische Wissen bis zu dem nötigen "Organisationstalent".

4) Jane Sunderland und Lenore Sarasan, Was muß man alles tun, um den Computer im Museum erfolgreich einzusetzen ? in: Materialien aus dem Institut für Museumskunde, Heft 30, S. 29 ff.

5) Eine in jedem Detail ernstzunehmende Übersicht über die Probleme, die die meisten Leute durch Aufsetzen von Scheuklappen zu lösen versuchen, finden Sie in: Lenore Sarasan, Why Museum Computer Projects Fail, Museum News, 1981, January/February, S. 40-49.

In diesem Punkte kann das Institut für Museumskunde nicht allzuviel tun, obwohl durchaus denkbar ist, daß etwaige Schwierigkeiten bei der Einführung der Informationstechnik hier ihre Wurzel haben könnten. Die Erfahrung zeigt nämlich, daß viele dem Computer angela-stete Probleme in Wirklichkeit andere Ursachen haben; wenn jemand Schwierigkeiten mit dem Computer bekommt, so liegt das oft genug daran, daß er das Problem, das ihm die Maschine lösen soll, noch nicht wirklich im Griff hat.

Man kann das auch positiv sehen: bei dem Versuch, den Computer einzusetzen, kann man eine Menge über konventionelle Dokumentationsverfahren lernen.

- **Meistens noch zu erarbeiten: Vernünftiger Einsatz der Informationstechnik.** Die hierfür notwendigen Kenntnisse und Fertigkeiten entsprechen spiegelbildlich dem oben skizzierten "Museumswissen". Das Fachwissen zu den Objekten muß man um Kenntnisse, was man von der Informationstechnik prinzipiell erwarten darf, ergänzen. Dem dokumentarischen Wissen entspricht eine nicht nur oberflächliche Auseinandersetzung mit Datenbanktechnik. Und last not least ist auch hier ein erhebliches Organisationstalent ("Planungswissen") gefragt.

In diesem Bereich kann Ihnen das Institut für Museumskunde helfen. Meistens wird es darum gehen, das nötige Wissen nicht abstrakt, sondern in einer für museale Zwecke spezifischen Form zu vermitteln.

Für die noch notwendige Fortbildung entwickelt das Institut für Museumskunde eine Reihe von Dienstleistungen, aus denen Sie sich Ihr persönliches Programm zusammenstellen können. Im Idealfall sollte man für jedes Thema mehrere Lernhilfen zur Verfügung stellen, das Spektrum sollte von der theoretischen Darstellung über Beispiele bis hin zu praktischen Übungen auf Papier oder am Computer reichen. Das Institut für Museumskunde kann Ihnen hier aber zur Zeit noch kein volles Programm anbieten. Bitte informieren Sie sich über entsprechende Hilfen also auch bei regionalen Einrichtungen (z.B. bei den Museumsämtern und -verbänden) sowie bei den jetzt vielerorts entstehenden lokalen und regionalen Arbeitskreisen.

Worum geht es bei diesem Handbuch ?

Das Thema lautet: **Wie muß man seine Daten formulieren bzw. strukturieren, damit ein Computer etwas Vernünftiges damit anfangen kann?** Die nicht zu umgehende strenge kausale Verknüpfung von formalen und inhaltlichen Eigenschaften der Daten mit den gewünschten Leistungen wird an ausgewählten Themen mit vielen Beispielen veranschaulicht.

In Betracht zu ziehen ist hier immer der gesamte Weg von der ersten Erzeugung der Daten bis zum fertigen Produkt. Bei vielen Stationen dieses Weges könnte man den Computer einsetzen, bei jeder gibt es bestimmte Probleme und zumeist mehrere alternative Lösungen. Da hinter den einzelnen Schritten dieses Weges oft ein erheblicher Arbeitsaufwand steht, sollte man in der Lage sein, Zeit und Kosten nicht nur einzelner Schritte, sondern des gesamten Weges zu kalkulieren. Es passiert hier leicht, daß man den Teufel mit Beelzebub austreibt.

Das erforderliche "Jonglieren" mit allen Elementen einer Computerisierung ist zugegebenermaßen ein ungemein schwieriges Unterfangen, das Problem wird aber durch Aufsetzen von Scheuklappen nicht gelöst.

Muß man das wirklich alles wissen ? Ich möchte Sie nicht unnötig quälen, aber Sie müssen zumindest ein sicheres Gefühl dafür entwickeln, welche Folgewirkungen auch kleine und leicht zu übersehende Dinge beim Arbeiten mit dem Computer haben können. Sie sind beim Arbeiten mit einem Computer grundsätzlich in der aus vielen Märchen sattem bekannten Situation, daß Sie z.B. drei Wünsche haben, die dann die gute Fee (oder auch ein böser

Zwerg) ganz wortwörtlich erfüllt. Bevor Sie sich also wünschen, daß alles, was Sie anfassen, zu Gold wird, müssen Sie sich klarmachen, was das für Ihre Ernährung und mancherlei andere wichtige Dinge bedeutet.

Ein solch stures "wortwörtliches" Vorgehen ist nun mal das grundlegende Merkmal solcher Maschinen. Ein "guter Riecher" oder (noch besser) solide Kenntnisse der möglichen Folgewirkungen sind für den Erfolg oder Mißerfolg einer Computerisierung entscheidend. Obwohl das Institut für Museumskunde hier bereits Lernhilfen bietet, haben wir uns dieses Thema hier noch einmal gesondert vorgenommen.

Grundlage sollte das bereits mehrfach zitierte **Heft 30 der "Materialien"** sein. Dort werden die ersten Schritte bei der Einführung des Computers im Museum in sehr praktischer Weise (und so knapp wie überhaupt möglich) beschrieben. Mit Hilfe dieses Heftes kann ein Museum eine professionelle Planung und Vorbereitung mehr oder weniger selbständig durchführen und dabei auch seine Anforderungen an den Computer genau spezifizieren. Im Zentrum steht ein Aufsatz von Lenore Sarasan ("Ein System zur Analyse von Museumsdokumentation"). Die Erstellung einer darauf basierenden **schriftlichen** Systemanalyse ist die sicherste Art, das dabei Gelernte zu kontrollieren. Wenn Sie dabei Schwierigkeiten hatten, wird Ihnen unser neues Handbuch helfen (s. Kapitel "Zur Wiederholung: IST-Analyse von Datenfeldern" auf Seite 97 ff.). Um zu kontrollieren, ob man die von Lenore Sarasan vorgeschlagene Systemanalyse wirklich kapiert hat, sollte zumindest auch ein ernsthafter Versuch, das dort beigegebene "Pflichtenheft" auszufüllen, gemacht werden.

Darauf aufbauend sollte man sein bisher mehr theoretisches Wissen durch praktische Arbeiten (neuhochdeutsch: "Learning by Doing") vertiefen. Bevor man sich allerdings selber ausdenkt, wie man etwas lernen könnte, wovon man keinen blassen Schimmer hat, sollte man von den in den letzten zwanzig Jahren beim Computereinsatz im Museum gemachten Erfahrungen profitieren und sicherstellen, daß das Lernprogramm nicht an der Sache vorbeigeht. Diesem Ziel dient das hier vorgelegte Lern- und Lesebuch.

Das hier vorgelegte Handbuch wendet sich in erster Linie an Leute, die die Informationstechnik in ihrem Museum einführen wollen, aber noch nichts von Datenbanken verstehen.

Das Handbuch gliedert sich also nicht (wie sonst meistens üblich) nach den Leistungen und den Bedienungsvorschriften eines bestimmten Programmsystems, sondern nach "Zusammenhängen", die einem beim Arbeiten mit solchen Maschinen vertraut sein sollten.

Dabei gehen wir grundsätzlich vom Speziellen zum Allgemeinen, vom konkreten Beispiel ("was passiert, wenn man . . .") zur allgemein verwendbaren Definition.

- **1. Hauptteil: Formale Grundlagen für die Definition von Datenfeldern.** Der 1. Hauptteil besteht im wesentlichen aus einer ausführlichen Auseinandersetzung mit der äußeren Form der zu erfassenden und verarbeitenden **Daten**. Es wird zunächst veranschaulicht, daß viele Produkte nur dann erzeugt werden können, wenn wir bei der Formulierung der Informationen bestimmte Regeln (Schreibanweisung, Syntax) einhalten. Hieraus lassen sich **grundlegende Anforderungen an die Definition von Datenfeldern** (vulgo: ein computergerecht definiertes "Häuschen" auf der Karteikarte) ableiten.

In diesem 1. Hauptteil wird dann auch die für viele Zwecke erforderliche Verknüpfung eines oder mehrerer solcher Datenfelder mit einer Hilfsdatei (vulgo "Nachschlagewerk") abgehandelt.

Die eigentlichen Feldinhalte spielen in diesem ersten Hauptteil nur eine untergeordnete Rolle, da der Computer sie nicht versteht und damit prinzipiell "nichts anfangen kann". Als eine Art von Paradebeispiel für diese stur formalistische Arbeitsweise des Computers behandeln wir in einem Anhang zum ersten Hauptteil die Verarbeitung von Inventarnummern und anderen Numerierungssystemen.

- **2. Hauptteil: Inhaltliche Grundlagen für die Definition von Datenfeldern und -strukturen.** Eine vernünftig strukturierte Datenbank bedarf zunächst einer klaren, die im 1. Hauptteil beschriebenen Fähigkeiten der Maschine geschickt einsetzenden, inhaltlichen Gliederung. Auch diese Gliederung muß sich wiederum an den Möglichkeiten der Maschine orientieren, wenn man inhaltliche Zusammenhänge nicht verlieren will.

Wir betrachten also zunächst die Frage, welche Inhalte man in ein einzelnes Datenfeld stecken kann. Wir lernen dabei, daß auch zwischen der **inhaltlichen Definition eines Datenfeldes** und der Möglichkeit, bestimmte Produkte zu erzeugen, klar beschreibbare, kausale Zusammenhänge bestehen.

In vielen Fällen wird aber eine einfache Aneinanderreihung von (formal und inhaltlich wohl definierten) einzelnen Datenfeldern nicht ausreichen. In einem zweiten Kapitel ("Was ist ein Museumsobjekt?") geht es dann um inhaltliche Zusammenhänge, die sich nicht durch eine einfache listenartige Aneinanderreihung von Datenfeldern darstellen lassen. Wir beschäftigen uns in diesem Teil also mit der **Struktur von "Datensätzen"** (vulgo: "Karteikarte").

- **Anhang: Arbeitsblätter für die produktorientierte Analyse von Textdaten.** Im Anhang werden - in Anlehnung an Heft 30 der "Materialien" - Arbeitsblätter für die Analyse von Textdaten beschrieben. Diese Arbeitsblätter fassen den Stoff des ersten und zweiten Hauptteils noch einmal in konzentrierter Form zusammen und können sowohl für eine Vertiefung der Systemanalyse als auch zur Kontrolle des Gelernten eingesetzt werden.
- **Glossar und Register:** In der Form eines Glossars werden die für das Thema wesentlichen Grundbegriffe der Datenverarbeitung noch einmal aus der Sicht der Informationstechnik erläutert. Es enthält also eine ganze Reihe von Informationen, die sonst nicht ausführlich erklärt werden und kann als selbständiges Kapitel gelesen werden.

Das daran anschließende **Register** soll das Nachschlagen erleichtern.

Ich gehe davon aus, daß nur wenige Leser die Zeit haben werden, dieses Handbuch systematisch (und in einem Stück) durchzuarbeiten. Die einzelnen Unterkapitel wurden daher so selbständig wie möglich gehalten. Zahlreiche Querverweise sollen es Ihnen erlauben, Ihren eigenen Weg durch dieses Labyrinth zu finden.

Über eine Sache sollte allerdings Klarheit herrschen: Wenn Sie sich auf das Abenteuer einer Computerisierung Ihres Museums einlassen wollen, so wird Sie diese "Rationalisierungs-Investition" zumindest in den ersten Jahren viel Zeit und Mühe kosten. "So nebenbei" läßt sich die Sache leider nicht machen. Wenn Sie diese Zeit nicht haben, sollten Sie vielleicht noch etwas warten.

1. Hauptteil: Formale Grundlagen für die Definition von Datenfeldern

Zweck dieses ersten Hauptteils ist eine erste Vertiefung der hier als bekannt vorausgesetzten "Pflichtlektüre"⁶. Die nicht zu umgehende strenge kausale Verknüpfung von formalen Eigenschaften der Daten mit den gewünschten Leistungen wird an geeigneten Beispielen veranschaulicht. Es geht also darum, wie man das "Rohmaterial", die Informationen, formulieren bzw. strukturieren muß, um dann damit bestimmte "Produkte" erzeugen zu können.

Wir werden das zunächst an einzelnen "Datenfeldern" erarbeiten. Was ein solches Datenfeld ist, können wir uns zunächst mehr gefühlsmäßig an einem Vergleich mit einer normalen Kartei klarmachen:

Heimatismuseum Rykenberg <hr/> Inventar-Nr: 78/1246 <hr/> Objekt: Glasvitrine
Heimatismuseum Emsdetten <hr/> Inventar-Nr: 78/127 <hr/> Objekt: Glasschrank
Museum der Stadt Bensberg <hr/> Inventar-Nr: K 12 <hr/> Objekt: Schrank <hr/> Beschreibung: Schrank, 18. Jh., Eichenholz

Abb. 1: Grundbegriffe: Datenfelder (= nicht mehr teilbare Elemente wie Museumsname, Inventarnummer, Objektbezeichnung usw., etwa "einzelnes Häuschen auf einer Karteikarte"), **Datensätze** (= Zusammenfassung von Datenfeldern als Attribute zu einer inhaltlich definierten Einheit, etwa "Karteikarte"), **Datei** (Zusammenfassung von gleichartigen Datensätzen, etwa "Kartei")

Wir werden dann im Laufe des 1. Hauptteils noch erkennen, daß dieser Vergleich mit einer normalen Kartei nicht ausreicht und daß wir zu einer Definition eines computergeeigneten Datenfeldes auch noch zusätzlich wissen müssen, wie dieses Datenfeld dann zu bestimmten Produkten verarbeitet werden soll.

Diese bewußt auf das einzelne Datenfeld beschränkte Sichtweise soll Ihnen den Einstieg in die Computerwelt erleichtern. Sie soll Sie nicht dazu verführen, mit dem Computer nun ganz genau das gleiche wie in einer "konventionellen" Dokumentation zu tun. Sie werden bald entdecken, daß ein wirkungsvoller Einsatz der Informationstechnik ein Umdenken erfordert. Auch konventionelle Verfahren sind stark durch ihr Medium ("Papier") bestimmt, dadurch existierende Beschränkungen sind uns aber oft so sehr in Fleisch und Blut übergegangen, daß wir sie kaum noch wahrnehmen oder uns nur schwer von ihnen trennen.

⁶ **Pflichtlektüre:** Heft 30 der "Materialien" S. 19 Übung 3: Das Datenfeld-Arbeitsblatt. Weitere Erläuterungen zu dieser Übung finden Sie hier auf Seite 97 ff.

Zum Inhalt: Computer können heute eine Menge überaus komplizierter Arbeiten erledigen. Solange das einwandfrei funktioniert, braucht man sich keine Gedanken darüber zu machen, wie ein Produkt zustandekommt.

Anders sieht das aus, wenn das gewünschte Produkt nicht, oder nur sehr umständlich oder gar fehlerhaft erzeugt wird. Dann muß man imstande sein, den Weg von den Ausgangsdaten (=Rohmaterial) zum fertigen Produkt in einer Weise zu beschreiben, die sich in Programme umsetzen läßt.

Oft genug wird man dabei feststellen, daß der "Fehler" in den Daten liegt, daß man vom Computer etwas verlangt, was er prinzipiell nicht leisten kann. Es ist daher für Systementwickler grundsätzlich nötig, die wenigen und an sich sehr einfach zu verstehenden Dinge, die ein Computer zuverlässig kann, zu kennen. Erst dann kann man - durch eine geeignete Kombination kleiner Einzelschritte - sinnvolle Produkte entwickeln.

In diesem 1. Hauptteil geht es um den Erwerb solcher Grundkenntnisse. Sie werden lernen, welche Anforderungen ein Produkt an die Daten stellt und durch welche Tricks man diesen Weg verkürzen oder erleichtern kann. Wir konzentrieren uns dabei auf eine beispielhafte Auswahl von Daten und Grundprodukten; erst wenn diese Grundlagen verstanden sind, hat es Sinn, zu komplexeren bzw. spezielleren Produkten überzugehen.

Zur Gliederung: Die Themen und die Reihenfolge ihrer Behandlung wurden so ausgewählt, daß jeweils eine schrittweise Vertiefung des notwendigen Wissens stattfinden kann. Dabei wurden aus Gründen der Verständlichkeit erhebliche Überschneidungen in Kauf genommen. Wenn bestimmte Kenntnisse vorausgesetzt werden, so wird im jeweiligen Abschnitt auf die nötige "Pflichtlektüre" hingewiesen.

Bei jedem Thema wird das Problem zunächst an einigen, besonders anschaulichen Beispielen erläutert. Diese Beispiele wurden sehr sorgfältig ausgewählt, sie können Ihnen u.a. auch beim Testen von Programmsystemen helfen.

Grundbegriffe: Saubere und schmutzige Daten

Wir wollen uns zunächst einmal mit dem weit verbreiteten Vorurteil beschäftigen, daß man seine Daten (z.B. "Karteikarten") nur "so wie sie sind" in den Computer einzufüttern brauche, um dann damit phantastische oder auch nur "ganz einfache" Dinge anstellen zu können. Das kann klappen oder auch nicht. In diesem Abschnitt bekommen wir eine erste Vorstellung davon, warum das so ist. Wir werden die Schuld dafür den Daten in die Schuhe schieben und sie je nachdem, ob das gewünschte Produkt erreicht werden kann, als "sauber" oder "schmutzig" bezeichnen. Später müssen wir das dann differenzieren.

Allgemeiner Hintergrund: Ein Computer speichert Informationen nicht als Text oder als Zahl, sondern in einer internen Codierung, die nur aus 0 und 1 besteht (Dual- oder Binärsystem). Ein Zeichen (z.B. ein Buchstabe) wird bei praktisch allen Computern durch eine Folge von 8 Binärziffern (auch Byte genannt) repräsentiert, wodurch 256 (2^8) verschiedene Zeichen darstellbar sind. Die Interpretation eines internen Codes durch ein bestimmtes Zeichen wird durch eine Code-Tabelle festgelegt, welche jedem internen Code ein Zeichen als äußere Repräsentation zuordnet. Wenn Sie Näheres dazu wissen wollen (Sie brauchen das aber jetzt noch nicht), lesen Sie im Anhang das Kapitel "Arbeitsblätter zum Zeichensatz" auf Seite 114 ff.

Folgerungen: Der Computer versteht unsere Daten nicht, er kann also z.B. nicht ohne zusätzliche Hilfen erkennen, daß in einem Inventar leicht unterschiedlich formulierte Begriffe wie *Bartmann*, *Bartmannkrug*, *Bartmannskrug*, *Bartmänner*, *Bartmannkrüge* und *Bartmannskrüge*

oder auch kleine "orthographische" Variationen wie *Blechtopf* bzw. *Blech-Topf* in einer Beschreibung die gleiche Art von Objekt signalisieren⁷.

Für den Computer sind Begriffe zunächst nur dann identisch, wenn sie **zeichengenau** übereinstimmen. Bis zu einem gewissen Grade können sich manche Programme diese Übereinstimmung "ausrechnen" (z.B. bei *Blechtopf* bzw. *Blech-Topf*)⁸, in vielen Fällen reicht das aber nicht aus und man muß entweder für völlig konsistente Daten sorgen oder z.B. mit den Synonymenverknüpfungen eines Thesaurus arbeiten.

Als Beispiel für die möglichen Auswirkungen selbst kleiner Unsauberkeiten geben wir einen Ausschnitt aus einem alphabetischen Register aus dem Projekt "Kleine Museen"⁹ (Abb. 2 auf Seite 16). In der linken Spalte sehen Sie die orthographisch nicht vereinheitlichten Objektbezeichnungen von *Festtagsrock* bis *Fibel*. Begriffe, die mal mit und mal ohne Bindestrich geschrieben wurden (mit * markiert, z.B. *Feuersteinbeil* bzw. *Feuerstein-Beil*), erscheinen doppelt (auf die Gründe gehen wir unten im Abschnitt "Sortierung" genauer ein) und weit auseinandergerückt. In der rechten Spalte sehen Sie ein mit genau den gleichen Daten erzeugtes Register, allerdings wurden die "per Programm" als äquivalent erkannten Bezeichnungen nur einmal abgedruckt und nun auch richtig einsortiert.

Dieses Beispiel soll zunächst nur veranschaulichen, warum bei der Computerei der Begriff der "Vereinheitlichung" eine so große Rolle spielt. Wie weit man seine Daten für die Datenverarbeitung vereinheitlichen muß, hängt davon ab, was man später mit diesen Daten anstellen möchte. Je nach dem gewünschten Produkt und den Fähigkeiten der hierfür verwendeten Programme können die gleichen Daten als "sauber" (geeignet) oder "schmutzig" (schlecht oder überhaupt nicht geeignet) angesehen werden. Es spielt dabei überhaupt keine Rolle, ob die Daten richtig oder falsch sind, entscheidend ist allein ihre äußere Form. Wir geben ein paar kurze, einleitende Beispiele für solche Produkte, die zugleich die weitere Gliederung dieses Kapitels signalisieren:

- **Wiedergabe von Texten:** Im Rahmen eines die Information nur wiedergebenden, sie aber nicht weiter verarbeitenden Produkts (z.B. Druck von Karteikarten) sind ungleichmäßige Formulierungen durchaus geeignet, d.h. "sauber". Sie können als "freier Text" gedruckt werden wie sie sind, es treten bestenfalls "Platzprobleme" auf (z.B. daß ein Text zu lang ist, um in ein bestimmtes "Häuschen" einer Karteikarte zu passen).
- **Alphabetische Register:** In einem alphabetischen Schlagwortregister sieht das gleichzeitige Vorkommen von Formulierungen wie *Bartmann*, *Bartmannkrug*, *Bartmannskrug*, *Bartmänner*, *Bartmannkrüge* und *Bartmannskrüge* dann natürlich ziemlich scheußlich aus (vgl. auch die Bindestriche in Abb. 2 auf Seite 16):

Bartaxt
Bartmänner
Bartmann
Bartmannkrüge
Bartmannkrug
Bartmannskrüge
Bartmannskrug
 Barttasse

Da die Wörter alle mehr oder weniger den gleichen Anfang haben, kann man sich in einem solchen Register zwar zurechtfinden und es z.B. als Arbeitsunterlage zur Vereinheitlichung der Daten benutzen, als Register für einen gedruckten Katalog ist eine solche

7) **Pflichtlektüre:** Heft 30 der "Materialien" S. 7-10: Was versteht der Computer von der deutschen Sprache ?

8) Vgl. z.B. die Tabelle äquivalenter Schreibweisen in Heft 30 der "Materialien" S. 12.

9) Die meisten der im Folgenden benutzten Beispiele stammen aus diesem Projekt. Ausgewählt wurden die Daten aus den folgenden Museen: Museum der Stadt Bensberg; Heimatmuseum, August-Holländer-Museum in Emsdetten; Niederrheinisches Museum für Volkskunde und Kulturgeschichte in Kevelaer; Heimatmuseum Rykenberg in Werl.

"Schlamperei" aber kaum noch akzeptabel. Wir lernen hieraus, daß auch bestimmte aesthetische Ansprüche zur technischen Definition eines Produkts (bzw. des Weges, der dort hinführt) gehören.

Abb. 2: Auswirkungen orthographischer Variationen (hier: Bindestrich) in einem Register (die mit * markierten Begriffe kommen "doppelt" vor). Die linke Spalte wurde mit einem einfachen Sortierprogramm erzeugt, infolgedessen erscheint der *Feuerstein-Abschlag* weit vor dem *Feuersteinabschlag*. Die rechte Spalte wurde mit einem Programm erzeugt, das den Bindestrich beim Sortieren ignoriert, es wird also nur jeweils eine Variation gedruckt. Der *Feuersteinbrocken* steht also "richtig" zwischen dem *Feuerstein-Bohrer* und dem *Feuerstein-Faustkeil*.

Objektdaten (wie erfaßt)

Festtagsrock
 Feuerblasrohr
 Feuerbock
 Feuereimer
 Feuerlunte
 Feuermeldetafel
 Feuerstahl
 Feuerstein
 Feuerstein-Abschlag *
 Feuerstein-Abspliss *
 Feuerstein-Beil *
 Feuerstein-Bohrer *
 Feuerstein-Faustkeil
 Feuerstein-Hochkratzer *
 Feuerstein-Klinge *
 Feuerstein-Klingenschaber
 Feuerstein-La-Gravette-Spitze
 Feuerstein-Levalloisklinge
 Feuerstein-Levalloisspitze
 Feuerstein-Messer *
 Feuerstein-Messerchen *
 Feuerstein-Nagelkratzer
 Feuerstein-Nucleus
 Feuerstein-Pfeilspitze *
 Feuerstein-Rundkratzer *
 Feuerstein-Rundscharer *
 Feuerstein-Scharer *
 Feuerstein-Spitze *
 Feuerstein-Stirnscharer
 Feuerstein-Tardenois-Spitze
 Feuersteinabschlag *
 Feuersteinabspliss *
 Feuersteinbeil *
 Feuersteinbohrer *
 Feuersteinbrocken
 Feuersteinfeilspitze
 Feuersteinhacke
 Feuersteinhochkratzer *
 Feuersteinklinge *
 Feuersteinknolle
 Feuersteinkratzer
 Feuersteinmesser *
 Feuersteinmesserchen *
 Feuersteinfeilspitze *
 Feuersteinrundkratzer *
 Feuersteinrundscharer *
 Feuersteinscharer *
 Feuersteinspandolch
 Feuersteinspitze *
 Feuersteinsplitter
 Feuerzange
 Feuerzeuglunte
 Fibel

Vereinheitlichung per Programm

Festtagsrock
 Feuerblasrohr
 Feuerbock
 Feuereimer
 Feuerlunte
 Feuermeldetafel
 Feuerstahl
 Feuerstein
 Feuerstein-Abschlag
 Feuerstein-Abspliss
 Feuerstein-Beil
 Feuerstein-Bohrer
 Feuersteinbrocken
 Feuerstein-Faustkeil
 Feuersteinfeilspitze
 Feuersteinhacke
 Feuerstein-Hochkratzer
 Feuerstein-Klinge
 Feuerstein-Klingenschaber
 Feuersteinknolle
 Feuersteinkratzer
 Feuerstein-La-Gravette-Spitze
 Feuerstein-Levalloisklinge
 Feuerstein-Levalloisspitze
 Feuerstein-Messer
 Feuerstein-Messerchen
 Feuerstein-Nagelkratzer
 Feuerstein-Nucleus
 Feuerstein-Pfeilspitze
 Feuerstein-Rundkratzer
 Feuerstein-Rundscharer
 Feuerstein-Scharer
 Feuersteinspandolch
 Feuerstein-Spitze
 Feuersteinsplitter
 Feuerstein-Stirnscharer
 Feuerstein-Tardenois-Spitze
 Feuerzange
 Feuerzeuglunte
 Fibel

- **Retrieval:** Hinter dieser Bezeichnung verbergen sich Methoden, mit deren Hilfe Sie nach den irgendwo in den Computer gesteckten **Informationen suchen** und ggf. auch finden können. Für ein zuverlässiges Retrieval sind Variationen, wie wir sie eben bei den "Bartmännern" sahen, nur schlecht geeignet. Woher soll man wissen, nach welchen der zahllosen denkbaren Formulierungen eines Begriffs gesucht werden soll ?

Man kann das bis zu einem gewissen Grad durch Tricks wie z.B. "Volltextrecherche" ausgleichen; dann müßte man nach Wörtern suchen, die mit "*Bartm*" beginnen und hoffen, daß es nicht zuviele Wörter gibt, in denen diese Zeichenfolge aus einem anderen Zusammenhang stammt (s.u. Abschnitt über Volltextrecherche auf Seite 30). Es macht auch hier wieder einen erheblichen Unterschied, ob man sich das Ergebnis nur auf dem Bildschirm anschauen will (für den Hausgebrauch kann man vieles akzeptieren) oder ob man es drucken und ggf. an andere weitergeben möchte (da könnte es dann doch recht peinlich werden).

Daten sind also nicht per se sauber oder schmutzig, erst die Kombination aus dem gewünschten Produkt (im Beispiel: einfache Wiedergabe eines Textes, alphabetisches Register und Retrieval von Schlagwörtern) und dem Leistungsumfang etwaiger Hilfsprogramme (z.B. automatisches Erkennen äquivalenter Schreibweisen, vgl. Abb. 2) läßt uns erkennen, wie die Daten formuliert werden müssen.

Hier liegt der Grund, warum Ihnen die Fachleute immer wieder sagen, daß man zunächst versuchen soll, die Ziele einer Dokumentation genau zu bestimmen. Solche Ziele muß man dann allerdings nicht nur "inhaltlich" (z.B. "Schneller Zugriff auf alle relevanten Objekte"), sondern "technisch" definieren (z.B. geeignet für alphabetische oder systematische Register).

Zur Veranschaulichung solcher "technischen Definitionen" stellen wir uns vor, Sie wollten mit den Objektdaten aus der linken Spalte von Abb. 2 auf Seite 16 ein Register wie in der rechten Spalte der gleichen Abbildung erzeugen. Sie müssen sich dann für eine der folgenden Alternativen bzw. eine geschickte Kombination derselben entscheiden:

- **Objektdaten einzeln vereinheitlichen:** Suchen Sie sich alle Objekte, bei denen eine nicht gewünschte Schreibweise vorkommt, einzeln heraus, und korrigieren Sie diese Daten. Nach jeder weiteren Erfassung ist dieser Vorgang ggf. zu wiederholen. Wenn Sie Pech haben, müssen Sie die gleiche Korrektur hundert- oder gar tausendmal durchführen.
- **Gleichbehandlung äquivalenter Schreibweisen per Programm:** Sie müssen sich hierfür ein Programm besorgen oder schreiben (lassen), das Ihnen diese Arbeit¹⁰ abnimmt. Jeder äquivalente Begriff erscheint dann nur noch in einer Schreibweise und wird alphabetisch richtig eingeordnet.

Damit können Sie dann das in der rechten Spalte von Abb. 2 abgebildete Register erzeugen. Wenn Ihnen das noch nicht genügt, wenn es Sie also z.B. in der rechten Spalte von Abb. 2 stört, daß *Feuerstein-Abschlag mit Bindestrich*, *Feuersteinpfeilspitze ohne Bindestrich* geschrieben wird, so können Sie das natürlich ebenfalls durch die mühsame Methode der Vereinheitlichung von Objektdaten erreichen, es gibt hier aber bessere Wege einer solchen "terminologischen Kontrolle". Aus der Fülle der Möglichkeiten seien hier zwei kurz vorgestellt (wir werden weiter unten genauer darauf eingehen):

- **Liste erlaubter Begriffe:** Die einfachere Form besteht in der Erstellung einer Liste erlaubter Begriffe (Näheres s. Kapitel "Präferenz- und Stopplisten" auf Seite 38). Mit ihrer Hilfe können Sie dafür sorgen, daß nur bestimmte Schreibweisen erfaßt werden können.

Wenn Sie also *Feuersteinpfeilspitze* eintippen, kann die Maschine feststellen, daß sie diesen Begriff nicht kennt. Durch Blättern in den Listen können Sie dann - falls schon vermerkt - die erlaubte Schreibweise *Feuerstein-Pfeilspitze* finden und noch einmal eintippen.

10) S. Heft 30, S. 12, Abb. 1.

Dies bedeutet also eine Art computerunterstützter Vereinheitlichung der Objektdaten. Wenn solche Listen sehr umfangreich sind, kann es Schwierigkeiten machen, sich in ihnen zurechtzufinden. Durch eine Kombination mit einem Programm zur Erkennung äquivalenter Schreibweisen können Sie diese Nachteile weitgehend vermeiden (Sie brauchen dann die unterschiedliche Schreibweise nicht herauszusuchen und noch einmal einzutippen).

- **Thesaurus** (Näheres s. Kapitel über Thesauri, Seite 41): Die elegantere Methode besteht in der Benutzung thesaurusähnlicher Systeme. Hierdurch können Sie die Leistungen einer Liste erlaubter Begriffe mit der Lösung zusätzlicher Probleme (wie Photographie - Fotografie) verbinden. Der Vorteil besteht darin, daß in einem solchen System nicht nur die "richtigen", sondern auch die "falschen" Schreibweisen erfaßt werden können, daß sich der Computer also leichter in einer solchen Datei zurechtfindet.

Selbst wenn Sie sich jetzt mit diesen Möglichkeiten noch nicht so richtig auskennen, werden Sie gemerkt haben, daß auch ganz "winzige" Anforderungen an ein Produkt große Wirkungen haben können. Mit der Definition eines Produkts oder der Entscheidung für bestimmte Programme können Sie - ohne es zu wollen - die dafür notwendigen Arbeitsgänge umständlich (teuer) oder rationell (billig) gestalten.

Aesthetische Gesichtspunkte können und soll(t)en trotzdem bei der Definition von Produkten eine erhebliche Rolle spielen. Bevor man ein Regelwerk (z.B. einen Datenkatalog wie in Abb. 32 auf Seite 61) entwirft (oder darüber ein Urteil abgibt), sollte man ganz genau wissen, was man im Computer mit wie formulierten Daten machen kann.

Die Formulierung von Textdaten (Schlagwörter, Namen, freie Texte)

Abgesehen von ganz wenigen Ausnahmen sind die meisten Informationen, die ein Datenbanksystem im Museum verarbeiten soll, sogenannte "Textdaten". Sie erscheinen in den verschiedensten Formen, meist aber als einzelne Wörter oder mehr oder weniger lange Sätze. Auch die meisten Inventarnummern sind - informationstechnisch betrachtet - "Textdaten", sie werden im Anhang zum ersten Hauptteil in einem eigenen Kapitel ("Inventarnummern und andere Numerierungssysteme", Seite 48 ff.) behandelt.

Solche Textdaten richten sich bei ihrer Formulierung durch den Inventarisator zunächst nach den Regeln der natürlichen Sprache, das Ziel ist in erster Linie Genauigkeit und Verständlichkeit. Daraus ergeben sich eine ganze Menge von Problemen¹¹, die der Benutzer verstehen sollte und für die er eine seinen Zielen angemessene Lösung finden muß.

Welche Arten von Textdaten gibt es ?

Es ist hier natürlich völlig unmöglich, eine inhaltlich vollständige Darstellung auch nur der wichtigsten oder häufigsten Textdaten, die bei der Inventarisierung oder Katalogisierung im Museum vorkommen können, zu versuchen. Sie haben das schon bei der "Do-it-Yourself-Systemanalyse (s. Heft 30 der "Materialien", S. 19, Übung 3: Das Datenfeld-Arbeitsblatt) für Ihr Museum versucht und haben also einen Überblick über Ihre Daten erarbeitet und eine Gliederung in "Datenfelder" vorgenommen.

Jetzt sollen Sie diese Daten also so formulieren, daß sie für bestimmte Produkte geeignet sind. Das muß man sich ganz genau (und am besten mit Hilfe von Beispielen) überlegen.

11) **Pflichtlektüre:** kurze Übersicht in Heft 30 S. 9-11: "Was versteht der Computer von der deutschen Sprache ?"

Ein Beispiel: vielleicht haben Sie dabei auch ein Feld "Datierung" - wir nehmen es hier als Beispiel, weil es keine langen Erklärungen braucht. An einer Reihe üblicher Datierungsangaben überlegen wir uns, was wir mit diesen machen wollen.

Eine häufige Forderung besteht z.B. darin, die Objekte nach ihrer Entstehungszeit chronologisch zu sortieren. Es liegt auf der Hand, daß eine alphabetische Sortierung das nicht leistet:

Abb. 3: Hinter dem allgemeinen Begriff der "Sortierung" verbergen sich ganz unterschiedliche Dinge. Im Falle der Datierung wird nicht der betreffende Text, sondern ein ihm zugeordnetes Hintergrundwissen (z.B. Jahreszahlen) sortiert.

Richtig: Datierungen chronologisch	Unsinnig: Datierungen alphabetisch
spätpaläolithisch	1.3.1986
kurz vor 480 v.Chr.	1230-50
augusteisch	18. Jh.
frühe Kaiserzeit	augusteisch
karolingisch	ca. 1450
1230-50	ca. 1830
ca. 1450	frühe Kaiserzeit
Mitte 15. Jh.	Frühjahr 1957
oberrheinisch um 1520	Jugendstil
18. Jh.	karolingisch
ca. 1830	kurz vor 1900
kurz vor 1900	kurz vor 480 v.Chr.
Jugendstil	Mitte 15. Jh.
Frühjahr 1957	oberrheinisch um 1520
1.3.1986	spätpaläolithisch

Was aber kann man tun? Soll man - wie häufig vorgeschlagen - jede dieser Datierungen vor der maschinellen Erfassung in Jahreszahlen umsetzen und dann nach diesen Zahlen sortieren? Um das in den Griff zu bekommen, machen Sie bitte die Übung auf Seite 20.

Sie werden dabei feststellen, daß das zahlreiche und zum Teil auch schwierige Entscheidungen erfordert (von wann bis wann dauerte das *Spätpaläolithikum*, war das in Dänemark zur gleichen Zeit wie im vorderen Orient, bedeutet *18. Jh.* in diesem Falle wirklich "zwischen 1701 und 1800"?); und wie sichert man, daß alle Mitarbeiter hier die gleichen Umsetzungen machen? Oder soll man eine zweite Datei anlegen, in der dann ganz schematisch jeder vorkommende Begriff nur einmal (allerdings ohne jede Rücksicht auf den jeweiligen Kontext) in Jahreszahlen umgesetzt zu werden braucht?

Jede dieser Entscheidungen impliziert eine Menge Arbeit (von der inhaltlichen Vorbereitung der maschinellen Datenerfassung bis zum Programmieren), jede hat ihre Vor- und Nachteile. Es wird hier aber deutlich, daß eine rein "inhaltliche" Analyse nicht ausreicht. Es genügt nicht, ein Datenfeld namens "Datierung" zu deklarieren und dann darauf aufbauend eine chronologische Sortierung zu fordern. Man muß dem Computer die Informationen so geben, daß er das gewünschte Produkt mit seinen Mitteln (und die sind bescheiden) erzeugen kann. Im Falle der chronologischen Sortierung braucht die Maschine also zusätzlich Jahreszahlen.

Fazit: Bei der Frage "Welche Arten von Textdaten gibt es?" geht es also um die Frage, was denn der Computer mit diesen Daten machen soll. Wir haben gesehen, daß ein enger Zusammenhang zwischen den formalen Eigenschaften und den Verarbeitungsmöglichkeiten besteht. In den folgenden Abschnitten gehen wir nun von den wichtigsten Arten dieser Datenverarbeitung aus und schauen uns an, welche Ansprüche sich daraus für die Formulierung der Daten ergeben.

Abb. 4: Übung zum Sortieren nach Hintergrundwissen		
Bitte setzen Sie die folgenden Datierungen in Jahreszahlen um, und zwar nach dem Schema <i>augusteisch</i> → 27 v. Chr. - 14 n. Chr. Und schauen Sie dabei bitte auf die Uhr!		
Datierung	von	bis
spätpaläolithisch		
kurz vor 480 v.Chr.		
augusteisch		
frühe Kaiserzeit		
karolingisch		
1230-50		
ca. 1450		
Mitte 15. Jh.		
oberrheinisch um 1520		
18. Jh.		
ca. 1830		
kurz vor 1900		
Jugendstil		
Frühjahr 1957		
1.3.1986		

Was kann ein "normales" Computerprogramm mit Textdaten machen ?

Ziel dieses Kapitels ist es zunächst, Ihnen eine anschauliche Vorstellung davon zu geben, was ein Computer überhaupt mit Textdaten machen kann. Es wird Sie nicht weiter verwundern, daß die Computerleute eine große Zahl von Tricks entwickelt haben, um bei der Säuberung von Daten bzw. bei der Benutzung nicht ganz sauberer Daten zu helfen.

Solche Hilfen spielen natürlich bei der im Museum meist wichtigen "retrospektiven" Datenerfassung (Benutzung bereits vorhandener, nicht für solche Zwecke erstellter Quellen) und wegen der aus diesen und anderen Gründen resultierenden großen Uneinheitlichkeit der Daten eine besonders große Rolle.

Bei der Beurteilung eines Systems sollten Sie vor Allem darauf achten, ob und wie weit der Hersteller klare (und ehrliche) Auskünfte darüber erteilt, wie die Daten formuliert sein müssen, damit die in dem betreffenden System zur Verfügung stehenden Methoden zuverlässig greifen.

Wiedergabe von Textdaten (Druck von Karteikarten, Katalogen usw.)

Unterschiede zwischen verschiedenen "Reportgeneratoren" (so nennt man Programme, die die Druckaufbereitung vornehmen) bestehen vor Allem in der Fähigkeit, das gewünschte Layout ohne Abstriche zu erzeugen¹². Hier können wir uns kurz fassen, denn es ist einem "Reportgenerator" im Prinzip völlig egal, wie die Informationen formuliert sind. Er behandelt sie alle als "freien Text". Ein paar Einschränkungen sind allerdings auch hierbei nötig:

- **Zeichensatz:** Sie sollten es nicht als gegeben annehmen, daß Ihr Computer alle für Sie wichtigen Zeichen vernünftig wiedergeben kann. Das gilt nicht nur für die zahlreichen Zeichen aus nationalen Alphabeten (wie z.B. Ç, ü, é, â, ä, à, å, ç, ê, ë, è, ì, î, ï, Ä, Å, É, æ, Æ, ô, ö, ò, û, ù, ÿ, Ö, Û, ø, £, ¥, P, f, á, í, ó, ú, ñ, Ñ usw.), auch ein "einfaches Umschalten" zwischen z.B. deutschen und griechischen Buchstaben ist zwar in manchen Textverarbeitungsprogrammen bereits möglich, bei Datenbanksystemen aber alles andere als selbstverständlich. Näheres dazu finden Sie im Anhang (Kapitel "Arbeitsblätter zum Zeichensatz", Seite 114 ff.).

Trotz vieler nationaler und internationaler Normen bzw. "De-facto-Normen" herrscht hier immer noch ein großes Durcheinander, denn mit einem "Byte" lassen sich nur 256 verschiedene Zeichen darstellen und das reicht oft nicht aus. Das gleiche Byte bedeutet in einem griechischen Computer etwas anderes als in einem deutschen. Eines Tages werden wir mit Computern arbeiten können, bei denen diese Einschränkungen entfallen; dann bekommen wir sicherlich auch internationale Normen für dieses heute immer noch außerordentlich knifflige Gebiet. Bis dahin muß man sich hier leider meistens einschränken.

- **Feldlänge:** Will man auf vorgedruckte Karteikarten oder Formulare drucken, muß man dafür sorgen, daß die Texte auch in die entsprechenden "Häuschen" passen. Bei Programmen, bei denen eine feste Feldlänge vorgegeben ist, ist das nicht weiter schwierig. Bei den an sich leistungsfähigeren Programmen, die solche Begrenzungen nicht brauchen, muß man darauf achten.
- **Reihenfolge:** Wünscht man, daß z.B. die Karteikarten in einer bestimmten Reihenfolge ausgedruckt werden, so muß das Programm "sortieren" können. Das wird im nächsten Abschnitt beschrieben.

Ein häufiges Mißverständnis muß hier noch sicherheitshalber angesprochen werden. Viele Leute verwechseln die Darstellung der Daten auf dem Bildschirm mit der Struktur der Datenbank bzw. den Ausgabeprodukten. Dieses Mißverständnis tritt besonders dann auf, wenn der Bildschirmaufbau dem der Karteikarte entspricht.

In manchen Fällen kann man diese Verwechslung daran erkennen, daß der Bearbeiter bei längeren Texten an den Zeilenenden Worttrennungen mit Bindestrichen eingetippt hat. In einem anderen Layout sieht das dann grauenvoll aus. Sie können sich diese Unterschiede leicht selber an ihrem Textverarbeitungsprogramm demonstrieren. Sie können den Inhalt eines Datenfeldes einer Datenbank genau wie einen Absatz in einem Textverarbeitungsprogramm nahezu beliebig "umformatieren" und vielleicht noch andere Schrifttypen oder -größen benutzen. Bitte beachten Sie, wie dämlich sich eine am Bildschirm (Abb. 5) vorgenommene Silbentrennung dann in anderen Layouts auswirkt (Abb. 6).

Bei der Wiedergabe von Textdaten müssen wir also nur auf wenige (und dazu leicht lernbare) Dinge achten. Falls Sie schon mit einem Textverarbeitungsprogramm gearbeitet haben, sind Ihnen diese Dinge vertraut.

12) **Pflichtlektüre:** Heft 30 der "Materialien" S. 44-46: Welche Möglichkeiten gibt es zur Herstellung von Druckausgaben?

Inventarnummer: M I 5
Objektbezeichnung: Schreibkabinett
Beschreibung: Der querrrechteckige Kasten zeigt außen unverzierte Holzwände. An den Schmalseiten ist jeweils ein Tragegriff angebracht. Hinter einer abschließbaren Schreibklappe befinden sich neun Schubkästen. Die Innenseite der Klappe, die Fronten der Schubladen und der Fächer sind mit Stabmosaik im Mudéjar-Stil furniert.

Abb. 5: Darstellung auf dem Bildschirm mit sinnloser Silbentrennung ("unver-zierte")

Beschreibung: Der querrrechteckige Kasten zeigt außen unver-zierte Holzwände. An den Schmalseiten ist jeweils ein Tragegriff angebracht. Hinter einer abschließbaren Schreibklappe befinden sich neun Schubkästen. Die Innenseite der Klappe, die Fronten der Schubladen und der Fächer sind mit Stabmosaik im Mudéjar-Stil furniert.

Der querrrechteckige Kasten zeigt außen unver-zierte Holzwände. An den Schmalseiten ist jeweils ein Tragegriff angebracht. Hinter einer abschließbaren Schreibklappe befinden sich neun Schubkästen. Die Innenseite der Klappe, die Fronten der Schubladen und der Fächer sind mit Stabmosaik im Mudéjar-Stil furniert.

Der querrrechteckige Kasten zeigt außen unver-zierte Holzwände. An den Schmalseiten ist jeweils ein Tragegriff angebracht. Hinter einer abschließbaren Schreibklappe befinden sich neun Schubkästen. Die Innenseite der Klappe, die Fronten der Schubladen und der Fächer sind mit Stabmosaik im Mudéjar-Stil furniert.

Abb. 6: Druck des Textes im Datenfeld "Beschreibung" in anderen Layouts: Die unnötige Trennung des Wortes "unver-zierte" rutscht bei einer anderen Zeilenbreite oder anderen Drucktypen leicht vom Ende der Zeile an eine andere Stelle.

Das Sortieren von Textdaten

Wir hatten bereits gesehen, daß "Sortieren" ein etwas problematischer und in vielen Fällen auch ein sehr weiter Begriff ist (s.o. Abb. 3 auf Seite 19), die meisten Sortiervorgänge lassen sich aber auf das zurückführen, was der Computer nun mal leisten kann und das sind grundsätzlich rein formalistische Vorgänge wie "alphanumerisches" (Buchstaben und Zahlen) oder "numerisches" (nur Zahlen) Sortieren. Auch für Dinge, bei denen eine alphabetische Sortierung völlig sinnlos ist (wie die oben besprochene chronologische Sortierung) müssen wir dem Computer zuerst irgendwie "sortierbare" Informationen (z.B. Jahreszahlen) zur Verfügung stellen.

Beim Sortieren werden wir eine Menge von Dingen lernen, die dann später für das Retrieval grundlegend sind, denn alle Methoden eines schnellen Retrieval beruhen letztlich auf einer internen, für den Benutzer unsichtbaren, Sortierung der Daten.

Wie sortiert ein Computer alphabetisch ?

Wir hörten schon bei der Diskussion der "sauberen" bzw. "schmutzigen" Daten (Seite 14 ff.), daß ein Computer Informationen nicht als Text oder als Zahl, sondern in einer internen Codierung speichert. Leider gibt es nicht nur eine, sondern mehrere solcher Codierungen oder Standards, die ihrerseits wiederum unzählige "Dialekte" haben.

Auf solche Fragen wird im Anhang detailliert eingegangen (Kapitel "Arbeitsblätter zum Zeichensatz", Seite 114 ff.), an dieser Stelle müssen Sie sich nur klarmachen, daß ein kleines und ein großes "A" oder ein Buchstabe mit bzw. ohne Akzent usw. nicht den gleichen internen Code haben und daß es Aufgabe der "Anwenderprogramme" ist, solche Dinge bei einer alphabetischen Sortierung in angemessener Weise zu berücksichtigen. Dies ist alles andere als einfach; wir werden im nächsten Kapitel an ein paar Beispielen besprechen, daß (und warum) es sehr verschiedene Ideen über eine richtige alphabetische Sortierung gibt.

Um der Sache näherzukommen, ist es gut sich klar zu machen, wie eine alphabetische Sortierung geschieht. Den meisten von uns ist diese Aufgabe viel zu vertraut, als daß wir sie noch genau beschreiben könnten. Grundlage aller Verfahren ist, daß man alle "Wörter" zuerst nach dem ersten, dann nach dem zweiten usw. Buchstaben in der Reihenfolge des Alphabets bzw. Zahlen nach ihrem Wert (0-9) sortiert.

Dieses sture spaltenweise Vorgehen kann bereits bei ganz einfachen Dingen (und ohne Störfaktoren wie Bindestriche) zu absurden Ergebnissen führen.

Beispiel: Wir nehmen uns als Beispiel ein paar aus Buchstaben und Zahlen zusammengesetzte "Wörter", z.B. Inventar- oder andere Nummern in der Reihenfolge A 1, A 2, A 10, A 20, A 100 und A 200. Gibt man diese Nummern einem einfachen Sortierprogramm, so entsteht das folgende Ergebnis (Abb 7):

	1	2	3	4	5	6	7	8	9	0	usw.
A	1										
A	1	0									
A	1	0	0								
A	2										
A	2	0									
A	2	0	0								

Abb. 7: Ein normales Sortierprogramm sortiert "spaltenweise"

In unserem Beispiel liegt der Fehler darin, daß ein normales Sortierprogramm im Computer ganz stur "spaltenweise" sortiert, d.h. zunächst nach dem Zeichen in der ersten Spalte ("A"), dann nach dem Zeichen in der zweiten Spalte (Leertaste), dann nach der dritten Spalte (erst "1", dann "2"), dann in der vierten Spalte (erst "nichts" bzw. eine "Leertaste", dann "0") usw. Das Programm unterscheidet also anders als wir nicht "automatisch" zwischen Zeichenarten, die wie hier Buchstaben und Zahlen bei einer Sortierung unterschiedlich behandelt werden sollten.

Das Vorkommen solch verschiedener Zeichenarten erklärt auch, warum in Abb. 2 auf Seite 16 ("Vergleich unvereinheitlichter und per Programm vereinheitlichter Daten") ein ganz "unwichtiges" Sonderzeichen wie ein Bindestrich die Sortierung stark beeinflussen kann. Da der Bindestrich eine andere ("kleinere") interne Codierung hat als die darauf folgenden Buchstaben und das Anwenderprogramm nicht "wußte", daß der Bindestrich bei einem solchen alphabetischen Register besser ignoriert werden sollte, entstand ein sehr unschönes Ergebnis.

Was da im einzelnen geschieht und was man machen kann, um solche Fehler zu vermeiden, wird im Anhang zum ersten Hauptteil im Kapitel "Inventarnummern und andere Nummerierungssysteme" (Unterkapitel "Was kann ein "normales" Computerprogramm mit Inventarnummern machen?" auf Seite 50 ff.) sehr ausführlich und mit vielen Abbildungen dargestellt. Wenn Sie sich für diese formalistischen Aspekte interessieren, sollten Sie sich das vor dem Weiterlesen anschauen.

Für normale Textdaten (Schlagwörter, Namen usw.) muß man sich zusätzlich die Frage stellen, wie eine gute alphabetische Sortierung aussieht. Wir werden gleich sehen, daß es da eine Menge unterschiedlicher und sich widersprechender Methoden gibt.

Was ist eigentlich eine richtige alphabetische Ordnung ?

Den meisten von uns ist es schon mal aufgefallen, daß ein Konversationslexikon und das Telefonbuch verschiedene "Sortierreihenfolgen" haben. Da sich die Problematik daran gut veranschaulichen läßt, konstruieren wir uns ein künstliches Beispiel, in dem die gleichen Einträge (historische und erfundene Personen und Institutionen) vorkommen (Abb. 8 auf Seite 25). Für das Lexikon nehmen wir uns zuerst den Meyer von 1905 als Vorbild, wir werden später sehen, daß das nicht ausreicht und daß wir da auch noch eine neuere Auflage brauchen, weil sich die Ideen darüber ganz offensichtlich mit der Zeit ändern.

Sucht man in diesem **Lexikon** (1905) nach *Goethe, Johann Wolfgang von*, so findet man ihn kurz **hinter** (aber noch auf der gleichen Seite mit) der *Gothaer Lebensversicherungs a.G.*, d.h. *oe* wird in diesem Fall (ebenso wie *ö*) wie ein einfaches *o* behandelt, für die Reihenfolge sind also die darauffolgenden Buchstaben (*tha* bzw. *the*) bestimmend. Im Berliner **Telefonbuch** von 1990 steht dagegen das *Goethe-Institut* (bzw. einige *Goethes* und *Göthes*) gut 13 Seiten **vor** der *Gothaer Lebensversicherungs a.G.*, weil das *ö* wie *oe* sortiert wird (die genauen Spielregeln findet man auf den ersten Seiten des Telefonbuchs unter dem Titel "Wie finde ich was im Telefonbuch?").

Schaut man sich die Sache dann genauer an, so entdeckt man eine Menge von "Ausnahmen". So findet man *Goes, Hugo van der* im alten Konversationslexikon (1905) zwischen *Godwin, Mary* und *Goffine, Leonhard*, da in diesem Fall das *oe* nicht als Umlaut, sondern als Doppelbuchstabe zu interpretieren ist. Im Telefonbuch wiederum werden Vornamen und Zusätze zu "Firmennamen" verschieden behandelt, wir finden also das *Goethe-Institut* nicht direkt hinter *Goethe, Johann Wolfgang von*, sondern erst hinter *Göthe, Zenzi*.

Die Ansetzung der Namen von "Körperschaften" gehört im übrigen zu den ganz besonders kniffligen Problemen¹³. Wir lernen daraus, daß sich Namen von Personen und von Körperschaften nur unter vielen Verrenkungen in ein gemeinsames Register (geschweige denn ein gemeinsames Datenfeld) bringen lassen. Wer etwas Erfahrung mit historischen oder aus anderen Kulturen stammenden Personennamen hat, wird bestätigen, daß solche Probleme auch bei diesen auftauchen.

Aber auch damit ist es noch nicht getan. In dem alten Lexikon wurde noch "von Hand" (und mit Verstand) sortiert, *Johann Wolfgang von Goethe* und das *Goethe-Institut* kommen **vor** *Goethe, Herrmann* bzw. *Rudolf*; *Godwin, Mary* kommt **nach** *Godwin, William*; dies liegt nicht an der Bedeutung des Dichters oder am immanenten Machismo der Lexikographen (Mary Godwin war eine frühe Frauenrechtlerin, Mutter von Mary Shelley, der Autorin des Frankenstein), sondern an einem ganz vernünftigen Prinzip: innerhalb des gleichen Namens gilt eine zeitliche Reihenfolge. Wir lernen hier zum ersten Mal ein Beispiel dafür kennen, daß verschiedene "Sortierkriterien" in einem Produkt kombiniert werden können.

Ganz anders sieht es bei einem neueren Lexikon aus. Im "Meyer" von 1974 wird bei *oe* nicht mehr unterschieden, ob das ein echter Doppelbuchstabe (*Goes*) oder ein Umlaut (*Goethe*) ist (also ganz genau so wie im Telefonbuch), aber noch immer wird das *ö* wie *o* behandelt. Zwischen *Goethe* und die *Gothaer Lebensversicherung*, die im alten Lexikon auf der gleichen Seite standen, schieben sich jetzt 75 Seiten ein:

13) Wer sich dafür interessiert, sollte sich das genauer anschauen: Regeln für die alphabetische Katalogisierung, Band 1: Regeln für wissenschaftliche Bibliotheken. Wiesbaden: Reichert, 1983, § 401-486.

Abb. 8: Alphabetische Sortierung nach verschiedenen Sortierregeln. Die Numerierung der beim Sortieren entstandenen Reihenfolge soll den Überblick erleichtern

Sortierung nach Lexikonregeln (1905)	Reihenfolge
Godwin, William	1
Godwin, Mary	2
Goes, Hugo van der	3
Goffine, Leonhard	4
Gothaer Lebensversicherungs a.G.	5
Goethe, Johann Wolfgang von	6
Goethe-Institut	7
Goethe, Herrmann	8
Goethe, Rudolf	9
Göthe, Zenzi	10
Sortierung nach Telefonbuchregeln	
Godwin, Mary	2
Godwin, William	1
Goes, Hugo van der	3
Goethe, Herrmann	8
Goethe, Johann Wolfgang von	6
Goethe, Rudolf	9
Göthe, Zenzi	10
Goethe-Institut	7
Goffine, Leonhard	4
Gothaer Lebensversicherungs a.G.	5
Sortierung nach Lexikonregeln (1974)	
Godwin, Mary	2
Godwin, William	1
Goes, Hugo van der	3
Goethe, Herrmann	8
Goethe, Johann Wolfgang von	6
Goethe, Rudolf	9
Goethe-Institut	7
Goffine, Leonhard	4
Gothaer Lebensversicherungs a.G.	5
Göthe, Zenzi	10

Dies ist natürlich außerordentlich verwirrend. Ein gewisser Trost besteht darin, daß die Art der Computer-Sortierung, wie sie für das Telefonbuch oder neuere Lexika eingesetzt wird, viel einfacher nachzuvollziehen ist als die durchaus vernünftige, aber vom Leser erhebliches Sprachwissen fordernde Regelung in einem älteren Lexikon.

Die Sache ist aber noch viel komplizierter, als es unser knappes Beispiel zeigt. Die "richtige" Sortierung ist zusätzlich auch noch eine höchst sprachspezifische Angelegenheit. In einem spanischen Lexikon z.B. wird *ll* als ein einziger Buchstabe behandelt und nicht nur anders ausgesprochen, sondern auch direkt vor *m* einsortiert, so daß ein Ortsregister die Reihenfolge

Lyon
Lluchmayor
Madrid

enthalten kann. Auch das häufige ñ steht nicht da, wo wir es erwarten würden. Der folgende Abschnitt ist direkt einem spanischen Wörterbuch entnommen.

manzana
mañana
mapa

Ich habe diese Beispiele mal spaßeshalber mit einem Textverarbeitungssystem (WORD) sortiert und dabei die folgenden Ergebnisse erhalten:

Luchmayor	manzana
Lyon	mañana
Madrid	mapa

In einem spanischen Lexikon wäre die erste Sortierung falsch, die zweite dagegen richtig. Nach deutschen Lexikonregeln (alt wie neu) hingegen ist es genau umgekehrt, d.h. das erste Beispiel ist richtig, das zweite falsch sortiert. Wenn Sie also Buchstaben aus anderen Alphabeten benutzen wollen, müssen Sie entscheiden, wie diese sortiert werden sollen¹⁴.

Diese wenigen Beispiele zeigen, daß eine "richtige alphabetische Sortierung" auch ohne Computer ein durchaus erklärungsbedürftiger Vorgang ist.

Alphabetische und systematische Register - wozu dienen sie eigentlich ?

Die beste Entscheidungshilfe bei solchen Fragen ist immer die Rückbesinnung auf die "Produkte" und das, was wir mit ihnen erreichen wollen.

Stellen wir uns also vor, wir wollten in unserem Museum alle **Schränke** suchen. Ein einfaches alphabetisches Register mit Verweisen auf die Inventarnummern (Abb. 9, Seite 27, linke Spalte) hilft uns da nicht viel weiter, es enthält nur wenige Einträge unter der Bezeichnung Schrank (sie werden in der Abbildung durch Fettdruck hervorgehoben).

Etwas besser geht das bereits in einem "permutierten" Register (Abb. 9, rechte Spalte), da finden wir (hier wiederum durch Fettdruck markiert) eine ganze Menge verschiedener Schränke.

Was ist ein permutiertes Register ? In einem permutierten Register erscheinen die Schlagwörter zunächst genau so, wie sie aufgeschrieben wurden. In Abb. 9 z.B. finden Sie den "Schreibschrank" zwischen "Schreibpult (Notariats-)" und "Schreibtischset". Zusätzlich erscheinen sie aber auch "permutiert", d.h. man hat dafür gesorgt, daß wichtige Wortelemente wie "Schrank" nach "vorne" kommen (z.B. "Schrank (Schreib-)") und dadurch bei einer normalen alphabetischen Sortierung dicht beieinander stehen.

Diese Art der Permutierung ist besonders im Deutschen mit seiner Vorliebe für zusammengesetzte Wörter eine außerordentlich große Hilfe. Sie wird auch außerhalb des Computers eingesetzt und kann durchaus auch "von Hand" erfolgen¹⁵. Ein solches Register ist auch eine gute Vorbereitung für systematische Arbeiten.

Eine **computergestützte Permutierung** hat viele Vorteile, denn jeder vorkommende Begriff braucht nur einmal permutiert zu werden. Sie erfordert allerdings auch die strikte Einhaltung von geeigneten Schreibanweisungen für das betreffende Datenfeld¹⁶.

14) Vgl. die "Arbeitsblätter zum Zeichensatz" im Anhang auf Seite 114 ff.

15) Vgl. die alphabetischen Register in Walter Trachsler, Systematik kulturhistorischer Sachgüter, Bern-Stuttgart 1981.

16) Ein Beispiel eines für Objektbezeichnungen geeigneten Regelwerks zur computergestützten Permutierung finden Sie in: Carlos Saro und Christof Wolters, EDV-gestützte Bestandserschließung in kleinen und mittleren Museen, Heft 24 der Materialien aus dem Institut für Museumskunde, Anhang S. 77-82.

Einfaches alphabetisches Register		Permutiertes Register	
Schnapshund	R03 K 64/100-101	Schottelplack	R03 K 71/67 b
Schnecke	R03 N 72/66 a-d	Schränkchen	R08 K 14
Schneckengehäuse	R03 E 72/181	Schränkchen (Geschirr-)	W36 78/839
Schneidemaschine (Tabak)	W01 78/198	Schränkchen (Nacht-)	W36 78/1642
Schneiderbügeleisen	W01 78/763	Schränkchen (Spielzeug-)	R03 K 64/406
Schneider-Elle	R03 J 64/578	Schrank	R08 K 12
	R03 J 64/579		W36 78/194
Schnelle	R03 K 65/103		W36 78/213
	R03 K 65/113		W36 78/824
	R03 K 65/116		W36 78/1283
	R03 K 65/119		W36 78/1630-1631
Schnibbelbohnenmaschine	W01 78/86		W36 78/1643
Schnitteisen	W36 78/619		W36 78/1801
	W36 78/621	Schrank (Baldachin-Paramenten-)	W36 78/1525
Schnupftabakdose	R03 K 64/629	Schrank (Chorbuch-)	R03 K 65/479
	R03 K 64/668	Schrank (Eck-)	R08 K 13
	R03 K 64/672		W36 78/215
	R03 K 65/410 a		W36 78/376
	R03 K 65/410 b		W36 78/438
	R03 K 70/312		W36 78/819
	R03 K 70/313		W36 78/1489
	R03 K 70/314	Schrank (Eichen-)	W36 78/234
	R03 K 70/315	Schrank (Einbau-)	W36 78/1264
	R03 K 70/316	Schrank (Glas-)	W01 78/127
Schnupftabaksdose	W36 78/151		W01 78/819
Schöpferstand	R08 HP 144		W36 78/164
Schöpfkelle	R08 K 116-117		W36 78/1232
	W36 78/709	Schrank (Holländischer Leinen-)	W36 78/848
Schöpföffel	W36 78/254	Schrank (Kasten-)	W36 78/820
Schöpfsieb	R08 HP 145	Schrank (Kleider-)	W01 79/206
	R08 HP 146		W36 78/1092
Schottelplack	R03 K 71/67 b		W36 78/1117
Schränkchen	R08 K 14		W36 78/1666
Schrank	R08 K 12	Schrank (Küchen-)	W36 78/1557
	W36 78/194	Schrank (Leinen-)	W36 78/1525
	W36 78/213	Schrank (Puppenstuben-Wäsche-)	R03 K 72/207
	W36 78/824	Schrank (Schreib-)	W36 78/365
	W36 78/1283	Schrank (Stern-)	W36 78/848
	W36 78/1630-1631	Schrank (Unter-)	W36 78/1258
	W36 78/1643	Schrank (Vitrinen-)	R03 K 64/292
	W36 78/1801		W36 78/718
Schrapper	W01 78/315		W36 78/1410
	W01 78/316	Schrank (Vitrinen-Eck-)	R03 K 64/399
	W01 78/317	Schrank (Wäsche-)	R08 K 11
	W01 78/318		W36 78/914
	W01 78/321 a	Schrank-Feston (Wäsche-)	R03 K 70/241 a-e
	W01 78/323 a	Schrapper	W01 78/315
	W01 79/101		W01 78/316
Schraubenzieher	W36 78/610 a-b		W01 78/317
Schreibmaschine	W01 79/93		W01 78/318
Schreibschrank	W36 78/365		W01 78/321 a
Schreibtischset	W36 78/1418		W01 78/323 a
Schreibzeug	W01 78/786-788		W01 79/101
Schreibzeughälter	R03 K 66/237 a-b	Schrapper (Riffel-)	W01 78/320 a
Schreiner-Hammer	R08 HH 112	Schraubenzieher	W36 78/610 a-b
Schröpfglas	R03 H 64/501 a-b	Schreiber (Kugel-)	R03 K 76/206 a-b
	R03 H 64/531	Schreibgerät (Reise-)	R03 K 70/269
Schröpfschnepper	R03 H 64/532 a-b	Schreibgerät (Steinzeug-)	R03 K 64/86-87
	R03 H 64/533 a-b		R03 K 64/234
Schublade	W36 78/761	Schreibmaschine	W01 79/93
Schüssel	R03 K 65/47	Schreibpult (Notariats-)	R08 K 16
	R03 K 65/48	Schreibschrank	W36 78/365
	R03 K 65/49	Schreibtischset	W36 78/1418
	R03 K 65/50	Schreibzeug	W01 78/786-788

Abb. 9: Suchen nach "Schrank" in einem einfachen (linke Spalte) und in einem permutierten Register (rechte Spalte)

Inneneinrichtung	
Möbel	
Regal	R08 K 43 Topfbank/Regal M.19.Jh. Material: Buchenholz
Eichen-Eck-Wandbrett	R08 K 42 Eichen-Eck-Wandbrett E.18.Jh.
Gewürzbord	W36 78/578 Gewürzbord mit 2 Laden Foto: 25/30
Gläserbord	W36 78/1467 Foto: 75/8
Tellerregal	W36 78/414 Tellerregal mit Löffelhalter Foto: 30/30
Wandbord	R03 K 64/289 Kleines zweizoniges Wandbord 19./20.Jh. W36 78/355 Wandbord/Handtuchhalter Foto: 17/30
Wandbretthalter	W36 78/386 4 Wandbretthalter Foto: 21/32
Schrank	R08 K 12 Schrank 18.Jh. Material: Eichenholz W36 78/194 Foto: 13/11 W36 78/213 Foto: 13/12 W36 78/824 Schrankfront Foto: 90/25 W36 78/1283 Schrank der Familie von Lilien Foto: 83/36 W36 78/1630-1631 Schrank, bestehend aus Unter- (1630) und Oberteil (1631) Foto: 98/11 Foto: 98/12 W36 78/1643 Schrankteile Foto: 96/27 W36 78/1801 Schrank mit zwei Türen; 1819 Foto: SB
Baldachin-Paramentenschrank	W36 78/1525 Leinenschrank/Baldachin-Paramentenschrank; 1786 Foto: 71/1
Geschirrschränkchen	W36 78/839 Foto: 61/17
Glazen Schapp	W01 78/127 Glasschrank, glazen Schapp Foto: 78/952
Glasschrank	W01 78/127 Glasschrank, glazen Schapp Foto: 78/952 W01 78/819 Foto: 78/969 W36 78/164 Foto: 32/13 W36 78/1232 Foto: 90/35
Glasvitrine	W36 78/1246 Foto: 79/35
Holländischer Leinenschrank	W36 78/848 Holländischer Leinenschrank/Sternschrank Foto: 87/34
Kannenstock	W36 78/674 Kannenstock, 1.H.18.Jh. Foto: 42/10
Kastenschrank	W36 78/820 Foto: 55/32
Kleiderschrank	W01 79/206 Kein Foto W36 78/1092 Foto: 63/2 W36 78/1117 Foto: 86/12 W36 78/1666 Foto: 96/27
Küchenschrank	W36 78/1557 Foto: 90/29
Leinenschrank	W36 78/1525 Leinenschrank/Baldachin-Paramentenschrank; 1786 Foto: 71/1
Nachtschränkchen	W36 78/1642 2 Nachtschränkchen Foto: 93/10
Schränkchen	R08 K 14 Schränkchen 2.H.19.Jh. Material: Eichenholz
Schreibschrank	W36 78/365 Schreibschrank, bäuerlich Foto: 12/9
Sternschrank	W36 78/848 Holländischer Leinenschrank/Sternschrank Foto: 87/34
Unterschrank	W36 78/1258 Foto: 73/10
Vitrinen-Eckschrank	R03 K 64/399 Vitrinen-Eckschrank (Eiche) um 1780 Foto: LBR 16/3735/2 Material: Eiche
Vitrinenschrank	R03 K 64/292 Kleiner Vitrinenschrank (Puppenhaus) 2.H.18.Jh. Foto: LBR 16/3720/3 W36 78/718 Foto: 42/3 W36 78/1410 Vitrinenschrank mit Aufsatz Foto: 75/3
Wäsche-Schrank	R08 K 11 Wäsche-Schrank M.18.Jh. Foto: LBR 28/15003 Material: Eichenholz
Sitzmöbel	
Armsessel	W36 78/737 Foto: 32/36
Armstuhl	R08 K 26 Armstuhl 1780 Foto: LBR 28/15008 Material: Eichenholz R08 K 27 Gepolsterter Armstuhl um 1840 Material: Mahagoni
Bogenstuhl	R08 HP 147 Bogenstuhl Nzt. Foto: LBR 28/15213
Filzenstuhl	R08 HP 149 Filzenstuhl Nzt. Foto: LBR 28/15216
Friesenstuhl	W36 78/359 Foto: 15/19
Fußbänkchen	W36 78/796 2 Fußbänkchen Foto: 28/31
Fußbank	W36 78/303 Foto: 16/30
Gautschstuhl	R08 HP 148 Gautschstuhl Nzt. Foto: LBR 28/15217
Gebetsbank	W36 78/1114 Foto: 63/8
Hängestuhl	R08 HH 89 Hängestuhl 1.H.20.Jh. Foto: LBR 28/15177 Material: Eisen Material: Holz
Jugendstilsofa	W36 78/1449-1450 Jugendstilsofa (1449) mit 2 Polstersesseln (1450) Foto: 84/28 Foto: 84/38
Kindersessel	R03 K 71/152 Kindersessel mit Flechtsitz 2.H.19.Jh. Foto: LBR 16/3832/4
Kinderstuhl	W36 78/795 Foto: 28/30
Kirschholzstuhl	R08 K 30 Kirschholzstuhl M.19.Jh.
Korblehnstuhl	W36 78/1108 Foto: 88/23

Abb. 10: Suchen nach "Schrank": Ausschnitt aus einem systematischen Register

Noch leichter wird die Suche durch ein "Systematisches Register" (Abb. 10, Seite 28). Dort wird man von den Wörtern völlig unabhängig, es spielt nun keine Rolle mehr, ob die Objektbezeichnung die Zeichenfolge *Schrank* enthielt; unter der Sachgruppe **Schrank** können anders gebildete Begriffe erscheinen (z.B. *Glasen Schapp*, *Glasvitrine*, *Kannenstock*), auch die Umlaute (*Schränkchen*) stören nicht mehr.

Die verschiedenen Wege, solch ein systematisches Register per Programm zu erzeugen, werden weiter unten im Zusammenhang mit dem Arbeiten mit Hilfsdateien beschrieben (Seite 33 ff.). Die Maschine tut auch dies mit Hilfe einer alphabetischen Sortierung, das Endprodukt weist aber darüber hinaus.

Zusammenfassung

Kleine Ursachen haben große Wirkungen. Sie können jetzt vielleicht schon etwas besser erkennen, was man alles zur Definition eines Registers braucht: Man muß wissen, wie ein Computer sortieren kann, man sollte eine angemessene Sortierreihenfolge der Zeichen festlegen, sich Gedanken über etwaige zusätzliche Sortierkriterien machen, für zusammengesetzte Begriffe vielleicht sogar eine Permutierung ins Auge fassen.

Die Gegenüberstellung von drei Produkten (Abb. 9-10), die alle aufgrund der gleichen Objektdaten (von vier Museen aus dem Projekt "Kleine Museen") erzeugt wurden, bildet den Abschluß des Kapitels "Sortieren von Textdaten" und den Übergang zum "Retrieval".

Bevor Sie allerdings zum nächsten Kapitel übergehen, empfehle ich zur Kontrolle des zur Sortierung Gelernten die folgende Übung:

Abb. 11: Übung zum Sortieren von Textdaten

Bitte nehmen Sie sich ein paar Lexika (z.B. Wörterbücher und Standardwerke Ihres Faches) vor und versuchen Sie, die darin verwendeten Sortierkriterien zu ermitteln (oft werden diese in "Hinweisen für die Benutzung" ausführlich erläutert) und ggf. zu begründen. Sie werden dabei noch eine ganze Reihe anderer Regeln (und Ausnahmen) finden.

Bitte schreiben Sie sich dabei eine Reihe von Begriffen auf, mit deren Hilfe Sie die **Sortierfähigkeiten eines Computerprogramms testen** können. Ein paar Anregungen dazu, was eine solche Liste enthalten sollte:

- Schreibweise mit und ohne **Bindestrich** (vgl. Abb. 2);
- **Zusammengesetzte Wörter**, bei denen sich die Schreibweise bei Benutzung eines Bindestrichs ändert (oder auch nicht) wie z.B. bei *Bettuch* (2 t), *Bett-Tisch* (3 t), *Betttruhe* (3 t); *Bett-Truhe* (3 t) usw.;
- **Umlaute** (sowohl am Anfang wie auch innerhalb eines Wortes) und ß;
- **Akzente** (é, è, â, á, à, ç);
- **Sonderzeichen** (d.h. Zeichen wie „(.)-;“, die ebenfalls eine Sortierung verändern können.

Sie werden dabei merken, daß Sie zwischen diese "Problemfälle" sorgfältig ausgewählte "normale" Schlagwörter stecken müssen, um die Unterschiede zwischen verschiedenen Sortierweisen zu erkennen.

Falls Sie bereits mit der "Textverarbeitung" vertraut sind, können Sie manche dieser Dinge damit ausprobieren (und damit gleich die Sortierregeln eines bestimmten Programms ermitteln).

Das Suchen von Textdaten (Retrieval)

Das Retrieval, das Suchen (aber nicht unbedingt Finden) steht auch im Museum bei den meisten Überlegungen zum Computereinsatz im Vordergrund. Für diesen "schnellen Zugriff auf alle relevanten Informationen" wurden viele Methoden entwickelt¹⁷, die Diskussion über Vor- und Nachteile bestimmter Verfahren ist nahezu unübersehbar.

Da das Retrieval kaum zusätzliche Anforderungen an die Formulierung der Daten stellt, beschränken wir uns in diesem Kapitel auf ganz wenige, aber grundlegende Dinge, die meisten wurden bereits erwähnt und erfahren hier nur noch eine Zusammenfassung.

Auch beim Retrieval kann man (allerdings nicht bei allen Programmen) das Ergebnis anschließend drucken.

Volltextrecherche

Mit dem Begriff "Volltextrecherche" ist gemeint, daß ein Programm die Daten nach bestimmten Zeichenfolgen ("strings"), also Wörtern oder Wortteilen durchsuchen kann. Dieses Verfahren benötigt eine Menge Rechenzeit und ist daher prinzipiell nur bei kleineren Datenmengen sinnvoll.

Abgesehen von solchen rein technischen Grenzen müssen wir uns aber auch fragen, wofür eine solche Recherche sinnvoll einzusetzen wäre. Dazu zunächst ein Beispiel (Abb. 12):

Abb. 12: Ergebnisse einer Volltextrecherche

Wenn Sie in unseren Testdaten aus dem Projekt "Kleine Museen" nach der Zeichenfolge "Schuh" suchen, bekommen Sie:

Damenhandschuh
Filet-Damenhandschuhe
Handschuh
Hemmschuh
Herrenlederhandschuh
Holzschuh
Holzschuhbank
Messing-Holzschuh
Messing-Schuhanzieher
Moorschuh
Porzellan-Frauenschuh
Puppenschuhe
Schuh
Schuhcremedose
Schuhkantenschneider
Schuhleisten
Schuhschnalle
Schuhsohle
Tierknochen-Schlittschuhe
Tischuhr
Wachs-Handschuhpaar

Man muß sich regelrecht Mühe geben, darunter auch "richtige" *Schuhe* zu finden.

¹⁷ **Pflichtlektüre:** Eine Übersicht über die vielen Tricks beim Retrieval gibt Heft 30 S. 41-44: "Welche Abfrage- bzw. Retrievalmöglichkeiten sind vorgesehen?"

Obwohl viele Leute das behaupten, wird die Sache auch dann nicht besser, wenn man nur Wörter sucht, die mit "Schuh" beginnen bzw. auf "Schuh" oder "Schuhe" enden (sogenanntes "Truncieren"). Sie können das an dem eben gegebenen Beispiel leicht selber nachprüfen.

Was Sie bei der Suche nach "Schuh" nicht finden sind Bezeichnungen wie *Holzklumpen*, *Lederstiefel*, *Schaftstiefel*, *Stiefel*, *Wasserstiefel*. Bei den als Beispiel dienenden 4 Museen gibt es nicht viele Schuhe, ein Ausschnitt aus dem systematischen Register zeigt nur fünf "Schuhe" (wobei der letzte auf mangelnde Beachtung des Homonymenproblems zurückzuführen ist und hier daher durchgestrichen wird):

Abb. 13: Schuhe im systematischen Register (oder Retrieval)

Bekleidung

- . Schuhe
- . . Holzklumpen R03 H 73/81 Holzklumpen zum Torf- und Tonstechen 2.H.19.Jh.
- . . Holzschuh W01 78/305 Foto: 78/207
- . . Holzschuh W01 78/306 Paar Holzschuhe Foto: 78/205
- . . Holzschuh W01 78/307-308 Holzschuh (307) mit Holzschuhbank (308)
- . . Holzschuh Foto: 78/945
- ~~.....Schuh.....W36 78/1767-Holzkrücke mit Schuh und Schiene-Foto: 99/22~~

Die Effizienz einer solchen Volltextrecherche ist stark sprachabhängig. Sprachen wie das Englische, in dem zusammengesetzte Begriffe nur selten zu einem "neuen" Wort zusammengesetzt werden und in dem diese Wörter durch die Zusammensetzung auch nicht verändert werden (keine "Fugenlaute" wie das "s" in *Bartmannskrug*), sind für eine Volltextrecherche viel besser geeignet als das Deutsche.

Wörter wie "Schuh" zeigen ganz deutlich die Möglichkeiten und Grenzen der Volltextrecherche in der deutschen Sprache¹⁸. Bei der Vorbereitung eines Wortschatzes für z.B. systematische Register ist es angenehm, mit einem einzigen Befehl alle Begriffe, die Zeichenfolgen wie *Schrank*, *Tisch*, *Uhr* oder dergleichen enthalten, zusammengestellt zu bekommen. Daß das dann nur zum Teil auch wirkliche *Schränke*, *Tische*, oder *Uhren* sind, spielt dabei eine nur untergeordnete Rolle. Für ein vernünftiges Retrieval kann dieses Verfahren aber nur eingesetzt werden, wenn der Wortschatz vorher einhundertprozentig vereinheitlicht wurde, ein Fall, bei dem man dann ebensogut auf die Volltextrecherche verzichten kann.

Suchen "im Dialog"

Immer wenn (z.B. bei der Vorführung eines Programmes) die Suchergebnisse blitzartig auf dem Bildschirm erscheinen und Sie dann mit dem Kopfe nicken ("genau das brauchen wir"), sollten Sie sich, bevor Sie das Scheckbuch zücken, kurz an das erinnern, was wir in den vorigen Kapiteln gelernt haben.

Was bei diesem "Dialog-Retrieval" in der Maschine abläuft entspricht nämlich im großen ganzen den Vorgängen, die wir schon oben bei den alphabetischen und systematischen Registern besprochen haben. Auch der Computer sucht seine Schlagwörter in alphabetischen oder systematischen Registern. Diese "invertierten" Register werden in vielen Programmsystemen automatisch aufgebaut und nach Änderungen in den Daten auch "aktualisiert". Damit sind aber die Probleme, die wir besprochen haben, noch lange (und vor Allem nicht "automatisch") gelöst.

18) Weitere Beispiele in Heft 30 S. 9-11: "Was versteht der Computer von der deutschen Sprache?"

Beim Suchen im Dialog läßt sich also nichts machen, was man nicht auch in solchen Registern machen könnte. Leider enthebt uns ein solches Programm also nicht der Aufgabe, unsere Daten vorher richtig zu formulieren und damit in die vom Programm "erwartete" Ordnung zu bringen.

Wenn Sie ein Retrieval-Programm auf Herz und Nieren prüfen wollen, dann nehmen Sie die Begriffsliste, die Sie in der Übung auf Seite 29 erarbeitet haben und tippen die Begriffe in ein dafür geeignetes Feld der Datenbank ein. Sie werden erstaunt sein, was Sie dann alles nicht wiederfinden.

Das heißt dann allerdings nicht, daß das Programm schlecht wäre. Sie müssen halt nur Vor- und Nachteile sorgfältig gegeneinander abwägen.

Welche Informationen gehören zum Objekt ? Das Arbeiten mit Hilfsdateien

Schauen wir uns die Daten zu einem Objekt genauer an, so werden wir oft genug feststellen, daß manche Daten eigentlich keine Aussagen zum Objekt, sondern nur Aussagen zu bestimmten Attributen enthalten.

Allgemeine Feststellungen wie z.B. die, daß *Rembrandt* von 1606-1669 lebte, eine *Taschenuhr* eine *Uhr* ist usw. sind zwar grundlegend für manche Arten von Retrieval oder systematischer Register, müssen aber deswegen nicht bei jedem Blatt des Meisters oder bei jeder *Taschenuhr* wiederholt werden. Dies führt uns zu dem schon öfter erwähnten Konzept der "Hilfsdateien", deren Vorteile vor Allem in einem Gewinn an Flexibilität und - oft noch wichtiger - in erheblichen Kosteneinsparungen bei der Datenerfassung liegen.

Ein weiterer (und oft übersehener) Aspekt ist, daß das Arbeiten mit Hilfsdateien dramatische Rückwirkungen auf die Datenstruktur haben kann. Diese Wirkungen gehen angenehmerweise grundsätzlich in Richtung einer Vereinfachung.

Statt langer Erläuterungen ein Beispiel: Erzeugung systematischer Register

Eine ganze Reihe der oben bereits gegebenen Beispiele für alphabetische und systematische Register (z.B. Abb. 2 auf Seite 16, rechte Spalte; Abb. 9 auf Seite 27 und Abb. 10 auf Seite 28) wurde mit Hilfe von "Hilfsdateien" erzeugt, ohne daß wir das näher erklärt hätten. Solche Hilfsdateien gibt es für sehr viele und - oberflächlich betrachtet - auch ganz unterschiedliche Zwecke. Bevor wir uns aber in eine Diskussion über so komplizierte Dinge wie Thesauri und dergleichen stürzen, soll zunächst gezeigt werden, was bei der Erzeugung eines systematischen Registers im Computer eigentlich vor sich geht.

Wir nehmen als Beispiel wieder das systematische Register in Abb. 10 auf Seite 28. Damit man das erzeugen kann, muß das Programm irgendwie erfahren, daß ein *Kleiderschrank* ein *Schrank* ist und dieser wiederum ein *Möbel*. Dafür gibt es zwei grundsätzlich verschiedene Methoden, die beide ihre Vor- und Nachteile haben:

- **1. Methode:** Man schreibt die immer gleiche Abfolge von diesen "Oberbegriffen" bei jedem einzelnen Objekt immer wieder neu auf. Um das zu veranschaulichen, bilden wir Ausschnitte aus ein paar dafür geeigneten Erfassungsbögen ab (Abb. 14 auf Seite 34 mit den uns schon bekannten Möbeln aus Abb. 10).

Dort wurden jeweils eigene Datenfelder (Klasse, Ordnung, Gattung und Art) für vier hierarchische Ebenen eingerichtet. Man kann das natürlich auch anders machen (vgl. die Erfassung geographischer Daten in Abb. 35 auf Seite 70), Vor- und Nachteile zwischen solchen Variationen stehen aber hier (noch) nicht zur Diskussion.

In jedem Falle sind nun alle für die Sortierung erforderlichen Informationen auf den betreffenden "Karteikarten" vollständig versammelt.

- **2. Methode:** Man schreibt solche Begriffshierarchien nicht bei jedem einzelnen Objekt, sondern nur einmal auf, dann allerdings in eine andere (z.B. Thesaurus-) Datei (Abb. 15, Seite 35 oberer Teil). Auf der "Karteikarte" zum Objekt braucht dann nur noch die Objektbezeichnung ohne Oberbegriffe zu stehen (Abb. 15 unterer Teil), sie dient als Verweis auf die Hilfsdatei.

Zur Erzeugung eines systematischen Registers wie in Abb. 10, Seite 28 muß ich nun bei beiden Methoden grundsätzlich drei Dinge tun, die wir uns kurz anschauen wollen, um ein besseres Gefühl für die Abwägung von Vor- und Nachteilen der beiden Methoden zu bekommen:

Inventar-Nr.: R08 K 27
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Sitzmöbel Art: Armstuhl

Inventar-Nr.: R03 K 64/292
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Schrank Art: Vitrinenschrank

Inventar-Nr.: W36 78/1092
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Schrank Art: Kleiderschrank

Inventar-Nr.: W36 78/848
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Schrank Art: Holländischer Leinenschrank

Inventar-Nr.: W36 78/1246
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Glasvitrine Art: Glasvitrine

Inventar-Nr.: W01 78/127
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Schrank Art: Glasschrank

Inventar-Nr.: R08 K 12
Klasse: Inneneinrichtung Ordnung: Möbel Gattung: Schrank Art:
Schrank, 18. Jh., Eichenholz

Abb. 14: Hintergrundwissen zur Erzeugung systematischer Register. 1. Methode: Die Oberbegriffe werden auf jeder einzelnen Karteikarte vollständig aufgeführt

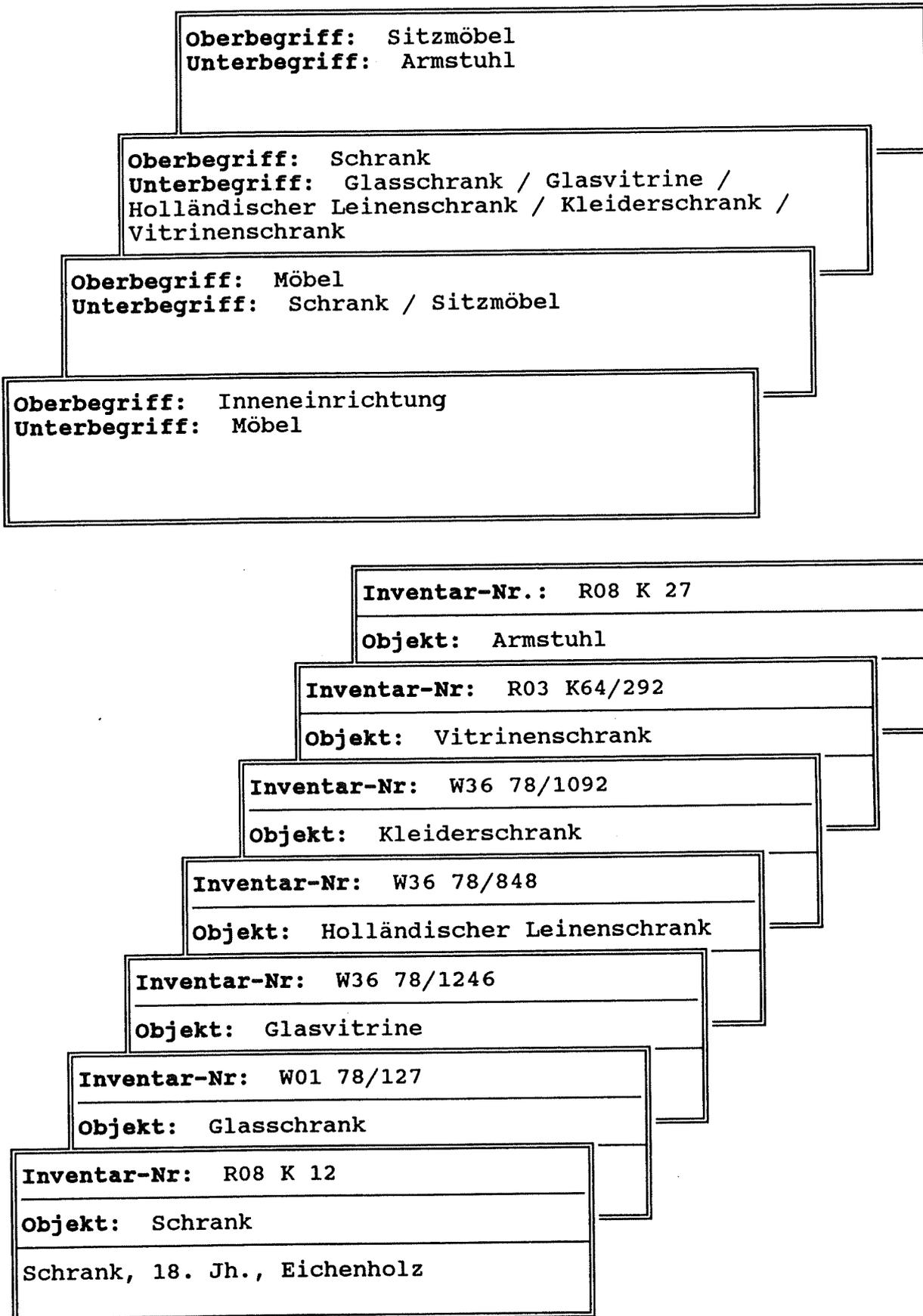


Abb. 15: Hintergrundwissen zur Erzeugung systematischer Register. 2. Methode: Eine leicht zu verändernde und zu ergänzende Datei mit Ober- und Unterbegriffen (oberer Teil) erspart die entsprechenden Angaben in der Objektdatei (unterer Teil)

Zunächst erzeugt man eine "Sortierdatei". Um die Begriffe in der hierarchisch richtigen Reihenfolge sortieren zu können, wird in dieser Datei für jede hierarchische Ebene eine ausreichend breite Spalte eingerichtet. Bei der ersten Methode enthalten diese Spalten also Klasse, Ordnung, Gattung und Art (s. Abb. 16). Bei der zweiten Methode werden die Spalteneinträge durch Verkettung der jeweiligen Oberbegriffe in die richtige Reihenfolge gebracht.

Abb. 16: Sortierdatei vor dem Sortieren (in der Reihenfolge der Inventarnummern)				
Klasse	Ordnung	Gattung	Art	Inventar-Nummer
Inneneinrichtung	Möbel	Schrank	Vitrinenschrank	R03 K 64/292
Inneneinrichtung	Möbel	Schrank		R08 K 12
Inneneinrichtung	Möbel	Sitzmöbel	Armstuhl	R08 K 27
Inneneinrichtung	Möbel	Schrank	Glasschrank	W01 78/127
Inneneinrichtung	Möbel	Schrank	Holländischer Leinenschrank	W36 78/848
Inneneinrichtung	Möbel	Schrank	Kleiderschrank	W36 78/1092
Inneneinrichtung	Möbel	Schrank	Glasvitrine	W36 78/1246

Die hier abgebildete Sortierdatei ist - nebenbei gesagt - gleichzeitig ein gutes Beispiel für eine Sortierung nach verschiedenen "Sortierkriterien" (s.o. die Reihenfolge der Einträge in einem alten Lexikon, s. Kapitel "Was ist eigentlich eine richtige alphabetische Ordnung?", Seite 24 mit Abb. 8). Sie können sich z.B. vorstellen, daß man zwischen "Art" und "Inventar-Nummer" noch die Datierung geschoben hätte, um innerhalb der gleichen Möbel eine chronologische Reihenfolge zu erzeugen.

Diese Sortierdatei wird dann ganz normal, d.h. "spaltenweise" sortiert (bei den Inventarnummern sind zusätzliche Regelungen erforderlich, die werden im Kapitel "Inventarnummern und andere Numerierungssysteme" auf Seite 48 ff. besprochen). Dies ergibt etwa das folgende Resultat (Abb. 17):

Abb. 17: Sortierdatei nach dem Sortieren				
Inneneinrichtung	Möbel	Schrank		R08 K 12
Inneneinrichtung	Möbel	Schrank	Glasschrank	W01 78/127
Inneneinrichtung	Möbel	Schrank	Glasvitrine	W36 78/1246
Inneneinrichtung	Möbel	Schrank	Holländischer Leinenschrank	W36 78/848
Inneneinrichtung	Möbel	Schrank	Kleiderschrank	W36 78/1092
Inneneinrichtung	Möbel	Schrank	Vitrinenschrank	R03 K 64/292
Inneneinrichtung	Möbel	Sitzmöbel	Armstuhl	R08 K 27

Sie können jetzt bereits erkennen, wie man durch eine geschickte Festlegung der Sortierreihenfolge durch alphabetisches Sortieren zu einem systematischen Register kommen kann.

In einem dritten Schritt wird diese korrekt sortierte, aber noch sehr unschön aussehende Datei für den Druck aufbereitet. In Abb. 18 auf Seite 37 sehen Sie eine relativ einfach zu erzeugende Druckaufbereitung (Unterdrücken ständig wiederholter Begriffe), mit einiger Mühe kann man das auch wie in Abb. 10 auf Seite 28 drucken.

Bei der zweiten Methode (Benutzung einer Hilfsdatei) gibt es mehrere mögliche Wege zur Erzeugung eines systematischen Registers wie in Abb. 10. Um die Erklärungen nicht zusätzlich zu komplizieren, nehmen wir an, daß wir auch hier die drei oben beschriebenen Schritte tun. Der grundlegende Unterschied zwischen den beiden Methoden besteht dann darin, daß

wir die Sortierdatei aus zwei Dateien zusammenstellen: für jede Objektbezeichnung werden die jeweiligen Oberbegriffe aus der Thesaurusdatei herausgesucht. Dieser kleine "technische" Unterschied kann dramatische Verbesserungen bei der Fehleranfälligkeit und damit den Kosten des Systems bewirken.

Abb. 18: Sortierte Datei nach einer einfachen Druckaufbereitung

Inneneinrichtung	Möbel	Schrank	R08 K 12
		Glasschrank	W01 78/127
		Glasvitrine	W36 78/1246
		Holländischer Leinenschrank	W36 78/848
		Kleiderschrank	W36 78/1092
		Vitrinenschrank	R03 K 64/292
	Sitzmöbel	Armstuhl	R08 K 27

Fehleranfälligkeit: Wir veranschaulichen das an einem kleinen "Schreibfehler". Stellen wir uns vor, wir hätten bei dem *Schrank* mit der Inventarnummer *R08 K 12* einen winzigen Fehler gemacht und bei der Klasse *Inneneinrichtung* (mit drei n) statt *Inneneinrichtung* geschrieben. Das Ergebnis der Sortierung sähe dann folgendermaßen aus (Abb. 19):

Abb. 19: Sortierte Datei mit Schreibfehler nach einer einfachen Druckaufbereitung

Inneneinrichtung	Möbel	Schrank	Glasschrank	W01 78/127
			Glasvitrine	W36 78/1246
			Holländischer Leinenschrank	W36 78/848
			Kleiderschrank	W36 78/1092
			Vitrinenschrank	R03 K 64/292
		Sitzmöbel	Armstuhl	R08 K 27
Inneneinrichtung	Möbel	Schrank		R08 K 12

Wenn es sich um einen solchen kleinen, nur bei einem Objekt vorkommenden Schreibfehler handelt, ist so ein Problem schnell bereinigt. Wenn ich aber, wie das bei einer solchen Arbeit üblich und sinnvoll ist, meine Systematik fortentwickeln und dabei schrittweise ändern will, wird es kompliziert. Stellen wir uns vor, wir wollten zwischen die Ebene von *Sitzmöbel* und *Armstuhl* noch eine zusätzliche Differenzierung (z.B. *ein- bzw. mehrsitzige Sitzmöbel*) einführen. Bei der ersten Methode muß ich alle betreffenden "Karteikarten" umschreiben und gegebenenfalls noch ein zusätzliches Feld in die Datenbank einführen, denn das hier abgebildete Schema erlaubt nur drei, aber nicht vier Oberbegriffe. Bei der zweiten Methode brauche ich nur eine einzige "Karteikarte" einzuschieben, die Zahl der hierarchischen Ebenen unterliegt keiner technischen Beschränkung.

Vor- und Nachteile der beiden Methoden liegen also auf ganz verschiedenen Ebenen. Für die erste Methode spricht eigentlich nur, daß ich sie auch in ganz primitiven, nicht "thesaurusfähigen" Datenbanksystemen anwenden kann. Gegen die erste Methode spricht, daß sie das Problem auf die Datenerfassung verlagert und daher teuer und schwer zu handhaben ist. Je nach der Zahl der hierarchischen Ebenen muß ich bei der ersten Methode die Menge der maschinellen Datenerfassung für das betreffende Datenfeld verdoppeln, verdreifachen (oder mehr) und dazu zusätzlich sichern, daß auch diese Datenfelder für die Oberbegriffe völlig einheitlich und konsistent ausgefüllt werden. Jede Änderung in der Systematik kann erhebliche Folgewirkungen haben.

Hinter der **zweiten Methode** steht der Versuch, unnötige "Redundanz" zu vermeiden. Man möchte die gleiche Information nicht mehr als einmal erfassen und - wenn nötig - auch nur an einer einzigen Stelle korrigieren müssen. Welche Rolle das in der Praxis spielt, erkennen wir daran, daß solche Hilfsdateien oder gar Hilfsdatenbanken heute ein zentrales Thema im Dokumentationswesen sind.

Die in diesem Kapitel als Beispiel benutzte Hilfsdatenbank war ein einfach strukturierter "Thesaurus". Bevor wir uns nun mit verschiedenen Arten von Hilfsdatenbanken beschäftigen, schieben wir ein paar Bemerkungen zu dem eben benutzten Grundbegriff der "Redundanz" ein.

Grundbegriffe: Vermeidung von "Redundanz" durch Hilfsdatenbanken

Die Verlagerung von häufig benutzten Informationen in Hilfsdateien ist in vielen Fällen außerordentlich arbeitssparend. Denken Sie neben dem eben gegebenen Beispiel z.B. an die Umsetzung von Datierungsangaben in Jahreszahlen, an die Lebensdaten von Künstlern und zahlreiche andere Fälle, wo sich eine Information gar nicht direkt auf das Sammlungsobjekt, sondern nur auf seine Eigenschaften bezieht. *Silber* ist immer ein *Edelmetall*, alle *Edelmetalle* sind *Metalle*; bei etwas Nachdenken werden Sie finden, daß sich ein solches Schema auf sehr viele Informationen erfolgreich (= arbeitssparend) anwenden läßt.

Hier sind nicht zuletzt Hilfsfunktionen wie das "Permutieren" von Schlagwörtern (z.B. Abb. 9, Seite 27 rechte Spalte) zu nennen; es wäre außerordentlich umständlich und teuer, solche Arbeiten auf jeder "Karteikarte" immer wieder neu durchzuführen. Immer wenn eine Information mehr als einmal vorkommt, sollte man sich die Frage stellen, ob man die Daten nicht so strukturieren könnte, daß man die damit verbundene Arbeit nur einmal machen muß.

Diese "Datenstrukturierung" ist natürlich - genau wie die berechtigte Unlust an repetitiven und immer langweiligen Arbeiten - im Grunde ein "alter Hut". Die den meisten Leuten noch nicht vertrauten "Hilfsdatenbanken" (die englische Bezeichnung "Resource Data Bases" gibt das besser wieder) enthalten weiterführende Informationen zu einzelnen Aspekten einer "Objektdatenbank", sie entsprechen also ziemlich genau den uns allen vertrauten Nachschlagewerken, Klassifikationen, Künstlerlexika, Ortsbüchern und dergleichen. Durch ihre Verknüpfung mit den Objektdatenbanken erlauben sie eine strengere terminologische Kontrolle, ein Sortieren oder Suchen nach übergeordneten Sachgruppen sowie eine ökonomischere Datenerfassung. Auch der "Computer" muß in solchen Werken nachschlagen können, sonst gibt es ein großes Durcheinander.

Diese Dinge sind also im Grunde ganz einfach, weil wir sie aus anderen Zusammenhängen bereits kennen oder sie fast instinktiv bei unserer eigenen Arbeit einsetzen. Eine "Umsetzung" in ein effizientes und leicht benutzbares Programm kann allerdings sehr schwierig werden, da oft die uns zur Verfügung stehende Standardsoftware nicht mitspielt.

Statt langer theoretischer Erörterungen wollen wir nun einfach ein paar Fälle, bei denen sich ein solches Vorgehen lohnt, mit Hilfe von Abbildungen veranschaulichen. Dabei werden wir auch ein paar weitere Grundbegriffe klären.

Präferenz- und Stopplisten

Die einfachsten Hilfsdateien haben die Form alphabetisch geordneter Wortlisten, sie enthalten also nur ein einziges Feld. Zwei Benutzungsarten kommen häufiger vor:

- **Positivmethode:** Die manchmal "Präferenzliste" genannte Hilfsdatei enthält nur die erlaubten Ausdrücke.

- **Negativmethode:** Die sogenannten "Stopplisten" (oder auch Stoppswortlisten) enthalten nur die unerwünschten Ausdrücke.

Mit ihrer Hilfe kann man sich die terminologische Kontrolle erleichtern.

Abkürzungsdateien

Die üblichste Art, bei ständig wiederholten Ausdrücken Schreibearbeit zu sparen, liegt in der Benutzung von Abkürzungen. Wenn man den Computer benutzt, kann man dann - wenn man will - diese Abkürzungen wieder durch den Klartext ersetzen.

Solche "Abkürzungsdateien" können mehrere Felder (z.B. die Abkürzung und den Klartext) enthalten (vgl. Abb. 20, Seite 40 oberer Teil). Abkürzungslisten können vielfältig eingesetzt werden, wir geben hier nur eine kurze Liste von (selbstverständlich miteinander kombinierbaren) Möglichkeiten:

- **Billige bzw. schnelle Datenerfassung:** Erfasst werden in dafür geeigneten Feldern nur die Abkürzungen (vgl. die den Inventarnummern vorangestellten Abkürzungen für das Museum in Abb. 20 unterer Teil). Diese kann man dann entweder in der Datenbank oder auch nur für den Druck durch den Klartext ersetzen (vgl. Abb. 21, Seite 41).
- **Terminologische Kontrolle:** Eine "Abkürzung" kann auch ein vollständiges Wort oder auch ein kurzer Satz sein. Mit Hilfe einer solchen Liste kann dann der Computer feststellen, ob es sich um einen "erlaubten" oder "verbotenen" Begriff handelt.

Abkürzungsdateien sind immer dann von Vorteil, wenn sie Zeit und Arbeit sparen. Man muß bei ihrer Benutzung aber auch bedenken, daß anderen Benutzern durch solche Abkürzungen die Arbeit erheblich erschwert werden kann.

Achtung: In vielen Programmsystemen werden solche Abkürzungsdateien großzügigerweise als "Thesaurus" bezeichnet. Sie müssen sich also immer über den betreffenden Leistungsumfang genau informieren !

Lexika

Abkürzungsdateien sind vergleichsweise einfach strukturierte Hilfsinstrumente, die sich in einer sehr einfachen und direkten Weise mit dem in der betreffenden Hauptdatei verwendeten Wortschatz auseinandersetzen.

Wenn es nicht nur um den Wortschatz, sondern um weiterführende "echte" Daten geht, wird man andere Hilfsmittel einsetzen. Solche "Lexika" sind meist fachbezogene Werke, Beispiele sind Künstlerlexika, ikonographische Hilfsmittel wie ICONCLASS oder der im "Inventaire Général" eingesetzte THESAURUS ICONOGRAPHIQUE von Garnier.

Zur Veranschaulichung können wir auf uns bereits bekannte Beispiele zurückgreifen. Die auf Seite 20 vorgeschlagene Übung zu einer Umsetzung von Datierungen in Jahreszahlen könnte Grundlage eines kleinen, aus nur wenigen Feldern bestehenden Lexikons werden. Ganz ähnlich kann man sich die Anfänge eines Künstlerlexikons vorstellen, bei dem man z.B. zu dem Namen des Künstlers noch zusätzliche Informationen wie Geburts- und Todesdatum usw. erfaßt.

Auch an sich selbständige Datenbanken (z.B. Bibliographien, Adressenlisten usw.) können als "Hilfsdatenbanken" eingesetzt werden.

Abkürzung: R03 Klartext: Niederrheinisches Museum für Volkskunde und Naturgeschichte, Kevelaer
Abkürzung: R08 Klartext: Museum der Stadt Bensberg
Abkürzung: W01 Klartext: Heimatmuseum Emsdetten
Abkürzung: W36 Klartext: Heimatmuseum Rykenberg
Inventar-Nr.: R08 K 27 Objekt: Armstuhl
Inventar-Nr.: R03 K64/292 Objekt: Vitrinenschrank
Inventar-Nr.: W36 78/1092 Objekt: Kleiderschrank
Inventar-Nr.: W36 78/848 Objekt: Holländischer Leinenschrank
Inventar-Nr.: W36 78/1246 Objekt: Glasvitrine
Inventar-Nr.: W01 78/127 Objekt: Glasschrank
Inventar-Nr.: R08 K 12 Objekt: Schrank Schrank, 18. Jh., Eichenholz

Abb. 20: Eine leicht zu verändernde und zu ergänzende Datei mit nur zwei Feldern (Abkürzung und Klartext, s. oberer Teil) erspart die entsprechenden Angaben in der Objektdatei (unterer Teil). Da in den erfaßten Museen (oder Abteilungen eines Museums) die gleiche Inventarnummer mehrfach vorkommen kann, wurde im Projekt "Kleine Museen" die Abkürzung zu einem Teil der Inventarnummer gemacht. Man hat dann die Wahl, ob man die Abkürzung (wie hier im unteren Teil) oder den Klartext (vgl. Abb. 21 auf Seite 41) benutzen will.

Museum der Stadt Bensberg
Inventar-Nr.: K 27
Objekt: Armstuhl

Niederrheinisches Museum für Volkskunde und Naturgeschichte, Kevelaer
Inventar-Nr: K 64/292
Objekt: Vitrinenschrank

Heimatismuseum Rykenberg
Inventar-Nr: 78/1092
Objekt: Kleiderschrank

Heimatismuseum Rykenberg
Inventar-Nr: 78/848
Objekt: Holländischer Leinenschrank

Heimatismuseum Rykenberg
Inventar-Nr: 78/1246
Objekt: Glasvitrine

Heimatismuseum Emsdetten
Inventar-Nr: 78/127
Objekt: Glasschrank

Museum der Stadt Bensberg
Inventar-Nr: K 12
Objekt: Schrank
Schrank, 18. Jh., Eichenholz

Abb. 21: Ständig wiederkehrende Begriffe (hier aus dem Projekt "Kleine Museen" der Name des Museums) können aus Abkürzungen erzeugt werden

Thesauri

Die wichtigsten Funktionen des Thesaurus haben wir bereits kennengelernt (im Kapitel "Statt langer Erläuterungen ein Beispiel: Erzeugung systematischer Register" auf S. 33 ff.).

Thesauri sind pragmatisch orientierte Instrumente, sie dienen in erster Linie dazu, einen Wortschatz zu strukturieren und zu kontrollieren, sie sind gerade in Fällen, in denen sich der Wortschatz hierarchisch strukturieren läßt (Sachgruppenordnungen, geographische Daten)

angemessen. Da solche Thesauri in vielen Fachbereichen seit Jahren mit Erfolg benutzt werden, sind ihre Vor- und Nachteile genau bekannt. Vergleichbar effiziente Alternativen stehen bisher nicht zur Verfügung.

Ein Thesaurus braucht nicht unbedingt eine in jeder Beziehung wissenschaftlich abgesicherte Sache zu sein. Die Erfahrung hat gezeigt, daß die mit einer praxisorientierten "Grobklassifikation" erreichbare Vorstrukturierung umfangreicher Sammlungsbestände eine durchaus attraktive und für Retrievalzwecke effiziente Zwischenlösung darstellt¹⁹.

Der Unterschied zwischen einer Thesaurus- und einer Abkürzungsdatei besteht im wesentlichen darin, daß ich zwischen den Begriffen inhaltliche Relationen wie z.B. Ober- und Unterbegriffe (wie in Abb. 15 sehr verkürzt dargestellt) definieren kann.

Dabei werden also nicht nur von irgendeinem Begriff aus weitere Informationen (Klartext zu Abkürzungen bzw. weiterführende Informationen) gefunden, sondern es können richtige Verkettungen von solchen Relationen verarbeitet werden. Solche Verweistechiken sind auch für andere Anwendungen von Interesse, wir werden das weiter unten im Kapitel "Darstellung von Zusammenhängen durch Verweise" (Seite 84 ff.) behandeln.

Mit Thesauri kann man also grundsätzlich alles tun, was oben schon bei den "Abkürzungen" beschrieben wurde, zusätzlich eröffnen sie außerordentlich effiziente Wege für z.B. die folgenden Aufgaben:

- **Systematische Register:** Hier sind nicht nur Produkte wie in Abb. 10 zu nennen, sondern auch die Möglichkeit zu Querverweisen, zur Benutzung von Synonymen usw.
- **Retrieval:** In manchen Programmsystemen kann der Thesaurus auch beim Retrieval von Informationen arbeitssparend eingesetzt werden.
- **Terminologische Kontrolle:** Ein Thesaurus enthält normalerweise nicht nur Deskriptoren (= erlaubte Begriffe), sondern auch Nicht-Deskriptoren (= Begriffe, die nicht benutzt werden sollen). Man kann dem System also z.B. beibringen, daß man nicht *Bartmannskrug*, sondern *Bartmannkrug* schreiben sollte. Dies ergänzt die beschränkten Möglichkeiten der Präferenz- und Stopplisten oder Lexika um eine wichtige Funktion.

Die Grundform solcher Thesauri ist seit vielen Jahren durch nationale und internationale Normen festgelegt²⁰, diese Normen können auch im Museumsbereich mit Profit eingesetzt werden. Es spricht im übrigen nichts dagegen, solche Normen durch zusätzliche Leistungen zu erweitern²¹.

Trotz dieser Normen ist der Gebrauch des Wortes "Thesaurus" offensichtlich in keiner Weise geschützt, man muß sich also bei jedem Programmsystem ganz genau darüber informieren, was für ein Thesaurus dort unterstützt wird.

Zusammenfassung: Wie benutzt der Computer Hilfsdateien ?

Mit ein paar kleinen, aber wichtigen Einschränkungen benutzt der Computer Hilfsdateien bzw. Hilfsdatenbanken ganz genau so, wie Sie ein Lexikon benutzen. Stellen Sie sich z.B. vor, er solle bei jedem neuen Eintrag im betreffenden Lexikon nachschauen, ob der jeweilige Begriff dort schon vorkommt.

Er nimmt sich also z.B. einen Künstlernamen aus der "Hauptdatei" (sagen wir "*Jan van Goyen*"), greift zur betreffenden "Hilfsdatei" (welche das ist, muß man ihm natürlich

19) Vgl. Anm. 24

20) Vgl. Anm. 29

21) Eine gute Übersicht über alle damit zusammenhängenden Fragen finden Sie in Gernot Wersig, Thesaurus-Leitfaden, 2. ergänzte Auflage, K.G.Saur 1985.

mitteilen) und sucht dort den Namen. Damit das schnell und effizient geschehen kann, sollte diese Hilfsdatei nach den Namen sortiert sein (das läßt sich auf verschiedene, hier nicht zu diskutierende Weisen, erreichen).

Wenn dieser Name ("*Jan van Goyen*") in Ihrem Künstlerlexikon nur einmal (und genau so geschrieben) vorkommt, ist der Computer zufrieden und wird entsprechend (wie vorprogrammiert) reagieren. Vielleicht sagt er gar nichts (und akzeptiert den formal korrekten Namen), vielleicht ist er geschwätzig und bringt Ihnen eine entsprechende Meldung auf den Bildschirm.

Gibt es in Ihrem Künstlerlexikon aber mehrere Künstler mit dem Namen "*Jan van Goyen*", so werden Sie merken, daß das so einfach nicht abläuft. Der Computer weiß sich nicht zu entscheiden, auch Sie können solche Fragen nicht ohne zusätzliche Informationen beantworten. Welche zusätzlichen Informationen da nun helfen könnten, können Sie entsprechend den jeweiligen Umständen ganz frei entscheiden, die Maschine kann das nicht.

Langer Rede kurzer Sinn: Die Einträge in einem Computerlexikon müssen formal eindeutig sein. Das erreicht man (auch der Thieme-Becker tut das so) durch sogenannte "Homonymenzusätze", beim Thieme-Becker z.B. werden mehrfach vorkommende Namen durch die Hinzufügung römischer Zahlen in runden Klammern eindeutig gemacht. Das geschah übrigens nicht in weiser Voraussicht auf das Computerzeitalter, sondern um die allfälligen Querverweise zu vereinfachen bzw. eindeutig zu machen. Und das ist das ganze Geheimnis beim Aufbau und bei der Benutzung von Hilfsdateien: **Formale Eindeutigkeit.**

Nicht alle Computerprogramme werden Ihnen hier einen optimalen Komfort bieten. Manche verlangen eine totale Vereinheitlichung der Begriffe, andere (thesaurusähnliche) werden vielleicht sogar mit verschiedenen Schreibweisen des gleichen Namens umgehen können. Dies können Sie im Zweifelsfalle leicht testen.

Stand der Forschung zu Fragen der Terminologie im Museum

Bei allen Anwendungen von Hilfsdateien spielt die terminologische Kontrolle eine wichtige, oft sogar die einzige Rolle. Dieser Aspekt soll hier nicht vollständig abgehandelt werden, es scheint daher nützlich, auf weiterführende Informationsmöglichkeiten hinzuweisen.

Eigene Erfahrungen des IfM: Für den hier interessierenden Bereich der terminologischen Kontrolle bzw. Normierung von Schlagwörtern hat das IfM ausgiebige Erfahrungen gemacht und diese auch publiziert²². Dabei ging es um ein schrittweises Vorgehen (Ausgangspunkt: bereits ausgefüllte Karteikarten - Verschlagwortung durch Hilfskräfte - Erschließung zunächst durch einfache, dann durch permutierte alphabetische Register - Aufbau einer pragmatisch orientierten Grobklassifikation durch Wissenschaftler - Übergang zu wissenschaftlich fundierten Thesauri für einzelne Themenbereiche). Aspekte hierbei sind Fragen des verwendeten Wortschatzes²³ sowie eine auf praktisch Anforderungen hin orientierte Einbindung in den Dokumentationsprozeß²⁴.

Das internationale Bild ist auf den ersten Blick etwas verwirrend. Begriffe wie Thesaurus, Lexikon, Klassifikation, Sachgruppenordnung, Nomenklatur, Index usw. werden fast wahllos benutzt. Das Spektrum reicht vom "Inventaire Général" in Frankreich (100 %-ig normierter

22) Carlos Saro und Christof Wolters, EDV-gestützte Bestandserschließung in kleinen und mittleren Museen, in: Materialien aus dem Institut für Museumskunde, Heft 24, Berlin 1988.

23) Christof Wolters, On the need to support language specific problems in terminology systems, in s.u.: Terminology for Museums, S. 561-567.

24) Christof Wolters, Object Databases and Thesauri for Small Museums, in: Hermann Bock, Peter Ihm (Editors), Classification, Data Analysis and Knowledge Organization, Proceedings of the 14th Annual Conference of the Gesellschaft für Klassifikation e.V., University of Marburg, March 12-14, 1990, Springer-Verlag, S. 205-214.

Wortschatz als Ausgangspunkt) bis zu Unternehmungen, die das Problem erst nachträglich entdecken.

Trotzdem gibt es seit ein paar Jahren ganz vorzügliche, auch für Außenstehende leicht nutzbare Informationsmöglichkeiten. Wer sich mit solchen Fragen auseinandersetzen möchte, sollte zunächst die folgende Liste abarbeiten:

- **Terminology for Museums**²⁵: Ende 1990 erschienen, gibt es den internationalen Stand der Forschung im Museum ausgezeichnet wieder.
- **CIDOC (Comité International pour la Documentation): Die Terminology Control Working Group**²⁶ hat im letzten Jahr eine Bibliographie begonnen, die nach einigen noch notwendigen Ergänzungen auch publiziert werden soll. Toni Petersen stellt z.Zt. Titel zu Objektbezeichnungen zusammen.
- **Census**²⁷: Eine mehr als 750 Seiten (Projektbeschreibungen und Register) umfassende Darstellung einzelner kunsthistorischer "Computer-Projekte".
- **MDA Book Catalogue / Book Sales Service**²⁸: Viele der wichtigsten Titel können Sie direkt bei der britischen Museum Documentation Association bestellen (aktuelle Unterlagen anfordern).
- **Normen**: Für Thesauri gibt es zuverlässige und auch im Museum gut brauchbare nationale und internationale Normen²⁹ sowie darauf aufbauende Handbücher³⁰.

Auch zu der Frage der praktischen Umsetzung irgendwelcher Pläne gibt es bereits eine Menge publizierter Erfahrungen. Diese sollten, auch wenn sie nicht direkt aus dem Museum stammen, außerordentlich ernst genommen werden:

- **Lokale Planung**: Dieser Bereich ist am besten bearbeitet. Es gibt sowohl handfeste "Gebrauchsanweisungen"³¹ als auch sehr ernstzunehmende Hinweise auf Gründe für das Scheitern solcher Vorhaben³².
- **Überregionale Kooperation**: Fragen einer nationalen bzw. internationalen Zusammenarbeit stellen sich allein schon aus Kostengründen³³.

25) Terminology for Museums, Proceedings of an International Conference held in Cambridge, England, 21-24 September 1988, The Second Conference of The Museum Documentation Association, Edited by D. Andrew Roberts, Published by The Museum Documentation Association, Cambridge 1990 (ISBN 0 905963 62 8).

26) Kontaktadresse: Toni Petersen, Art and Architecture Thesaurus, 62 Stratton Road, Williamstown, Massachusetts 01267, USA, Tel. 413 458-2151, Fax 413 458-3757

27) SN/G: Report on Data Processing Projects in Art, Scuola Normale Superiore, Pisa, Italy and The J. Paul Getty Trust, Los Angeles, CA, USA, 1988.

28) The Museum Documentation Association, Building O, 347 Cherry Hinton Road, Cambridge CB1 4DH, Tel. 0044-223-242848

29) DIN 1463: Teil 1, Erstellung und Weiterentwicklung von Thesauri (Einsprachige Thesauri), Teil 2, Erstellung und Weiterentwicklung von Thesauri (Mehrsprachige Thesauri), ISO 5964: Documentation - Guidelines for the establishment and development of multilingual thesauri. Alle diese Normen können beim Beuth-Verlag GmbH, Burggrafenstraße 6, 1000 Berlin 30 bestellt werden.

30) Z.B. Gernot Wersig, Thesaurus-Leitfaden, 2. ergänzte Auflage, K.G.Saur 1985; Für Fragen der Organisation (und damit der Kosten) s. besonders Kapitel 5.2.6, S. 239-242: Organisationsform der Thesaurus-Arbeit.

31) Jane Sunderland und Lenore Sarasan, Was muß man alles tun, um den Computer im Museum erfolgreich einzusetzen? in: Materialien aus dem Institut für Museumskunde, Heft 30 (1990).

Zum gleichen Thema erschienen sind D. Andrew Roberts, Planning the Documentation of Museum Collections, MDA 1985, 568 Seiten und David Bearman, Automated Systems for Archives and Museums: Acquisition and Implementation Issues, Pittsburgh 1987.

32) Lenore Sarasan, Why Museum Computer Projects Fail, publiziert in: Museum News, 1981, January/February, S. 40-49: Eine in jedem Detail ernstzunehmende Übersicht über die Probleme, die die meisten Leute durch Aufsetzen von Scheuklappen zu lösen versuchen.

33) Vgl. hierzu Christof Wolters, Vorschläge zur Planung zentraler Dienstleistungen für Museen in der Bundesrepublik Deutschland und Berlin (West), in: Museumsblatt 2, 1990, Hrsg. Landesstelle für Museumsbetreuung in Zusammenarbeit mit dem Museumsverband Baden-Württemberg e.V. Neckarhalde 30 A, D 7400 Tübingen, S. 6-26 (s. Publikationen auf Seite 133).

- **Schulung und Betreuung:** Anders als im Bibliotheks- und Dokumentationswesen gibt es im Museumsbereich noch keine spezifischen Ausbildungsgänge. Daraus ergeben sich grundlegende Probleme³⁴.

Praktisch-organisatorische Erfahrungen zu diesen zentralen Fragen werden leider viel zu selten diskutiert, können aber weitgehend aus anderen Anwendungsbereichen übertragen werden³⁵.

Der zentrale Punkt in diesem Zusammenhang ist - wie bei allen Projekten - die Qualifikation der mit der Durchführung eines solchen Projekts beauftragten Mitarbeiter sowie die Schaffung geeigneter Arbeitsbedingungen³⁶. Hier wird viel gesündigt. Eine Einrichtung, die diese Probleme löst, schafft damit eine ausgezeichnete Grundlage für einen Erfolg, wer sich darum drückt, soll sich über den dann unausweichlichen Flop nicht wundern.

Zusammenfassung: Stufenmodell für die Formulierung von Textdaten

Das Ziel des 1. Hauptteils war, die nicht zu umgehende strenge kausale Verknüpfung von formalen Eigenschaften der Daten mit den gewünschten Leistungen an geeigneten Beispielen zu veranschaulichen.

Wir taten das zunächst in einer selbstgewählten Beschränkung auf einzelne Datenfelder. Wir betrachteten, was eine Maschine mit den in einem solchen Datenfeld gespeicherten Zeichenfolgen tun kann und kamen dabei auf grundlegende Produkte wie eine einfache Wiedergabe der Daten oder auch eine Sortierung in Form alphabetischer Register. Wir sahen auch, daß man für kompliziertere Produkte wie systematische Register ggf. auf mehr als ein einziges Datenfeld zurückgreifen muß. Nicht zuletzt wurde erklärt, daß die meisten Arten des "Retrieval" letztlich nichts anderes sind als eine geschickte "interne" Benutzung solcher alphabetischer oder systematischer Register. Die formalen Anforderungen an die Daten sind bei beiden grundsätzlich die gleichen.

Daraus wollen wir nun Konsequenzen ziehen. Ziel dieses Abschnitts ist die Vorbereitung einer auf Heft 30 der Materialien aufbauenden Analyse des einzelnen Datenfelds in der Form von Arbeitsblättern. Sie finden diese Arbeitsblätter im Anhang (Seite 93 ff.).

Aus dem bisher Gelernten können wir eine Art **Stufenmodell für die Definition einzelner Datenfelder** ableiten, das uns dabei helfen soll, etwas Ordnung in den Lernstoff zu bringen.

Diese Ordnung entspricht einer schrittweisen Aufbereitung der Daten mit dem Ziel, bestimmte Produkte per Programm mit diesen Daten zu erzeugen. Hierbei soll vor Allem klar werden, daß der Übergang zur jeweils nächsten Stufe auch eines strengeren Regelwerks für Definition und Kontrolle der Daten bedarf.

- **Wiedergabe der Daten:** Die Behandlung von Daten als "Freier Text" scheint die geringsten Ansprüche an die Formulierung von Daten zu stellen. Speichermedien (incl. Papier) sind geduldig. Trotzdem muß man schon hierfür an ein paar ganz formalistische Dinge denken:
 - **Zeichensatz:** Eine Klärung des erforderlichen Zeichensatzes ist bereits für die einfachsten Produkte grundlegend. Folgewirkungen auf die folgenden Stufen (z.B. Sortierung) sind zu beachten.

34) s. Anm 33 S. 19-21.

35) Eine kurze und auch für Museen voll zutreffende Darstellung von Grundfragen der komplexen Problematik von IT-Schulung und IT-Benutzerbetreuung gibt Volker Bihl, Überlegungen zur Organisation des Einsatzes der Informationstechnik in der Bundesverwaltung (Teil II), VOP 6/1988, 306-308.

36) Ausführliche Hinweise s.o. Anm. 33 S. 21-23

- **Feldlänge:** Für bestimmte Produkte wie den Druck von Formularen und Karteikarten hat man nur einen beschränkten Platz zur Verfügung. Das sollte bereits vor der Datenerfassung geklärt werden.
- **Sortieren der Daten:** Die meisten etwas komplizierteren Produkte bauen auf der Sortierbarkeit der Daten, auf ihrer Benutzung als "**Schlagwörter**" auf. Das Thema ist daher nahezu unerschöpflich. Um hier Klarheit zu gewinnen, wurden zwei aufeinander aufbauende Stufen bzw. Arten von Registern ausgewählt:
 - **Alphabetische Register** (bzw. ein auf einzelne Begriffe bezogenes Retrieval): Solche Register sind nicht nur ein mögliches "Endprodukt" (z.B. Namensregister), sie sind vor Allem auch ein den meisten Benutzern verborgen bleibendes Zwischenprodukt für weiterführende Leistungen.

Klärung der Rechtschreibung: Probleme können im Deutschen z.B. daraus entstehen, daß ältere Inventare noch mit Schreibmaschinen ohne Umlaute und ß geschrieben wurden. Bei "normalen" Wörtern macht eine Umsetzung in eine modernere Schreibweise dem Datentypisten nicht allzuviel Mühe, was aber ist mit Personen- oder geographischen Namen ?

Der früher unausweichlich notwendigen strengen Vereinheitlichung der Daten vor der maschinellen Datenerfassung stehen heute komfortablere Methoden (teilweise Vereinheitlichung per Programm, Thesauri) gegenüber.

Regelwerke für alphabetische Register werden meist themenbezogen formuliert und können bei historischen Daten außerordentlich kompliziert werden. So einfach es erscheint, ein Telefonbuch zu sortieren, so schwierig wird das bei mittelalterlichen oder zahlreiche Zusätze enthaltenden Namen.

Ein alphabetisches Register wird zumindest "*Schmidt, E.*" und "*Schmidt, Egon*" nahe zueinanderbringen. Ob das die gleiche Person ist, erfahren wir daraus zunächst nicht.

- **Systematische Register** (bzw. ein entsprechendes, auf zusammenhängende Gruppen von Informationen bezogenes Retrieval): Solche Register bauen zwar grundsätzlich auf alphabetisch sortierten Zwischenprodukten (und den entsprechenden Anforderungen an die Daten) auf, können aber eine Menge "inhaltlicher" Zusammenhänge darstellen.

Grundlage für systematische Ordnungen ist immer ein "**Hintergrundwissen**", das der Maschine in einer für sie geeigneten Weise mitgeteilt werden muß. Wir werden diesen Punkt im 2. Hauptteil noch vertiefen (Kapitel "Was kann ich eigentlich alles in ein einzelnes Datenfeld stecken ?" auf Seite 64 ff.).

Je nach der Qualität der gewünschten alphabetischen oder systematischen Register muß ich also bei der Formulierung der Daten bestimmte Regeln beachten. Der kritische Punkt liegt hier im Grad der terminologischen Kontrolle. Bei der Benutzung der hierfür optimalen Hilfsdateien müssen die Daten nicht nur richtig formuliert, sondern auch inhaltlich eindeutig definiert werden. Dieses bedeutet eine weitere Stufe in unserem Modell:

- **Erzeugung von Eindeutigkeit:** Für viele Zwecke wird schon aus inhaltlichen Gründen eine über jeden Zweifel erhabene Eindeutigkeit der Dateninhalte zu fordern sein. Ein Name wie *Jan van Goyen* reicht nicht aus, um diesen Künstler eindeutig zu identifizieren. Bei vielen Daten kann das nur durch künstliche (Homonymen- bzw. Homographen-) Zusätze erreicht werden.

Bei der Benutzung von Hilfsdateien muß auch aus technischen Gründen für eine formale Eindeutigkeit bei den "Koppelbegriffen" gesorgt werden, damit ggf. weitere Daten aus der betreffenden Hilfsdatei geholt werden können.

Dieses stark vereinfachte Stufenmodell soll es Ihnen erleichtern, die sich aus den von Ihnen gewünschten Produkten unausweichlich ergebenden Anforderungen an die Formulierung der Daten zu erkennen.

Obwohl Sie an den beschriebenen Zusammenhängen nichts ändern können, ist es doch Ihr gutes Recht, daraus ganz verschiedene, meinetwegen sogar sich gegenseitig zunächst ausschließende Konsequenzen zu ziehen. Der eine möchte vielleicht erst alle diese Probleme für alle nur denkbaren Daten klären, der andere wird eine stufenweise Überarbeitung bereits erfaßter Daten für weitergehende Produkte vorziehen.

Betrachtet man die Geschichte der Datenverarbeitung im Museum, so gibt es hierbei eigentlich nur zwei völlig unrealistische Extrempositionen: Einerseits ist es noch niemandem gelungen, alles vorherzusehen, andererseits hat sich aber das Vertrauen, daß der Computer solche Probleme schon in naher Zukunft von alleine lösen würde, bisher noch nicht sonderlich bewährt.

Anhang zum 1. Hauptteil: Komplizierte Formen der Syntax eines Datenfeldes

Dieser Anhang zum ersten Hauptteil soll Ihnen dabei helfen, eine anschauliche Vorstellung von der ganzen "Trickkiste", die man gegebenenfalls beim Sortieren von Textdaten einsetzen muß, zu bekommen.

Wir werden das an Numerierungssystemen exemplifizieren, wollen aber dabei immer daran denken, daß entsprechende Dinge immer dann auf uns zukommen können, wenn wir mehr als eine einzige Information in ein Datenfeld stecken oder irgendwelche "Codes", die aus mehreren Elementen zusammengesetzt sind, nicht auf verschiedene Felder verteilen möchten. Dieser Anhang stellt daher auch eine Art Vorbereitung des zweiten Hauptteils dar, in dem wir uns mit solchen Fragen beschäftigen werden.

Numerierungssysteme spielen in der Museumsdokumentation eine so zentrale Rolle, daß wir uns in jedem Falle mit ihrer Verarbeitung im Computer intensiv auseinandersetzen müssen.

Inventarnummern und andere Numerierungssysteme

Inventarnummern können - je nach ihrem Aufbau - eines der leichtesten oder auch eines der lästigsten und schwierigsten Probleme beim Computereinsatz im Museum sein. Der Grund dafür ist schlicht und einfach, daß die meisten Computersysteme zwar mit reinen Zahlen prima umgehen können, aber mit der Kombination von Buchstaben und Zahlen, die sich manche Museumsleute für ihre Sammlungen ausgedacht haben, ohne zusätzliche Programmierung nicht viel anfangen können.

Die Existenz solcher komplizierter Nummernsysteme hat die verschiedensten Gründe. Im Rahmen dieses Handbuchs können wir darauf nur so weit eingehen (s. das nächste Kapitel), als das zum Verständnis der daraus resultierenden Probleme nötig ist.

Probleme mit Inventarnummern werden zumeist beim "Sortieren" sichtbar. Eine Sortierung kann z.B. nach aufsteigenden Inventarnummern gewünscht werden (z.B. für den Druck von Inventarverzeichnissen). Auch bei Korrekturen am Bildschirm darf man erwarten, daß die Inventarnummern in der "richtigen" Reihenfolge kommen.

Um die dabei entstehenden Probleme verstehen zu können, schauen wir uns zunächst an ein paar einfachen Beispielen an, wie die meisten Computerprogramme sortieren (Kapitel "Was kann ein "normales" Computerprogramm mit Inventarnummern machen?", Seite 50 ff.). Dies ist zudem ein gutes Übungsbeispiel für die rein formalistische Arbeitsweise aller Computerprogramme und damit auch eine Grundlage für ein besseres Verständnis dafür, daß manche Dinge nicht allein durch geeignete Schreibanweisungen gelöst werden können, sondern noch einer zusätzlichen Programmierung bedürfen.

Die in diesem Handbuch vorgeschlagene Analyse von Inventarnummern und anderen im Museum verwendeten Nummernsystemen hat den Zweck, Minimalanforderungen an das Programmsystem festzustellen, bevor es zu spät ist (Kapitel "Arbeitsblätter zur Beschreibung von Numerierungssystemen" auf Seite 106 ff.). Sie ist also zunächst eine **Ergänzung zu Heft 30 der Materialien aus dem Institut für Museumskunde**. Dort werden Probleme mit einzelnen Datenfeldern zwar erwähnt (z.B. beim Datenfeld-Arbeitsblatt, S. 19, 25 und in der System-Checkliste Seite 15 unter "Richtiges Sortieren nach Erwerbungs- bzw. Katalognummern"), aber schon aus Raumgründen nicht im Detail erläutert. Das hier vorgelegte Kapitel setzt die Kenntnis von Heft 30 der Materialien voraus, es bewahrt Sie vor unliebsamen Überraschungen beim Computereinsatz.

Bitte führen Sie eine Analyse Ihrer Inventarnummern vor dem Beginn der maschinellen Erfassung durch, sonst kann es leicht passieren, daß Sie die Erfassung wiederholen müssen.

Was sind Inventarnummern ?

Zum Begriff: Die Inventarnummer ist die Nummer, unter der ein Objekt im Inventar aufgeführt ist. Der Begriff wird leicht unterschiedlich gebraucht, je nachdem ob die Buchführung des Museums nur aus einem einzigen Inventar besteht oder sich in mehrere gliedert, also z.B. neben dem Eingangsinventar ("Eingangsbuch") noch einen wissenschaftlichen Katalog (heute oft Karteikarten) und gegebenenfalls weitere schriftliche Quellen enthält.

Im Zusammenhang dieses Handbuchs benutzen wir zwar den üblichen Begriff der Inventarnummer, die Ausführungen gelten aber ohne Abstriche auch für etwaige Katalog-, Akten- und andere Nummern, für Nummern in Bibliothek und Archiv und nicht zuletzt für Abbildungen (z.B. "Foto-Nummer").

Alle diese Numerierungssysteme haben die gleiche Aufgabe: die eindeutige Identifizierung durch Numerierung (und gegebenenfalls Beschriftung) des Objekts sowie den Verweis auf weitere Informationen.

Dies klingt einfach und hat doch recht oft zu erstaunlich komplizierten Systemen geführt. Die Gründe hierfür sind vielfältig und sind eng mit der Geschichte des Museums, ganz besonders mit Umorganisationen aller Art verbunden. Komplikationen entstanden immer dann, wenn man der Inventarnummer noch weitergehende Informationen hinzufügen wollte oder gelegentlich sogar mußte. Ein paar Beispiele hierfür sind:

- **Nennung des Inventars:** Bei der Zusammenlegung bereits inventarisierter Bestände wird durch zusätzliche (meist vorangestellte Buchstaben-) Kürzel die Eindeutigkeit der sonst evtl. doppelt vorkommenden Nummern gesichert; auch die Information, aus welcher Sammlung ein Stück kommt, wird gelegentlich als Grund genannt.
- **Abteilungs-Kürzel:** Ähnliches kann aus internen organisatorischen Gründen geschehen - wenn Abteilungen getrennt inventarisieren, ist ein gemeinsames Numerierungssystem umständlich.
- **Sachgruppen - Klassifikationssysteme:** In vielen Museen hat man nach "Sachgruppen" inventarisiert bzw. durch bestimmte Kürzel das Objekt bestimmten Sammelgebieten zugewiesen.
- **Erwerbungsjahr:** Gerade bei Eingangsinventaren ist die Verbindung von Jahreszahl mit einer laufenden Nummer üblich.
- **Zusammengehörigkeit von Objekten:** Nicht zuletzt hat man häufig versucht, durch Nummern mit Unternummern in irgendeiner Weise zusammengehörige Objekte zusammenzufassen.

Alle diese "Zusatzinformationen" können durch die Entwicklung überholt oder gar falsch werden: Inventar- und Abteilungskürzel werden durch Umorganisation der Sammlungen irreführend, Sachgruppen werden unsinnig, wenn bedeutende Neuerwerbungen die Sammlungsschwerpunkte verändern, das Erwerbungsjahr macht Probleme, wenn man vorher nie inventarisierte Altbestände erfaßt. Auch die in der Inventarnummer ausgedrückte Zusammengehörigkeit von Objekten wird zumeist für so unterschiedliche Dinge gebraucht, daß sie jeden präzisen Sinn verliert: Ensembles aller Art (gleicher Vorbesitzer, gleiche Wohnungseinrichtung, gleicher Fundplatz usw.) werden so auf platz- und arbeitssparende, aber letztlich ineffiziente Weise abgehandelt.

Heute weiß man, daß man Zusatzinformationen nach Möglichkeit nicht in die Inventarnummer integrieren sollte, in den meisten Fällen kommt diese Erkenntnis aber zu spät. Wir müssen uns dann nolens volens mit den bestehenden Systemen auseinandersetzen.

Was kann ein "normales" Computerprogramm mit Inventarnummern machen ? Numerische und alphanumerische Sortierung

Vorbemerkung: Der Begriff "normales" Computerprogramm ist ganz bewußt gewählt worden - er soll Sie darauf hinweisen, daß Probleme, deren Lösung Sie für ganz selbstverständlich oder zumindest leicht ("da schreibt man schnell ein Programm") ansehen, sich in vielen Programmen als recht schwierig herausstellen können.

Es ist also leider immer nötig, sich über die Leistungen eines Programmes völlige Klarheit zu verschaffen. Ein Hinweis hierzu am Anfang: es macht einen Unterschied (aus Gründen, die hier nicht näher erläutert werden), ob z.B. die in diesem Kapitel behandelte Sortierung nur für die **Druckausgabe** funktioniert oder auch für eine "richtige" Reihenfolge der Objekte beim Korrigieren am **Bildschirm**. Achten Sie auf solche Dinge, bevor Sie sich für ein Programm entscheiden.

Zur Sache: Wenn es sich bei den Inventar- (oder auch anderen) Nummern nur um **einfache arabische Zahlen ohne Kürzel oder Buchstabenzusätze und ohne "Trenner"** handelt (und zwar ohne eine einzige Ausnahme!), dann kann man die Inventarnummer in der Datenbank als rein numerisches Feld deklarieren. Eine **"numerische Sortierung"** macht dann prinzipiell keine Schwierigkeiten.

Zum Begriff Trenner: Trenner sind meist "Sonderzeichen", mit denen man selbständige Elemente eines Textes voneinander trennt, um die Leserlichkeit zu verbessern und Mißverständnisse zu vermeiden. In einem normalen Text haben Leertaste und Satzzeichen diese Funktion, bei Numerierungssystemen werden solche Zeichen (Leertaste, Punkt, Schrägstrich, Bindestrich usw.) in ähnlicher Weise, aber ohne "Duden" eingesetzt (z.B. M 4711.73/1 a-c).

Wenn man aber (z.B. wegen solcher Ausnahmen oder weil das System das nicht erlaubt) solche Nummern in ein normales Datenfeld schreiben muß, so wird man eine sogenannte **"alphanumerische Sortierung"** vornehmen müssen. Eine solche Sortierung ist uns allen aus alphabetischen Registern vertraut. Es wird da zunächst nach dem ersten Zeichen (erst alle Wörter, die mit A anfangen ...), dann nach dem zweiten Zeichen usw. sortiert. Bei kompliziert aufgebauten Inventarnummern kann das zu absurden Ergebnissen führen.

Wir zeigen die bei einer standardmäßigen alphanumerischen Sortierung nicht vermeidbaren Fehler mit Hilfe einer tabellenartigen Darstellung, in die wir Beispiele der üblichen Arten von Inventarnummern eintragen (Abb. 22-30). Sie können das übrigens, wenn Sie über ein moderneres Textverarbeitungsprogramm verfügen, leicht selber "durchspielen".

Einfache arabische Zahlen mit Zusätzen (Abb. 22)

Wir nehmen uns ein paar Zahlen (z.B. 1, 2, 5, 10, 20 a, 1618 ...), wegen gelegentlich vorkommender Zusätze (20 a) müssen wir alphanumerisch sortieren.

Das Ergebnis (Abb. 22 auf Seite 51) zeigt sehr schön das sture spaltenweise Vorgehen: 1, 10, 1618, 2, 20, 5 ... Die erste Spalte ist aufsteigend sortiert, die zweite wird innerhalb der in der ersten Spalte entstehenden Gruppen wiederum aufsteigend sortiert usw. Wir sehen dabei auch, daß eine "Leertaste" (bzw. eine "kürzere" Nummer) vor einer Zahl (bzw. einer "längeren" Nummer) einsortiert wird.

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1														
2														
5														
1 0														
2 0		a												
1 6 1 8														

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1														
1 0														
1 6 1 8														
2														
2 0		a												
5														

Abb. 22: Einfache arabische Zahlen mit Zusätzen

Buchstaben mit Zahlen (Abb. 23)

Solche Buchstabenkürzel mit darauffolgenden Zahlen (z.B. A 20, L 5, M 1, M 2, M 10, Z 1618) sind in vielen Museen üblich. Zumeist entsprechen ihnen bestimmte vordefinierte Sachgruppen, manchmal wurden Kürzel eingeführt, weil man bereits mit Inventarnummern versehene ganze Sammlungen übernahm und diese nicht mit neuen Nummern versehen wollte.

Das Ergebnis einer alphanumerischen Sortierung (Abb. 23) ähnelt sehr dem vorigen Beispiel. Bei Buchstaben mit darauffolgenden Zahlen wird - wie zu erwarten - der Anfang zwar richtig sortiert (A - L - M - Z), die uns schon bekannten Fehler entstehen dann aber innerhalb der in sich richtig (alphabetisch) sortierten Kürzel (M 1 - 10 - 2):

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A	2	0												
L	5													
M	1													
M	2													
M	1	0												
Z	1	6	1	8										

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A	2	0												
L	5													
M	1													
M	1	0												
M	2													
Z	1	6	1	8										

Abb. 23: Buchstaben mit Zahlen

Kombination aus mehreren Zahlengruppen (Abb. 24-26)

Aus zwei oder mehr Gruppen zusammengesetzte arabische Zahlen mit "Trennern" (Schrägstriche, Punkte, Leertaste etc) sind ebenfalls sehr üblich, zumindest bei moderneren Inventaren.

Wir nehmen als erstes (und relativ einfaches) Beispiel die häufig verwendete Kombination von Jahreszahl und laufender Nummer, zunächst die üblichere Reihenfolge "Jahreszahl - laufende Nummer" (z.B. 1969.1618, 1973.1, 1973.2, 1973.5, 1973.10, 1973.20).

Das Ergebnis (Abb. 24) entspricht wieder ganz dem vorigen Beispiel. Der gleichartige Anfang wird "richtig" sortiert, weil die Jahreszahlen aus dem Zeitraum der Inventarisierung alle gleich lang sind. Danach geht es wieder munter durcheinander:

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	9	6	9	.	1	6	1	8						
1	9	7	3	.	1									
1	9	7	3	.	2									
1	9	7	3	.	5									
1	9	7	3	.	1	0								
1	9	7	3	.	2	0								

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	9	6	9	.	1	6	1	8						
1	9	7	3	.	1									
1	9	7	3	.	1	0								
1	9	7	3	.	2									
1	9	7	3	.	2	0								
1	9	7	3	.	5									

Abb. 24: Kombination aus mehreren Zahlengruppen (1): Jahreszahl und laufende Nummer

Das Durcheinander wird noch wesentlich größer, wenn es sich bei der ersten Zahlengruppe nicht um eine Jahreszahl, sondern ebenfalls um eine laufende Nummer handelt (z.B. 26.1618, 173.1, 173.2, 173.15, 297.20, 297.103). Solche Zahlengruppen können natürlich auch mit Buchstabenkürzeln kombiniert vorkommen, hier reicht uns bereits der einfachere Fall (Abb. 25).

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
2	6	.	1	6	1	8								
1	7	3	.	1										
1	7	3	.	2										
1	7	3	.	1	5									
2	9	7	.	2	0									
2	9	7	.	1	0	3								

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	7	3	.	1										
1	7	3	.	1	5									
1	7	3	.	2										
2	6	.	1	6	1	8								
2	9	7	.	1	0	3								
2	9	7	.	2	0									

Abb. 25: Kombination aus mehreren Zahlengruppen (2): laufende Nummer mit Unter-
nummer

Völlig katastrophal wirkt sich die seltenere Reihenfolge "laufende Nummer - Jahreszahl" (z.B. 20/1953, 10/1964, 1/1973, 2/1973, 5/1973, 1618/1988) aus (Abb. 26).

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
2	0	/	1	9	5	3								
1	0	/	1	9	6	4								
1	/	1	9	7	3									
2	/	1	9	7	3									
5	/	1	9	7	3									
1	6	1	8	/	1	9	8	8						

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
1	/	1	9	7	3									
1	0	/	1	9	6	4								
1	6	1	8	/	1	9	8	8						
2	/	1	9	7	3									
2	0	/	1	9	5	3								
5	/	1	9	7	3									

Abb. 26: Kombination aus mehreren Zahlengruppen (3): laufende Nummer - Jahreszahl

Wenn - wie leider oft üblich - die Jahreszahl dann noch in einer abgekürzten Form geschrieben wird, sind solche Nummern übrigens auch für einen Menschen schwer zu interpretieren, was bedeutet z.B. 73/64, 12/49 usw. ? Daß das kein "Computerproblem" ist, kann man meist leicht an der Menge der falsch zurückgestellten Karteikarten erkennen.

Römische Zahlen (Abb. 27)

Solche Zahlen werden kaum jemals als selbständige laufende Nummern benutzt. Oft dienen sie (gegebenenfalls in Kombination mit Buchstaben) als Kürzel (z.B. K I 1, K II 2, M I 5, M I 10, M VII 20, M IX 4711 usw.), auch als "Unternummern" kommen sie gelegentlich vor.

Für den Computer sind das allerdings keine Zahlen, sondern Buchstaben. Das Ergebnis ist entsprechend (Abb. 27):

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
K	I			1										
K	I	I			2									
M	I				5									
M	I			1	0									
M	V	I	I			2	0							
M	I	X			4	7	1	1						

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
K	I			1										
K	I	I			2									
M	I			1	0									
M	I			5										
M	I	X			4	7	1	1						
M	V	I	I			2	0							

Abb. 27: Römische Zahlen: Buchstabe mit römischer Zahl als Kürzel

Eine kleine Erleichterung besteht darin, daß durch Zufall die römischen Zahlen I - VIII bei alphanumerischer Sortierung in die richtige Reihenfolge kommen. Wenn Ihre römischen Zahlen nicht über VIII hinausgehen, können sie für die Sortierung als Buchstaben behandelt werden.

Weitere Fehlerquellen, Gegenmittel (Abb. 28-30)

Das alphanumerische Sortieren basiert auf einem sturen spaltenweisen Vorgehen, wobei in den meisten Fällen Leertasten vor Buchstaben und Buchstaben vor Zahlen einsortiert werden.

Darin liegt eine zusätzliche, bisher noch nicht besprochene Fehlerquelle. Vergißt man bei der Erfassung eine Leertaste (Abb. 28: bei M 1), so wird die betreffende Nummer dadurch verschoben (vgl. mit den entsprechenden, aber korrekt erfaßten Nummern in Abb. 23 auf Seite 51).

Richtige Reihenfolge

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A		2	0											
L			5											
M		1												
M			2											
M		1	0											
Z		1	6	1	8									

Alphanumerische Sortierung

1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A		2	0											
L			5											
M		1	0											
M			2											
M		1												
Z		1	6	1	8									

Abb. 28: Leertaste vergessen

Manche Software-Hersteller empfehlen daher ihren Kunden, Inventarnummern "spaltengenau" zu schreiben und Zahlen linksbündig mit Nullen aufzufüllen. Das heißt, daß man für jedes Element bestimmte Spalten reservieren soll.

Der Benutzer muß dann also für jedes Element der Nummer die maximale Zeichenzahl ermitteln und dafür eine ausreichende Zahl von Spalten reservieren und bei der Erfassung streng einhalten. Zahlen müssen entsprechend ergänzt werden, eine Sache, die schon bei einfachen Zahlen sehr umständlich, unanschaulich und infolgedessen auch fehleranfällig ist und von der bei längeren Folgen von Buchstaben und Ziffern abzuraten ist (Abb. 29). Der schwarze Peter (und die ganze Arbeit) liegt beim Kunden:

Richtige Reihenfolge

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A					2	0									
L					0	5									
M					0	1	/	0	0	0	3				
M					0	1	/	0	0	2	0				
M					0	1	/	0	0	2	2				
Z	0	1	6	1	8										

Alphanumerische Sortierung

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A					2	0									
L					0	5									
M					0	0	1	/	0	0	0	3			
M					0	0	0	1	/	0	0	2	0		
M					0	0	0	1	/	0	0	2	2		
Z	0	1	6	1	8										

Abb. 29: Zahlen linksbündig mit Nullen auffüllen

Aber auch das ist keine narrensichere Methode. Wenn man aus Schlamperei (oder weil es nun mal so im Inventar oder auf dem Objekt steht) einmal den einen und einmal den anderen "Trenner" benutzt (oder vergißt - vgl. Abb. 28), werden die Sortierergebnisse wieder total verfälscht (vgl. Abb. 30 mit Abb. 29).

Richtige Reihenfolge

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A					2	0									
L					0	5									
M -					0	0	1	/	0	0	0	3			
M					0	0	0	1	/	0	0	2	0		
M					0	0	0	1	-	0	0	2	2		
Z	0	1	6	1	8										

Alphanumerische Sortierung

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
A					2	0									
L					0	5									
M					0	0	0	1	-	0	0	2	2		
M					0	0	0	0	1	/	0	0	2	0	
M -					0	0	0	0	1	/	0	0	0	3	
Z	0	1	6	1	8										

Abb. 30: Auswirkungen wechselnder Trenner

Wir hatten das schon bei den Bindestrichen in Abb. 2 auf Seite 16 gesehen. Weitere Informationen zur Sortierung von Sonderzeichen finden Sie im Anhang (Kapitel "Arbeitsblätter zum Zeichensatz" auf Seite 114 ff.)

Für **römische Zahlen** und **nachgestellte Sortierbegriffe** gibt es im Bereich der normalen alphanumerischen Sortierung nur die Möglichkeit, römische Zahlen durch arabische zu ersetzen bzw. die Inventarnummern in die "richtige Reihenfolge" zu bringen.

Alle diese Gegenmittel laufen letztlich darauf hinaus, die Inventarnummern mehr oder weniger stark zu verändern. Dieses kann - wenn man diese Änderungen nicht auch bei den Sammlungsobjekten vornimmt - zu einer nicht endenwollenden Serie von Mißverständnissen führen.

Kombination mehrerer Arten von Inventarnummern in einem Museum

Es ist gar nicht so selten, daß mehrere Arten von Inventarnummern nebeneinander im gleichen Museum vorkommen oder daß sich ein Projekt mit mehreren Arten solcher

Nummern auseinandersetzen muß. Die bei der Besprechung von Zusatzprogrammen abgebildete Liste (Abb. 31 auf Seite 57) mit den schon besprochenen Beispielen aus Abb. 22-27 reicht aus, um dieses zusätzliche Problem zu veranschaulichen.

Zusätzliche Probleme: Sammel- und Bereichsangaben

In vielen Inventaren werden mehrere Objekte - nicht selten sogar ganze Gruppen oder Konvolute - nicht einzeln aufgeführt, sondern gemeinsam abgehandelt. Daraus können ungemein schwierig zu lösende Probleme entstehen (vgl. Kapitel "Was ist ein Museumsobjekt?" auf Seite 75 ff.), an dieser Stelle beschränken wir uns auf die maschinelle Verarbeitung der Inventarnummern. Dieses ist an ein paar Beispielen leicht zu erklären:

- **A 317 - A 320:** Eine solche Angabe bedeutet, daß die dann folgenden Ausführungen für die Nummern A 317, A 318, A 319 und A 320 gelten. Es wird hier nicht weiter hinterfragt, ob diese Stücke wirklich zusammengehören (z.B. 4 Teile eines Kostüms) oder nur aus Bequemlichkeit zusammen erfaßt wurden (z.B. 4 Webgewichte verschiedener Provenienz), es interessiert auch nicht, ob sie einwandfrei identifiziert werden können (das geht nur, wenn die Beschreibung darauf eingeht bzw. die Stücke entsprechend beschriftet sind).

Für eine **Sortierung** entsteht kein zusätzliches Problem, die Angabe wird nach A 316 ... und vor A 321 ... einsortiert. Für ein **Retrieval** können Probleme entstehen. Sucht jemand nach A 318, dann wird er dieses Stück in den meisten Systemen nicht finden.

In der **Schreibweise** sind erhebliche Variationen möglich und üblich, A 317 - A 320 (wie oben), A 317-320, A 317-20, manchmal auch A 317/20. Es liegt auf der Hand, daß man sich für über das Sortieren hinausgehende Anforderungen auf zusätzliche Programmierung einstellen und dafür eine einheitliche Schreibweise festlegen muß.

- **1954/336 a-c:** Solche Inventarnummern werden im Regelfall (mit vielen Ausnahmen) eher für zusammengehörige Objekte vergeben. Die Problematik entspricht genau dem vorigen Beispiel, auch hier kann ein Programm die nicht genannte Nummer 1954/336 b normalerweise nicht finden, auch hier kann die Schreibweise variieren.

Eine häufige zusätzliche Schwierigkeit kann entstehen, wenn der Inventarisator nur einen von zwei leicht verwechselbaren Buchstaben (z.B. I bzw. J) benutzte (das gibt es auch bei Buchstaben-Kürzeln). Dann können weder Mensch noch Maschine diese Zwischenwerte ausrechnen.

- **1954/ 336 a - 338 c:** Dies sieht zunächst wie eine Variation der beiden vorigen Beispiele aus, bringt aber zusätzliche Schwierigkeiten. Kein Mensch (und damit auch keine Maschine) kann ohne zusätzliche Informationen feststellen, welche Nummern sich außer den genannten dahinter verbergen. Gab es das Stück 1954/336 k ?
- **M VII 478 und M IX 12:** Dieser Fall ist relativ selten und findet sich fast nur in Katalogen oder auf Karteikarten. Gründe hierfür können sein: nachträgliche Zusammensetzung von bereits nummerierten Fragmenten eines Objekts, Arbeitersparnis (eine Karteikarte für mehrere ähnliche Objekte), Orientierung an einem Sammelfoto mit mehreren Objekten.

Sortierung und Retrieval machen erhebliche Schwierigkeiten.

Die meisten Programmsysteme setzen sich mit dem Problem der Sammel- und Bereichsangaben nicht auseinander.

Tricks zur Umgehung solcher Probleme: "Die Dokumenten-Nummer"

Wegen dieser Schwierigkeiten schlagen viele Software-Systeme vor, die Objekte bzw. die Inventareinträge mit einer zusätzlichen, solche Probleme vermeidenden, Nummer zu

versehen. Bei manchen Systemen werden diese Nummern in Bereiche aufgeteilt und können dann z.B. als Grundlage für eine hierarchische Gliederung komplexer Objekte benutzt werden (z.B. MIDAS-HIDA).

Dieses Verfahren einer unabhängigen Numerierung hat Vor- und Nachteile:

- **Vorteile:** das Problem der Sortierung bei der Druckausgabe oder auch bei der "richtigen" Reihenfolge für Korrekturen am Bildschirm kann dadurch gelöst werden, daß man die Stücke in der entsprechenden Reihenfolge eingibt.
- **Nachteile:** diese richtige Reihenfolge gilt allerdings nur für **eine** Nummer, ein Sortieren nach mehreren Nummern (z.B. Inventar-, Katalog- und Negativnummer) ist nach wie vor ausgeschlossen. Auch beim Retrieval einzelner Nummern kann es Schwierigkeiten geben (Variationen der Schreibweise vgl. Abb. 28 bzw. Abb. 30), ähnliches gilt natürlich für Sammel- und Bereichsangaben.

Eine vom Zufall der überlieferten Numerierung unabhängige Vergabe von Nummern kann trotzdem notwendig werden. Dieser Fall tritt immer dann auf, wenn Sie für die Dokumentation Ihre Objekte anders definieren, als das die bereits vorliegende Numerierung nahelegt oder diese Objekte in "Teilobjekte", die dann keine eigene Nummer haben, zerlegen.

Diese Fragen der "Datenstrukturierung" werden im zweiten Hauptteil behandelt, Aussagen zu einer von der Inventarnummer unabhängigen Numerierung finden Sie im Kapitel "Darstellung von Zusammenhängen durch Verweise" auf Seite 84 ff.

Was muß ein Programm können, um Inventarnummern richtig zu sortieren ?

Da solche Numerierungssysteme nun einmal bei allen Inventarisierungsarbeiten eine grundlegende Rolle spielen, sollte man ernsthaft überlegen, ob man das Problem nicht durch Zusatzprogramme lösen kann. Dabei ist aber immer zu beachten: Was bei der Druckausgabe funktioniert, kann bei der Bildschirmausgabe trotzdem Schwierigkeiten machen (umgekehrt ist das kaum der Fall).

Sortieren für Druckausgaben

Die wichtigste Eigenschaft für das Sortieren von Inventarnummern besteht in der Fähigkeit des Programms, den Wechsel von Buchstaben zu Zahlen (mit oder ohne Trenner) "selbständig" zu erkennen und solche komplexen Gebilde daher auch ohne das lästige Auffüllen mit Nullen und dazu ohne Vereinheitlichung der Trenner richtig sortieren.

Das entspricht - programmtechnisch - einer Zerlegung des Datenfeldes "Inventarnummer" in mehrere Datenfelder, die dann jeweils alphanumerisch oder numerisch zu sortieren wären³⁷. Wie eine solche, aus mehreren "Sortierbegriffen" bestehende "Sortierdatei" aussieht, sahen wir bereits bei den systematischen Registern (Abb. 16 auf Seite 36). Die ja auch für viele andere Felder geltende Frage, was man eigentlich in ein einziges Datenfeld stecken könne, wird im zweiten Hauptteil systematisch behandelt (auf Seite 64 ff.).

Solange die normale Sortierreihenfolge ("von links nach rechts") gilt, machen die meisten Arten von Inventarnummern keine Probleme - man kann sie erfassen "wie sie sind" und trotzdem richtig verarbeiten. Die in den Abbildungen 22-25 und 28-30 gegebenen Beispiele machen dann keine Probleme.

³⁷) Diese Art der Zerlegung machen sich manche museumsspezifischen Programmsysteme zunutze; so können z.B. bei MIMSY (Willoughby Associates) die Inventarnummern bereits bei der Erfassung durch den Benutzer in mehrere Datenfelder zerlegt werden.

Wie weit man mit Hilfe standardisierbarer Zusatzprogramme kommen kann zeigt die Abb. 31 auf Seite 57 (rechte Spalte). Das dort für die rechte Spalte verwendete Zusatzprogramm weiß zwischen Buchstaben und Zahlen zu unterscheiden.

Abb. 31: Kombination mehrerer Arten von Inventarnummern in einem Museum (Beispiele aus Abb. 22-27):

Linke Spalte: Richtig sortiert (unterschiedliche Nummernarten voneinander getrennt)
Mittlere Spalte: Falsche alphanumerische Sortierung (die Nummern gehen wahllos durcheinander).
Rechte Spalte: Sortierung mit Zusatzprogramm (Standard-GOS), das Zahlen und Buchstaben zu unterscheiden weiß. Nachgestellte Sortierbegriffe (z.B. 1/1973) und römische Zahlen (z.B. M IX 4711) würden eine zusätzliche Programmierung erfordern.

Richtige Reihenfolge	Alphanumerische Sortierung	Zusatz-Programm
1	1	1
2	1/1973	1/1973
5	10	2
10	10/1964	2/1973
20 a	1618	5
1618	1618/1988	5/1973
A 20	173.1	10
L 5	173.15	10/1964
M 1	173.2	20/1953
M 2	1969.1618	20 a
M 10	1973.1	26.1618
Z 1618	1973.10	173.1
1969.1618	1973.2	173.2
1973.1	1973.20	173.15
1973.2	1973.5	297.20
1973.5	2	297.103
1973.10	2/1973	1618
1973.20	20 a	1618/1988
26.1618	20/1953	1969.1618
173.1	26.1618	1973.1
173.2	297.103	1973.2
173.15	297.20	1973.5
297.20	5	1973.10
297.103	5/1973	1973.20
20/1953	A 20	A 20
10/1964	L 5	L 5
1/1973	L I 1	L I 1
2/1973	L II 2	L II 2
5/1973	M 1	M 1
1618/1988	M 10	M 2
L I 1	M 2	M 10
L II 2	M I 10	M I 5
M I 5	M I 5	M I 10
M I 10	M IX 4711	M IX 4711
M VII 20	M VII 20	M VII 20
M IX 4711	Z 1618	Z 1618

Trotzdem gibt es auch dann noch bestimmte Grenzen, die nur mit Hilfe zusätzlicher, gegebenenfalls museumsspezifischer Programme überschritten werden können. Dabei handelt es

sich grundsätzlich um Dinge, die auch ein menschlicher Bearbeiter der einzelnen Inventarnummer nicht ansehen kann und die sich daher jeder Analyse per Programm entziehen. Die wichtigsten Beispiele hierfür sind:

- **Buchstabe oder römische Zahl ?** Niemand kann bei Inventarnummern wie I 147.11 oder V 26 a aus der einzelnen Nummer erkennen, ob es sich bei den vorangestellten Kürzeln um Buchstaben oder römische Zahlen handelt. Die beiden Arten werden also miteinander vermischt.
- **Jahreszahl oder laufende Nummer ?** Bei Inventarnummern wie 73/64 oder 12/49 ist nicht zu erkennen, ob sich hinter einer dieser Zahl eine Jahreszahl versteckt (die dann gegebenenfalls durch 18 .. oder 19 .. zu ergänzen wäre) und ob die Jahreszahl voran- oder nachgestellt ist. Auch diese beiden Arten werden miteinander vermischt.
- **Exoten aller Art.** Beispiele sind selten, aber einprägsam. Wer hätte gedacht, daß die Inventarnummern B 421 A und B 121 B in einem rheinischen Museum zunächst nach dem nachgestellten Buchstaben, dann nach dem vorangestellten und erst zuletzt nach der Zahl sortiert gehören ? Was soll das arme Computerprogramm machen, wenn ein Bearbeiter aus einer nicht mehr rekonstruierbaren Laune (?) heraus plötzlich römische (statt der sonst durchgehend verwendeten arabischen) Zahlen als "Unternummern" benutzt ? Wie ist die Inventarnummer K 4711 a-z + 1 zu interpretieren ?

Soweit man solche Fälle überhaupt per Programm lösen will, muß man lernen, Inventarnummern zu analysieren und daraus Tricks bzw. Spielregeln für die Maschine zu entwickeln. Daraus kann sich im Extremfall ergeben, daß man verschiedene Arten von Nummern nicht in das gleiche Datenfeld stopfen darf.

Voraussetzung für jede vernünftige Ordnung ist also, daß das Programm erfährt, welche Inventarnummer wie zu behandeln ist. Das Programm kann dann - ohne daß das irgendwo sichtbar ist - die Nummern intern so umformen, daß sie vernünftig sortiert werden können. Solche Anwendungen sollten nur nach Rücksprache mit den Programmentwicklern versucht werden.

Zusatzprogramme für Sammel- und Bereichsangaben

Ein Zusatzprogramm kann Sammel- und Bereichsangaben nur dann verarbeiten, wenn sie in einer bestimmten Weise gemacht werden. Die folgenden Beispiele bzw. Konventionen wurden dem Projekt "Kleine Museen" entnommen:

- **Sammelangaben:** Die Inventarnummern mußten vollständig erfaßt und durch ein in keiner Inventarnummer vorkommendes Zeichen ("&") voneinander getrennt werden, also z.B. M VII 478 & M IX 12 (und nicht M VII 478 & IX 12).

Für die **Druckausgabe** wurde dann nach allen angegebenen Nummern sortiert, auf dem **Bildschirm** nach der "kleinsten".

- **Bereichsangaben:** Als Trenner diente das Minuszeichen ("-"), das Programm versuchte dann, die nicht genannten Zwischenwerte auszurechnen. Das ging natürlich nur, wenn die Angaben in aufsteigender Reihenfolge gemacht wurden (also z.B. A 317 - A 320 oder 1954/336 a-c, nicht aber A 320 - A 317 oder 1954/336 c-a). Die eventuell vorangestellten Teile der Inventarnummer, die nicht durch das Minuszeichen verbunden sind, wurden diesen (ausgerechneten) Zwischenwerten vorangestellt. Diese vorangestellten Elemente durften entweder nur bei der ersten Angabe vorkommen (z.B. 1954/336 a-c) oder mußten identisch sein (z.B. A 317 - A 320). Unverständliche Angaben wie 1954/336 a - 338 c wurden zurückgewiesen.

Für die **Druckausgabe** wurde nach der "kleinsten" Nummer sortiert, für das **Retrieval** standen auch die **Zwischennummern** zur Verfügung.

Beispiel für das richtige Sortieren von Bildschirmausgaben

Für das Projekt "Kleine Museen" wurden Programme entwickelt, um auch (aus Computersicht) total "verrückte" Arten von Inventarnummern für die Druckausgabe wie für den Bildschirm richtig sortieren zu können.

Mit diesen Programmen konnten auch nachgestellte Kürzel und römische Zahlen in denselben richtig verarbeitet werden. Die Voraussetzung war eine sehr ins Detail gehende Systemanalyse, deren Ergebnisse dann in speziellen Dateien gespeichert und für diese Aufgaben dann von einem Standardprogramm benutzt werden konnten. Ein durchaus erwünschter Nebeneffekt dieses Vorgehens war, daß eine gewisse "Wortschatzkontrolle" erfolgte: Nummern, die dem für ein bestimmtes Museum ermittelten Schema nicht entsprachen (z.B. unbekannte Kürzel) wurden bereits bei der Eingabe zurückgewiesen.

Wer solche Leistungen von einem Programm erwartet, kommt um eine detaillierte Systemanalyse nicht herum. Wie man das macht, wird im Anhang ("Arbeitsblätter zur Beschreibung von Numerierungssystemen", Seite 106 ff.) genau beschrieben.

2. Hauptteil: Inhaltliche Grundlagen für die Definition von Datenfeldern und -strukturen

Aus dem 1. Hauptteil ("Formale Grundlagen für die Definition von Datenfeldern") sollten ein paar wichtige Erkenntnisse hängengeblieben sein:

- Wenn der Computer etwas Vernünftiges mit unseren Daten machen soll, dann müssen wir die Daten so formulieren, daß der Computer etwas damit anfangen kann.
- Computer ist nicht gleich Computer - an vielen Stellen wurde auf die unterschiedliche Leistungsfähigkeit von Anwenderprogrammen hingewiesen und erklärt, wie man solche Programme gegebenenfalls testen kann.

Diese Erkenntnisse bezogen sich - trotz kleinerer Ausflüge in Themen wie "Hilfsdateien" - grundsätzlich auf die verschiedenen Arten (Drucken, Sortieren, Retrieval) der Verarbeitung einzelner Datenfelder. Im Zentrum stand die Formulierung von Daten, die "Syntax" bzw. die Schreibweise.

Wir müssen jetzt also versuchen, das im 1. Hauptteil erarbeitete Wissen über die formalen Grundlagen für die Definition von einzelnen Datenfeldern mit den Inhalten unserer Dokumentation zu verbinden und dafür zu sorgen, daß uns der Computer auch dort nicht im Stich läßt.

Wenn Sie den ersten Hauptteil dieses Handbuchs und die **Pflichtlektüre** (Heft 30 der "Materialien") nicht nur oberflächlich angesehen, sondern wirklich studiert haben, werden Ihnen viele der jetzt zu behandelnden Dinge schon weitgehend bekannt sein. Nehmen wir also an, Sie hätten sich mit dem Konzept des Datenfeldes wirklich vertraut gemacht und bei dem dafür wichtigen Teil der Systemanalyse³⁸ eine **Liste von Datenfeldern** und (hoffentlich !) auch Erläuterungen dazu erarbeitet.

Solche "Datenkataloge" gibt es natürlich bereits in großer Zahl; abgesehen von Spezialdokumentationen zu einzelnen Objektgattungen weisen sie alle eine große Familienähnlichkeit auf. Vielleicht haben Sie bei der Erarbeitung Ihres Datenkatalogs sogar versucht, hier nicht alles selber zu erfinden, sondern sich mit anderen abzustimmen.

Für die Zwecke dieses Handbuchs nehmen wir uns einen solchen gemeinsam erarbeiteten Datenkatalog, der sich ganz bewußt auf wenige Datenfelder beschränkt. Es geht uns bei dieser Wahl eines Beispiels natürlich nicht darum, diesen speziellen Datenkatalog nun im einzelnen zu analysieren und zu kritisieren. Wir brauchen aber ein konkretes Beispiel dafür, wie sich Kollegen, die zum Teil bereits erste Erfahrungen mit dem Computer haben, die Sache vorstellen (s. Abb. 32 auf Seite 61).

Wenn Sie schon einmal versucht haben sollten, ein solches "Fangblatt" auszufüllen, müßten Sie neben geklärten oder ungeklärten Fragen der Syntax, die wir hier nicht wieder aufwärmen wollen, eigentlich auch auf ein paar Dinge gestoßen sein, die im 1. Hauptteil dieses Handbuchs noch nicht besprochen wurden. Was macht man z.B. wenn:

- mehr als eine Angabe erfaßt werden soll (z.B. "Material: *Gold, Silber, Porzellan*"),
- Angaben nur für Teile eines Objekts gelten ("*Kaffeekanne aus Porzellan mit Griff aus Silber*"),
- Angaben nicht wirklich in das Feld passen (z.B. beim Fundort: schreibt man da auch Dinge wie "*Höhle im So-und-so-Berg*" ?),
- mehrere leicht unterschiedliche Objekte unter einer Inventarnummer erfaßt wurden.

38) **Pflichtlektüre:** Einen guten Einstieg zu diesen Fragen gibt Heft 30 S. 19: "Übung 3: Das Datenfeld-Arbeitsblatt". Weitere Erläuterungen zu dieser Übung finden Sie hier auf Seite 97 ff.

Abb. 32: Beispiel für einen bewußt knapp gehaltenen Datenkatalog

Arbeitskreis Computereinsatz in Münchner Museen

Gemeinsamer Datenkatalog

(Stand: 18.09.1990)

Ort des Museums

Museumsname

Standort

Inventar-Nummer

Objektbezeichnung

Sachgruppeneinteilung

Darstellung

Titel

Funktion

Herstellungsort

Fundort

Gebrauchsort

Erwerbungsart

Provenienz

Entwerfer

Hersteller/Ausführung

Verleger/Vertrieb

Hersteller der Vorlage

Datierung

von

bis

verbale Angaben

Material

Technik

Maße

Foto-Nummer

Vielleicht würden Sie diese Angaben ganz einfach "so wie sie sind" in den Datenkatalog eintragen und darauf vertrauen, daß der Computer damit schon irgendwie klar kommen wird. Papier ist geduldig. Vielleicht hat die Lektüre des ersten Hauptteils aber bereits erste Zweifel an diesem Computervertrauen gesät, wenn ja, dann sind Sie auf dem richtigen Weg. Es gibt hier nämlich eine Menge Dinge, die man beim Ausfüllen eines Fangblatts gar nicht merken kann und die dann später trotzdem zu Schwierigkeiten führen.

Der wirkliche Test eines Datenkatalogs besteht nicht im Ausfüllen (das natürlich ein wichtiger erster Schritt ist), sondern in der Erzeugung der gewünschten Produkte.

Historische Daten haben nämlich dummerweise ein paar Eigenschaften, die bei kommerziellen Anwendungen praktisch keine Rolle spielen und für die die meisten handelsüblichen Anwenderprogramme infolgedessen keine praktikablen oder gar komfortablen Lösungen anbieten.

Wir werden zunächst versuchen, einen Überblick über die im 2. Hauptteil zu behandelnden Probleme zu bekommen. Die Schwierigkeit dabei ist, daß die jetzt zu betrachtenden "inhaltlichen" Zusammenhänge viel komplexer sind als die im 1. Hauptteil beschriebenen. Es war relativ einfach zu erklären, daß man bestimmte Daten als Schlagwörter formulieren muß, damit die Maschine daraus ein alphabetisches Register machen kann. Es ist wesentlich schwerer zu erklären, was die Maschine an "Datenstruktur" braucht, damit sie soetwas dann auch zuverlässig tun kann. Nicht jedes Datenbanksystem wird von alleine verstehen, daß Angaben wie "Material: Gold/Silber" erst auseinandergenommen werden müssen, bevor man ein vernünftiges alphabetisches Register machen kann.

Damit die Maschine mit solchen Angaben umgehen kann, braucht sie entweder Zusatzprogramme oder eine geeignete "Datenbankstruktur" (bei dem eben erwähnten Beispiel sogenannte "Mehrfachfelder"). Geschwindigkeit und Effizienz der Verarbeitung von Daten sind von solchen Dingen stark abhängig und wir müssen versuchen zu verstehen, was man mit welcher Datenstruktur machen kann und wie sich das auf die Definition der Datenfelder auswirkt. Der Ihnen jetzt wahrscheinlich noch etwas unklare Begriff "Datenstruktur" wird sich am Ende des 2. Hauptteils mit Leben gefüllt haben.

Selbst (oder gerade) wenn Sie "nur ganz einfache Sachen" wollen, werden Ihnen zwei eng zusammenhängende Fragenkomplexe begegnen, die sich in keinem Falle umgehen lassen und die in eigenen Kapiteln veranschaulicht werden:

- **Was kann ich alles in ein einzelnes Datenfeld stecken ?** Hier geht es zunächst um die grundsätzlichen Anforderungen an die inhaltliche Definition von Datenfeldern (Seite 64 ff.). Wir betrachten die Datenstruktur aus der Sicht des einzelnen Datenfeldes, ohne uns um größere Zusammenhänge zu kümmern.

Im Vordergrund steht die Auseinandersetzung mit den "real existierenden" Daten, die sich oft nur schwer in ein vorgegebenes Schema pressen lassen. Diese Daten sind von sehr unterschiedlicher Qualität (Unterkapitel "Was tut man mit ungenauen Daten?") bzw. liegen in sehr unterschiedlichen Formulierungen vor (Unterkapitel: "Was macht man mit historischen Daten?").

Diese erste inhaltliche Auseinandersetzung mit dem Datenkatalog kann verschiedene Auswirkungen haben: sie kann zu einer Vermehrung der Datenfelder führen oder auch zu der Erkenntnis, daß man halt auf ein paar Produkte verzichten muß. Egal, wie Sie sich entscheiden, bleiben wir zunächst bei dem Modell einer relativ einfachen, listenmäßigen Auflistung von Datenfeldern, einer sogenannten "**flachen**" Datenstruktur, wie sie zahlreiche kommerzielle (= im Computershop um die Ecke kaufbare) allgemeine Datenbanksysteme bieten.

- **Was ist ein Museumsobjekt ?** In einem zweiten Schritt (Seite 75 ff.) betrachten wir die zum einzelnen Objekt vorliegenden Daten unter dem Gesichtspunkt weiterer, bisher noch

nicht besprochener, inhaltlicher Zusammenhänge wie z.B. zwischen "Tasse" und "Service", "Fibel" und "Graburne". Wir entfernen uns also vom einzelnen Datenfeld und kümmern uns um die Struktur von "Datensätzen".

Wir werden dabei feststellen, daß es einen wesentlichen Unterschied macht, ob man seine Daten selber definiert und erzeugt (wie bei einem Warenlager) oder sie von der Geschichte in mehr oder weniger vollständigem bzw. fragmentarischem Zustand überliefert bekommt. Menge, Differenziertheit und auch die Form dieser Daten lassen sich oft nicht mehr in eine einfache, "flache" Datenstruktur pressen.

Obwohl die beiden Fragenkomplexe sehr eng zusammenhängen, werden die Auswirkungen auf die Datenstruktur zunächst in getrennten Kapiteln anhand konkreter Beispiele besprochen. Nach dem Durcharbeiten dieser Kapitel sollten Sie eine klarere Vorstellung von den Vor- und Nachteilen bestimmter, heute üblicher Lösungen haben.

Sie werden dabei nicht zuletzt die Grenzen fest vordefinierter Datenkataloge erkennen. Dinge, die man selbst bei der detailliertesten Systemanalyse leicht übersehen kann, die Einbeziehung weiterer Quellen oder auch Neuerwerbungen können die Anforderungsprofile drastisch verändern. Diese Situation hat schon früh zu einer **Suche nach einfacheren und flexibleren Datenstrukturen**, in denen solche Probleme leichter zu lösen sind, geführt (Seite 87 ff.).

Fazit: Die Auseinandersetzung mit Datenstrukturen ist nicht nur für den "Anfänger" eine schwierige Sache, auch die Fachleute werden hier oft verschiedener Meinung sein. Dies ist aber kein Grund, sich hier vor Entscheidungen zu drücken. **Die Wahl der richtigen Datenstruktur ist ein wesentlicher Faktor für Kosten und Effizienz einer Computerisierung.**

Was kann ich eigentlich alles in ein einzelnes Datenfeld stecken ?

Wir müssen zunächst die Frage klären, was man sinnvollerweise in ein einzelnes Datenfeld stecken kann. Bisher haben wir das Datenfeld sehr formalistisch betrachtet, jetzt müssen wir uns verstärkt um "Inhalte" kümmern. Gibt es hier ähnlich strenge Abhängigkeiten wie die zwischen "Syntax" und Produkt ?

Die Datenfelder, über die wir im 1. Hauptteil sprachen, waren sogenannte "Gebundene Deskriptoren", sie bestanden grundsätzlich aus zwei Teilen, dem **Feldnamen** und dem **Feldinhalt**, also z.B. "Objektbezeichnung: *Kleiderschrank*".

Im 1. Hauptteil haben wir uns im wesentlichen um die produktgerechte Formulierung der Feldinhalte gekümmert. Wir betrachteten sie als Zeichenfolgen, die gedruckt, sortiert und gegebenenfalls gesucht und gefunden werden sollten. Je nach dem gewünschten Produkt mußten bestimmte Regeln eingehalten werden. Daraus ergab sich, etwas vereinfacht gesagt, daß ein Feld weitgehend durch seine Syntax bestimmt wurde. Inhalte mit verschiedener Syntax konnten nicht in das gleiche Feld gesteckt werden (z.B. die Datierungen auf Seite 20) oder bedurften komplizierter Zusatzprogramme (z.B. die Inventarnummern im Anhang zum ersten Hauptteil).

Welche Funktion hat nun eigentlich der Feldname ?

- **Technisch betrachtet** ist er nichts anderes als der Name einer "Adresse", unter der die Maschine die betreffenden Angaben finden kann. Seine einzige Funktion besteht also darin, Feldinhalte gezielt aufsuchen und gegebenenfalls bearbeiten zu können.

Es ist der Maschine im Prinzip völlig egal, ob wir das Feld "XYZ", "123" oder "Objekt" nennen, sie versteht Feldnamen ebeno wenig wie Feldinhalte. Daran ändert auch die Tatsache nichts, daß manche Anwenderprogramme hier Restriktionen haben (z.B. begrenzte Zeichenzahl, nur Buchstaben, nur Zahlen, 1. Zeichen Buchstabe usw.).

- **Inhaltlich betrachtet** vertritt der Feldname ein Konzept, mit dessen Hilfe ich die Daten in einer angemessenen Weise erfassen **und** die Produkte inhaltlich sinnvoll gestalten kann.

Ein relativ einfacher Fall liegt z.B. dann vor, wenn alle Feldinhalte eines einzigen Feldes zu einem in sich abgeschlossenen alphabetischen Register umgeformt werden sollen (z.B. Künstlerregister). Etwas komplizierter liegt der Fall, wenn Inhalte mehrerer Felder in einem solchen Register vereinigt werden sollen (z.B. allgemeine Namensregister) oder wenn sogar verschiedene Sortierkriterien wirksam werden sollen (vgl. das alte Lexikon von 1905, in dem zunächst nach dem Nachnamen, dann nach den Lebensdaten sortiert wurde, auf Seite 24).

Durch den Bezug auf bestimmte Produkte, egal ob die nun gedruckt werden oder nur als Ergebnis eines Retrieval auf dem Bildschirm erscheinen, bekommen wir ein paar erste Anhaltspunkte für Kriterien, die uns bei der inhaltlichen Definition von Datenfeldern helfen sollten:

Die 1. Faustregel für die inhaltliche Definition von Datenfeldern ist "produktbezogen": Alle Inhalte, die bei der Herstellung irgendeines gewünschten Produkts eine selbständige Rolle spielen, sollten eigene Datenfelder bekommen.

Mit dieser produktbezogenen Faustregel können wir bereits eine Menge anfangen. Wenn wir - um bei den oben genannten Beispielen zu bleiben - ein Künstlerregister wollen, muß die Maschine irgendwie erkennen können, wo die Namen der Künstler stehen. Die einfachste (aber natürlich nicht die einzige) Weise das zu erreichen, besteht darin, den Künstlernamen ein eigenes Datenfeld zu geben und dieses also von den Datenfeldern für andere Hersteller-namen zu unterscheiden.

Wir müssen uns dabei aber auch an die vielen im 1. Hauptteil genannten Hinweise auf die unterschiedliche Leistungsfähigkeit aller Anwenderprogramme erinnern. Obwohl es schön und wünschenswert wäre, eine formale und inhaltliche Datenstrukturierung ohne jeden Gedanken an bestimmte Anwenderprogramme vorzunehmen, so ist das (bisher ?) noch nicht möglich.

Wenn Ihnen jemand erzählt, seine Datenstrukturierung sei völlig unabhängig von irgendwelchen Computerprogrammen (und damit noch in hundert Jahren aktuell), dann versteht er entweder nicht viel vom Geschäft oder er schwindelt.

Bevor wir uns aber mit solchen komplizierten Sachen auseinandersetzen, machen Sie erst mal eine kleine Übung. Wir nehmen uns die Datenfelder zu Ortsangaben aus dem kleinen Münchner Datenkatalog (Abb. 32 auf Seite 61) und schauen, was wir damit machen können.

Mit dieser Übung überprüfen Sie, ob Sie zwischen "Grobkategorien" und "Datenfeldern" unterscheiden können³⁹. Je nachdem, welche Produkte Sie erzeugen möchten, wird sich der betreffende Teil ihres Datenkatalogs als das eine oder das andere erweisen. Wenn Sie dann eine erste Vorstellung über ganz konkrete Abhängigkeiten zwischen inhaltlicher Datenstruktur und Produkten haben, hat die Übung ihren Sinn erfüllt. Wir werden diese Übung dann am Ende des Kapitels noch einmal aufgreifen.

Abb. 33: Übung Von der Grobkategorie zum Datenfeld (I)

Stellen Sie sich vor, Sie hätten bei Ihren Ortsangaben zu irgendeinem Thema (z.B. Fundort) Inhalte wie *Oberaichbach, Niederaichbach, Landshut, Niederbayern, Freistaat Bayern* usw., also Gemeindeteile, Gemeinden, Kreise bzw. kreisfreie Städte, Regierungsbezirke, Bundesländer usw.

Welche Produkte kann man erzeugen (oder auch nicht), wenn man

- Alle diese Angaben in ein einziges Feld (z.B. "Fundort") steckt ?
- Gemeinden, Kreise, Bundesländer usw. zu eigenen Feldern macht ?

Es reicht, wenn Sie hier einfache (alphabetische) und systematische Register bzw. Suchanfragen wie "alle Stücke aus Niederbayern" als Beispiel nehmen.

Haben Sie daran gedacht, daß *Landshut* sowohl eine Gemeinde als auch ein Kreis ist ?

Und noch eine wichtige Frage: Welche dieser Datenfelder müssen Sie grundsätzlich immer ausfüllen, um welches Produkt zu erzeugen oder welche Art von Suchanfrage stellen zu können ?

Die Richtigkeit Ihrer Antworten können Sie im Kapitel "Was kann man mit einfachen ("flachen") Datenstrukturen machen ?" (Seite 67 ff.) überprüfen.

Was tut man mit ungenauen Daten ? (Rembrandt oder niederländisch 17. Jh. ?)

Egal, welche Datenfelder Sie sich ausgedacht haben - spätestens bei der Erfassung des einzelnen Objekts werden Sie feststellen, daß die Ihnen vorliegenden Informationen oft nicht

39) Pflichtlektüre: Zum Begriff "Grobkategorie" siehe Heft 30 S. 18-19, "Übungen 1 bzw. 3".

genau in diese vorbereiteten Schubladen passen. Über manche Stücke wissen Sie viel, über andere wenig. Daraus ergeben sich sowohl inhaltliche als auch technische Probleme.

Nehmen wir zunächst ein **Beispiel aus einem Kunstmuseum** (wir werden später sehen, daß das Problem auch anderswo auftaucht), einem Museum also, das in jedem Falle wissen will, wer das gute Stück "gemacht" hat, weil der Künstler nun mal das grundlegende Ordnungskriterium darstellt.

Von diesen Künstlern wissen wir oft eine ganze Menge. Der Münchner Datenkatalog (Seite 61) geht darauf ein, indem er verschiedene Felder für "Entwerfer", "Hersteller/Ausführung", "Verleger/Vertrieb" und "Hersteller der Vorlage" vorsieht. Oft wissen wir aber nur sehr wenig, z.B. kennen wir nicht immer den "Namen". Solche Fälle umschreiben wir dann traditionellerweise mit Dingen wie "Notnamen" ("Meister des Paradiesgärtleins", "Meister E.S."), Angaben zur "Werkstatt" ("Rembrandt-Werkstatt"), zum "Umkreis" oder zur "Schule" ("Schule von Barbizon"), vielleicht muß man sogar auf eine "Kunstlandschaft" ("oberrheinisch um 1520") oder eine "Stilbezeichnung" ausweichen.

Wir brauchen dieses Beispiel hier nicht weiter zu vertiefen, wir wollen nur kurz skizzieren, was sich daraus für die von Ihnen zu bestimmende Datenstruktur ergibt. Sie haben hier ein paar sehr schwierige Entscheidungen zu treffen; wenn Ihnen das das Gefühl gibt, den Teufel mit Beelzebub auszutreiben, haben Sie nicht so unrecht:

- **Inhaltlich:** Wenn Sie differenziert vorgehen wollen, werden Sie eine komplexe Datenstruktur bzw. eine große Zahl von Datenfeldern brauchen. Wenn Sie "nur ganz einfache Dinge" wollen, muß der mit der maschinellen Erfassung beauftragte Mitarbeiter jeweils entscheiden, welche Differenzierungen er nolens volens "wegschmeißt" oder wie er die Daten so "umformuliert", daß sie doch noch irgendwie in Ihr Schema passen (das wiederum auch "technisch" stimmen muß).
- **Technisch:** Im 1. Hauptteil haben wir gelernt, daß unsere Datenfelder **formal** so präzise definiert sein müssen, daß man daraus vernünftige Produkte erzeugen kann. Daraus ergab sich, daß man nur solche Informationen in die gleiche Kategorie "stopfen" sollte, die nach den gleichen Kriterien sortiert werden können und im gleichen Register erscheinen sollten. Die oben genannten Arten, einen Künstler so genau wie möglich einzugrenzen, sollten sicherlich nicht in ein einziges Register und daher auch nicht in ein einziges Datenfeld gepreßt werden.

Wahrscheinlich ist Ihnen längst klar, daß die bunte Mischung aus präzisen und unscharfen Daten, mit denen wir nun mal arbeiten müssen, nicht nur bei den "Künstlern" auftritt.

Weitere Fälle sind leicht zu finden. Sie entstehen quasi automatisch, wenn die gesuchten Informationen nicht unabhängig vom Objekt überliefert sind (extrinsische Daten wie "Rembrandt lebte von dann bis dann"), sondern wenn man versucht, die Antwort auf eine Frage durch eine fachmännische Bestimmung des Objekts so genau wie möglich "einzukreisen". Denken Sie also nicht nur an die Frage nach dem Künstler, sondern auch an die bei fast allen Gattungen gewünschten Datierungen, an die Schwierigkeit, Material und Technik bei einer alten Handzeichnung oder einem Steinbeil genau zu bestimmen usw.

Immer zu bedenken: Die "Paßgenauigkeit" zwischen den vorliegenden Daten und dem Datenkatalog sollte auch unter praktischen, letztlich finanziellen Aspekten der maschinellen Datenerfassung geprüft werden. Immer wenn eine Umformung der Daten notwendig ist, sollten Sie sich überlegen, wer diese Arbeit machen soll und welche Kenntnisse die dafür bestimmte Person haben müßte.

Fazit: Ungenaue, nicht präzise in einen Datenkatalog passende Daten sind unser täglich Brot. Für den Computer ist es natürlich völlig egal, ob die Daten differenzierter oder gröber als die Datenfelder in Ihrem Datenkatalog sind.

Was macht man mit "historischen" Daten ? (Chemnitz oder Karl-Marx-Stadt ?)

Ähnliches wird Ihnen auch bei ganz "genauen" Daten begegnen. Ein besonders anschauliches Beispiel bieten hier Ortsangaben, die im übrigen zu den ganz besonders kniffligen Fragen gehören. Wenn Ihnen jemand sagt, er habe das Problem der Ortsangaben gelöst, so wird Ihnen dieses Kapitel reichlich Material bieten, um seine Aussagen nachzuprüfen.

Das Problem ist, kurz gesagt, daß sich der Geltungsbereich eines Begriffes historisch verändern kann und daß sich solche Veränderungen durchaus in den von Ihnen benutzten Quellen niedergeschlagen haben können.

Das in der Kapitelüberschrift gegebene Problem (Chemnitz oder Karl-Marx-Stadt ?) ist noch relativ leicht zu lösen, denn bei den meisten Dokumentationen spielen solche Unterschiede keine wichtige Rolle. Man kann die beiden Namen in den meisten Fällen als Synonyme behandeln.

Schwieriger wird die Sache, wenn ein Statuswechsel vorliegt, also z.B. aus einer selbständigen Gemeinde ein Ortsteil wird. Wirklich dramatisch wird es aber erst bei den Namen von Ländern und Regionen. Gehört "Straßburg" zu "Deutschland", wo ordnen Sie "Ostpreußen" ein ?

Das Problem wird Ihnen in zwei Formen begegnen:

- **Bei der Datenerfassung:** Wenn der "engste" oder genaueste Begriff eine solche, d.h. eine ganze "historische" Region betreffende, Ortsangabe ist.
- **Bei systematischen Registern bzw. beim Retrieval:** Wenn Sie auf der Basis eines solchen "Oberbegriffs" etwas drucken oder suchen wollen.

"Krieg oder Kreisreform" sind vielleicht die anschaulichsten, bestimmt aber nicht die einzigen Gründe für solche historisch bedingten Komplikationen. Beispiele für sich entwickelnde Bedeutungen bestimmter Begriffe werden Sie leicht auch bei Objektbezeichnungen, Material, Technik usw. finden.

Ansätze zu einer Lösung gibt es natürlich. Man kann hier z.B. analog zu den oben (Kapitel "Zusammenfassung: Wie benutzt der Computer Hilfsdateien?" auf Seite 42) diskutierten Künstlernamen verfahren und damit solche vieldeutigen Begriffe durch geeignete Homonymenzusätze "eindeutig" machen. Man kann auch dafür sorgen, daß die Information über die Quelle dieser Bezeichnung miterfaßt wird, denn in einem alten Buche stört uns eine Bezeichnung wie "Ostpreußen" nicht. Eine einfache und elegante Lösung dieses Problems ist nicht bekannt, Vor- und Nachteile der üblichen Lösungen müssen Sie aber genau kennen (s. das folgende Kapitel).

Fazit: Auch hier ist es dem Computer natürlich völlig egal, warum ein Begriff einmal die eine und ein anderes mal eine andere Bedeutung hat. Wenn Sie bei der Definition von Datenfeldern nicht auf solche Dinge achten, vermischen sich im Computer "ungenaue" und "genaue, aber historisch bedingte" Daten auf eine für spätere Benutzer oft unentwirrbare Weise.

Listenartige Datenkataloge mit einfacher ("flacher") Datenstruktur

Spätestens an dieser Stelle sollten Sie gemerkt haben, daß einfache Dinge schwerer zu machen sind als komplexe. Wenn Sie einfache und gut überschaubare Datenkataloge benutzen wollen, müssen Sie das Verbiegen von Daten und eine Beschränkung auf wenige Produkte in Kauf nehmen. Wenn Sie umgekehrt jeder nur möglichen Bedeutungsschattierung ein eigenes Datenfeld geben, werden Sie schnell hunderte oder gar tausende von Datenfeldern bekommen. Solche aufgeblähten Regelwerke sind dann allerdings meist nur noch von dafür extra ausgebildeten Dokumentaren zu verstehen.

Ein Ausweg aus diesem Dilemma wird seit langer Zeit gesucht und natürlich gibt es da auch ein paar recht elegante Lösungsansätze. Sie verlangen alle, daß wir uns von dem bisher benutzten starren Konzept einer "flachen", listenartig aufgebauten Datenstruktur mit autonomen, nur aus Feldname und Feldinhalt bestehenden Datenfeldern entfernen.

Bevor wir solche flexibleren Datenstrukturen beschreiben und uns überlegen, ob es überhaupt eine Software gibt, die mit solchen Datenstrukturen arbeiten kann, konstruieren wir uns eine Serie von aufeinander aufbauenden Beispielen. Wir gehen dabei von den gewünschten Produkten aus:

- **Gewünschte Produkte:** Als Produkte möchten wir entsprechend dem "Stufenmodell für die Formulierung von Textdaten" (Seite 45 ff.) die beiden, die für die Definition von Datenstrukturen am wichtigsten sind:
 - **Alphabetische Register** (bzw. ein auf einzelne Begriffe bezogenes Retrieval) der in den Beispielen verwendeten Einzelangaben (z.B. "Herstellungsort: Oberaichbach").
 - **Systematische Register** (bzw. ein entsprechendes, auf zusammenhängende Gruppen von Informationen bezogenes Retrieval) von der Art "Alle Stücke aus Bayern". Diese Produkte bzw. Fragen wollen wir sowohl differenziert (nach den Feldern des Münchner Datenkatalogs auf Seite 61, also nach **Museumsort, Herstellungsort, Fundort, Gebrauchsort, Erwerbungsart**) als auch allgemein (also "Stücke, die im Lauf ihrer Geschichte was mit Niederbayern zu tun hatten") erzeugen bzw. beantworten können.

Aus mnemotechnischen Gründen erfinden wir nun Objekte, bei dem alle der im Datenkatalog vorkommenden Ortsangaben vorkommen, aber (aus weiter unten zu diskutierenden Gründen) nur einmal. Dabei werden wir schrittweise vorgehen.

- **Das erste beschriebene Objekt:** Nehmen wir an, es handelt sich um ein aus Familienbesitz in **Landshut** 1912 vom Bayerischen Nationalmuseum in **München** erworbenes Schmuckstück (Armreif), dessen Geschichte sich aus der glücklicherweise noch erhaltenen Familienchronik genau rekonstruieren läßt: Es war ein eigenhändig hergestelltes Verlobungsgeschenk des Dorfschmieds von **Oberaichbach** (heute ein Ortsteil von **Niederaichbach**), der mit seiner Frau direkt nach der Heirat nach **Neunkirchen** zog. Seine Frau zog nach seinem Tode zu ihrer Verwandtschaft nach **Miltenberg**, wo sie alsbald starb und auf dem Peters-Friedhof begraben wurde. Der Armreif wurde bei Ausschachtungsarbeiten daselbst gefunden und der inzwischen nach **Landshut** verzogenen Familie übergeben.
- **Hintergrundwissen:** Je nach Datenkatalog benötigt der Bearbeiter bestimmte Kenntnisse. In unserem Beispiel sollte er etwas zur Verwaltungszugehörigkeit⁴⁰ der vorkommenden Ortsnamen wissen. Nehmen wir also jemanden an, der fleißig und diszipliniert genug ist, die Angaben in einem Ortsbuch zu überprüfen und immer nach dem gleichen Schema zu machen. In dem hier gewählten geographischen Namenbuch finden wir für die Verwaltungszugehörigkeit Angaben in der Reihenfolge Land, Regierungsbezirk, Kreis (bzw. kreisfreie Stadt), Gemeinde (bei Gemeindeteilen die zugehörige Gemeinde):

Landshut: Bayern, Niederbayern, Kreisfreie Stadt, Landshut

Miltenberg: Bayern, Unterfranken, Miltenberg, Miltenberg

München: Bayern, Oberbayern, Kreisfreie Stadt, München

Neunkirchen: Nordrhein-Westfalen, Arnsberg, Siegen, Neunkirchen

Neunkirchen: Rheinland-Pfalz, Koblenz, Westerwaldkreis, Neunkirchen

Neunkirchen: Rheinland-Pfalz, Trier, Bernkastel-Wittlich, Neunkirchen

Neunkirchen: Baden-Württemberg, Karlsruhe, Neckar-Odenwald-Kreis, Neunkirchen

40) Nach: Geographisches Namenbuch Bundesrepublik Deutschland, Band 1, bearbeitet und herausgegeben vom Institut für Angewandte Geodäsie, Frankfurt am Main 1981.

Neunkirchen: Bayern, Unterfranken, Miltenberg, Neunkirchen
Neunkirchen: Baden-Württemberg, Stuttgart, Schwäbisch Hall, Michelfeld
Oberaichbach: Bayern, Niederbayern, Landshut, Niederaichbach

Die Zusammenhänge zwischen bestimmten Produkten, den Daten von Objekten und dem im Kopfe des Bearbeiters oder in der Maschine zur Verfügung stehenden Hintergrundwissen werden wir nun an verschiedenen flachen Datenstrukturen durchspielen.

Datenstruktur I: Flache Datenstruktur mit je einer Angabe pro Datenfeld

Als Ausgangspunkt nehmen wir Ortsangaben aus dem einfachen Datenkatalog auf Seite 61 und die Übung auf Seite 65. Davon ausgehend werden wir versuchen, etwaige Defizite durch eine andere Syntax bzw. eine andere Datenstruktur auszugleichen (Abb. 34):

Abb. 34: Datenstruktur I. Flache Datenstruktur mit je einer Angabe pro Datenfeld	
Vorkommende Ortskategorien (vorschriftsmäßig, d.h. nach Kontrolle der Schreibweise im Ortsbuch, ausgefüllt):	
Ort des Museums:	München
Herstellungsort:	Oberaichbach
Fundort:	Miltenberg
Gebrauchsort:	Neunkirchen
Erwerbungsort:	Landshut
Vorteile: Leicht (= ohne jedes Hintergrundwissen) auszufüllen Alphabetische Register können hergestellt werden Systematische Register können nicht hergestellt werden.	

Dem großen Vorteil eines mühelosen Ausfüllens steht hier eine eng begrenzte Verarbeitungsmöglichkeit entgegen. Man kann das drucken, man kann auch alphabetische Register erzeugen, anspruchsvollere Produkte sind zunächst nicht möglich. Ich möchte aber schon jetzt darauf hinweisen, daß sich auf dieser Datenstruktur sehr gut weitere Bearbeitungsstufen aufbauen lassen (vgl. mit der weiter unten auf Seite 73 abgebildeten "Datenstruktur IV").

Man kann die oben gewünschten systematischen Register also nur dann erzeugen, wenn man die dafür nötigen Informationen der Maschine in irgendeiner Form zur Verfügung stellt. Drei häufig gewählte Möglichkeiten werden im folgenden vorgestellt:

- **Benutzung einer komplizierten Syntax:** (s.u. "Datenstruktur II" auf Seite 70)
- **Vermehrung der Datenfelder:** (s.u. "Datenstruktur III" auf Seite 71 f.)
- **Benutzung einer Hilfsdatenbank:** (s.u. "Datenstruktur IV" auf Seite 73)

Sie werden sehen, daß das gar nicht so schwer ist.

Datenstruktur II: Flache Datenstruktur und komplizierte Syntax

Wir veranschaulichen zunächst, welche Probleme man durch einen Übergang zu einer sehr differenzierten Syntax zumindest teilweise lösen kann. Solche Verfahren erfordern, daß vor der Datenerfassung alle für ein systematisches Register notwendigen Informationen vollständig und in einer immer gleichen Reihenfolge eingetragen werden müssen. Wir nehmen also an, daß der betreffende Bearbeiter die Ortsangaben mit Hilfe eines bestimmten Ortsbuches ergänzt (incl. an sich umzuarbeitender Angaben wie "Kreisfreie Stadt") und dabei vieldeutige Angaben wie "Neunkirchen" klärt (Abb. 35):

Abb. 35: Datenstruktur II. Flache Datenstruktur und komplizierte Syntax (mehr als eine Angabe pro Datenfeld)	
Vorkommende Ortskategorien (vorschriftsmäßig ausgefüllt):	
Ort des Museums:	Bayern, Oberbayern, München
Herstellungsort:	Bayern, Niederbayern, Landshut, Niederaichbach, Oberaichbach
Fundort:	Bayern, Unterfranken, Miltenberg
Gebrauchsort:	Bayern, Unterfranken, Miltenberg, Neunkirchen
Erwerbungsort:	Bayern, Niederbayern, Landshut
Auswirkungen auf die Produkte: Die gewünschten Produkte können mit Einschränkungen hergestellt werden:	
<ul style="list-style-type: none"> - Alphabetische Register: Nur durch die Schlagwörter voneinander trennende Zusatzprogramme - Retrieval nach Einzelbegriffen: Nur durch Volltextrecherche bzw. die Schlagwörter voneinander trennende Zusatzprogramme - Systematische Register: Durch einfaches Sortieren der betreffenden Felder - Systematisches Retrieval: Nur durch Volltextrecherche bzw. die Schlagwörter voneinander trennende Zusatzprogramme - Defizite: Hohe Kosten durch Redundanz und Fehleranfälligkeit der Datenerfassung 	

Man kann also leicht den Teufel mit Beelzebub austreiben, durch Erleichterungen für das systematische Register das alphabetische erschweren. Eine Verbesserung wäre hier die Einführung von "Mehrfachfeldern" in die Datenstruktur, dazu Näheres bei dem an anderen Beispielen erklärten Vergleich von unserer Datenstruktur II (Abb. 40 auf Seite 78) mit der solche Anforderungen erfüllenden Datenstruktur V (Abb. 41 auf Seite 79).

Die eben beschriebenen, heute altertümlich anmutenden Datenstrukturen werden immer noch in vielen Projekten benutzt. Oft werden die Angaben nicht in absteigender, sondern aufsteigender "Sortierreihenfolge" gemacht, manchmal wird der Benutzer aufgefordert, zumindest die Eindeutigkeit der Angaben zu sichern (im Beispiel: Kreis oder kreisfreie Stadt, welches *Neunkirchen* usw.). Solche "Vereinfachungen" ändern aber nichts an den grundsätzlich beschränkten Verarbeitungsmöglichkeiten.

Datenstruktur III: Flache, aber sehr differenzierte Datenstruktur

Rein technisch betrachtet ist eine **inhaltliche Differenzierung der Datenfelder** wesentlich effizienter als die in der Datenstruktur II verwendete komplizierte Syntax. Wir müssen dafür den bestehenden Datenkatalog mit Ortsangaben für jedes Thema (Museum, Herstellung, Fund, Gebrauch und Erwerb) um Dinge wie Gemeindeteile, Gemeinden, Kreise bzw. kreisfreie Städte, Regierungsbezirke, Bundesländer usw. ergänzen. Das bläht die Datenstruktur zwar gewaltig auf (s. Abb. 36, wir benötigen hierfür mehr als eine Seite), hat aber später bei der Erzeugung von Produkten erhebliche Vorteile.

Abb. 36: Datenstruktur III. Flache, aber sehr differenzierte Datenstruktur und einfache Syntax (Blatt 1)
Vorkommende Ortskategorien (vorschriftsmäßig ausgefüllt):
Ort des Museums (Land): Bayern
Ort des Museums (Regierungsbezirk): Oberbayern
Ort des Museums (Kreis bzw. kreisfreie Stadt): München
Ort des Museums (Gemeinde): -
Ort des Museums (Gemeindeteil): -
Herstellungsort (Land): Bayern
Herstellungsort (Regierungsbezirk): Niederbayern
Herstellungsort (Kreis bzw. kreisfreie Stadt): Landshut
Herstellungsort (Gemeinde): Niederaichbach
Herstellungsort (Gemeindeteil): Oberaichbach
Fundort (Land): Bayern
Fundort (Regierungsbezirk): Unterfranken
Fundort (Kreis bzw. kreisfreie Stadt): Miltenberg
Fundort (Gemeinde): -
Fundort (Gemeindeteil): -
(Fortsetzung siehe nächste Seite)

Datenstruktur III: Fortsetzung (Blatt 2)
Gebrauchsort (Land): Bayern
Gebrauchsort (Regierungsbezirk): Unterfranken
Gebrauchsort (Kreis bzw. kreisfreie Stadt): Miltenberg
Gebrauchsort (Gemeinde): Neunkirchen
Gebrauchsort (Gemeindeteil): -
Erwerbungsart (Land): Bayern
Erwerbungsart (Regierungsbezirk): Niederbayern
Erwerbungsart (Kreis bzw. kreisfreie Stadt): Landshut
Erwerbungsart (Gemeinde): -
Erwerbungsart (Gemeindeteil): -
<p>Auwirkungen auf die Produkte: Die gewünschten Produkte können ohne Einschränkungen hergestellt werden:</p> <ul style="list-style-type: none"> - Alphabetisches Register bzw. Retrieval nach Einzelbegriffen: Durch Einbeziehung der gewünschten Datenfelder - Systematische Register: Durch einfaches Sortieren der betreffenden Felder in einer geeigneten Sortierreihenfolge - Systematisches Retrieval: Durch Direktretrieval in den betreffenden Datenfeldern - Defizite: Hohe Fehleranfälligkeit und Kosten der Datenerfassung durch Redundanz

Diese Art der technisch begründeten Vermehrung von Datenfeldern ist heute noch bei vielen Unternehmungen üblich. Immer wenn in einem Datenkatalog Datenfelder erscheinen, deren Inhalte sich wie z.B. Oberbegriff zu Unterbegriff ("*Bayern*" zu "*Oberbayern*") oder wie Synonyme (z.B. hochsprachliche und mundartliche Bezeichnungen) verhalten, sollte man sich die Frage stellen, ob das wirklich nötig ist bzw. ob man solche Informationen nicht in irgendwelche Hilfsdateien auslagern könnte.

Datenstruktur IV: Flache Datenstruktur und Benutzung einer Hilfsdatenbank

Wie schon an anderer Stelle am Beispiel der Erzeugung systematischer Register zu Objektbezeichnungen (Seite 33 ff.) ausführlich dargelegt, lassen sich viele solcher Defizite durch die **Benutzung von Hilfsdatenbanken** (z.B. Thesauri) ausgleichen. Wenn wir also das oben zitierte Ortsbuch in eine solche Hilfsdatenbank umformen und dabei mehrfach vorkommende Ortsnamen nach einheitlichen Regeln eindeutig machen, können wir auf etwas zurückkommen, was auf den ersten Blick genauso aussieht wie die knappe Datenstruktur I auf Seite 69), sich aber durch die Verknüpfung mit der Hilfsdatenbank doch grundsätzlich unterscheidet: Diese "Datenstruktur IV (Abb. 37 auf Seite 73) vermeidet die oben festgestellten Defizite.

Abb. 37: Datenstruktur IV. Flache Datenstruktur und Benutzung einer Hilfsdatenbank	
Vorkommende Ortskategorien (vorschriftsmäßig, d.h. incl. Klärung etwaiger Homonyme, ausgefüllt):	
Ort des Museums:	München (kreisfreie Stadt)
Herstellungsort:	Oberaichbach
Fundort:	Miltenberg (kreisfreie Stadt)
Gebrauchsort:	Neunkirchen (5)
Erwerbungsort:	Landshut (kreisfreie Stadt)
Auswirkungen auf die Produkte: Die gewünschten Produkte können durch Benutzung der Hilfsdatenbank ohne Einschränkungen hergestellt werden	
Defizite: Bei dem als Beispiel gewählten Objekt keine.	

Man darf sich das Arbeiten mit Hilfsdateien allerdings auch nicht zu leicht vorstellen. Die Probleme, die wir vorher in der "Objektdatei" lösten (oder auch nicht), werden zunächst nur auf eine andere Datei verlagert. Diese Hilfsdatei braucht also ebenfalls eine angemessene, die Erfassung der Daten und die Erzeugung der gewünschten Produkte erlaubende, Datenstruktur.

Die Benutzung von Hilfsdateien ist für viele Probleme eine sehr elegante Lösung, stellt aber vergleichsweise hohe Anforderungen an das Datenbanksystem. Selbst wenn Sie nur nach einem einzelnen Begriff suchen (z.B. "alle Objekte, die im Kreis Landshut hergestellt wurden"), muß diese Suche über die Hilfsdatenbank laufen

Zusammenfassung: Was sind die Grenzen fest vordefinierter Datenkataloge ?

Sie können jetzt also einen Teil der in der Übung auf Seite 65 gestellten Fragen ganz präzise beantworten und damit erste kausale Zusammenhänge zwischen den bisher besprochenen einfachen ("flachen") Datenstrukturen und grundlegenden Produkten wie alphabetischen und systematischen Registern erkennen. Die Benutzung von in solcher Weise inhaltlich fest vordefinierten Datenstrukturen funktioniert so lange, bis Sie auf Informationen stoßen, die nicht in die vorbereiteten Schubladen passen. Ein solches Objekt erfinden wir nun:

Das zweite beschriebene Objekt: Es handelt sich um ein Stück Felsmalerei aus der Höhle Labaume-Latrone bei Sainte-Anastasie im französischen Département Gard, das 1954 dem Heimatmuseum in Oberaichbach (heute ein Ortsteil von Niederaichbach) aus einer Privatsammlung in Miltenberg überlassen wurde.

Das oben beschriebene **Hintergrundwissen** bzw. die Datenfelder, die es aufnehmen sollen, müssen nun um ein paar Dinge ergänzt werden, denn die Felder für Land, Regierungsbezirk, Kreis (bzw. kreisfreie Stadt), Gemeinde (bei Gemeindeteilen die zugehörige Gemeinde) stellen "Schubladen" dar, in die unsere neuen Daten nur schlecht passen.

Ich möchte Ihnen nicht die ganze Arbeit abnehmen und schlage Ihnen daher vor, die Auswirkungen dieses neuen Objekts auf die Datenstrukturen I-IV selber zu erarbeiten (s.u. "Übung: Von der Grobkategorie zum Datenfeld (II)" auf Seite 74).

Abb. 38: Übung Von der Grobkategorie zum Datenfeld (II)

Wiederholen Sie die Übung auf Seite 65 unter Benutzung der Datenstrukturen I-IV. Sie sollten dabei die folgenden Fragen zusätzlich klären:

Was passiert, wenn Sie Funde aus anderen Ländern haben? Schreiben Sie ein französisches "département" in das Feld "Bundesland" oder "Regierungsbezirk" oder machen Sie dafür ein eigenes Datenfeld auf? Wirkt sich diese Entscheidung auf Ihre Produkte aus und gegebenenfalls wie?

Was machen Sie mit den Namen von Höhlen oder anderen, verwaltungsmäßig nicht eindeutig zuzuordnenden Angaben?

Wie muß man den Datenkatalog bzw. das komplementäre Hintergrundwissen (= Hilfsdatenbank) strukturieren?

Bitte beantworten Sie diese Fragen unter dem Aspekt der gewünschten Produkte (alphabetische Register, Retrieval von Einzelbegriffen, systematisches Register bzw. Retrieval).

Wir stoßen hier auf das bei historischen Daten zentrale Problem, daß wir zwar bereits vorliegende Quellen sehr sorgfältig analysieren, deshalb aber noch lange nicht zuverlässig vorher-sagen können, welche Informationen wir in unseren Quellen finden werden. Der Ankauf neuer Objekte oder auch die Erforschung der bisher schon vorhandenen können eine Datenstruktur schnell veralten lassen. Wir merken daran, wie starr selbst noch ganz modern anmutende Lösungen sind.

Auswirkungen auf die Produkte: Leicht auszufüllende Datenstrukturen wie die "Datenstruktur I" auf Seite 69 erlauben nur die Erzeugung sehr einfacher Produkte. Bis zu einem gewissen Grad kann man das durch eine komplizierte Syntax ausgleichen ("Datenstruktur II" auf Seite 70), solche Lösungen sind aber teuer und fehleranfällig. Eine sichere Erzeugung von etwas anspruchsvolleren Produkten ist erst dann möglich, wenn man ausreichend differenzierte Strukturen in der betreffenden Datei ("Datenstruktur III" auf Seite 71 f.) oder in einer Hilfsdatenbank ("Datenstruktur IV" auf Seite 73) einsetzt. Die Lösung mit Hilfsdatenbanken führt uns auf ein sehr einfaches und im Grunde auch sehr flexibles Erfassungsschema zurück, dafür muß aber der Preis der Homonymenkontrolle bzw. einer systematischen Gliederung des Wortschatzes bezahlt werden.

Fazit: Die inhaltliche Definition von Datenfeldern sollte also die folgenden Faustregeln beachten:

- Alle Inhalte, die bei irgendeinem gewünschten Produkt eine selbständige Rolle spielen, sollten eigene Datenfelder bekommen.
- Das Simulieren solcher Felder durch "Trenner" (wie in der Datenstruktur II (Abb. 35 auf Seite 70) ist bestenfalls ein Notbehelf mit erheblichen Nachteilen.
- Bei allen Inhalten, die nicht das einzelne Objekt, sondern nur Zusatzinformationen zu einzelnen Attributen desselben betreffen, sollte die Auslagerung in eine Hilfsdatei versucht werden.

Im nächsten Kapitel werden wir uns überlegen was passiert, wenn die benötigten Datenfelder vom rein Inhaltlichen her zwar alle vorhanden sind, die Daten aber trotzdem nicht so recht in diese Schubladen passen wollen.

Was ist ein Museumsobjekt ?

Anders als bei einem Warenlager, können Sie das nicht ganz frei nach Ihren eigenen Regeln definieren. Manchmal haben Sie nur eine "Kaffeetasse", manchmal das ganze "Kaffeegeschirr"; Sie haben vielleicht den "Kopf eines Heiligen", vielleicht aber auch die ganze Figur des "St. Florian" oder sogar den ganzen "Hochaltar" mit Skulpturen, Ornamenten und Gemälden. Das "Steinbeil" ist nur ein Teil eines Fundzusammenhangs, die "Skizze" nur ein Teil eines "Skizzenblatts" oder eines "Skizzenbuchs" oder gar eines Konvoluts.

Wieso ist das ein Problem ? Oberflächlich betrachtet können wir in ein Datenfeld Objektbezeichnung natürlich genau so gut "Kaffeetasse" wie "Service", "Kopf eines Heiligen" oder "Hochaltar" schreiben. Nur was passiert, wenn wir wissen, wozu unser Stück gehört, wovon es ein Teil ist ? Wohin schreiben wir das ? Solche Zusammenhänge sind zwar oft nicht bekannt und bei manchen Gattungen auch unüblich (ein neuzeitliches Tafelbild wird nur selten ein Teil eines größeren Werks sein), in anderen Fällen aber sind sie wichtig, vielleicht sogar "wichtiger" als das Einzelobjekt ("*Fragment einer Schlange vom Pergamonaltar*").

Komplexe Strukturen sind also kein Privileg bestimmter Sammelgebiete oder besonders detaillierter Dokumentationen. Immer wenn Sie sich für Zusammenhänge interessieren und diese dokumentieren wollen, haben Sie das Problem. Informationstechnisch betrachtet ist es allerdings ganz egal, aus welchen Gründen Museumsobjekte komplex strukturiert sein können. Die folgenden Beispiele sollen ein Gefühl dafür vermitteln, daß solche Dinge ebenso gut ganz banal wie hochwissenschaftlich begründet sein können:

- **Banal: Vergabe von Inventarnummern.** Sie definieren das Museumsobjekt dadurch, daß Sie ihm eine Nummer geben. Dabei werden Sie eine Reihe von Entscheidungen zu treffen haben. Soll man jedem einzelnen Teil des Geschirrs eine selbständige Nummer geben, vergibt man besser "Unternummern" oder wie oder was ? Wie bringt man es fertig, den folgenden Widerspruch zu beseitigen: in einem Falle habe ich ein einzelnes Steinbeil (ganz selbstverständlich mit eigener Inventarnummer und Fundplatz), in einem anderen Falle habe ich einen großen Fundzusammenhang mit vielleicht hunderten von einzelnen Stücken (incl. Steinbeile); muß ich die nun ebenfalls Stück für Stück einzeln inventarisieren ?

Solche Entscheidungen trifft zunächst der "Zufall", der das gute Stück in's Museum brachte. Anschließend haben dann oft schon Ihre Vorgänger durch ihre Art der Inventar-nummervergabe zur Klärung oder Verwirrung beigetragen. Jetzt sitzen Sie auf einem Berg unterschiedlich strukturierter Daten und sollen ein vernünftiges Ablageschema entwickeln.

- **Wissenschaftlich: Sie haben "zu viele" Informationen.** Ein vorzüglich erforshtes und dokumentiertes Sammlungsobjekt hat z.B. nicht nur einen Hersteller, sondern mehrere. Das kann ein "Auto" oder ein "Altar" oder ein signierter "*Barockschmuck mit antiken Gemmen*" (welche meinetwegen ebenfalls signiert sein sollen) sein. Diese namentlich bekannten Hersteller haben eventuell nicht nur an verschiedenen Orten und zu verschiedenen Zeiten gelebt, jeder hat bestimmte Teile unseres Sammlungsobjekts gemacht, die wiederum aus ganz verschiedenen Materialien sind. Selbst der bewußt ganz einfach gehaltene Münchner Datenkatalog auf 61 greift dieses Problem auf. Wenn Sie "nur ganz einfache Sachen" wollen und infolgedessen versuchen, diese Informationen in ein einfaches Schema (z.B. je ein Feld für Hersteller, Objektbezeichnung, Datierung, Lokalisierung und Material) zu pressen, entsteht Unfug.

Wenn im übrigen Ihr seliger Vorgänger eine Zigarrenschachtel mit "*Webgewichten*" oder "*Münzen*" aus verschiedenen Zeiten unter einer Inventarnummer abgelegt und damit wie oben beschrieben als ein Sammlungsobjekt definiert hat, werden Sie mit Ihrem "einfachen" Erfassungsschema auf ganz ähnliche Schwierigkeiten stoßen. In dem Moment, in dem Sie Aussagen zum einzelnen Stück machen wollen, müssen Sie das Problem anpacken. Was

schreiben Sie bei der Zigarrenschachtel mit "*Webgewichten*" oder "*Münzen*" in das Feld Datierung ? "*Antike bis Neuzeit*" ?

Diese Phänomene werden oft übersehen, oft verdrängt man sie auch mit dem einfach zu postulierenden, aber schwer zu realisierenden Anspruch, daß man ja nur ganz einfache Sachen machen wolle. Die Auswirkungen auf die Datenstruktur werden im nächsten Kapitel anhand konkreter Beispiele beschrieben.

Dem Computer ist es egal, ob zu viele unterschiedliche Objekte unter einer Inventarnummer zusammengefaßt wurden oder ob es sich um komplex strukturierte Objekte handelt. Informationstechnisch sind die beiden Fälle identisch.

Komplexe Datenstrukturen (Mehrfach- und Gruppenfelder, Verweise)

Im einleitenden Kapitel "Was ist ein Museumsobjekt ?" hatten wir ein paar Fälle kennengelernt, bei denen aus inhaltlichen Gründen mehrere, gelegentlich sogar widersprüchliche Informationen zu einem Datenfeld vorkommen. Solche Fälle treten immer dann auf, wenn wir über ein Objekt viel aufschreiben wollen oder wenn in einer Dokumentation mehrere, nicht identische Objekte als "Konvolut" behandelt werden sollen.

Wir behandeln diese Fälle erst jetzt, weil sich die daraus entstehenden Probleme zu einem guten Teil an den oben besprochenen Datenstrukturen I-IV veranschaulichen lassen. Um hier Lösungen zu finden, müssen wir allerdings das Spektrum der Möglichkeiten zur Datenstrukturierung erweitern, eine Sache, die leider in vielen Programmsystemen nicht vorgesehen ist und daher gegebenenfalls durch Zusatzprogramme "simuliert" werden muß.

Auch hier werden wir versuchen, die gewünschten Datenstrukturen mit Hilfe einiger weniger Beispiele zu erklären. Damit Sie nun nicht den Eindruck bekommen, daß solche Dinge nur bei Ortsangaben auftauchen, nehmen wir uns Beispiele mit anderen Datenarten; wir wählen uns **Objektbezeichnung, Material und Datierung** und konstruieren ein paar Beispiele:

- **Beispiel 1: Einfach strukturierte Objekte mit je einer einzigen Information pro Datenfeld**
- **Beispiel 2: Einfach strukturierte Objekte mit mehreren Informationen pro Datenfeld**
- **Beispiel 3: Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern**
- **Beispiel 4: Konvolut aus einfach strukturierten Objekten**

Die für die Erfassung bzw. Verarbeitung solcher Fälle erforderliche größere Flexibilität hat allerdings zwei Seiten. Um hier mit den Füßen auf dem Boden zu bleiben, müssen wir uns immer wieder klarmachen, daß es ja nicht ausreicht, komplexe Strukturen zu erlauben bzw. zur Verfügung zu stellen, wir müssen auch dafür sorgen, daß die Maschine diese Strukturen sinnvoll verarbeiten kann. Hierbei spitzt sich alles auf die Frage zu, wie die Maschine erkennen kann, welche Daten zusammengehören und welche nicht. Bisher war das ganz einfach: Alle in einem Datensatz vorkommenden Informationen gehörten zusammen und Datensätze sind für die Maschine leicht zu erkennen. Wir werden weiter unten schnell erkennen, daß wir dieses einfache Prinzip peu-à-peu aufweichen müssen, um komplexere Sachverhalte beschreiben zu können.

Um zu zeigen, wie grundlegend solche Probleme sind, setzen wir unsere Anforderungen an die Produkte herab und betrachten nur die alphabetischen Register. Die Einbeziehung systematischer Register würde unsere Erklärungen nur unnötig komplizieren ohne neue Erkenntnisse beizutragen; die Anforderungen systematischer Register an die Daten(felder) bleiben die gleichen, egal ob diese in einfache oder komplexe Datenstrukturen "verpackt" sind.

Wir kümmern uns in den folgenden Beispielen also nicht um das Problem der Unterscheidung von mehreren Ebenen von "Oberbegriffen" (z.B. Gemeinde, Kreis, Regierungsbezirk usw.). Wir wollen hier "nur" alphabetische Register bzw. ein Retrieval, bei dem ich einzelne oder auch mehrere Datenfelder gleichzeitig benutzen kann⁴¹. Bei jedem Beispiel werden wir uns fragen, wie sich unsere jeweilige Datenstruktur auf diese Produkte auswirkt.

Wir wollen nun zunächst versuchen, die als Beispiele gewählten Objekte mit Hilfe der uns schon bekannten (I-IV) und - wenn nötig - zu erweiternden Datenstrukturen (V-VII) zu erfassen. Wir wählen Objektbezeichnung, Material und Datierung (angelehnt an den Münchner Datenkatalog, ohne bei der Datierung "von" bzw. "bis" jeweils auszufüllen, weil das das Beispiel nur unnötig komplizieren würde).

Beispiel 1: Einfach strukturierte Objekte mit je einer einzigen Information pro Datenfeld

Die Objekte: Wir haben in unserer Auswahl einen Schmuckkasten mit einer Perlenstickerei aus der Mitte des 19. Jahrhunderts. Dieser enthält einen Halsschmuck aus Silber ("Jugendstil"), je eine Brosche aus Silber ("20-er Jahre") und Gold ("Biedermeier"), einen einzelnen goldenen Ohrring ("kaiserzeitlich"), eine ungefaßte Karneol-Kamee ("kaiserzeitlich") sowie einen Ehering aus Platin ("1954").

Diese Objekte können wir in einer flachen Datenstruktur mit einfacher Syntax ("Datenstruktur I" wie auf Seite 69) leicht erfassen, jedenfalls solange wir sie als voneinander völlig unabhängige Einzelobjekte behandeln. Das Ergebnis sehen Sie in Abb. 39. Daß zwischen diesen Objekten Zusammenhänge bestehen, erkennt man hier nur an den Inventar-nummern, welcher Art diese Zusammenhänge sind, muß man sich "denken" (dazu Näheres in den Datenstrukturen VI bzw. VII auf den Seiten 83 bzw. 84).

Abb. 39: Datenstruktur I. Flache Datenstruktur mit je einer Angabe pro Datenfeld (wie Abb. 34 auf Seite 69)

Beispiel 1: Einfach strukturierte Objekte mit je einer einzigen Information pro Datenfeld

Objekt-Nr.:	1
Objektbezeichnung:	Schmuckkasten
Material:	Perlenstickerei
Datierung:	Mitte 19. Jh.

Objekt-Nr.:	1 a
Objektbezeichnung:	Halsschmuck
Material:	Silber
Datierung:	Jugendstil

Objekt-Nr.:	1 b
Objektbezeichnung:	Brosche
Material:	Silber
Datierung:	20-er Jahre

(Fortsetzung siehe nächste Seite)

41) Pflichtlektüre zu grundlegenden Retrievalarten: Heft 30 S. 41 ff.

Datenstruktur I. Flache Datenstruktur mit je einer Angabe pro Datenfeld : Fortsetzung von Seite 77

Objekt-Nr.:	1 c
Objektbezeichnung:	Brosche
Material:	Gold
Datierung:	Biedermeier

Objekt-Nr.:	1 d
Objektbezeichnung:	Ohrring
Material:	Gold
Datierung:	Kaiserzeit

Objekt-Nr.:	1 e
Objektbezeichnung:	Kamee
Material:	Karneol
Datierung:	Kaiserzeit

Objekt-Nr.:	1 f
Objektbezeichnung:	Ehering
Material:	Platin
Datierung:	1954

Ausfüllbarkeit: Keine Probleme

Auswirkungen auf die Produkte: Das gewünschte Retrieval sollte keine Probleme machen

Beispiel 2: Einfach strukturierte Objekte mit mehreren Informationen pro Datenfeld

Die Objekte: Wir wählen eine Brosche aus Gold und Silber ("1872") und einen Halsschmuck mit dazugehörigen Ohrringen aus Gold (1986).

Beim Eintragen in unseren Datenkatalog müssen wir nun versuchen, pro Datenfeld mehr als eine Information unterzubringen. Wir benutzen hierfür zunächst eine uns bereits bekannte Möglichkeit, die "Datenstruktur II" (Abb. 40), bei der man durch die Definition von "Trennern" mehrere Informationen in einem einzigen Datenfeld unterbringen kann. Dem Computer ist es natürlich völlig egal, ob sich hinter einer solchen Syntax Ober- und Unterbegriffe (wie Bundesland, Regierungsbezirk, Kreis usw.) oder Begriffswiederholungen (wie Gold / Silber) verbergen.

Abb. 40: Datenstruktur II. Flache Datenstruktur und komplizierte Syntax (wie Abb. 35 auf Seite 70)**Beispiel 2: Einfach strukturierte Objekte mit mehreren Informationen in einem Datenfeld**

Objekt-Nr.:	2 a
Objektbezeichnung:	Brosche
Material:	Gold / Silber
Datierung:	1872

(Fortsetzung siehe nächste Seite)

Datenstruktur II. Flache Datenstruktur und komplizierte Syntax: Fortsetzung von Seite 78	
Objekt-Nr.:	2 b
Objektbezeichnung:	Halsschmuck / Ohrring
Material:	Gold
Datierung:	1986
Ausfüllbarkeit: Nach Definition eines geeigneten "Trenners" (hier: Schrägstrich) keine Probleme	
Auswirkungen auf die Produkte: Nur mit sehr starken Einschränkungen erzeugbar (Volltextrecherche bzw. die Schlagwörter voneinander trennende Zusatzprogramme)	

Wie schon oben festgestellt, kann eine komplizierte Syntax zwar eine Erfassung solcher Daten ermöglichen, eine Erzeugung etwas anspruchsvollerer Produkte wird aber Schwierigkeiten machen.

Eine bei manchen Anwendungen übliche Lösung dieses Problems besteht in der Definition zusätzlicher Datenfelder, etwa "Material 1", "Material 2", "Material 3" usw. Dies ist erforderlich, wenn die Software nur eine Information pro Datenfeld erlaubt und auch die Verarbeitungsprogramme so etwas nicht simulieren können. Hier treibt man natürlich den Teufel mit Beelzebub aus, denn man muß nicht nur von vornherein festlegen, wieviel Feldwiederholungen man erlauben will, man muß diese Felder dann auch bei allen Verarbeitungsprogrammen angemessen berücksichtigen.

Die für solche Fälle besser geeignete Lösung besteht in etwas, das uns stark an die oben besprochene "Datenstruktur III" (Abb. 36 auf Seite 71 f.) erinnern sollte, nur daß statt einer inhaltlichen Differenzierung (wie Bundesland, Regierungsbezirk, Kreis usw.) nun erlaubt sein soll, das gleiche Datenfeld mehrfach zu benutzen. Diese neue "Datenstruktur V" enthält also bestimmte Datenfelder (in den Beispielen "Material", "Objektbezeichnung") mehrfach, diese werden als selbständige Felder behandelt (Abb. 41):

Abb. 41: Datenstruktur V. Flache Datenstruktur mit Mehrfachfeldern und einfacher Syntax	
Beispiel 2: Einfach strukturierte Objekte mit mehreren Informationen in einem Datenfeld	
Objekt-Nr.:	2 a
Objektbezeichnung:	Brosche
Material:	Gold
Material:	Silber
Datierung:	1872
Objekt-Nr.:	2 b
Objektbezeichnung:	Halsschmuck
Objektbezeichnung:	Ohrring
Material:	Gold
Datierung:	1986
Ausfüllbarkeit: Keine Probleme	
Auswirkungen auf die Produkte: Die gewünschten Produkte können ohne Einschränkung erzeugt werden	

Merke: Mehrfachfelder sind Datenfelder, die mehr als eine Angabe enthalten dürfen⁴².

Die gewünschten Produkte lassen sich erzeugen, weil die Daten relativ einfach strukturiert sind: "Gold" und "Silber" beziehen sich beide gleichermaßen auf "Brosche", "Halsschmuck" und "Ohrring" sind beide aus "Gold". Entscheidend ist hier nicht die Zahl der Angaben, sondern daß alle Angaben auch für alle anderen Angaben gelten.

Die Grenze der Wirksamkeit von Mehrfachfeldern werden wir beim folgenden Beispiel erkennen.

Beispiel 3: Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern

Die Objekte: Wir haben zunächst einen modernen Ring aus Weißgold ("1943") mit einer griechischen Silbermünze ("hellenistisch"), dann einen historistischen ("ca. 1870") Halsschmuck aus Gold mit sechs antiken Gemmen aus Karneol ("kaiserzeitlich") und sechs modernen Kameen aus Lava ("Ende 18. Jh."). Die Angabe "Weißgold" gilt hier nur für den "Ring", nicht aber für die "Münze", die Datierung "kaiserzeitlich" nur für "Gemme", die Datierung "Ende 18. Jh." nur für "Kamee" usw. Das ergibt das folgende Bild (Abb. 42):

Abb. 42: Datenstruktur V. Flache Datenstruktur mit Mehrfachfeldern und einfacher Syntax

Beispiel 3: Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern

Objekt-Nr.:	31
Objektbezeichnung:	Ring
Objektbezeichnung:	Münze
Material:	Weißgold
Material:	Silber
Datierung:	1943
Datierung:	hellenistisch

Objekt-Nr.:	32
Objektbezeichnung:	Halsschmuck
Objektbezeichnung:	Gemme
Objektbezeichnung:	Kamee
Material:	Gold
Material:	Karneol
Material:	Lava
Datierung:	historistisch
Datierung:	kaiserzeitlich
Datierung:	Ende 18. Jh.

Ausfüllbarkeit: Keine Probleme

Auswirkungen auf die Produkte: Die gewünschten Produkte können nur mit einem erheblichen "Ballast" erzeugt werden

42) Pflichtlektüre: Heft 30 S. 38

Wir konnten diese jetzt doppelt oder sogar dreifach vorkommenden Schlagwörter nun zwar einfach in die "Datenstruktur V" eintragen, werden dann aber bei den Produkten unnötigen "Ballast" bekommen: Auf die Frage nach "*hellenistischen Ringen*" würde das Objekt Nummer 31 ebenso gefunden werden wie auf die Frage nach "*Münzen von 1943*". Es würden also neben wenigen richtigen eine Menge falscher Antworten stehen, weil der Computer nicht erkennen kann, was zu was gehört.

Merke: Wenn bei einem Retrieval unzutreffende Stücke gefunden werden, so nennt man das "Ballast".

Für solche Probleme hilft uns auch die weiter oben besprochene "Datenstruktur IV" (s. Abb. 37 auf Seite 73) mit seiner Benutzung einer Hilfsdatenbank nicht weiter, denn Zusammenhänge wie z.B. "*Ring, Weißgold, Silber*" und "*Münze, Silber, hellenistisch*" gelten nur für dieses eine Objekt und können daher nicht sinnvoll in eine Hilfsdatenbank ausgelagert werden.

Wir müssen uns für solche Fälle also etwas grundsätzlich anderes ausdenken. Wir brauchen eine Möglichkeit, solche Zusammenhänge innerhalb eines Datensatzes in einer Weise wiederzugeben, daß der Computer beim Retrieval etwas damit anfangen kann.

Zur Lösung solcher Probleme gibt es eine Reihe mehr oder weniger umständlicher oder auch eleganter Möglichkeiten, wir besprechen hier nur die bekannteste und üblichste, die "**hierarchische Datenstruktur**" und die sie charakterisierenden "**Gruppenfelder**".

Merke: Gruppenfelder haben die Aufgabe, mehrere Datenfelder ("Unterfelder") zusammenzufassen⁴³, sie enthalten also statt der Daten sogenannte "Unterfelder".

Für unser Beispiel 3 konstruieren wir uns nun ein solches Gruppenfeld (nennen wir es "Angaben zum Objekt"). Wir müssen natürlich auch dafür sorgen, daß dieses Gruppenfeld selber wiederholbar ist. Diese neue "Datenstruktur VI" könnte z.B. wie in Abb. 43 aussehen.

Um die Übersicht zu erleichtern, haben wir die Unterfelder leicht eingerückt. Wir erkennen, daß wir mit Hilfe solcher Gruppenfelder zusammengehörige Datenfelder gruppenweise zusammenfassen können. Letztlich läuft das darauf hinaus, das Objekt in bestimmte, historisch bedingte Teile zu zerlegen.

Abb. 43: Datenstruktur VI. Wiederholbare Gruppenfelder mit "Unterfeldern" und einfacher Syntax

Beispiel 3: Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern

Objekt-Nr.:	31
Angaben zum Objekt	
Objektbezeichnung:	Ring
Material:	Weißgold
Datierung:	1943
Angaben zum Objekt	
Objektbezeichnung:	Münze
Material:	Silber
Datierung:	hellenistisch

(Fortsetzung siehe nächste Seite)

Datenstruktur VI. Wiederholbare Gruppenfelder mit "Unterfeldern" und einfacher Syntax: Fortsetzung von Seite 81	
Objekt-Nr.:	32
Angaben zum Objekt	
Objektbezeichnung:	Halsschmuck
Material:	Gold
Datierung:	historistisch
Angaben zum Objekt	
Objektbezeichnung:	Gemme
Material:	Karneol
Datierung:	kaiserzeitlich
Angaben zum Objekt	
Objektbezeichnung:	Kamee
Material:	Lava
Datierung:	Ende 18. Jh.
Ausfüllbarkeit: Keine Probleme	
Auswirkungen auf die Produkte: Die gewünschten Produkte können ohne jeden störenden "Ballast" erzeugt werden	

Gießen wir auch hier etwas Wasser in den Wein: Es ist außerordentlich schwer, Gruppenstrukturen so festzulegen, daß sie für alle Objekte passen. In dem eben gebrachten Beispiel haben die Teilobjekte keine Inventarnummern. Falls sie welche hätten (und der Fall kommt natürlich häufig vor), müßte die Objekt-Nummer in das Gruppenfeld "Angaben zum Objekt" mit einbezogen werden (vgl. hierfür das Konvolut Beispiel 4 in Abb. 45 auf Seite 84). In einer anderen, weiter unten zu besprechenden, Verweis-Datenstruktur (Abb. 46 auf Seite 86) wird eine solche Einzelnumerierung sogar zur "technischen Notwendigkeit".

Beispiel 4: Konvolute aus einfach strukturierten Objekten

Wir hatten oben schon festgestellt, daß es dem Computer ganz egal ist, warum komplexe Datenstrukturen entstehen (Kapitel "Was ist ein Museumsobjekt", Seite 75 ff.). Um das zu veranschaulichen, ergänzen wir das eben behandelte Beispiel 3 um eines, bei dem die Komplexität aus ganz banalen Gründen entstand.

Die Objekte: Wir nehmen also an, daß die Objekte aus dem Beispiel 1 (Abb. 34 auf Seite 69) ein in sich geschlossenes Konvolut darstellen. Um das Problem stärker herauszuarbeiten und um zu zeigen, daß man bei diesem Vorgehen keine Objekt-Nummern braucht, stellen wir uns für einen Moment vor, daß das ganze Konvolut nur eine einzige Inventarnummer erhielt (die Objekt-Nummern, die wir uns hierbei "wegdenken", sind deswegen in Abb. 44 auf Seite 83 eingeklammert). Das ist zwar alles andere als eine korrekte Vergabe von Inventarnummern, wir finden solche Fälle aber sehr häufig.

Wenn wir nun die "Datenstruktur VI" für dieses Konvolut benutzen, so entsteht ein Bild, das der "Datenstruktur I" mit den gleichen Objekten (Abb. 39 auf Seite 77) sehr ähnlich sieht. Der Hauptunterschied zur "Datenstruktur I" liegt darin, daß wir statt sieben selbständiger Datensätze nun nur noch einen einzigen haben.

Die Einführung von hierarchischen Datenstrukturen mit ihren wiederholbaren Gruppenfeldern bedeutet einen erheblichen Gewinn gegenüber den vorher behandelten "flachen" Datenstrukturen. Man kann hier natürlich noch viel weiter gehen und z.B. ineinander verschachtelte Gruppenfelder erlauben. Ein wesentlicher Nachteil dieser Datenstrukturen soll aber nicht verschwiegen werden: es gibt nur wenige hierarchische Datenbanksysteme auf dem Markt.

Abb. 44: Datenstruktur VI. Wiederholbare Gruppenfelder mit "Unterfeldern" (Mehrfachfelder)

Beispiel 4: Konvolut aus einfach strukturierten Objekten (die bei Konvoluten oft nicht für jedes Einzelteil vergebenen Objekt-Nummern sind eingeklammert)

Angaben zum Objekt
Objekt-Nr.: 1
Objektbezeichnung: Schmuckkasten
Material: Perlenstickerei
Datierung: Mitte 19. Jh.

Angaben zum Objekt
(Objekt-Nr.: 1 a)
Objektbezeichnung: Halsschmuck
Material: Silber
Datierung: Jugendstil

Angaben zum Objekt
(Objekt-Nr.: 1 b)
Objektbezeichnung: Brosche
Material: Silber
Datierung: 20-er Jahre

Angaben zum Objekt
(Objekt-Nr.: 1 c)
Objektbezeichnung: Brosche
Material: Gold
Datierung: Biedermeier

Angaben zum Objekt
(Objekt-Nr.: 1 d)
Objektbezeichnung: Ohrring
Material: Kaiserzeit
Datierung: Gold

Angaben zum Objekt
(Objekt-Nr.: 1 e)
Objektbezeichnung: Kamee
Material: Karneol
Datierung: Kaiserzeit

Angaben zum Objekt
(Objekt-Nr.: 1 f)
Objektbezeichnung: Ehering
Material: Platin
Datierung: 1954

Ausfüllbarkeit: Keine Probleme

Auswirkungen auf die Produkte: Das gewünschte Retrieval sollte keine Probleme machen

Eine andere Anwendung von Gruppenfeldern wird weiter unten im Kapitel "Die Suche nach einfacheren und flexibleren Datenstrukturen" (87 ff.) besprochen.

Darstellung von Zusammenhängen durch Verweise

Wir haben im Zusammenhang mit Hilfsdateien schon eine ganze Reihe von Beispielen für Verweisstrukturen erwähnt (Seiten 33 ff. und 41). Diese Verweise betrafen typische "Thesaurus-Relationen" wie Ober- und Unterbegriffe, Synonyme und dergleichen.

Verweise können aber natürlich auch andere Inhalte betreffen, denn der Maschine sind solche inhaltlichen Dinge ganz egal. Sie brauchen sich auch nicht auf Hilfsdateien zu beziehen, sie können durchaus innerhalb der gleichen Datei benutzt werden.

Als bewußt einfach gehaltenes Beispiel nehmen wir den Fall eines Konvoluts noch einmal wieder auf. Wir haben dabei schon mehrere Möglichkeiten, solche Zusammengehörigkeiten darzustellen, kennengelernt:

- **Datenstruktur I** (Abb. 39 auf Seite 77): Die Zusammengehörigkeit war nur aus der Inventarnummer zu erschließen.
- **Datenstruktur VI** (Abb. 44) auf Seite 81: Die Zusammengehörigkeit wurde durch wiederholbare Gruppenfelder innerhalb eines einzigen Datensatzes dargestellt.

Wir erfinden nun einen Verweis "zusammengehörig mit" (wie wir ihn auf vielen Karteikarten finden) und kommen damit zur **Datenstruktur VII**. Um zu unterstreichen, daß solche Verweise ganz beliebige Datensätze betreffen können, schließen wir den Verweis auf ein Abendkleid ein. Wir stellen uns z.B. vor, daß dieses Abendkleid zwar getrennt von der Textilabteilung inventarisiert wurde, daß aber die Zusammenhänge (der ganze Kram stammt aus dem Nachlaß der berühmten Sängerin Bianca Castafiore) erhalten bleiben sollen (Abb. 45):

Abb. 45: Datenstruktur VII. Flache Datenstruktur mit Verweisen

Beispiel 4: Konvolut aus einfach strukturierten Objekten

Objekt-Nr.:	1	Zusammengehörig mit:	1 a-f
		Zusammengehörig mit:	T 74
Objektbezeichnung:	Schmuckkasten		
Material:	Perlenstickerei		
Datierung:	Mitte 20. Jh.		

Objekt-Nr.:	1 a	Zusammengehörig mit:	1
Objektbezeichnung:	Halsschmuck		
Material:	Silber		
Datierung:	Jugendstil		

Objekt-Nr.:	1 b	Zusammengehörig mit:	1
Objektbezeichnung:	Brosche		
Material:	Silber		
Datierung:	20-er Jahre		

Objekt-Nr.:	1 c	Zusammengehörig mit:	1
Objektbezeichnung:	Brosche		
Material:	Gold		
Datierung:	Biedermeier		

(Fortsetzung siehe nächste Seite)

Datenstruktur VII. Flache Datenstruktur mit Verweisen: Fortsetzung von Seite 84			
Objekt-Nr.:	1 d	Zusammengehörig mit:	1
Objektbezeichnung:	Ohrring		
Material:	Kaiserzeit		
Datierung:	Gold		

Objekt-Nr.:	1 e	Zusammengehörig mit:	1
Objektbezeichnung:	Kamee		
Material:	Karneol		
Datierung:	Kaiserzeit		

Objekt-Nr.:	1 f	Zusammengehörig mit:	1
Objektbezeichnung:	Ehering		
Material:	Platin		
Datierung:	1954		

Und dazu ein "neues" Objekt aus dem gleichen Zusammenhang:

Objekt-Nr.:	T 74	Zusammengehörig mit:	1
Objektbezeichnung:	Abendkleid		
Material:	Seide		
Datierung:	1942		
Provenienz:	Bianca Castafiore		
Ausfüllbarkeit: Keine Probleme			
Auswirkungen auf die Produkte: Das gewünschte Retrieval sollte keine Probleme machen			

Nun sind Verweise wie "Zusammengehörig mit" sicher nicht besonders aussagekräftig und sollten nur dann benutzt werden, wenn man die "Relation" nicht genauer definieren kann. Besser sind "qualifizierte Verweise", bei denen man dann auch leichter entscheiden kann, wie sie später verarbeitet werden sollen.

Als Beispiel für mögliche Anwendungen solcher qualifizierten Verweise nehmen wir das Beispiel 3 wieder auf (s.o. Datenstruktur VI, Abb. 43 auf Seite 81) und prüfen, ob sich auch hier die Gruppenstruktur der Datenstruktur VI durch eine Verweisstruktur ersetzen läßt.

Bei diesem Versuch begegnen wir sofort einer unabdingbaren Anforderung an solche Verweise: Sie müssen auf etwas verweisen, das die Maschine finden und eindeutig identifizieren kann. In einer Datei, in der die Objekt-Nummer diese Rolle des "Schlüssel" ("Key") spielt, muß so ein Verweis also auf die betreffende Objekt Nummer geschehen. Die Teilobjekte in unserem Beispiel 3 hatten aber bisher keine eigenen Objekt-Nummern.

Bevor wir solche Verweisstrukturen einsetzen können müssen wir also dafür sorgen, daß alle Teilobjekte (und damit "Datensätze") eine eindeutige Objekt- oder Dokumenten-Nummer erhalten. In unserem Beispiel (Abb. 46) haben wir einfach weitere Inventarnummern verteilt. Diese neuen Nummern haben in dem Beispiel noch die Form von "Sammelnummern" (32 a-f bzw. 32 g-l); wenn wir diese Kameen und Gemmen einzeln beschreiben wollten, wäre es sicherlich günstiger, sie auch einzeln zu numerieren.

Abb. 46: Datenstruktur VII. Flache Datenstruktur mit Verweisen		
Beispiel 3: Komposite Objekte mit mehreren Informationen in mehreren Datenfeldern		
Objekt-Nr.:	31	Teilobjekt: 31 a
Objektbezeichnung:	Ring	
Material:	Weißgold	
Datierung:	1943	
Objekt-Nr.:	31 a	Gesamtobjekt: 31
Objektbezeichnung:	Münze	
Material:	Silber	
Datierung:	hellenistisch	
Objekt-Nr.:	32	Teilobjekt: 32 a-f
		Teilobjekt: 32 g-1
Objektbezeichnung:	Halsschmuck	
Material:	Gold	
Datierung:	historistisch	
Objekt-Nr.:	32 a-f	Gesamtobjekt: 32
Objektbezeichnung:	Gemme	
Material:	Karneol	
Datierung:	kaiserzeitlich	
Objekt-Nr.:	32 g-1	Gesamtobjekt: 32
Objektbezeichnung:	Kamee	
Material:	Lava	
Datierung:	Ende 18. Jh.	
Ausfüllbarkeit: Keine Probleme		
Auswirkungen auf die Produkte: Die gewünschten Produkte können ohne jeden störenden "Ballast" erzeugt werden		

Mit Verweisen kann man also auf sehr einfache Weise selbständige Einzelsätze einer Datei explicit zusammenfassen, ohne solche Zusammenhänge aus zufälligen Dingen wie der Numerierung erschließen oder mit komplizierten Gruppenstrukturen darstellen zu müssen.

Verweise sind dazu außerordentlich flexibel. Wenn wir z.B. nachträglich feststellen sollten, daß auch der historistische Halsschmuck mit der Objekt-Nummer 32 (Beispiel 3) aus dem Besitz der "Castafiore" stammt, brauchen wir unsere Verweise nur zu ergänzen, umzustrukturieren brauchen wir nichts.

Zusammenfassung: Die Suche nach einfacheren und flexibleren Datenstrukturen

Im ersten Hauptteil ging es um die Formulierung der Daten. Der Computer ist eine Rechenmaschine - wie bei einer Registrierkasse muß man eine ganze Reihe von letztlich mechanischen Regeln beachten, wenn man ein korrektes Ergebnis erzielen will. Je nach Leistungsumfang der eingesetzten Programme, kann eine konsequente Umsetzung dieser Regeln selber wieder durch Hilfsprogramme erleichtert werden. Diese Hilfen für eine "terminologische Kontrolle" bleiben trotz ihres unbestreitbaren Nutzens allerdings ganz mechanisch, je nach der Intelligenz des Anwenders wird daraus eine Zwangsjacke oder ein Gewinn an Bedienungskomfort.

Aufgabe des zweiten Hauptteils war, die Wechselbeziehungen zwischen den zu dokumentierenden Inhalten und der Datenstruktur zu veranschaulichen und dabei so weit wie möglich Ursache und Wirkung herauszuarbeiten. Im Vordergrund der Betrachtung stand wie im 1. Hauptteil eine fehlerfreie Erzeugung der gewünschten Produkte. Gesucht waren möglichst einfache und flexible Datenstrukturen.

Im Kapitel "Was kann man mit einfachen ("flachen") Datenstrukturen machen?" (Seite 67 ff.) stand eine für den Computer geeignete inhaltliche Definition des einzelnen Datenfeldes im Vordergrund der Betrachtung. Das Ergebnis war, stark vereinfacht, daß eine solche Definition am leichtesten zu gewinnen ist, wenn man von den Produkten ausgeht. Alle Inhalte, die bei der Herstellung solcher Produkte eine selbständige Rolle spielen, sollten ein eigenes Datenfeld erhalten.

Dabei konnten wir feststellen, daß es aus praktischen Gründen oft angezeigt ist, etwaige inhaltliche Differenzierungen und Zusatzinformationen in **Hilfsdateien** zu verlagern.

Das in Bezug auf einzelne Datenfelder und ihre Verwendbarkeit festgestellte gilt natürlich auch, wenn diese Datenfelder selber in komplexere Datenstrukturen eingebunden sind. Solche komplexen Datenstrukturen betreffen also nicht mehr das einzelne Datenfeld, sondern die Zusammenfassung mehrerer Datenfelder in einem Datensatz bzw. die Verknüpfung mehrerer Dateien. Das Ziel dieser komplexen Datenstrukturen war doppel: Einerseits möchte man die Daten erfassen können, ohne sie zu verbiegen, andererseits sollen die gewünschten Produkte ohne unnötigen Ballast erzeugt werden können.

Im Vergleich zu einfachen ("flachen") Datenstrukturen gewinnen wir durch komplexe Datenstrukturen einen erheblichen Gewinn an Flexibilität, wenn wir zusätzlich **Mehrfachfelder** wie in der "Datenstruktur V" (Abb. 42) auf Seite 80 oder gar wiederholbare **Gruppenfelder** wie in der "Datenstruktur VI" (Abb. 43 auf Seite 81) einführen. Wir können die als Beispiele genommenen Objekte und Konvolute selbst dann damit beschreiben, wenn diese ganz unsachgemäß numeriert wurden (Abb. 44 auf Seite 83).

Eine grundsätzlich andere Datenstruktur entsteht, wenn wir ähnlich wie bei der Benutzung eines Thesaurus **explizite Verweise auf andere Datensätze** (statt der Gruppenfelder innerhalb eines Datensatzes) einsetzen. Neben Verweisen von Gesamt- auf Teilobjekte bzw. umgekehrt (Abb. 46 auf Seite 86) nahmen wir als Beispiel den Fall relativ unspezifischer Verweise vom Typ "Zusammengehörig mit" (Abb. 45 auf Seite 84). Dieses Vorgehen eröffnet uns also zusätzliche Möglichkeiten, die die hierarchische Ordnung nicht kennt.

Das regt natürlich zum Nachdenken an. Es ist an dieser Stelle aber außerordentlich nützlich, sich mal wieder daran zu erinnern, daß man sich bei der inhaltlichen Definition von Feldern immer ganz genau überlegen soll, was die Maschine mit diesem Feld bei der Verarbeitung nun eigentlich machen soll. Soll sie bei einer Suchfrage (z.B. nach "Eheringe aus Platin") nicht nur die betroffenen Sätze (im Beispiel das Objekt mit der Nr. 1 f), sondern alle "zusammengehörigen" (im Beispiel den "Schmuckkasten" und das "Abendkleid") bringen ?

Wie setzt man das um ?

Es wurde in diesem zweiten Hauptteil dargestellt, daß es für ein Problem oft mehrere Lösungen gibt und es wurde versucht, deren jeweiligen Vor- und Nachteile zu beschreiben. Es wird Ihnen jetzt noch etwas schwer fallen, daraus nun sofort konkrete Handlungsanweisungen zu destillieren, denn es bedarf einer großen Übung, mit allen diesen Möglichkeiten zu jonglieren.

Trotz all dieser Schwierigkeiten sollten Sie nun aber in der Lage sein, sich zumindest mittelfristig zuverlässige Strategien zu der Frage zu erarbeiten, mit welchen Datenfeldern und Datenstrukturen Sie welche Art von Daten erfassen und daraus bestimmte Produkte erzeugen können.

Es ist ganz dezidiert nicht Aufgabe dieses Buchs, Ihnen nun ein bestimmtes Datenbanksystem oder auch nur eine bestimmte Art eines solchen vorzuschlagen. Wie man so etwas macht, wird im Heft 30 der Materialien aus dem Institut für Museumskunde ausführlich⁴⁴ erklärt. Ich möchte hier nur davor warnen, sich durch Wörter wie "modern", "hierarchisch", "relational" usw. beeindrucken und von einer durchdachten Systemanalyse und ihrer Umsetzung in ein Pflichtenheft abhalten zu lassen.

Letztlich muß sich Ihr Vorgehen ja auch an den Kosten orientieren. Eine Standard-Software, die alle in diesem zweiten Hauptteil beschriebenen Arten der Datenstrukturierung unterstützt, werden Sie so leicht nicht finden. Es wird Ihnen dabei klar werden, daß es noch keine idealen, alle Wünsche erfüllenden Vorgehensweisen und Programmsysteme gibt und daß hier noch viel Entwicklungsarbeit zu leisten ist.

Denken Sie dabei bitte immer daran, daß man bei solchen Dingen immer nur schrittweise vorangehen sollte. Das kann z.B. so aussehen, daß man nur die Daten erfaßt, von denen man schon genau sagen kann, was mit ihnen dann geschehen soll. Wenn man gleich "alles" erfassen möchte, muß man sich halt klar machen, daß solche Daten dann einer späteren Überarbeitung bedürfen werden.

Die Anpassung des Wünschbaren an das Machbare ist also Ihre Aufgabe. Wenn Sie jetzt in der Lage sein sollten, hier nicht mehr mit Meinungen und Hoffnungen argumentieren zu müssen, sondern ganz rationale Entscheidungen zu treffen, sind Sie auf dem richtigen Weg.

Entwicklungstendenzen

Noch vor wenigen Jahren waren die meisten Leute davon überzeugt, daß die Daten zu Sammlungsobjekten relativ stabil seien; einmal erfaßt, würde sich an diesen Daten nicht mehr viel ändern. Mittlerweile hat man gelernt, daß das so nicht stimmt, daß Museumsdokumentation eine viel lebendigere und dynamischere Sache ist, als man dachte. Sicherlich trägt der erleichterte Zugriff auf die Daten zu dieser für viele unerwarteten Dynamik bei.

Immer wenn man sich mit bisher unbearbeiteten Objekten oder Informationsquellen auseinandersetzt wird man auf Dinge stoßen, an die man bei der Erstellung seiner Datenstruktur noch nicht gedacht hat. Jede wirklich neue Frage wird die Erhebung neuer Daten erfordern.

Dies ist zunächst eine Übung in Bescheidenheit. So wie es nicht möglich ist, die definitive Monographie zu einem Museumsobjekt zu schreiben, so ist es auch nicht denkbar, eine definitive Datenstruktur zu erarbeiten, die noch in hundert Jahren gültig ist. Die Auseinandersetzung mit solchen Fragen hat aber trotzdem eine ganze Menge wichtiger Erkenntnisse gebracht, die sich in ein paar Entwicklungstendenzen niedergeschlagen haben, die wir jetzt kurz skizzieren wollen.

44) Pflichtlektüre: Heft 30 S. 29 ff.

Wenn man die Geschichte des Computereinsatzes in den "Objektwissenschaften" zurückverfolgt, so muß man sich manchmal verwundert fragen, wo heute eigentlich ein paar der noch vor wenigen Jahren leidenschaftlich diskutierten, damals als grundlegend betrachteten Wünsche und Hoffnungen geblieben sind.

Vor 20 Jahren stand das "Problem der Beschreibung" noch ganz im Vordergrund. Der Wissenschaftler erhoffte sich durch den schnellen Zugriff auf die im Computer detailliert zu beschreibenden Objekte neue Einsichten und eine beträchtliche Erleichterung seiner Arbeit. Manche versprachen sich von der neuen Art der Dokumentation eine reinigende Wirkung auf die Geisteswissenschaften, eine eindeutige Nomenklatur wurde in Anlehnung an die Naturwissenschaften als unabdingbare Voraussetzung für die "Wissenschaftlichkeit" angesehen; andere sahen allein im Zusammenbringen aller relevanten Daten in ein gemeinsames System Ansätze für eine neue heuristische Methode.

Mehr am Rande dachte man an Informationen, die nicht so sehr durch das Objekt und seine formalen Eigenschaften, als durch seine Geschichte bestimmt sind. Darin sah man von Anfang an den für alle beteiligten Disziplinen gemeinsamen Teil, während die eigentliche Objektbeschreibung natürlich gattungsspezifisch blieb; eine griechische Münze ist anders zu beschreiben als eine gotische Kathedrale.

Diese Informationen zur Geschichte der Objekte stehen heute - zumindest im Museum - im Vordergrund aller Diskussionen. Man kann diese pragmatische Beschränkung auf das kurzfristig Machbare als einen Gewinn an "Augenmaß" ansehen, wir müssen aber auch beachten, daß wir bei den Benutzern der Daten eine Verschiebung vom Wissenschaftler auf die Verwaltung feststellen können. Es ist überhaupt kein Zufall, wenn Lenore Sarasan diese Entwicklung (und ihre Auswirkungen auf die Programmsysteme) als den Weg vom Retrievalsystem zur Sammlungsverwaltung charakterisiert.

Diese Interessenverschiebung von der "Speziellen" zur "Allgemeinen Erfassung"⁴⁵, von neuen Impulsen für die Wissenschaft zu einer rationelleren Sammlungsverwaltung, hat uns einiges über den Umgang mit dem Formenreichtum "historischer" Informationen gelehrt. Auch diese Daten stellen ein "Problem der Beschreibung", mit dem wir uns nun in den folgenden Abschnitten beschäftigen wollen.

Einfachere und leichter zu verstehende Datenmodelle und -strukturen

Hinter den meisten Informationen, die wir zu einem Objekt ermitteln und aufschreiben möchten, stehen Fragen des folgenden Typs: Wer, wann, wo, was, wie (usw.) in Auftrag gegeben, hergestellt, benutzt, verändert, gefunden, gekauft, restauriert, ausgestellt (usw.) hat. So formuliert, könnte man denken, daß man eine inhaltliche Definition von Datenstrukturen aus einer gar nicht so großen Zahl von Elementen zusammensetzen könnte.

Es hat infolgedessen auch an Versuchen nicht gefehlt, solche einfach zu verstehenden "Datenmodelle" in Datenbankstrukturen umzusetzen. Alle diese Versuche laufen letztlich darauf hinaus, statt einer Unzahl von Einzelfeldern die oft sehr komplexen, bisher meist in Feldnamen verpackten Informationen genauer zu analysieren und auf wenige, miteinander kombinierbare Inhalte zu reduzieren. Statt unzähliger Felder zu Ortsangaben wie z.B. "Herstellungsort (Gemeinde)", "Herstellungsort (Bundesland)", "Fundort (Gemeinde)", "Vorbesitzer (Gemeinde)" usw. wird man z.B. "Vorgänge" (Prozesse, Situationen) wie *Herstellung, Fund, Besitz*, und "geographische Einheit" wie *Gemarkung, Gemeinde, Kreis, Regierungsbezirk, Bundesland* unterscheiden und daraus die passende Kategorie kombinieren.

Ein paar dieser Versuche möchte ich kurz charakterisieren, weil die Zukunft ganz sicher in einer Kombination der dabei vorgeschlagenen Methoden bestehen wird:

45) So die sprachliche Charakterisierung in s.u. Anm. 48.

- **MDA-Standard⁴⁶**: Das Ende der 60-er Jahre in der "Information Retrieval Group of the Museum Association" (IRGMA) konzipierte und dann in den 70-er Jahren von der Museum Documentation Association (MDA) in einen "Data Standard" (Datenkatalog) umgearbeitete "Communication Format" ist der bekannteste und auch in der Praxis am weitesten verbreitete Versuch dieser Art.

In dem dann von der MDA angebotenen "hierarchischen" Datenbanksystem (Ende der 70-er Jahre das eigens dafür entwickelte GOS, heute MODES) wurden die oben erwähnten "Vorgänge" (Prozesse, Situationen) wie "Herstellung", "Fund" usw. zu Gruppenfeldern, in denen dann wiederholbare Datenfelder wie "Name", "Ort", "Datierung" untergebracht wurden.

Wir sehen in Abb. 47 einen Ausschnitt aus diesem MDA-Standard. Man erkennt, daß die (fast) immer wieder gleichen Datenfelder unter verschiedenen Gruppenfeldern wiederholt werden können. Diese Art der Felder nennt man nicht "Mehrfachfelder" (die wir oben auf Seite 78 behandelten), sondern "Wiederholungsfelder"⁴⁷. Wenn Sie solche Felder benutzen wollen, müssen Sie sich einer Software bedienen, die damit umzugehen vermag. Dies ist durchaus nicht selbstverständlich.

Abb. 47: Ausschnitt aus dem MDA-Standard: Angaben zur "Herstellung" bzw. zum "Fund" mit standardisierten Wiederholungsfeldern

PRODUCTION	FIELD COLLECTION
PART	PART
METHOD	METHOD
PERSON	PERSON
CORPORATE_BODY	CORPORATE_BODY
DATE	DATE
PERIOD	PERIOD
PLACE	PLACE
TECHNICAL_DATA	COLLECTION_NUMBER
AUTHORITY	AUTHORITY
DOCUMENTATION	DOCUMENTATION
NOTE	NOTE

- **Bericht der Arbeitsgruppe Museumsdokumentation ("Grünes Papier"⁴⁸)**: In parallelen und z.T. auf den englischen Vorarbeiten aufbauenden Bemühungen einer Arbeitsgruppe des Deutschen Museumsbundes ging man einen Schritt weiter: Auch die "Vorgänge" wurden nun zu Datenfeldern, die in einer hierarchischen Beziehung (heute würden wir sagen "wie Unterbegriffe eines Thesaurus") zu den Hauptthemen Entstehung, Veränderungen am Objekt, Herkunft und Aufbewahrung, Wissenschaftliche Bearbeitung und Dokumentation sowie Bezeichnung standen. Zusätzlich war vorgesehen, diese Vorgänge durch Verweise (wie oben auf Seite 84 beschrieben) frei kombinierbar (und natürlich wiederholbar) zu gestalten, wodurch dieses Modell ganz erheblich an Flexibilität gewann.

Dieses Modell wurde nie in ein Datenbanksystem umgesetzt - zur Zeit seiner Entstehung wäre das technisch noch außerordentlich schwierig gewesen, im Rückblick muß man heute feststellen, daß wir in Deutschland die Entwicklung verschlafen haben.

46) Die neueste Version kam im Frühjahr 1991 heraus: The MDA Data Standard, ISBN 0 905963 74 1

47) **Pflichtlektüre**: Heft 30 S. 38.

48) H. Oehler und C. Wolters, Anlage 1 zum Bericht der Arbeitsgruppe Museumsdokumentation: Regeln für die "allgemeine" und "spezielle" Erfassung von Museumsobjekten, in: Museumskunde, Heft 3, 1971, s. 121-161.

- **Zahlreiche neuere Projekte:** Andere konzentrierten sich auf die zahlreichen Arten von Daten, die man innerhalb eines Themas wie "wer", "wann", "wo" finden kann, also z.B. bei Ortsangaben Verwaltungseinheiten wie "*Gemarkung, Gemeinde, Kreis, Regierungsbezirk, Bundesland*". Die Information um welche dieser Einheiten es sich jeweils handelt, wird als ein eigenes Datenfeld (mit Namen wie "Rolle" oder "Typ") behandelt. Wenn dann neue Einheiten auftreten (z.B. "*département*") braucht man an der Datenstruktur nichts zu ändern. Auch hierdurch gewinnt man also erheblich an Flexibilität.

Wenn man diese Dinge miteinander kombiniert, kommt man auf Datenmodelle, bei denen der Feldname in mehrere, vielleicht sogar frei miteinander kombinierbare, Teile zerlegt wird und von dem es natürlich zahlreiche, meist noch experimentelle, Varianten gibt⁴⁹. Mit solchen Methoden kann eine Datenstruktur erheblich an Flexibilität gewinnen. Vieles wird hier allerdings von einer verständlichen Darstellung und nicht zuletzt der konkreten Umsetzbarkeit in Bezug auf existierende Datenbanksoftware abhängen. Wenn Sie mehr darüber erfahren wollen, schauen Sie bitte in das Glossar (s.v. "Datenmodelle"). Hier bleibt noch viel zu tun.

Wachsender Einsatz von Verweistechniken zur Datenstrukturierung

Verweistechniken werden in den allermeisten Fällen für Aufgaben wie "Terminologische Kontrolle" einzelner Datenfelder eingesetzt. Wenn man die betreffenden Hilfsdateien angemessen strukturiert, kann man weitergehen und durch Verlagerung von Hintergrundwissen in die Hilfsdatei eine redundante Datenerfassung umgehen. Wenn solche Hilfsdateien wie z.B. die Thesauri selber eine entsprechende Datenstruktur aufweisen, kann man durch Verkettung von Verweisen dann auch kompliziertere Produkte wie z.B. Systematische Register erzeugen. Diese Fälle wurden im 1. Hauptteil im Kapitel über das Arbeiten mit Hilfsdateien dargestellt (Seite 33 ff.).

Solche Verweistechniken spielen auch bei der Datenstrukturierung eine immer größere Rolle. Wir haben das oben (Seite 84 ff.) dargestellt, ohne die "relationalen" Datenbanksysteme, die gerade die Vermeidung von Redundanz zum obersten Ziel haben, zu erwähnen. Dies hat den einfachen Grund, daß Sie sich bei der Ermittlung der für Sie günstigsten Datenstrukturen nicht durch irgendwelche Aussagen über die "Modernität" eines Systems beeinflussen lassen sollten. Das von Lenore Sarasan als entscheidendes Kriterium für die Auswahl eines Computersystems herausgestellte Konzept der "Paßgenauigkeit"⁵⁰ beschreibt ganz präzise, worauf Sie achten sollten. Lassen Sie sich also nicht durch irgendwelche Wörter beeindrucken, sondern fragen Sie ganz genau, was ein System für Sie und Ihr Museum tut.

Wir haben deshalb versucht, das Funktionieren der Verweise von den gewünschten Produkten (z.B. systematische Register) oder Beispielen kompliziert strukturierter Objekte (bzw. Konvolute von solchen) zu entwickeln. Wir haben dabei gesehen, daß die Benutzung von Verweisen neben der immer wünschenswerten Vermeidung von Redundanz nicht nur zu einfacheren ("flacheren"), sondern auch zu flexibleren Datenstrukturen führen kann.

Die durch Verweise innerhalb der gleichen Datei gegebenen vielfältigen Strukturierungsmöglichkeiten wurden im Museum bisher nur selten intensiv genutzt, sind aber in hohem Maße "zukunftsträchtig".

49) Z.B. das neue "Data Administration Standarda Manual" der Smithsonian Institution, dazu Jane Sledge, Smithsonian Strategies for Terminology Control: A People-Dependant Process, in: Terminology for Museums, 95-104 (s.o. Anm 25).

Wer sich hier über die neuesten Entwicklungen informieren möchte, sollte sich an die "Reconciliation of Standards Working Group" des CIDOC wenden.

50) Pflichtlektüre: Heft 30 S. 33

Ausblick: Quellennahe Erfassung

Jeder, der mal eine Datenbank aufgebaut hat, ist mit dem Problem vertraut, daß er die in der Quelle (Inventare, Publikationen usw.) gefundenen Informationen oft formal und inhaltlich stark verändern muß, bevor der Computer damit etwas anfangen kann. Richtschnur für eine solche Überarbeitung ist letztlich immer das vom gewählten Datenbanksystem unterstützte Datenmodell. Man muß also seine Daten in eine flache, eine hierarchische oder gar relationale Struktur bringen. Die für solche Vereinheitlichungen (oder "Normalisierungen") nötige Überarbeitung ist nicht nur außerordentlich teuer, die dabei fast immer zu schließenden Kompromisse und Fehlinterpretationen führen auch zu einer Verarmung der Daten oder sogar zu Fehlern in denselben.

Diese Art, alles über einen Kamm zu scheren und damit Widersprüche zwischen den Informationsquellen zu beseitigen ist, obwohl man sich daran gewöhnt hat, natürlich auf die Dauer unbefriedigend. Ein Historiker verschwendet seine kostbare Zeit nicht mit der Vereinheitlichung der oft ungenügenden und sich widersprechenden Quellen, er versucht vielmehr unter Einbeziehung dieser Quellen und neuer Ideen zu neuen Erkenntnissen zu gelangen und daraus wieder neue Quellen zu schaffen.

Infolgedessen verstärkt sich heute wieder die Diskussion über eine "quellennahe" Erfassung. Statt sich vor der Datenerfassung mühsam zu überlegen, welcher Formulierung oder welchem Inhalt der sich oft widersprechenden Quellen man den Vorzug geben soll, sucht man Wege, um zumindest bei der Datenerfassung die in der Quelle getroffenen Entscheidungen akzeptieren zu können. Das setzt natürlich voraus, daß eine Datenstruktur die Erfassung solcher Widersprüche erlaubt. Statt "*Hersteller: Rembrandt*" muß man also erfassen können, daß im Inventar "*Rembrandt*" steht, im alten Katalog "*Rembrandt-Nachfolge*" und daß das gute Stück vom Rembrandt-Projekt dem "*Govert Flinck*" namentlich "zugeschrieben" wird. Vielleicht findet sich ja auch mal wieder eine Autorität, die es wieder als eigenhändiges Werk des Meisters ansieht.

Das ist nicht einfach durch "Mehrfachfelder" oder andere entsprechende Tricks zu erreichen, denn es liegt auf der Hand, daß der Rembrandt des Inventars mit dem des Rembrandt-Projekts - rein informationstechnisch betrachtet - etwa soviel zu tun hat wie das Deutschland vor dem letzten Weltkrieg mit dem nach demselben.

Die Zurückführung des bei historischen Daten praktisch überall auftretenden Problems der unscharfen Informationen wie "*Rembrandt*" oder "*Deutschland*" auf Widersprüche zwischen in sich ja meist konsistenten Informationsquellen stellt - rein gedanklich - eine sehr elegante Lösung des Unschärfeproblems dar.

Das Interesse an der quellennahen Erfassung ist aber nicht nur rein wissenschaftlicher Art. Man erhofft sich von ihr auch eine drastische Senkung der vor der Datenerfassung zu leistenden, in ihrem Umfang meist unterschätzten Arbeiten für die Vereinheitlichung der Daten.

Man muß dann aber auch Wege schaffen, mit diesen ganz ungleichmäßigen oder sogar widersprüchlichen Daten etwas anfangen zu können. Es genügt also nicht, solche Daten erfassen zu können, man muß auch sagen, wie sie verarbeitet werden sollen. Das ist alles andere als einfach: Wie beziehe ich z.B. bei einem Retrieval nach Objekten aus dem Bezirk Cottbus die davon im Prinzip betroffenen, aber unter "Brandenburg" inventarisierten, Objekte ein? Hier sind also noch erhebliche Entwicklungsarbeiten zu leisten.

Die Tendenz zur quellennahen Erfassung ist letztlich in der Arbeitsweise des Historikers begründet und daher alles andere als neu. Ihre Realisierung im Computer war bisher aber nur sehr unvollkommen möglich. Die Entwicklung der Softwaretechnik hat nun manche dieser Ziele aus dem Bereich des Utopischen in greifbare Nähe gerückt. Peu-à-peu lernen wir heute, wie man hier weiterkommen kann.

Anhang: Arbeitsblätter für die produktorientierte Analyse von Textdaten

Die in diesem Anhang zu Heft 33 der Materialien aus dem Institut für Museumskunde zusammengestellten Arbeitsblätter sind gewissermaßen eine Fortsetzung der immer wieder als Pflichtlektüre zitierten "Do-it-Yourself-Systems-Analysis" von Lenore Sarasan. Trotz kleinerer Überschneidungen enthalten sie im wesentlichen eine Differenzierung der dort oft nur "angetippten" Themen. Da sie nun aber nicht mehr das "IST", sondern ganz dezidiert das "SOLL" beschreiben, sind sie zusätzlich auch eine gute Überleitung zu dem im gleichen Heft abgedruckten "Pflichtenheft".

Ein paar Worte zu dem Sinn von Arbeitsblättern

Die in Heft 30 und 33 der Materialien aus dem Institut für Museumskunde vorgeschlagenen Arbeitsblätter (natürlich haben wir das Wort "Fragebogen" gemieden) sollen eine Arbeitshilfe sein. Dadurch, daß Sie selber etwas tun, kontrollieren Sie, ob und wie weit Sie die Sache im Griff haben ("Learning by Doing"). Gleichzeitig schaffen Sie eine solide Grundlage für die zukünftige Arbeit. So weit die guten Absichten der Autoren (die nicht wenig stolz sind, eine komplizierte Materie auf so wenig Raum noch einmal konzentriert zusammengefaßt zu haben).

Eine Systemanalyse besteht nun aber nicht darin, hunderte von Arbeitsblättern mit irgendwelchen Angaben, von denen man nicht so recht weiß, wozu sie dann später dienen sollen, auszufüllen. Denken Sie also bitte auch an die Leute, die das dann später lesen sollen. Lange Serien von solchen Blättern, auf denen jeweils nur ganz wenige Informationen stehen, sind eine außerordentlich entmutigende Lektüre. Bitte benutzen Sie zumindest einen Stift, der sich auch auf Fotokopien stark vom Schriftbild der Arbeitsblätter abhebt.

Wenn Sie ein gutes Computersystem haben wollen, dann ist es Ihre Aufgabe, Ihre Anforderungen so klar wie möglich herauszuarbeiten und sie nicht irgendwo in einem Wust von Papier zu verstecken. Sie müssen sich also Mühe geben, das Material so klar zu gliedern, daß sich z.B. ein Software-Anbieter leicht darin zurechtfinden kann. Wenn Sie also - um ein Beispiel zu geben - wie im Regelfalle für die meisten Felder den immer gleichen Zeichensatz benötigen und trotzdem für jedes dieser Felder individuelle Angaben machen (oder gar eigene Arbeitsblätter zum Zeichensatz ausfüllen), dann können Sie noch dazulernen (höflicher kann man das wirklich nicht sagen).

Zu den "neuen" Arbeitsblättern

Die hier vorgestellten Arbeitsblätter versuchen also, die verwirrende Vielfalt von Eigenschaften einzelner Datenfelder oder auch Datensätze noch einmal in einer knappen, listenartigen Form zusammenzufassen und damit gleichzeitig Unterlagen für ein detaillierteres Handbuch bzw. "Pflichtenheft" zu schaffen.

Dabei wird das im 1. Hauptteil nur skizzierte Stufenmodell für die "Syntax" (Seite 45) um inhaltliche, die Struktur von Dateien und Datensätzen betreffende, Stufen ergänzt. Wir beschränken uns hierbei auf die wichtigsten Eigenschaften, eine wie auch immer geartete "Vollständigkeit" wird nicht angestrebt.

Die Reihenfolge der Arbeitsblätter entspricht ganz bewußt nicht der Gliederung des Handbuchs, sie ist auch kein Vorschlag für eine Arbeitsreihenfolge. Sie ist aber ein Anhaltspunkt für eine gute Gliederung des aus einer schriftlichen Systemanalyse zu entwickelnden "Anforderungsprofils". Sie werden also am Anfang Dinge finden, über die Sie sich vielleicht erst relativ spät klar werden. Aussagen zu Themen wie "Struktur von Dateien und

Datensätzen" erlauben es aber dem Leser, sehr schnell zu entscheiden, ob eine bestimmte Software für seine Zwecke geeignet ist.

Behandelt werden die folgenden Themen bzw. Arbeitsblätter:

- **Struktur von Dateien und Datensätzen** (Seite 95 f.): Dieses Arbeitsblatt schließt direkt an die Übungen 1 und 2 (Datei-Arbeitsblatt, Eingangs-/Ausgangs-Arbeitsblatt) von Lenore Sarasan an. Es schlägt Ihnen eine übersichtliche Zusammenfassung zu strukturellen Merkmalen von Dateien und Datensätzen vor. Alle hierfür nötigen Kenntnisse wurden im zweiten Hauptteil beschrieben.
- **Zur Wiederholung: IST-Analyse von Datenfeldern** (Seite 97 ff.). Das DIYSA - Datenfeld - Arbeitsblatt aus Heft 30 der Materialien aus dem Institut für Museumskunde hat bei manchen Kollegen, die die Arbeitsweise des Computers noch nicht kennen, zu Mißverständnissen geführt. Es wird hier also noch einmal ausführlich und mit Beispielen versehen erläutert.
- **Beschreibung einzelner Felder** (Seite 99 ff.): Diese Arbeitsblätter schließen direkt an das Datenfeld-Arbeitsblatt von Lenore Sarasan an. Behandelt werden die Einbindung in die Datenstruktur sowie die für die Herstellung grundlegender Produkttypen notwendigen Unterscheidungen wie Freier Text, Schlagwörter für alphabetische und systematische Register sowie Verweise.
Zusätzlich werden einige Möglichkeiten der automatischen Bearbeitung eines Datenfeldes behandelt; dies geschieht an Beispielen, d.h. ohne jeden Anspruch auf Vollständigkeit.
- **Beschreibung von Numerierungssystemen** (Seite 106 ff.): Die Arbeitsblätter enthalten zusätzliche Hilfen für die im Anhang zum 1. Hauptteil beschriebenen Eigenschaften von Numerierungssystemen.
- **Zeichensatz** (Seite 114 ff.): Fragen zum Zeichensatz wurden bei Sarasan nur am Rande behandelt. Es schien daher günstig, dieses gerade bei wissenschaftlichen Dokumentationen wichtige Thema auf eigenen Arbeitsblättern abzuhandeln. Im Zentrum stehen die Darstellbarkeit der Zeichen beim Druck bzw. auf dem Bildschirm, die Art der Eingabe (Tastatur bzw. Ersatzdarstellung) sowie Fragen der Sortierung.

Ein systematisches Ausfüllen dieser Arbeitsblätter ist nur dann sinnvoll, wenn Sie wirklich ein eigenes System entwerfen wollen. In jedem Falle sollten Sie aber mit Hilfe der Arbeitsblätter auf spielerische Weise kontrollieren, wie weit Sie die Sache verstanden haben.

Arbeitsblatt zur Struktur von Dateien und Datensätzen

Diesem Kapitel wird nur ein sehr summarisches Arbeitsblatt beigegeben. Seine Aufgabe ist, Ihnen Vorschläge für eine übersichtliche Darstellung der Strukturen von Dateien und Datensätzen zu machen.

Wir haben bereits gesehen, daß man ein Problem oft auf verschiedene Weisen (= mit verschiedenen Datenstrukturen) lösen kann, Sie müssen sich also immer darüber klar sein, daß es hier nicht um den technischen Entwurf eines völlig neuen Datenbanksystems geht, sondern um eine quasi metaphorische Darstellung Ihrer individuellen Anforderungen.

Als Anhaltspunkt für Ihre Darstellung nehmen Sie bitte die entsprechenden Abbildungen aus dem zweiten Hauptteil (und nicht einen bereits als "Fangblatt" aufbereiteten Datenkatalog wie Abb. 32 auf Seite 61).

Versuchen Sie dann, für jede Datei ein eigenes Arbeitsblatt zu erstellen. Die dabei benutzten Informationen sind oft nicht mehr als eine Wiederholung von Angaben, die bei der Beschreibung des einzelnen Datenfeldes bereits gemacht wurden (s.u. Seite 99 ff., wo Sie auch die nötigen Begriffsdefinitionen finden). Diese "Redundanz" erleichtert aber das Verständnis. Bitte geben Sie also die folgenden Informationen:

- **Dateiname:** Leicht merkbare Kurzbezeichnung
- **Charakterisierung:** Hier genügen allgemeine Angaben wie "Objektdatei", "Thesaurus von Objektbezeichnungen", "Bibliographische Angaben zu den Abkürzungen der Standardliteratur" usw. Sie sollten dafür sorgen, daß aus der Charakterisierung anschaulich wird, wozu diese Datei benutzt werden soll. Wenn es sich um eine Hilfsdatei handelt, sollten Sie auch die folgenden Angaben zur "Hauptdatei" ausfüllen:
 - **Name der Hauptdatei:** Im allgemeinen ist das nur eine einzige, es kann in manchen Fällen aber praktisch sein, z.B. Abkürzungen für mehrere Felder oder gar Dateien in einer Datei zusammenzufassen.
 - **Verweisfeld:** Name des Datenfeldes, das in der Hauptdatei als Verweis dient (das ist dann automatisch der "Satzschlüssel" Ihrer Datei), braucht aber natürlich nicht den gleichen Feldnamen zu haben.
- **Feldnamen in der gewünschten Reihenfolge:** Sie erleichtern die Lektüre, wenn Sie diese Liste der Feldnamen anschaulich gliedern; ein paar Vorschläge dazu:
 - **Reihenfolge:** Sie beginnen am besten mit dem Satzschlüssel.
 - **Leerzeilen:** Trennung zusammenhängender Gruppen von Feldnamen (wie Abb. 36 auf Seite 71),
 - **Unterfelder zu Gruppenfeldern leicht einrücken** (wie Abb. 43 auf Seite 81)
 - **Abgekürzte Zusatzinformationen zum Feldtyp beigegeben:** Hier reichen die beim nächsten Punkt aufgeführten groben Unterscheidungen völlig aus, also z.B.:
 - Objekt-Nr. (KEY)
 - Angaben zum Objekt (G)
 - Objektbezeichnung (MD)
 - Material (MD)
 - (usw.)
- **Benötigte Arten von Feldern:** Diese Liste faßt die auf den Arbeitsblättern zur Beschreibung von einzelnen Feldern bereits erarbeiteten Merkmale übersichtlich zusammen.

Zur Wiederholung: IST-Analyse von Datenfeldern nach Sarasan

Eine ausgezeichnete Grundlage für die IST-Analyse von Datenfeldern ist die "Übung 3: Das Datenfeld-Arbeitsblatt" aus Heft 30 der Materialien aus dem Institut für Museumskunde. Diese Übung wurde dort aber nicht sehr ausführlich erklärt und hat daher bei manchen mit der Materie noch nicht vertrauten Anwendern zu Schwierigkeiten und Mißverständnissen geführt.

Bevor wir uns den nächsten Arbeitsblättern zuwenden, schieben wir hier also etwas ausführlichere Erläuterungen mit Verweisen auf den Text dieses Handbuchs ein. Dies ist gleichzeitig eine gute Gelegenheit, ein paar zusätzliche, für ein späteres Handbuch nützliche, Merkmale zu beschreiben:

Feldname

Vorausgesetzt wird die Lektüre des Kapitels "Was kann ich eigentlich alles in ein einziges Datenfeld stecken?" auf Seite 64 ff. Die für den Computer geeignete Definition muß den dort entwickelten Kriterien genügen.

Als Feldnamen benutzt man in diesem Stadium der Arbeit praktischerweise möglichst kurze, aber dennoch leicht zu erinnernde Kurzformen wie z.B. INVNR (falls es nur eine Art gibt), INVNR1, INVNR2 usw. (bei mehreren), OBJBEZ, MAT, NEGNR und dergleichen. Wichtig ist hier nur, daß Sie und die mit Ihnen zusammenarbeitenden Kollegen damit "klarkommen".

Sie können hier sowieso nur eine vorläufige Form wählen, da verschiedene Programmsysteme bei den **programminternen Feldnamen** unterschiedliche Konventionen für die Schreibweisen vorsehen (manche nur Zahlen, manche Buchstaben mit Zahlen, manche nur Großbuchstaben, manche nur eine Form, andere dazu noch "Aliasnamen"). In jedem Falle sollte der programminterne Feldname möglichst kurz und prägnant sein.

Diese Kurzformen sind allerdings für Leute, die das System nicht ständig benutzen, sehr unangenehm und verwirrend. Sie sollten sich also schon bald Gedanken über eine möglichst verständliche Formulierung des Feldnamens machen. Dieser **Feldname im Klartext** kann dann frei formuliert werden, statt INVNR können Sie "Inventar-Nummer", statt OB "Oberbegriff" schreiben. Diese Formulierungen können Sie dann für die Darstellung auf dem Bildschirm oder für Standardausdrucke verwenden.

Bei manchen Feldern, besonders bei den **Numerierungen** werden Sie zumindest am Anfang nur schwer entscheiden können, ob äußerlich unterschiedliche Nummern bzw. Variationen der Schreibweise nach den gleichen Regeln verarbeitet bzw. im gleichen Datenfeld erfaßt werden sollten.

Faustregel: Nummern mit gleichem Inhalt und grundsätzlich gleicher Struktur können Sie den gleichen Feldnamen geben (wie man so etwas analysiert, wird im Kapitel "Beschreibung von Numerierungssystemen" auf Seite 106 ff. genau erläutert, ist in den meisten Fällen aber ganz leicht zu erkennen).

Beispiel: für Nummern wie z.B. A 317, ZF 2351 a usw. gilt durchgehend: 1. Element immer Großbuchstabe(n) als Sachgruppenkürzel, 2. Element immer laufende arabische Nummer, 3. und letztes Element fakultativ: Kleinbuchstabe als "Unternummer".

Es ist möglich, daß ein Durcharbeiten des nächsten Kapitels ("Beschreibung von Datenfeldern") Rückwirkungen auf den Feldnamen hat. Es kann nicht schaden, das betreffende Kapitel vor dem Ausfüllen dieses Arbeitsblattes durchzulesen.

Obwohl das auf dem betreffenden Arbeitsblatt nicht aufgeführt ist, sollte(n) beim Feldnamen in jedem Falle auch die Datei(en) angegeben werden!

Form

Bitte die Alternative **alphanumerisch** (Buchstaben und Zahlen bzw. mehrere Zahlengruppen) oder **numerisch** (nur einfache Zahl ohne Punkt und Komma) ankreuzen. Diese Frage wird wegen ihrer Wichtigkeit (und um unnötiges Blättern zu vermeiden) auch auf den neuen Arbeitsblättern aufgeführt.

Definition

Vielleicht werden Sie am Anfang Schwierigkeiten haben, hier brauchbare Texte zu entwerfen. Stellen Sie sich also bitte einen Benutzer vor, der mit Ihrem System noch nicht vertraut ist und es mit Hilfe eines Handbuchs gerne lernen möchte. Wenn Sie die Verständlichkeit Ihrer Texte an Kollegen testen, werden Sie sehr schnell erkennen, daß selbst ganz einfache Dinge erklärungsbedürftig sind.

Zu empfehlen ist also ein stufenweises Vorgehen. Dabei kann aus einfachen Dingen wie "Objektbezeichnung" sehr bald ein eher zu langer Text werden, in dem Sie z.B. erläutern,

- daß Sie einen möglichst "engen" Begriff (nicht "*Tisch*", sondern "*Küchentisch*") wünschen,
- in welchem Maße beschreibende Merkmale in die Objektbezeichnung gehören oder nicht (wollen Sie wirklich Formulierungen wie "*Küchentisch, weiß gestrichen*" oder "*Küchentisch aus Eichenholz*" ?),
- was mit fremdsprachigen oder mundartlichen Bezeichnungen geschehen soll und
- ob und wie Sie das Feld von verwandten Feldern wie "Zweck/Funktion" oder "Sachgruppe" abgrenzen usw.

Wenn Sie mit einem kontrollierten Wortschatz oder einem Thesaurus arbeiten, sollte das hier erläutert werden (z.B. Was macht man, wenn der betreffende Sachverhalt dort nicht vorgesehen ist ?)

Verwendungen

Aufschlußreich ist hier in vielen Fällen die **Verwendungsgeschichte**, besonders wenn sich daraus interessante Informationen zum Verständnis von Änderungen bei Form oder Inhalt ergeben. Bei einer Inventar- oder Negativ-Nummer möchte man z.B. wissen, ob die Nummer zur Beschriftung der Objekte oder Negative dient. Bitte prüfen Sie auch nach (Stichproben), ob die dort benutzte Schreibweise mit der schriftlichen Quelle genau übereinstimmt.

Interessant ist auch der **Verwendungszweck**: Wozu dient dieses Datenfeld ? Wenn man den Feldinhalt z.B. in ein bestimmtes "Häuschen" auf einer Karteikarte oder eine bestimmte Stelle eines Formulars drucken möchte, muß man entweder die Feldlänge entsprechend einschränken oder regeln, was gegebenenfalls mit dem "überlaufenden" Rest geschehen soll.

Hierzu gehören auch Feststellungen wie "nur für hausinterne Benutzung" (z.B. Preis- und Wertangaben) usw.

Schreibweise (Syntax) usw.

Die dann auf dem DIYSA-Datenfeld-Arbeitsblatt folgenden Angaben (Schreibweise, Wortschatzkontrolle, Beispiele und Feldlänge) sollten in den meisten Fällen auf dem nächsten Arbeitsblatt näher erläutert werden. Denken Sie aber daran, daß Sie für Ihr Handbuch überzeugend formulierte Erklärungen zu diesen Themen brauchen.

Arbeitsblätter zur Beschreibung von einzelnen Feldern

Um ein Feld für den Computer ausreichend genau beschreiben zu können, muß man genau festlegen, was die Maschine mit diesem Datenfeld machen soll. Auf diesem Arbeitsblatt werden nur sehr wenige, dafür aber wichtige Eigenschaften des Feldes abgefragt.

Seite 1: Grundlegende Angaben (Seite 104)

Feldname und Datei

Mit diesen Angaben finden Sie den Anschluß an das DIYSA-Datenfeld-Arbeitsblatt (siehe Kapitel "IST-Analyse von Datenfeldern" auf Seite 97 ff.).

Einbindung in die Datenstruktur

Hier wird die Rolle des Feldes in einem Datensatz zumindest soweit beschrieben, wie das für eine angemessene Datenerfassung bzw. die Erzeugung bestimmter Produkte erforderlich ist. Die grundsätzlichen Erläuterungen finden Sie im 2. Hauptteil (besonders im Kapitel "Was ist ein Museumsobjekt?" auf Seite 75 ff.), im folgenden wird dann jeweils auf die entsprechenden Unterkapitel verwiesen.

In jedem Falle sollten Sie diese Angaben schon aus Gründen der Übersichtlichkeit dann auf dem Arbeitsblatt für die Beschreibung einer Datei (Seite: 95 f. noch einmal zusammenfassen. Hier geht es also nur um Angaben aus der Sicht des einzelnen Feldes.

- **Datenfeld als Satzschlüssel (KEY):** Da die Inhalte mehrerer Sätze übereinstimmen können (z.B. mehrere identische Objekte) benötigt der Computer in jedem Falle ein eindeutiges, nur für diesen Datensatz geltendes Merkmal, z.B. eine in der Datei nur einmal vorkommende "Laufende Nummer". Wenn der Benutzer dieses Merkmal überhaupt zu sehen bekommt bzw. dieses Merkmal sogar selber bestimmen kann, erscheint es häufig unter Namen wie "Dokumenten-Nummer".

In vielen Fällen möchte man aber den Inhalt eines bestimmten Datenfeldes als Satzschlüssel benutzen bzw. ihm nach außen (unabhängig von systeminternen Regelungen) diese Funktion geben. Üblich ist das z.B. bei Inventarnummern oder den Satzschlüsseln von Hilfsdateien (z.B. Orts- oder Personennamen), die ja zeichengenau mit den Inhalten bestimmter Datenfelder übereinstimmen müssen.

In solchen Fällen ist das Datenfeld schon aus rein technischen Gründen eine s.u. "Mußkategorie". Ein gutes Programmsystem sichert, daß man weitere Daten nur dann eingeben kann, wenn dieses Feld ausgefüllt wurde.

- **Verweis auf einen Satzschlüssel der gleichen Datei (VW):** Erläuterungen s. Kapitel "Darstellung von Zusammenhängen durch Verweise" auf Seite 84 ff.
- **Verweis auf eine Hilfsdatei (VW-HILF):** Erläuterungen s.u. bei den "Produkttypen".
- **Einfaches Datenfeld (D):** Datenfeld zur Aufnahme einer einzigen Information.
- **Mehrfachfeld (MD)** (s. "Datenkatalog V", Abb. 41 auf Seite 79): Das Datenfeld soll mehrere gleichartige und gleich zu behandelnde Informationen aufnehmen können.

Falls das Programmsystem Mehrfachfelder nicht kennt und sie also "simulieren" muß, sollte ein "Trenner" (z.B. ein Sonderzeichen, das in den Daten auf keinen Fall vorkommen kann) definiert werden.

- **Gruppenfeld (G):** Zu Gruppenfeldern siehe die Erläuterungen zu Abb. 43 auf Seite 81 ff.

- **Wiederholungsfeld (WD):** Dies ist ein Datenfeld, das an verschiedenen Stellen eines Datensatzes, also z.B. als Unterfeld zu verschiedenen Gruppenfeldern vorkommen kann (Beispiel: MDA-Standard s. Abb. 47 auf Seite 90).

Wenn das Wiederholungsfeld ein Mehrfachfeld sein soll, bitte beide Positionen anstreichen.

- **Falls das Feld ein Element einer Gruppenstruktur ist:** Wir unterscheiden hier nur zwei Fälle:

- **Falls Unterfeld: Name des Gruppenfeldes.** Bei "verschachtelten" Datenstrukturen kann ein Unterfeld selber ein Gruppenfeld sein.

Sollen Unterfelder wie z.B. beim MDA-Standard (Abb. 47 auf Seite 90) eingesetzt werden, ist zu klären, ob das Programmsystem sogenannte "**Wiederholungsfelder**" verarbeiten kann.

- **Falls Gruppenfeld: Name der Unterfelder:** Liste der Feldnamen.

Definition des Datenfeldes für die grundlegenden Produkttypen

Die Liste möglicher "Produkttypen" reicht von Wiedergaben (Druckprodukte aller Art wie z.B. Karteikarten und Kataloge) über alphabetische und systematische Register bis zum Einsatz von Hilfsdateien zur terminologischen Kontrolle. Da die betreffenden Kriterien ausführlich diskutiert wurden, können wir uns bei den Erläuterungen kurz fassen.

Maßgeblich ist hier der 1. Hauptteil (besonders das Kapitel "Zusammenfassung: Stufenmodell" auf Seite 45 ff.). Es geht zunächst um die grundsätzlichen Anforderungen, die man als **Freier Text**, **Schlagwort** und **Verweis auf Hilfsdatei** charakterisieren kann (wobei die "höheren" Anforderungen die jeweils "niedrigeren" selbstverständlich einschließen):

- **Freier Text:** Der Feldinhalt soll nicht für eine weitere Verarbeitung, sondern nur für "Wiedergaben" dienen (s. Kapitel "Wiedergabe von Textdaten" auf Seite 21 f.).
- **Schlagwort:** Der Feldinhalt dient zusätzlich für Produkte wie alphabetische Register oder ein schnelles Retrieval. Eine terminologische Kontrolle per Programm findet nicht statt (s. Kapitel "Das Sortieren von Textdaten" auf Seite 22 ff. bzw. die Erzeugung systematischer Register ohne Hilfsdateien auf Seite 33 ff.).
- **Verweis auf Hilfsdatei:** Der Feldinhalt verweist auf einen Datensatz, bei dem er selber als "**Satzschlüssel**" (Key) dient (s. Kapitel zum Arbeiten mit Hilfsdateien auf Seite 33 ff.). Wir beschränken uns hier auf eine kurze Liste von Möglichkeiten, ohne die unzähligen Variationen zu berücksichtigen. Im Zentrum steht immer die Frage "Was soll das Programm mit den Inhalten (Werten) dieses Datenfeldes machen ?":
 - **Präferenzliste (Positiv-Methode):** Man kann in das Datenfeld nur "Werte" eintragen, die in der Präferenzliste aufgeführt sind. Sinnvoll bei Feldern, bei denen nur wenige, häufig wiederholte Werte vorkommen können (z.B. "Erwerbsart: Kauf").
 - **Stoppliste (Negativ-Methode):** Man kann nur Werte benutzen, die nicht in dieser Liste vorkommen (z.B. "Erwerbsart: gekauft").
 - **Abkürzungsdatei:** Man kann den Inhalt des Datenfeldes durch einen anderen, den "Klartext" ersetzen. Dies ist unmittelbar einsichtig und sinnvoll bei Abkürzungen. Man muß festlegen, ob der Klartext sofort bei der Eingabe in das Datenfeld eingetragen oder ob das bei der späteren Verarbeitung nur simuliert werden soll. Eine Abkürzungsdatei kann wie eine Präferenzliste benutzt werden.

- **Lexikon:** Die Nutzung entspricht - technisch betrachtet - der der Abkürzungsdatei, die dort abgelegten Daten werden meistens den Charakter von "Zusatzinformationen" haben (z.B. "Lebensdaten von Künstlern") und daher erst bei einer späteren Verarbeitung benutzt werden. Auch ein Lexikon kann wie eine Präferenzliste benutzt werden.

Wenn Daten des Lexikons beim Retrieval benutzt werden sollen (z.B. "Objekte von Künstlern, die zwischen 1500 und 1530 geboren wurden"), so muß das im Programmsystem vorgesehen sein.

- **Thesaurus:** In einem Thesaurus werden die Inhalte eines Datenfeldes durch "Relationen" mit anderen Inhalten des gleichen Datenfeldes verknüpft.

Ein Thesaurus erlaubt eine gleichzeitige Verwendung der Positiv- und der Negativmethode und kann für die Herstellung systematischer Register eingesetzt werden.

Auch bei einem Thesaurus muß man klären, was mit "neuen" Begriffen geschehen soll und ob er beim Retrieval einzusetzen ist.

Für alle drei Syntax-Typen sind der **Zeichensatz** und die **Feldlänge** zu bedenken, selbst wenn das Programmsystem hier keine Einschränkungen vorsieht:

- **Freier Text:** Falls der Feldinhalt für den Druck von Karteikarten und Formularen mit einem beschränkten Platz dienen soll.
- **Schlagwörter:** Aus praktischen und ästhetischen Gründen sollten hier gewisse Grenzen eingehalten werden. Dienen die Schlagwörter als Vorstufe für später einzuführende Verweise, so gelten zusätzlich die dort genannten Einschränkungen.
- **Verweise:** Die Verweise müssen aus technischen Gründen mit dem **Satzschlüssel** des Satzes, auf den verwiesen wird, zeichengenau übereinstimmen. Für die meisten Programmsysteme bestehen dabei technische Grenzen.

Zu beachten sind hier aber auch praktische Grenzen: Die Formulierung von Schlagwörtern und Verweisen als "Roman" ist arbeitsaufwendig und fehleranfällig (Tipfehler).

Form

Wiederholung der Angaben vom DIYSA-Datenfeld-Arbeitsblatt: **alphanumerisch** (Buchstaben und Zahlen bzw. mehrere Zahlengruppen) oder **numerisch** (nur einfache Zahl ohne Punkt und Komma).

- **Begrenzter Zeichensatz:** Ausführliche Erläuterungen zu diesem Thema finden Sie im Kapitel "Arbeitsblätter zum Zeichensatz" auf Seite 114 ff.

Es kann sinnvoll sein, den Zeichensatz für manche Datenfelder auf bestimmte Zeichen oder Zeichenklassen einzuschränken. Das gilt z.B., wenn Sie oben beim Mehrfachfeld einen "Trenner" definiert haben (das braucht hier aber nicht wiederholt zu werden) oder bei manchen Abkürzungen (nur Großbuchstaben und Zahlen) usw.

Für die meisten Datenfelder wird die Angabe der Form "alphanumerisch" ausreichen.

Seite 2: Automatische Bearbeitung eines Datenfeldes (Seite 105)

Auf Seite 2 des Arbeitsblattes zur Beschreibung von einzelnen Feldern können Sie Angaben zu einer automatischen Bearbeitung machen. Behandelt wird eine **kleine Auswahl** solcher Möglichkeiten. Selbst wenn Sie hier nicht alles, was Ihr Herz begehrt, finden, so gibt Ihnen das Arbeitsblatt doch eine ganze Reihe von Hinweisen auf Fragen, die für eine entsprechende Programmierung geklärt werden müssen.

Bearbeitung des Zeichensatzes per Programm

Der bei den meisten Datenfeldern nicht von vornherein festzulegende Wortschatz und die große Variationsbreite der Schreibweisen schränkt die Möglichkeiten einer Syntaxkontrolle per Programm in der Museumsdokumentation stark ein.

Da solche Methoden aber in manchen Fällen außerordentlich wirksam sein können, wird hier auf ein paar grundlegende, zum Teil bereits erwähnte Möglichkeiten hingewiesen. Für Numerierungssysteme finden Sie weiterführende Informationen auf Seite 106 ff. bzw 112 f. (Arbeitsblätter).

- **Kontrolle des vorgesehenen Zeichensatzes:** Im Prinzip kann der Computer den verwendeten Zeichensatz relativ leicht kontrollieren. Manche Programmsysteme unterstützen zumindest eine Unterscheidung zwischen erlaubten und verbotenen Zeichen nach bestimmten "Zeichenklassen". Wenn dies nicht ausreicht, benutzen Sie bitte die betreffenden Arbeitsblätter zum Zeichensatz (Seite 117-121, Erläuterungen dazu auf Seite 114 ff.).
- **Umformung einzelner Zeichen per Programm:** Es kommt vor, daß man zur Erleichterung der Eingabe eine automatische Umformung bestimmter Zeichen wünscht. Viele Programme sehen solche Dinge z.B. für das Sortieren vor (interne Umformung von Klein- in Großbuchstaben, gegebenenfalls Unterdrückung von Akzenten).

Wenn das "klassenweise" geschehen soll, machen Sie die Angaben hier, sonst benutzen Sie die Spalte "Bemerkungen" der Arbeitsblätter zum Zeichensatz (Seite 117-121, Erläuterungen dazu auf Seite 114 ff.).

- **Zeitpunkt der Umformung:** "Sofort bei der Dateneingabe" dient einer Vereinheitlichung der Daten und ist besonders bei Satzschlüsseln (z.B. Inventarnummern) sinnvoll. "Nur für Sortierprogramme" bedeutet, daß der Feldinhalt zwar unverändert übernommen werden soll, daß er aber für Sortierprogramme (bzw. Retrieval) noch bearbeitet werden muß.
- **Art der Umformung:** Mit "Standard" bezeichnen wir hier die bei den meisten Sortierprogrammen übliche Umformung von Klein- in Großbuchstaben (wie auf dem Arbeitsblatt für Deutsche Kleinbuchstaben (Seite 119) in der Spalte "Sortierung"). Werden andere Umformungen gewünscht, so ist das extra zu erläutern (bei Numerierungssystemen können solche Angaben auf dem betreffenden Arbeitsblatt auf Seite 112 gegeben werden).
- **Zielfeld der Umformung:** Unterschieden werden hier nur zwei Fälle: "intern" (für den Benutzer unsichtbar, z.B. bei Satzschlüssel) und "extern" (gleiches Datenfeld).

Terminologische Kontrolle durch Hilfsdateien

Hier werden ein paar übliche Reaktionen von Programmen zur terminologischen Kontrolle aufgeführt. Sie müssen sich solche Dinge sehr sorgfältig überlegen:

- **Behandlung "neuer" Begriffe:** Mit Ausnahme ganz weniger Datenfelder wird es Ihnen sicher sehr schwer fallen, gleich zu Beginn der Arbeit alle erlaubten Werte (Feldinhalte) vollständig in einer Hilfsdatei aufzuführen. Versuchen Sie genau festzulegen, was das Programm tun soll.

Bei "Zurückweisung" (Sie können den Begriff nicht eingeben) müssen Sie eine Möglichkeit schaffen, den Begriff auf eine Art "Warteliste" zu bringen bzw. ihn sofort in die betreffende Hilfsdatei einzuführen.

Bei "Fehlermeldung" sollten Sie sich nicht nur eine geeignete Formulierung derselben überlegen, sondern auch, wie Sie dann überhaupt noch eine ernsthafte terminologische Kontrolle erreichen wollen.

Entscheidend sind hier die bei Ihrer Dokumentation vorgesehenen Entscheidungsmechanismen: Ein für mehrere Museen gemeinsamer Thesaurus stellt ganz andere Anforderungen als ein auf ein Museum oder gar einen einzigen Mitarbeiter beschränktes Projekt.

- **Übernahme der Schreibweise:** Soll die Schreibweise des Begriffs in der Hilfsdatei zeichengenau übernommen werden ?
- **Ersetzen von Abkürzungen:** Sollen Abkürzungen durch den Klartext ersetzt werden ?

Bearbeitung durch Spezialprogramme (Auswahl)

Hier können nur ein paar Einzelfälle genannt werden, eine fragebogenmäßige Darstellung ist nicht möglich. Solche Anforderungen sollte man auf einem zusätzlichen Blatt erläutern.

- **Mußkategorie:** Dieser etwas salopp klingende Ausdruck bezeichnet Datenfelder, die aus formalen oder inhaltlichen Gründen in jedem Falle ausgefüllt werden müssen. So etwas kann per Programm relativ leicht nachgeprüft werden.

Wenn das Datenfeld als Satzschlüssel dienen soll, ist es quasi automatisch eine "Mußkategorie" (dieser Fall wurde bereits oben abgehandelt). Solche Dinge können aber auch aus inhaltlichen Gründen gewünscht und bei geeigneter Programmierung auch kontrolliert werden. Wenn Sie solche Kontrollen einbauen (lassen), überlegen Sie sich bitte rechtzeitig, was Sie machen, wenn Sie dieses Feld aus irgendwelchen Gründen nicht sofort, sondern erst später ausfüllen können.

- **Überprüfen der Zulässigkeit im Kontext anderer Felder:** In einem Thesaurus kann ein Begriff z.B. nicht gleichzeitig Unter- und Oberbegriff zu einem anderen sein. In solchen Fällen sollte die Zulässigkeit einer Angabe also im Kontext anderer Felder überprüft werden. Diese Technik kann auch für eine differenzierte Definition von Mußkategorien eingesetzt werden (z.B. "Wenn Gemeinde, dann auch Kreis bzw. kreisfreie Stadt").
- **Automatischer Rückverweis:** Dies sichert die für das Funktionieren unabdingbare 100 %ige Konsistenz.
- **Programmunterstützte Permutation:** Die Permutationen im alphabetischen Register Abb. 9 auf Seite 27 wurden auf diese Weise erzeugt¹. Hier ist je nach den Daten ein detailliertes Regelwerk zu entwerfen bzw. zu übernehmen.
- **Erzeugen von Satzdatum:** Das sogenannte "Satzdatum" speichert automatisch das Datum der Eingabe bzw. der letzten Änderung. Intern sollte das Satzdatum eine total vereinheitlichte und leicht sortierbare Form wie "Jahr-Monat-Tag" (z.B. "1991-08-13") haben, auf dem Bildschirm oder beim Druck sind leserlichere Formen vorzuziehen.
- **Andere:** Hier dürfen Sie Ihrer Phantasie freien Lauf lassen. Denken Sie aber auch daran, daß Sie solche Zusatzprogramme wahrscheinlich bezahlen müssen.

1) **Pflichtlektüre:** Heft 30 der "Materialien" S. 7-10: Was versteht der Computer von der deutschen Sprache ?

Arbeitsblätter zur Beschreibung einzelner Felder
Seite 2: Automatische Bearbeitung des Datenfeldes

Feldname:

Datei:

Bearbeitung des Zeichensatzes per Programm

- **Kontrolle des vorgesehenen Zeichensatzes (s. Seite 1):** ja / nein

- **Umformung einzelner Zeichen per Programm:** ja / nein

Zeitpunkt der Umformung:

Nur für Sortierprogramme Sofort bei der Dateneingabe

Art der Umformung: Standard / s. Arbeitsblätter zum Zeichensatz

Zielfeld der Umformung:

intern (z.B. Satzschlüssel) extern (gleiches Datenfeld)

Terminologische Kontrolle durch Hilfsdateien

- **Behandlung "neuer" Begriffe:** Zurückweisung / Fehlermeldung

- **Übernahme der Schreibweise:** ja / nein

- **Ersetzen von Abkürzungen:** ja / nein

Bearbeitung durch Spezialprogramme (Auswahl)

- **Mußkategorie** ja / nein

- **Prüfung der Zulässigkeit im Kontext anderer Felder:** ja / nein

Erläuterungen:

- **Automatischer Rückverweis:** ja / nein **auf Datenfeld (Name):**

- **Permutation:** ja / nein **Zielfeld (Name):**

- **Erzeugen von Satzdatum:** ja / nein **Ausgabeform (Beispiel):**

- **Andere:** ja / nein **Erläuterungen:**

Name:

Datum:

Arbeitsblätter zur Beschreibung von Numerierungssystemen

Die im folgenden beschriebenen Arbeitsblätter setzen sich mit einem bereits im Anhang zum 1. Hauptteil (Seite 48 ff.) ausführlich besprochenen Fall komplexer Syntax eines Datenfeldes auseinander. Sie sollten direkt im Anschluß an die im vorigen Kapitel behandelten Arbeitsblätter benutzt werden.

Die Hauptfrage bei allen Numerierungssystemen lautet natürlich: Kann das infrage kommende Programmsystem mit den in meinem Museum verwendeten Nummern vernünftig umgehen?

Voraussetzung für eine Antwort ist eine detaillierte Kenntnis dieser Numerierungs-Systeme (incl. der im Laufe der Zeit darin entstandenen Variationen und Ausnahmen). Dies erfordert eine kritische Durchsicht aller Inventare bzw. anderer zu erfassender Quellen (Katalognummern, Negativnummern usw.) und ein darauf basierendes "inhaltliches" Verständnis derselben.

Ergebnis dieser Analyse sollten Antworten auf die folgenden Fragen sein:

- Wieviel unterschiedliche Systeme gibt es und wie sind diese im Regelfalle aufgebaut (z.B. Buchstabenkürzel - laufende Nummer mit Kleinbuchstaben als Unternummern)?
- Was bedeuten die einzelnen Elemente? Sind z.B. Großbuchstaben Kürzel für bestimmte Sammelgebiete bzw. Inventare, handelt es sich dabei evtl. um römische Zahlen? Sind die arabischen Zahlen "laufende Nummern" bzw. Unternummern oder Jahreszahlen (Jahr der Erwerbung oder der Inventarisierung)? Die Antworten sind grundlegend für die spätere Verarbeitung.
- Gibt es Abweichungen von diesen Regeln (z.B. Unternummern gelegentlich als arabische oder römische Zahlen)?
- Ist die Sortier-Reihenfolge einheitlich von links nach rechts (z.B. Jahreszahl - laufende Nummer)?
- Sind die Sortierreihenfolgen der gefundenen Systeme einheitlich oder widersprüchlich (z.B. früher Buchstabenkürzel mit laufender Nummer, jetzt laufende Nummer - Jahreszahl)?
- Gibt es römische Ziffern und welche Funktion haben sie (vorangestelltes Kürzel, Kombination mit Buchstaben als Kürzel, Unternummern und dergl.)?
- Gibt es Mischungen von Buchstaben und römischen Ziffern und finden sich diese im gleichen Inventar (z.B. A VII 4711) oder in verschiedenen?
- Gibt es "Standard-Trenner" (z.B. zwischen Buchstabe und Zahl immer Leertaste, zwischen laufender Nummer und Unternummer immer ein Punkt)? Sind diese Trenner in der Quelle einheitlich durchgehalten und - falls nicht - sollen sie vereinheitlicht werden?

Diese Arbeitsblätter dienen zunächst einer übersichtlichen Materialsammlung. In jedem Falle sollte den ausgefüllten Arbeitsblättern nach Abschluß der Arbeit ein Kommentar vorangestellt werden, der auf die oben gestellten Fragen im Zusammenhang (und ausreichend detailliert bzw. mit Beispielen versehen) eingeht.

Sie können gleichzeitig als abschreckendes Beispiel für alle Fälle dienen, in denen Sie vielleicht meinen, daß eine "komplizierte" Syntax die elegante Lösung eines Problems sein könnte.

Erläuterungen zu Seite 1: Syntax

Die Seite 1 (abgebildet auf Seite 112) sollte bei alphanumerischen Nummern (Buchstaben und Zahlen bzw. mehrere Zahlengruppen) in jedem Falle ausgefüllt werden:

Feldname und Datei

Wiederholung der Angaben von den vorigen Arbeitsblättern

Beispiel

Ein Ziel dieser Übung ist, möglichst viele Arten von Nummern mit der gleichen Syntax beschreiben und damit auch im gleichen Datenfeld erfassen zu können. Benutzen Sie an dieser Stelle also ein Beispiel, das möglichst alle vorkommenden Elemente enthält. Etwaige Bereichsangaben können sofort eingeführt werden, obwohl sie eigentlich nur abgekürzte Formen für eine Liste von Nummern sind.

Wählen Sie also eine für Ihr Museum charakteristische Nummer aus und tragen Sie diese in das Arbeitsblatt ein (z.B. eine "INVNR1" wie in Abb. 51). Bitte lassen Sie zunächst "exotische" Einzelfälle und Ausnahmen beiseite. Sie müssen dann am Ende Ihrer Analyse entscheiden, ob es einfacher ist, diese Nummern ggf. zu ändern oder eigene Datenfelder dafür einzuführen.

Abb. 51: Übungsbeispiel für die Auswahl von charakteristischen Nummern	
Feldname: INVNR1	Datei: OBJEKT
Beispiel: M IX 357 a	

Zerlegung in Elemente (1. Element 2. Element 3. Element ...)

Bitte zerlegen Sie die Nummer in ihre Elemente und tragen Sie das oben gewählte Beispiel in die entsprechenden Häuschen ein (ohne Trenner). Wichtig ist hierbei, daß die **Zerlegung nach Sortierkriterien** (s.u. "Sortierung") erfolgt. Falls es also Kürzel vom Typ A 1, A 2 usw. gibt, dann sollten diese in allen Fällen, bei denen durch eine alphanumerische Sortierung eine falsche Reihenfolge entstehen würde (ab A 10), in ihre (nach verschiedenen Regeln zu sortierenden) Elemente zerlegt werden. Wenn es also nur die drei Kürzel A 1, A 2 und A 3 gibt, kann man sie gemeinsam behandeln, bei dem Übungsbeispiel ist das nicht sinnvoll (Abb. 52).

Abb. 52: Übungsbeispiel für die Zerlegung in Elemente					
Feldname: INVNR1			Datei: OBJEKT		
Beispiel: M IX 357 a					
1. Element M	2. Element IX	3. Element 357	4. Element a	5. Element	6. Element

Sortier-Reihenfolge

In den meisten Fällen braucht man nur "von links nach rechts" anzukreuzen (wie in Abb. 53).

Abb. 53: Übungsbeispiel für normale Sortierreihenfolge					
Feldname: INVNR1			Datei: OBJEKT		
Beispiel: M IX 357 a					
1. Element M	2. Element IX	3. Element 357	4. Element a	5. Element	6. Element
Sortier-Reihenfolge: <input checked="" type="checkbox"/> von links nach rechts <input type="checkbox"/> andere (Reihenfolge der Element-Nummern): _____					

Bei z.B. der Reihenfolge "Laufende Nummer - Jahreszahl" kreuzt man "andere" an und gibt die Sortier-Reihenfolge der Elementnummern (z.B. "2 - 1 - 3") an (wie in Abb. 54). **Kommt diese andere Sortierreihenfolge im gleichen Museum vor, muß hierfür ein zweites Datenfeld vorgesehen werden (z.B. "INVNR2").**

Abb. 54: Übungsbeispiel für eine ungewöhnliche Sortierreihenfolge					
Feldname: INVNR2			Datei: OBJEKT		
Beispiel: 47/1968 a					
1. Element 47	2. Element 1968	3. Element a	4. Element	5. Element	6. Element
Sortier-Reihenfolge: <input type="checkbox"/> von links nach rechts <input checked="" type="checkbox"/> andere (Reihenfolge der Element-Nummern): 2 - 1 - 3					

Im folgenden werden abwechselnd die einzelnen Elemente und die dafür benutzten Trenner beschrieben.

Element Nr. (Abb. 55 auf Seite 109)

Angaben zur maximalen Zeichenzahl, zur Sortierung und der Funktion der oben definierten und numerierten Elemente. Bitte tragen Sie zunächst die oben benutzten Elementnummern ein. Falls es mehr als 6 Elemente geben sollte, bitte Nachfolgeblatt verwenden:

- **Maximale Zeichenzahl:** Wieviel Zeichen kann das Element maximal enthalten ?
- **Sortierung:** Bitte Betreffendes ankreuzen. Hier ist nur eine einzige Angabe sinnvoll, weil die Art der Sortierung ja als Kriterium für die Zerlegung der Nummer in Elemente dient:
 - **Alphanumerisch:** Eine solche Sortierung ist sinnvoll, wenn es sich um Groß- oder Kleinbuchstaben (auch mehrere, wenn sie direkt und ohne Trenner aufeinanderfolgen) handelt, z.B.: "A", "Ma", "a" usw.
 - **Numerisch:** Nur ganze arabische Zahlen ohne irgendwelche Trenner
 - **Römische Zahl:** Bedarf eines eigenen "Sortierprogramms"

- **Andere:** Hier können z.B. griechische Buchstaben und dergleichen aufgeführt werden.
Vorsicht: Bei Zeichen, die nicht aus lateinischen Buchstaben oder arabischen Zahlen bestehen, muß man sich eventuell Gedanken zu einer "Ersatzdarstellung" machen; auch bei der Sortierung kann es Schwierigkeiten geben. Solche Fälle sollten mit Hilfe der Arbeitsblätter zum Zeichensatz geklärt werden.
- **Funktion:** "Inhaltliches" zur Funktion der Elemente. Falls es für eine Art von Inventarnummern eine ganze Reihe von Kürzeln gibt (s.o. "Faustregel"), kann man Zeit und Papier sparen, indem man diese in Form einer Liste erfaßt (auf Seite 2 der Arbeitsblätter):
 - **Vollständiges Kürzel:** Dies können nur Elemente sein, die auch das dämlichste Sortierprogramm noch richtig sortieren kann. Wenn das vollständige Kürzel das nicht erlaubt, hätte es bereits oben in sortierbare Elemente zerlegt werden müssen (inhaltliche Erläuterung desselben auf der Seite 2 des Arbeitsblattes geben).
 - **Teil von Kürzel:** Für Fälle, in denen ein Kürzel aus mehreren, nach verschiedenen Kriterien zu sortierenden, Teilen besteht (z.B. "Buchstabe und arabische Zahl größer als 9" wie M 17, "Mal ein, mal zwei Buchstaben" (z.B. AA 1 und A 1), "Buchstabe und römische Zahl größer als VIII" wie z.B. K IX).
 - **Jahreszahl:** Falls diese Zahlen nicht "vollständig ausgeschrieben" sind, sollten Sie sich Gedanken über eine Normalisierung machen. Spätestens hundert Jahre nach Beginn der Numerierung wird die Sache doppeldeutig.
 - **Laufende "Nummer":** Dies ist meistens eine arabische Zahl (z.B. M 317), häufig haben aber auch Kleinbuchstaben eine solche Funktion (z.B. M 317 a). Selten (gottseidank!) sind römische Zahlen (z.B. M 317 IX). Wenn Sie die letzte Form als seltene Ausnahme in Ihren Quellen finden, sollten Sie nicht nur Ihren Vorgängern danken, sondern auch überlegen, ob sich das Problem ohne zusätzliche Datenfelder irgendwie lösen läßt (z.B. durch das gefährliche "Ummumerieren").

Abb. 55: Übungsbeispiel für die Beschreibung von Elementen					
1. Element	2. Element	3. Element	4. Element	5. Element	6. Element
M	IX	357	a		
Element Nr. (1)			Maximale Zeichenzahl: 1		
Sortierung: (<input checked="" type="checkbox"/>) alphanumerisch () numerisch () römische Zahl () Andere: _____					
Funktion: () vollst. Kürzel (<input checked="" type="checkbox"/>) Teil von Kürzel () Jahreszahl () laufende "Nr."					
Element Nr. (2)			Maximale Zeichenzahl: 5		
Sortierung: () alphanumerisch () numerisch (<input checked="" type="checkbox"/>) römische Zahl () Andere: _____					
Funktion: () vollst. Kürzel (<input checked="" type="checkbox"/>) Teil von Kürzel () Jahreszahl () laufende "Nr."					
Element Nr. (3)			Maximale Zeichenzahl: 4		
Sortierung: () alphanumerisch (<input checked="" type="checkbox"/>) numerisch () römische Zahl () Andere: _____					
Funktion: () vollst. Kürzel () Teil von Kürzel () Jahreszahl (<input checked="" type="checkbox"/>) laufende "Nr."					
Element Nr. (4)			Maximale Zeichenzahl: 1		
Sortierung: (<input checked="" type="checkbox"/>) alphanumerisch () numerisch () römische Zahl () Andere: _____					
Funktion: () vollst. Kürzel () Teil von Kürzel () Jahreszahl (<input checked="" type="checkbox"/>) laufende "Nr."					

Trennzeichen (Abb. 56)

In diesem Abschnitt geht es um die Frage, woran ein Programm den Wechsel von einem Element zum nächsten erkennen kann und wie dieser Wechsel zu verstehen ist.

- Bei der zunächst erfolgenden **Beschreibung der Trennzeichen** sind mehrere Angaben möglich (wenn die Schreibweise ungleichmäßig ist), in diesem Falle sollte aber der gewünschte Standardtrenner herausgehoben werden. Auf dem Arbeitsblatt sind die häufigsten Trenner bereits aufgeführt und brauchen nur angestrichen zu werden:

Achtung: Innerhalb des gleichen Datenfeldes darf ein Trenner nur eine einzige Funktion haben:

- **Leertaste:** Üblich und sinnvoll (und ausreichend), wenn ein Wechsel zwischen Zeichenklassen erfolgt (z.B. zwischen Buchstaben und arabischen Zahlen).
 - **Schrägstrich, Punkt, Komma, Doppelpunkt, Semikolon, Minuszeichen (Bindestrich);** auf dem Arbeitsblatt werden aus Platzgründen nur diese Zeichen selber aufgeführt.
 - **Andere:** Hier soll z.B. vermerkt werden, wenn es keinen Trenner gibt (nur wenige Programme können damit sinnvoll umgehen) oder weitere, oben nicht aufgeführte, Zeichen verwendet wurden.
- **Für:** Hier soll die **Funktion** des betreffenden Trenners erläutert werden. Gefragt wird also, was der Trenner nun wirklich trennt.
 - **Elemente einer selbständigen Nummer:** Normalfall (eine einzige Nummer)
 - **Bereiche:** Für den häufigen Fall (s.o. "Bereichsangaben"), daß mehrere, direkt aufeinander folgende Nummern bzw. "Unternummern" zusammengefaßt werden (z.B. A 317 - A 320 oder 1954/336 a-c)
 - **Mehrfachangaben:** Für den seltenen Fall (s.o. "Sammelangaben"), daß mehrere, nicht direkt aufeinanderfolgende Nummern genannt werden **M VII 478 & M IX 12** . Wenn die zweite usw. Nummer der gleichen Art angehört, braucht man diese dann nicht auch noch detailliert zu analysieren.

Abb. 56: Übungsbeispiel für die Beschreibung von Trennern					
Feldname: INVNR1			Datei: OBJEKT		
Beispiel: M IX 357 a - c					
1. Element 47	2. Element 1968	3. Element a	4. Element c	5. Element	6. Element
Trennzeichen: () Leertaste / <input checked="" type="checkbox"/> . , : ; - () Andere: _____ für: (X) Elemente einer selbständigen Nr. () Bereiche () Mehrfachangaben					
Trennzeichen: (X) Leertaste / . , : ; - () Andere: _____ für: (X) Elemente einer selbständigen Nr. () Bereiche () Mehrfachangaben					
Trennzeichen: () Leertaste / . , : ; -X () Andere: _____ für: () Elemente einer selbständigen Nr. (X) Bereiche () Mehrfachangaben					

Seite 2: Kürzel und ihre Auflösung

Die Seite 2 der Arbeitsblätter (abgebildet auf Seite 113) dient der Auflistung bzw. Auflösung von Kürzeln. Sie bedarf keiner umständlichen Erläuterung, ein paar Beispiele mögen genügen; die ersten beiden (A bzw. BS) für die üblichen, meist der Nummer vorangestellten Sachgruppenkürzel, die nächsten zwei (M I bzw. M IX) für die ebenfalls nicht seltenen, aus Buchstaben und römischen Zahlen zusammengesetzten Kürzel, das letzte (C) für den seltenen Fall eines "nachgestellten" Sachgruppenkürzels (Abb. 57).

Abb. 57: Übungsbeispiel für eine Auflistung von Kürzeln		
Feldname: INVNR1		Datei: OBJEKT
Element Nr.	Kürzel	Erläuterungen
1	A	Klassisches Altertum
"	BS	Blechspielzeug
1-2	M I	Provinz Brandenburg
"	M IX	Baltikum
3	C	Frühmittelalter

Arbeitsblätter für Numerierungssysteme					
Seite 1: Syntax					
Feldname:			Datei:		
Beispiel:					
1. Element	2. Element	3. Element	4. Element	5. Element	6. Element
Sortier-Reihenfolge: () von links nach rechts () andere (Reihenfolge der Element-Nummern): _____					
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Element Nr. ()			Maximale Zeichenzahl: _____		
Sortierung: () alphanumerisch () numerisch			() römische Zahl () Andere: _____		
Funktion: () vollst. Kürzel () Teil von Kürzel			() Jahreszahl () laufende "Nr."		
Trennzeichen: () Leertaste / . , : ; -			() Andere: _____		
für: () Elemente einer selbständigen Nr.			() Bereiche () Mehrfachangaben		
Name:			Datum:		

Arbeitsblätter zum Zeichensatz

Zur Wiederholung: Ein Computer speichert Informationen nicht als Text oder als Zahl, sondern in einer internen Codierung, die nur aus 0 und 1 besteht (Dual- oder Binärsystem). Ein Zeichen (z.B. ein Buchstabe) wird bei praktisch allen Computern durch eine Folge von 8 Binärziffern (auch Byte genannt) repräsentiert, wodurch 256 (2^8) verschiedene Zeichen darstellbar sind. Die Interpretation eines internen Codes durch ein bestimmtes Zeichen wird durch eine Code-Tabelle festgelegt, welche jedem internen Code ein Zeichen als äußere Repräsentation zuordnet.

Leider gibt es nicht nur einen, sondern mehrere solcher Codierungen oder Standards, die ihrerseits wiederum unzählige "Dialekte" haben. Das wirkt sich nicht nur auf die Darstellung ("Welche Zeichen gibt es?"), sondern auch auf die Sortierung der Zeichen aus:

- **Im PC-Bereich** ist ein erweiterter ASCII-Code (American Standards Code of Information Interchange) weitgehend Standard. Leider ist beim ASCII-Code nur die Bedeutung von 128 Zeichen normiert, die anderen 128 Zeichen wurden von IBM mit speziellen Grafikzeichen und mit nationalen Sonderzeichen "landesspezifisch" belegt, woraus für PC's eine Art "De-facto-Standard" wurde.

Die Sortierreihenfolge der wichtigsten Zeichen in ASCII ist:

Zahlen
Großbuchstaben
Kleinbuchstaben

- **Im Großrechnerbereich** wird meist der EBCDIC-Code benutzt, der ebensowenig eine Festlegung für nationale Sonderzeichen enthält. Im Vergleich zu ASCII ist dort die Reihenfolge vertauscht, d.h.:

Kleinbuchstaben
Großbuchstaben
Zahlen

- **Programmiersprachen:** Die meisten in Programmiersprachen vorhandenen Sortierprogramme sortieren Texte einfach nach der Wertigkeit ihrer Zeichen (d.h. nach der Interpretation des internen Codes als Zahl), wodurch also Groß- und Kleinbuchstaben getrennt werden.
- **Anwenderprogramme:** Wichtiger als diese internen Codes ist für Sie, welche Sortierreihenfolge die von Ihnen benutzten Anwenderprogramme haben. Selbstverständlich wird vom Computer erwartet, daß er ganz unabhängig von irgendwelchen internen Codes "richtig" sortiert. Sie können solche Dinge bereits mit "Textverarbeitung" machen, dürfen sich aber nicht darauf verlassen, daß das Programm genau das tut, was Sie wollen.

Die folgenden Arbeitsblätter sollen Ihnen dabei helfen, dieses Problem etwas besser in den Griff zu bekommen und dabei genau zu definieren, was Sie für Ihre Arbeit brauchen. Als Beispiel wird der in Deutschland auf den meisten Personal Computern benutzte ASCII-Zeichensatz benutzt. Auf den Arbeitsblättern finden Sie eine Auswahl der für die meisten Anwendungen im Museum wichtigsten Zeichen:

- **Sonderzeichen** (Seite 117)
- **Großbuchstaben** (Seite 118): Aus praktischen Gründen incl. der im deutschen üblichen "nationalen" Sonderzeichen (Umlaute).
- **Kleinbuchstaben** (Seite 119): Ebenfalls incl. Umlaute und "ß"
- **Nationale Alphabete** (Seite 120)
- **Zahlen** (Seite 121)

- **Beschreibung weiterer Zeichen** (Seite 121): Leeres Arbeitsblatt. Gedacht ist auch hierbei zunächst an Zeichen aus dem in Deutschland üblicherweise verwendeten ASCII-Code. Natürlich kann man diesen Code auch mit anderen, z.B. griechischen oder tibetanischen Zeichen belegen, das ist aber bei Datenbankanwendungen heute immer noch eine knifflige Angelegenheit und wird hier nicht in Form von Arbeitsblättern abgehandelt.

Im **Kopf der Arbeitsblätter** wird zunächst nach den betroffenen Datenfeldern gefragt. Dies berücksichtigt, daß besonders bei der Sortierung verschiedene Regelungen erforderlich sein können. Das gilt z.B. für Inventarnummern, bei denen nationale Alphabete normalerweise nicht vorkommen und die Sonderzeichen oft anders zu behandeln sind als bei normalen Schlagwörtern. Je nach Lage der Dinge füllen Sie also nur einen oder auch mehrere Sätze dieser Arbeitsblätter aus.

Die **Spalten der Arbeitsblätter** beschränken sich auf wenige, aber grundlegende Informationen. Um Ihnen die Arbeit zu erleichtern, wurden Antworten, die für praktisch alle Systeme gelten, bereits ausgefüllt, dabei allerdings eingeklammert:

- **Code:** Wenn nichts anderes im Programm festgelegt wurde, dient der Zahlenwert des Codes für die Sortierung. Wir erkennen jetzt, warum im ASCII-Code die Zahlen (48-57) vor den Buchstaben (Großbuchstaben: 65-90) sortiert werden.

Schaut man sich den internen Code der Sonderzeichen (Seite 117) an, so wird die folgende Sortierung verständlich:

Nummer (1)
 Nummer -1-
 Nummer 1
 Nummer <1>
 Nummer ?
 Nummer «1»
 Nummer unbekannt
 Nummer: 1
 Nummer: unbekannt

Wenn Sie das z.B. in einem Textverarbeitungssystem selber ausprobieren, werden Sie evtl. eine andere Reihenfolge erhalten (z.B. alle Sonderzeichen vor den Zahlen).

Merke: Ein Anwenderprogramm kann eine vom internen Code abweichende Sortierreihenfolge festlegen. Bei Kleinbuchstaben ist das üblich (s.u. "Sortierung").

- **Zeichen:** Übliche Darstellung der Zeichen (falls nötig, Erläuterung bei "Bemerkungen").
- **Darstellung:** Die Zeichen, die wir beim Druck oder auf dem Bildschirm sehen, werden vom Programm der internen Codierung zugeordnet. Es ist also durchaus üblich, daß z.B. ein amerikanisches Programm keine Umlaute kann bzw. den entsprechenden Code mit einem anderen Zeichen belegt hat. Im Extremfall kann Ihnen ein Wort wie "Österreich" ein Programm "knacken" (in manchen Systemen entspricht das "Ö" der programmintern häufig verwendeten "eckigen Klammer").

Bei "normalen" Buchstaben und Zahlen ist die Sache weitgehend genormt, bei "nationalen" Zeichen (und dazu gehören die Umlaute und das "ß") muß man darauf achten.

- **Druck:** Ob das Zeichen gedruckt werden kann, hängt vom Drucker ab.
- **Bildschirm:** Das gleiche gilt für den Bildschirm. Textverarbeitungsprogramme sind hier meist "besser" als Datenbanksysteme. Manchmal werden Sie einer "Ersatzdarstellung" (meist Kombination mehrerer Zeichen) begegnen.
- **Eingabe:** Dieser Punkt ist wichtig für nicht genormte Zeichen, d.h. besonders für ein paar Sonderzeichen und nationale Alphabete. Falls ein Zeichen nicht auf der Tastatur

vorkommt, kann eine Eingabe oft durch eine "Ersatzdarstellung" (s.o. "Bildschirm") oder durch gleichzeitiges Drücken mehrerer Tasten erfolgen (z.B. ALT+Code).

Falls das Zeichen bei den meisten in Deutschland benutzten Geräten auf der Tastatur vorkommt, wird das durch "(Ts)" gekennzeichnet.

- **Daten:** Hier sollen Sie die Frage beantworten, ob das betreffende Zeichen in den Daten vorkommt oder nicht. Wichtig ist das besonders bei den "Sonderzeichen": Wenn in Ihrer Inventarnummer Schrägstriche vorkommen können, dann darf man dieses Zeichen nicht als "Trenner" verwenden.
- **Sortierung:** Die Sortierung erfolgt in erster Linie nach dem Code. Kleinbuchstaben werden aber im allgemeinen wie die entsprechenden Großbuchstaben sortiert. Bei manchen nationalen Zeichen geschieht die Sortierung, als ob es sich um einen Doppelbuchstaben handeln würde (dazu ausführlich das Kapitel "Was ist eigentlich eine richtige alphabetische Ordnung" auf Seite 24 ff.).

Hier müssen Sie eindeutige Entscheidungen treffen, denn in einem deutschen Register wird z.B. "ñ" wie "N", in einem spanischen als eigener Buchstabe zwischen "N" und "O" sortiert. Wenn Sie ein Datenfeld haben, aus dem Sie ein spanisches Register erzeugen wollen, brauchen Sie dafür also eine andere Sortierreihenfolge der Zeichen als in einem deutschen bzw. gemischten Register.

Die in deutschsprachigen Registern üblichen Sortierwerte sind bereits eingetragen: "(Cd)" für Code, sonst der entsprechende Großbuchstabe, z.B. "(N)".

- **Bemerkungen:** Erläuterungen soweit nötig (z.B. "Leertaste"). Die Spalte kann zur Definition von Dingen wie der automatischen Ersetzung eines Zeichens durch ein anderes benutzt werden (dazu Kapitel "Produktorientierte Beschreibung von Datenfeldern" auf Seite 99 ff.).

Arbeitsblätter für Zeichensatz (Seite 1)					Sonderzeichen (ASCII)		
Datenfeld (Name bzw. "alle"): _____							
Code	Zeichen	Darstellung		Eingabe	Daten	Sortierung	Bemerkungen
		Druck	Bildsch.	Taste/Ersatz	ja/nein	Code/Zeichen	
32		(j)	(j)	(Ts)	.	.	Leertaste
33	!	(j)	(j)	(Ts)	.	.	
34	"	(j)	(j)	(Ts)	.	.	
35	#	(j)	(j)	(Ts)	.	.	
36	\$	(j)	(j)	(Ts)	.	.	
37	%	(j)	(j)	(Ts)	.	.	
38	&	(j)	(j)	(Ts)	.	.	
39	'	(j)	(j)	(Ts)	.	.	
40	((j)	(j)	(Ts)	.	.	
41)	(j)	(j)	(Ts)	.	.	
42	*	(j)	(j)	(Ts)	.	.	
43	+	(j)	(j)	(Ts)	.	.	
44	,	(j)	(j)	(Ts)	.	.	
45	-	(j)	(j)	(Ts)	.	.	
46	.	(j)	(j)	(Ts)	.	.	
47	/	(j)	(j)	(Ts)	.	.	
58	:	(j)	(j)	(Ts)	.	.	
59	;	(j)	(j)	(Ts)	.	.	
60	<	(j)	(j)	(Ts)	.	.	
61	=	(j)	(j)	(Ts)	.	.	
62	>	(j)	(j)	(Ts)	.	.	
63	?	(j)	(j)	(Ts)	.	.	
156	£	
174	«	
175	»	

Arbeitsblätter für Zeichensatz (Seite 2)				Deutsche Großbuchstaben (ASCII)			
Datenfeld (Name bzw. "alle"): _____							
Code	Zeichen	Darstellung		Eingabe	Daten	Sortierung	Bemerkungen
		Druck	Bildsch.	Taste/Ersatz	ja/nein	Code/Zeichen	
65	A	(j)	(j)	(Ts)	(j)	(Cd)	
66	B	(j)	(j)	(Ts)	(j)	(Cd)	
67	C	(j)	(j)	(Ts)	(j)	(Cd)	
68	D	(j)	(j)	(Ts)	(j)	(Cd)	
69	E	(j)	(j)	(Ts)	(j)	(Cd)	
70	F	(j)	(j)	(Ts)	(j)	(Cd)	
71	G	(j)	(j)	(Ts)	(j)	(Cd)	
72	H	(j)	(j)	(Ts)	(j)	(Cd)	
73	I	(j)	(j)	(Ts)	(j)	(Cd)	
74	J	(j)	(j)	(Ts)	(j)	(Cd)	
75	K	(j)	(j)	(Ts)	(j)	(Cd)	
76	L	(j)	(j)	(Ts)	(j)	(Cd)	
77	M	(j)	(j)	(Ts)	(j)	(Cd)	
78	N	(j)	(j)	(Ts)	(j)	(Cd)	
79	O	(j)	(j)	(Ts)	(j)	(Cd)	
80	P	(j)	(j)	(Ts)	(j)	(Cd)	
81	Q	(j)	(j)	(Ts)	(j)	(Cd)	
82	R	(j)	(j)	(Ts)	(j)	(Cd)	
83	S	(j)	(j)	(Ts)	(j)	(Cd)	
84	T	(j)	(j)	(Ts)	(j)	(Cd)	
85	U	(j)	(j)	(Ts)	(j)	(Cd)	
86	V	(j)	(j)	(Ts)	(j)	(Cd)	
87	W	(j)	(j)	(Ts)	(j)	(Cd)	
88	X	(j)	(j)	(Ts)	(j)	(Cd)	
89	Y	(j)	(j)	(Ts)	(j)	(Cd)	
90	Z	(j)	(j)	(Ts)	(j)	(Cd)	
142	Ä	(j)	(j)	(Ts)	(j)	(AE)	
153	Ö	(j)	(j)	(Ts)	(j)	(OE)	
154	Ü	(j)	(j)	(Ts)	(j)	(UE)	

Arbeitsblätter für Zeichensatz (Seite 3)		Deutsche Kleinbuchstaben (ASCII)				
Datenfeld (Name bzw. "alle")						
Code	Zeichen	Darstellung		Eingabe	Daten	Sortierung
		Druck	Bildsch.	Taste/Ersatz	ja/nein	Code/Zeichen
						Bemerkungen
97	a	(j)	(j)	(Ts)	(j)	(A)
98	b	(j)	(j)	(Ts)	(j)	(B)
99	c	(j)	(j)	(Ts)	(j)	(C)
100	d	(j)	(j)	(Ts)	(j)	(D)
101	e	(j)	(j)	(Ts)	(j)	(E)
102	f	(j)	(j)	(Ts)	(j)	(F)
103	g	(j)	(j)	(Ts)	(j)	(G)
104	h	(j)	(j)	(Ts)	(j)	(H)
105	i	(j)	(j)	(Ts)	(j)	(I)
106	j	(j)	(j)	(Ts)	(j)	(J)
107	k	(j)	(j)	(Ts)	(j)	(K)
108	l	(j)	(j)	(Ts)	(j)	(L)
109	m	(j)	(j)	(Ts)	(j)	(M)
110	n	(j)	(j)	(Ts)	(j)	(N)
111	o	(j)	(j)	(Ts)	(j)	(O)
112	p	(j)	(j)	(Ts)	(j)	(P)
113	q	(j)	(j)	(Ts)	(j)	(Q)
114	r	(j)	(j)	(Ts)	(j)	(R)
115	s	(j)	(j)	(Ts)	(j)	(S)
116	t	(j)	(j)	(Ts)	(j)	(T)
117	u	(j)	(j)	(Ts)	(j)	(U)
118	v	(j)	(j)	(Ts)	(j)	(V)
119	w	(j)	(j)	(Ts)	(j)	(W)
120	x	(j)	(j)	(Ts)	(j)	(X)
121	y	(j)	(j)	(Ts)	(j)	(Y)
122	z	(j)	(j)	(Ts)	(j)	(Z)
129	ü	(j)	(j)	(Ts)	(j)	(UE)
132	ä	(j)	(j)	(Ts)	(j)	(AE)
148	ö	(j)	(j)	(Ts)	(j)	(OE)
225	ß	(j)	(j)	(Ts)	(j)	(SS)

Arbeitsblätter für Zeichensatz (Seite 4)					Nationale Alphabete (ASCII)	
Datenfeld (Name bzw. "alle")						
Code	Zeichen	Darstellung		Eingabe	Daten	Sortierung
		Druck	Bildsch.	Taste/Ersatz	ja/nein	Code/Zeichen
						Bemerkungen
128	Ç	(C)
130	é	(E)
131	â	(A)
133	à	(A)
134	â	(A)
135	ç	(C)
136	ê	(E)
137	ë	(E)
138	è	(E)
139	ï	(I)
140	î	(I)
141	ì	(I)
143	Å	(A)
144	É	(E)
145	æ	(AE)
146	Æ	(AE)
147	ô	(O)
149	ò	(O)
150	û	(U)
151	ù	(U)
152	ÿ	(Y)
155	ø	(C)
160	á	(A)
161	í	(I)
162	ó	(O)
163	ú	(U)
164	ñ	(N)
165	Ñ	(N)

Glossar

Aliasnamen (alias names): Durch Vergabe von Aliasnamen zu einem → Feldnamen kann ein → Datenfeld unter verschiedenen Namen angesprochen werden. Zum Feldnamen "Objektbezeichnung" können z.B. die Aliasnamen OBJBEZ und O vergeben werden, wodurch der geübte Benutzer einige Schreibarbeit spart. Auch für den Datenaustausch kann die Vergabe von Aliasnamen wichtig sein, da gleichwertige, in verschiedenen Datenbanken aber unterschiedlich benannte, Datenfelder dadurch vereinheitlicht werden können.

Betriebssystem (operating system): Zusammenfassende Bezeichnung für verschiedene Programme, die zur Ausführung von Benutzerprogrammen erforderlich sind. Das Betriebssystem stellt elementare Funktionen für die Ein- und Ausgabe von Daten zur Verfügung, welche eine Anpassung an die unterschiedlichen Ein- bzw. Ausgabe-medien (Tastatur, verschiedene Bildschirme, Disketten, Festplatten) vornehmen. Ein Benutzerprogramm braucht daher nicht an eine spezielle Gerätekonfiguration angepaßt zu werden.

Weiterhin gehören zu einem Betriebssystem noch verschiedene Dienstprogramme, welche Standardprobleme wie Sortieren, Kopieren, Datensicherung usw. lösen.

Während im Großrechnerbereich jeder Hersteller ein eigenes Betriebssystem anbietet, haben sich im PC- und Kleinrechnerbereich herstellerunabhängige Systeme etabliert.

Derartige Betriebssysteme sind:

- **MS-DOS:** Industriestandard im PC-Bereich
- **OS/2:** Von IBM 1987 als Nachfolger für MS-DOS vorgestelltes Betriebssystem, welches sich (noch) nicht durchsetzen konnte.
- **UNIX:** Ein vor allem im Bereich der mittleren Datentechnik ("Workstations") sehr verbreitetes Betriebssystem, welches aber in diversen Versionen vom PC bis zum Großrechner verfügbar ist.

Datei (file): Eine Datei enthält gleichartig strukturierte → Datensätze und entspricht

etwa einem Karteikasten in der konventionellen Dokumentation.

Dateien werden vom → Betriebssystem verwaltet, d.h. sie können z.B. gelöscht oder auf einen anderen Datenträger kopiert werden.

Datenbank (data base): Eine Datenbank ist für den Benutzer eine vollständige Anwendung, bestehend aus den anwendungsspezifischen → Dateien und den zugehörigen Verwaltungs- und Auswertungsprogrammen. Physikalisch besteht eine Datenbank meist aus zahlreichen Dateien, deren Verwaltung ausschließlich vom Datenbankprogramm vorgenommen werden sollte.

Datenbankbeschreibung (data dictionary): Die Datenbankbeschreibung enthält eine vollständige Definition aller in einer Datenbank enthaltenen → Dateien, deren → Datenfelder und deren Relationen untereinander. Jeder verändernde Zugriff des Datenbankprogramms auf eine Datenbank sollte durch die Datenbankbeschreibung gesteuert werden (z.B. strikte Einhaltung von Konsistenzregeln bei Datenänderungen).

Datenbankprogramm (data base management system - DBMS): Das Datenbankprogramm stellt die Schnittstelle zwischen dem Benutzer und der Datenbank zur Verfügung. Es enthält Programme für alle zur Bearbeitung einer Datenbank notwendigen Aktionen wie Eingabe bzw. Korrektur von Daten, Retrieval innerhalb der Daten, Sortierung nach vorgebbaren Kriterien sowie Ausgabe ausgewählter Daten nach bestimmten Formatierungsanweisungen.

Einige Datenbankprogramme enthalten eine Programmschnittstelle, welche einen Zugriff auf die Daten auch durch externe Programme ermöglicht.

Datenfeld (data field, data item): Ein Datenfeld ist die kleinste von einem → Datenbankprogramm anzusprechende Informationseinheit eines → Datensatzes. Es besteht aus dem → Feldnamen und einem Wert.

Beispiele: "Museumsname *Heimatismuseum Emsdetten*", "Inventarnummer 78/127", "Objektbezeichnung *Glasschrank*".

In der → Datenbankbeschreibung wird für jedes Datenfeld außer dem → Feldtyp (in den obigen Beispielen alphanumerisch) festgelegt, ob das Feld ein → Mehrfachfeld, ein → Indexfeld oder ein → Wiederholungsfeld ist, ob das Feld in jedem Datensatz einen Wert haben muß (obligatorisches Feld, Muß-Kategorie) und ob die Eingabe anhand einer Liste oder einer → Hilfsdatei überprüft werden soll.

Datensatz (record): Ein Datensatz faßt die zu einem Beschreibungsobjekt gehörenden → Datenfelder zusammen. Er entspricht damit etwa einer Karteikarte bei der konventionellen Dokumentation.

Datenmodell: Ziel einer Datenbankanwendung ist die Nachbildung eines Teils der realen Welt, z.B. die wissenschaftliche Beschreibung eines Museumsobjekts in einem Inventarisierungsprojekt. Zur Nachbildung der Realität in einer Datenbank bedient man sich eines Datenmodells.

Ein Datenmodell beschreibt den formalen Aufbau aller in einer Datenbank enthaltenen → Datenfelder sowie ihre Beziehungen untereinander (Die häufig verwendete "flache" Datenstruktur, d.h. eine listenartige Aufzählung von miteinander nicht verbundenen Datenfeldern, stellt in diesem Sinne kein Datenmodell dar).

Man unterscheidet das → hierarchische Datenmodell, das → Netzwerkdatenmodell und das → relationale Datenmodell.

In letzter Zeit werden in der Literatur häufig auch → objektorientierte Datenmodelle und → Entity-Relationship-Datenmodelle beschrieben, sie spielen aber mangels Realisierung in am Markt erhältlichen Datenbankprogrammen noch keine Rolle.

DBMS (data base management system)
→ Datenbankprogramm

Entity-Relationship-Modell: Das Entity-Relationship-Modell ist ein Hilfsmittel für den konzeptuellen Entwurf eines Datenmodells.

Es geht davon aus, daß der von einem Datenmodell zu beschreibende Teil der realen Welt aus unterscheidbaren Dingen, Konzepten oder Ereignissen besteht, welche Entities genannt werden (der eingedeutschte Begriff Entität ist nicht sehr

verbreitet). Beispiele für Entities sind Rembrandt, "Der Mann mit dem Goldhelm" oder die Dahlemer Gemäldegalerie.

Die für eine spezielle Anwendung "ähnlichen" Entities faßt man zu einem Entity Set zusammen, als z.B. alle Künstler, die im Thieme-Becker erwähnt sind oder alle Ölgemälde einer Gemäldegalerie.

Zwischen den Entity Sets einer Anwendung bestehen Beziehungen, die sog. Relationships (z.B. der Künstler A ist Entwerfer des Bildes B). Relationships sind auch innerhalb eines Entity Sets möglich (Künstler A ist Lehrer des Künstlers B).

Die Relationships zwischen den Entity Sets einer Anwendung können in standardisierten Entity-Relationship-Diagrammen graphisch dargestellt und nach festen Regeln z.B. in ein relationales Datenmodell übertragen werden.

Feldnamen: Über den Feldnamen spricht der Benutzer eines Datenbanksystems die verschiedenen → Datenfelder an. Der Benutzer muß also nicht wissen, an welcher Stelle eines → Datensatzes sich z.B. die Objektbezeichnung verbirgt, das → DBMS erledigt dies. Feldnamen sollten kurz sein und mnemotechnisch geschickt formuliert werden. Viele Datenbanksysteme erlauben die Vergabe von → Aliasnamen, d.h. von mehreren gleichwertigen Feldnamen für ein Datenfeld.

Feldtyp: In der → Datenbankbeschreibung wird zu jedem → Datenfeld ein Feldtyp festgelegt. In diesem wird angegeben, welche Art von Daten in dieses Feld eingetragen werden dürfen. Verbreitete Feldtypen sind alphanumerisch, numerisch, Boolean und Datum. Die Unterstützung der verschiedenen Feldtypen hängt sehr stark vom verwendeten Datenbankprogramm ab. Einige Datenbankprogramme erlauben zusätzlich die Definition eigener Feldtypen.

Gruppenfelder (group fields): Gruppenfelder fassen inhaltlich zusammengehörende → Datenfelder zu Einheiten zusammen, die vom → Datenbanksystem gemeinsam angesprochen werden können. So kann z.B. eine Adresse (bestehend aus Straße, Postleitzahl, Ort, Zustellbezirk, Nationalitätenkennzeichen usw.) zu einem Gruppenfeld

erklärt werden, welches komplett kopiert, gelöscht usw. werden kann.

Beim → hierarchischen Datenmodell ist die Gruppenbildung das entscheidende Strukturierungsmerkmal.

Hierarchisches Datenmodell: Grundlage des hierarchischen Datenmodells ist die Baumstruktur. Dabei werden → Datenfelder zu → Gruppenfeldern zusammengefaßt, welche ihrerseits wieder Teile von übergeordneten Gruppenfeldern sein können. Z.B. werden die zu einer Literaturangabe erforderlichen Datenfelder wie Autor, Zeitschrift, Jahrgang usw. zu einem Gruppenfeld Literatur zusammengefaßt, welches zusammen mit der Inventarnummer, der Objektbezeichnung usw. ein weiteres Gruppenfeld Inventar bildet.

Durch die beliebige Wiederholbarkeit von Datenfeldern (→ Mehrfachfelder) und die Wiederholung von Gruppenfeldern (z.B. mehrere Literaturangaben zu einem Objekt) lassen sich Museumsobjekte im hierarchischen Datenmodell relativ "natürlich" beschreiben. Relationen zwischen verschiedenen Dateien sind möglich, müssen aber für die einzelne Anwendung angepaßt werden, da kein vollständiges mathematisches Modell dahintersteht (wie beim relationalen Datenmodell).

Hilfsdateien (Sekundärdateien) werden in einer Datenbankanwendung zumeist bei der Ein- oder Ausgabe benutzt. Bei der Eingabe können sie z.B. der Überprüfung des zulässigen Wortschatzes für ein Datenfeld dienen. Bei einem Ausdruck können Daten aus einer Hilfsdatei eingefügt werden (z.B. die Lebensdaten eines Künstlers oder die Oberbegriffe eines Thesaurus, welche nicht bei allen Datensätzen, sondern nur einmal in einer Hilfsdatei gespeichert wurden).

Indexfelder (indexed field): Ein Indexfeld erlaubt den direkten (und damit schnellen) Zugriff auf eine Datenbank in einer anderen Sortierreihenfolge, als diese vom → Satzschlüssel vorgegeben ist. Wird z.B. in einer Inventardatei die Objektbezeichnung zu einem Indexfeld erklärt, können die → Datensätze am Bildschirm (oder in einem Ausdruck) auch in der alphabeti-

schen Reihenfolge der Objektbezeichnungen angezeigt werden.

Indexfelder beschleunigen das Retrieval erheblich, verlangsamen aber die Eingabe bzw. Korrektur, da die dafür notwendigen invertierten Register bei jeder Änderung angepaßt werden müssen.

Bei der Definition eines Indexfeldes wird angegeben, ob ein Wert nur einmal in der Datenbank vorkommen darf ("unique"), wie z.B. eine Inventarnummer, oder nicht (z.B. Objektbezeichnung).

Mehrfachfelder (multivalued fields) sind → Datenfelder, die mehrere gleichberechtigte Werte enthalten können, z.B. mehrere Materialangaben zu einem Objekt.

Netzwerkdatenmodell: Beim Netzwerkdatenmodell können beliebige Beziehungen zwischen → Datenfeldern verschiedener Dateien definiert werden. Bei einer Unterteilung eines Museumsobjekts in mehrere Teilobjekte lassen sich z.B. Literaturangaben zu mehreren dieser Teilobjekte in Relation setzen.

Da die Abbildung von netzwerkartigen Datenstrukturen auf reale Speichermedien erhebliche Schwierigkeiten macht, spielt dieses Datenmodell im PC-Bereich keine Rolle.

Objektorientiertes Datenmodell: Beim objektorientierten Datenmodell werden für zusammengehörende Informationen sog. Klassen gebildet, z.B. eine Klasse Literaturangaben.

Bei der Bildung einer Klasse werden auch alle Methoden definiert, mit denen auf Objekte der Klasse zugegriffen werden kann. Diese Methoden sind Bestandteil der Klasse, nicht der Verarbeitungsprogramme (Prinzip der Kapselung - encapsulation). Eine Ausdruckenweisung für die Literaturangabe ist also Bestandteil der entsprechenden Klassendefinition und wird nicht für jedes Ausgabeprodukt neu formuliert. Da der Zugriff auf Objekte einer Klasse ausschließlich über die in ihr definierten Methoden möglich ist, lassen sich Konsistenzbedingungen für die enthaltenen Daten leicht formulieren und überprüfen.

Eine einmal definierte Klasse kann einfach in andere Anwendungen übernommen wer-

den, wobei vorhandene Methoden bei Bedarf überladen werden (z.B. durch andere Ausgabeanweisungen). Die Klassen eines objektorientierten Datenmodells können hierarchisch gegliedert werden. In einer abgeleiteten Klasse müssen nur die Methoden definiert werden, die sich von den entsprechenden Methoden der Ausgangsklasse unterscheiden (Prinzip der Vererbung - Inheritance).

Beim objektorientierten Datenmodell haben die Klassen eine ähnliche Funktion wie die Gruppen beim hierarchischen Modell, jedoch ist durch die feste Einbindung der zugehörigen Methoden in die entsprechenden Klassendefinitionen die technische Realisierung eleganter.

Vom Ansatz her ist dieses Datenmodell für die museumsüblichen Daten hervorragend geeignet, entsprechende Datenbanksysteme befinden sich aber noch im Entwicklungsstadium.

Relationales Datenmodell: Grundlegende Struktur des relationalen Datenmodells ist die Tabelle. Die Zeilen der Tabelle entsprechen den → Datensätzen und die Spalten den → Datenfeldern. Die Reihenfolge der Spalten oder Zeilen einer Tabelle ist im relationalen Datenmodell unwesentlich, die Identifizierung einer Zeile erfolgt ausschließlich über den Satzschlüssel und die Identifizierung einer Spalte über den Feldnamen. Jeder Feldname muß innerhalb der gesamten Datenbank eindeutig sein, wodurch → Wiederholungsfelder nicht möglich sind. Mehrere Tabellen (Dateien) einer relationalen Datenbank können durch gemeinsame → Datenfelder miteinander verknüpft werden. Dazu muß der Satzschlüssel der einen Tabelle ein Datenfeld der anderen Tabelle sein, z.B. kann in einer Inventarisierungsdatei über ein Museumskürzel eine Relation zu einer Museumsdatei hergestellt werden (in welcher das Museumskürzel der Satzschlüssel ist).

Durch die Tabellenstruktur einer einzelnen Datei ist der zulässige Wertebereich (domain) mathematisch als Menge beschreibbar und auch alle Verknüpfungen verschiedener Tabellen miteinander sind mengentheoretisch darstellbar. Für das relationale Datenmodell existiert mit SQL

(structured query language) eine weitgehend standardisierte Datenbankbeschreibungssprache.

Obwohl die Tabelle als Grundstruktur des relationalen Datenmodells leicht verständlich ist, erfordert die Gesamtkonzeption einer relationalen Datenbank einen erheblichen Planungsaufwand, da zur Vermeidung von Redundanzen einige Normalisierungen vorgenommen werden müssen. Redundante Daten sind extrem fehleranfällig bei Korrekturen, da eine Änderung immer bei allen zugehörigen Sätzen durchgeführt werden muß.

Die z.B. bei einer Inventarisierung sinnvollen → Mehrfachfelder stehen im Widerspruch zum relationalen Datenmodell und lassen sich in diesem nur über programmtechnische Tricks realisieren.

Viele im PC-Bereich verbreiteten Datenbanksysteme (dBase, Paradox, Clipper, DataEase usw.) werden zwar als relational bezeichnet, haben aber mit dem Datenmodell oft nur die tabellenartige Dateistruktur gemeinsam.

Satzschlüssel (key field, primary key): Ein oder mehrere → Datenfelder, welche einen → Datensatz eindeutig identifizieren. Bei einem Inventar kann dies z.B. die Inventarnummer (für mehrere Museen evtl. eine Zusammensetzung aus einem Museumskürzel und der Inventarnummer) sein.

Häufig wird auch eine künstliche **Dokumentnummer** vergeben, welche keinerlei inhaltliche Bedeutung hat und nur zur Identifizierung der Datensätze dient.

SQL (structured query language) → Relationales Datenmodell

Wiederholungsfelder (repeatable fields) sind → Datenfelder, die innerhalb eines → Datensatzes an verschiedenen Stellen vorkommen können (besonders beim → hierarchischen Datenmodell).

Register

- Abkürzungsdateien 39
- Alphabetische Ordnung 24
- Alphabetische Register 26
- Alphanumerische Sortierung 22, 50
- Ballast 81
- Betriebssysteme 123
 - MS-DOS 123
 - OS/2 123
 - UNIX 123
- Dateien 13, 123
 - Hilfsdateien 33, 125
 - Abkürzungsdateien 39
 - Lexika 39
 - Präferenz- und Stopplisten 38
 - Thesauri 41, 72
- Datenbankbeschreibung 123
- Datenbanken 123
- Datenbankprogramm 123
- Datenfelder 13, 123
 - Aliasnamen 123
 - Dokumenten-Nummer 55, 85
 - Feindefinition 64
 - Feldname im Klartext 97
 - Feldnamen 97, 124
 - Feldtyp 124
 - Grobkategorien 65
 - Gruppenfelder 81, 124
 - Indexfelder 125
 - Inhaltliche Definition 98
 - Komplizierte Syntax 48, 50, 78
 - Mehrfachfelder 79, 125
 - Mußkategorie 99, 103
 - Programminterne Feldnamen 97
 - Satzschlüssel 85, 99, 126
 - Verweise auf andere Dateien 33, 73
 - Verweise auf die gleiche Datei 84
 - Wiederholungsfelder 90, 126
- Datenkataloge 60, 73, 90
- Datenmodelle 89, 124
 - Entity-Relationship-Modell 124
 - Hierarchisch 125
 - Netzwerk 125
 - Relational 126
- Datensätze 13, 124
- Datenstrukturen 60
 - Arbeitsblatt 95
 - Flach 67, 77
 - Hierarchisch 81
 - Komplex 76
 - Verweise auf andere Dateien 33, 73
 - Verweise innerhalb einer Datei 84
- Datierung (Beispiele) 19
- DBMS 123
- Dialog-Retrieval 31
- Dokumenten-Nummer 55, 85
- Dokumentenummer 126
- Drucken von Textdaten 21
- Entity-Relationship-Modell 124
- Feldlänge 21
- Feldname im Klartext 97
- Feldnamen 97, 124
- Feldtyp 124
- Flache Datenstrukturen 67, 77
- Freier Text 21
- Grobkategorien 65
- Grundbegriffe
 - Ballast 81
 - Freier Text 21
 - Redundanz 38
 - Saubere und schmutzige Daten 14
 - Textdaten 18
 - Vereinheitlichung der Daten 15
- Gruppenfelder 81, 124
- Hierarchische Datenstrukturen 81
- Hierarchisches Datenmodell 125
- Hilfsdateien 125
 - Abkürzungsdateien 39
 - Funktionsweise 33
 - Lexika 39
 - Präferenz- und Stopplisten 38
 - Thesauri 41, 72
- Indexfelder 125
- Inventarnummern (Beispiele) 48
- Invertierte Register 31, 125
- Komplexe Datenstrukturen 76
- Komplizierte Syntax 50, 70, 78
- Lexika 39
- MDA-Data Standard 90
- Mehrfachfelder 79, 125
- MS-DOS 123
- Mußkategorie 99, 103
- Negativmethode 39
- Netzwerkdatenmodell 125
- Numerierungssysteme 48
- Objektbezeichnungen (Beispiele) 33
- Objektorientiertes Datenmodell 125
- Ortsangaben (Beispiele) 68
- OS/2 123
- Permutierte Register 26
- Positivmethode 38
- Präferenz- und Stopplisten 38
- Programminterne Feldnamen 97

- Redundanz 38
- Register
 - alphabetisch 26
 - invertiert 31, 125
 - permutiert 26
 - Sortierkriterien 24, 29, 36, 64, 107
 - Sortierreihenfolge 25, 36, 56, 70, 72, 106, 108, 114
 - systematisch 26, 33
- Relationales Datenmodell 126
- Reportgeneratoren 21
- Retrieval
 - Allgemeines 30
 - Dialog 31
 - Volltextrecherche 30
- Satzschlüssel 85, 99, 126
- Saubere und schmutzige Daten 14
- Sekundärdateien 125
- Sortieren
 - Alphanumerisch 22, 50
 - Chronologisch 19
 - Komplizierte Syntax 50, 70, 78
 - Lexika, Telefonbücher 24
 - Sortierkriterien 24, 29, 36, 64, 107
 - Sortierreihenfolge 25, 36, 56, 70, 72, 106, 108, 114
 - Textdaten (allgemein) 22
- SQL 126
- Stopplisten 38
- Stufenmodell 45
- Systematische Register 26, 33
- Terminologie 43
- Textdaten 18
 - Beispiele
 - Datierung 19
 - Inventarnummern 48
 - Kombination mehrerer Felder 76
 - Objektbezeichnungen 33
 - Ortsangaben 68
 - Sortieren
 - Allgemein 22
 - Komplizierte Syntax 50, 70, 78
 - Wiedergabe (Druck) 21
- Thesauri 41, 72
- Trenner 50, 78
- UNIX 123
- Vereinheitlichung der Daten 15
- Verweise auf andere Dateien 33, 73
- Verweise auf die gleiche Datei 84
- Volltextrecherche 30
- Wiederholungsfelder 90, 126
- Zeichensatz 21, 102, 114

VERÖFFENTLICHUNGEN AUS DEM INSTITUT FÜR MUSEUMSKUNDE

Staatliche Museen Preußischer Kulturbesitz

MATERIALIEN AUS DEM INSTITUT FÜR MUSEUMSKUNDE

(zu beziehen durch: Institut für Museumskunde, In der Halde 1, D - 1000 Berlin 33)

Heft 1 - 3 in einem Band (2. Aufl. 1984):

Heft 1: Christof Wolters

Benutzerhandbuch Datenerfassung und Datenkorrektur
(215 S.)

VERGRIFFEN

Heft 2: Peter - Georg Hausmann

Beispiele von Korrekturprodukten

Beiheft zum Benutzerhandbuch Datenerfassung und Datenkorrektur
(125 S.)

VERGRIFFEN

Heft 3: Christof Wolters

Informationssystem Museumsobjekte

Bericht über das 1978 - 1980 im Auftrag des Deutschen Museumsbundes e.V.
durchgeführte Pilotprojekt

Mit einem Vorwort von Stephan Waetzoldt
(94 S.)

VERGRIFFEN

Heft 4

Erhebung der Besuchszahlen

an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1981
Berlin 1982 (30 S.)

Heft 5

Günter S. Hilbert

Eine neue konservatorische Bewertung der Beleuchtung in Museen
Berlin 1983 (69 S.)

VERGRIFFEN

Heft 6

Erhebung der Besuchszahlen

an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1982
Berlin 1983 (25 S.)

Heft 7

Andreas Grote

Materialien zur Geschichte des Sammelns

Zwei Vorträge in Israel 1982 und 1983
Englisch mit deutschen Resümees
Berlin 1983 (63 S.)

Heft 8

Erhebung der Besuchszahlen

an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1983
Berlin 1984 (25 S.)

Heft 9

Hans - Joachim Klein

Analyse der Besucherstrukturen an ausgewählten Museen
in der Bundesrepublik Deutschland und in Berlin (West)
Berlin 1984 (220 S.)

VERGRIFFEN

Heft 10

Eintrittsgeld und Besuchsentwicklung an Museen
der Bundesrepublik Deutschland mit Berlin (West)
Berlin 1984 (36 S.)

- Heft 11 **Bibliographie - Report zu den Gebieten Museologie, Museumspädagogik und Museumsdidaktik**
Berlin 1984 (160 S.) erw. Neuauflage Heft 19
- Heft 12 **Wissenschaftliche Volontäre**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
Berlin 1984 (96 S.) erw. Neuauflage Heft 20
- Heft 13 Carlos Saro und Christof Wolters
Handbuch Datenerfassung - Kleine Museen
Berlin 1985 (209 S. und 140 S. Anhang) **VERGRIFFEN**
(wird in dieser Form nicht neu aufgelegt)
- Heft 14 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1984
Berlin 1985 (32 S.)
- Heft 15 **Entwicklung von Museumskonzeptionen**
in der Bundesrepublik Deutschland und Berlin (West) 1974 - 1985
Berlin 1985 (46 S.)
- Heft 16 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1985
Including an English Summary
Berlin 1986 (39 S.)
- Heft 17 **Gutachten zur Änderung der Öffnungszeiten**
an den Staatlichen Museen Stiftung Preußischer Kulturbesitz
Erstellt von Hans - Joachim Klein
Berlin 1986 (77 S.)
- Heft 18 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1986
Including an English Summary
Berlin 1987 (40 S.)
ISSN 0931-7961 Heft 18 **VERGRIFFEN**
- Heft 19 **Bibliographie - Report 1987 zu den Gebieten Museologie, Museumspädagogik und Museumsdidaktik**
Berlin 1987 (203 S.)
ISSN 0931-7961 Heft 19 **VERGRIFFEN**
(erweiterte Neuauflage Heft 29)
- Heft 20 **Wissenschaftliche Volontäre**
an den Museen und Denkmalämtern der Bundesrepublik Deutschland samt
Berlin (West)
Berlin 1987 (131 S.)
ISSN 0931-7961 Heft 20
- Heft 21 Petra Schuck - Wersig, Martina Schneider und Gernot Wersig
Wirksamkeit öffentlichkeitsbezogener Maßnahmen für Museen und kulturelle Ausstellungen
Berlin 1988 (64 S.) erweiterte Neuauflage 1989
ISSN 0931-7961 Heft 21

- Heft 22 Traudel Weber, Annette Noschka
Texte im Technischen Museum
Textformulierung und Gestaltung, Verständlichkeit, Testmöglichkeiten
Including an English Summary
Berlin 1988 (72 S.)
ISSN 0931-7961 Heft 22
- Heft 23 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1987
Including an English Summary
Berlin 1988 (46 S.)
ISSN 0931-7961 Heft 23
- Heft 24 Carlos Saro und Christof Wolters
EDV - gestützte Bestandserschließung in kleinen und mittleren Museen
Bericht zum Projekt "Kleine Museen" für den Zeitraum 1984 - 1987.
Including an English Summary
Berlin 1988 (135 S.)
ISSN 0931-7961 Heft 24
- Heft 25 Petra Schuck - Wersig, Gernot Wersig
Museen und Marketing
Marketingkonzeptionen amerikanischer Großstadtmuseen als
Anregung und Herausforderung
Including an English Summary
Berlin 1988 (112 S.)
ISSN 0931-7961 Heft 25
- Heft 26 Andrea Prehn
Versicherung in Museen und Ausstellungen
Berlin 1989 (103 S.)
ISSN 0931-7961 Heft 26
- Heft 27 Annette Noschka-Roos, Monika Hagedorn-Saupe
Museumspädagogik in Zahlen - Erhebungsjahr 1988
Berlin 1989 (77 S.)
ISSN 0931-7961 Heft 27
- Heft 28 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West)
für das Jahr 1988
Including an English Summary
Berlin 1989 (56 S.)
ISSN 0931-7961 Heft 28 VERGRIFFEN
- Heft 29 **Bibliographie - Report 1987 zu den Gebieten Museologie,**
Museumspädagogik und Museumsdidaktik
mit referierendem Bibliographie - Teil
zum Sachgebiet Besucherforschung
Berlin 1989 (289 S.)
ISSN 0931-7961 Heft 29 VERGRIFFEN
- Heft 30 Jane Sunderland und Lenore Sarasan
Was muß man alles tun, um den Computer
im Museum erfolgreich einzusetzen?
Mit einer Einleitung von Christof Wolters
Berlin 1989 (79 S.)
ISSN 0931-7961 Heft 30

- Heft 31 **Erhebung der Besuchszahlen**
an den Museen der Bundesrepublik Deutschland samt Berlin (West) mit
Besuchszahlenangaben zu den Museen der (ehemaligen) DDR
für das Jahr 1989
Berlin 1989 (64 S.)
ISSN 0931-7961 Heft 31
- Heft 32 Hans-Joachim Klein und Barbara Wüsthoff-Schäfer
Inszenierung an Museen und ihre Wirkung auf Besucher
Karlsruhe 1990 (141 S.)
ISSN 0931-7961 Heft 32
- Heft 33 Christof Wolters
**Wie muß man seine Daten formulieren bzw. strukturieren,
damit ein Computer etwas Vernünftiges damit anfangen kann ?**
Berlin 1991 (133 S., 64 Abb.)
ISSN 0931-7961 Heft 33

BERLINER SCHRIFTEN ZUR MUSEUMSKUNDE

Bei Bezug über das Institut für Museumskunde (Bestellkarte) räumen die Verlage einen Rabatt ein.

Band 1-7 zu beziehen durch: Gebr. Mann Verlag, Berlin

- Bd. 1 Günter S. Hilbert
Sammlungsgut in Sicherheit
Teil 1: Sicherheitstechnik und Brandschutz
Berlin 1981
ISBN 3-7861-1288-6
- Bd. 2 Hans - Joachim Klein und Monika Bachmayr
Museum und Öffentlichkeit
Fakten und Daten - Motive und Barrieren
Berlin 1981
ISBN 3-7861-1276-2
- Bd. 3 **Ausstellungen - Mittel der Politik ?**
Internationales Symposium
10. - 12. September 1980 in Berlin, veranstaltet vom Institut für Museumskunde,
Staatliche Museen Preußischer Kulturbesitz Berlin
und vom Institut für Auslandsbeziehungen Stuttgart
Red. Klaus Bleker und Andreas Grote
Berlin 1981
ISBN 3-7861-1316-5
- Bd. 4 Bernhard Graf und Heiner Treinen
Besucher im Technischen Museum
Zum Besucherverhalten im Deutschen Museum München
Berlin 1983
ISBN 3-7861-1378-5

- Bd. 5 **Wolfger Pöhlmann**
Ausstellungen A - Z
Gestaltung, Technik, Organisation
Berlin 1988
ISBN 3-7861-1453-6
- Bd. 6 **Günter S. Hilbert**
Sammlungsgut in Sicherheit
Teil 2: Lichtschutz und Klimatisierung
Berlin 1987
ISBN 3-7861-1452-8
- Bd. 7 **Martin Roth**
Heimatmuseum
Zur Geschichte einer deutschen Institution
ISBN 3-7861-1452-8

Zu beziehen über den Buchhandel:

- Bd. 8 **Hans-Joachim Klein**
Der gläserne Besucher
Publikumsstrukturen einer Museumslandschaft
ISBN 3-7861-1452-8

WEITERE PUBLIKATIONEN

Zu beziehen durch Bleicher Verlag, Gerlingen:

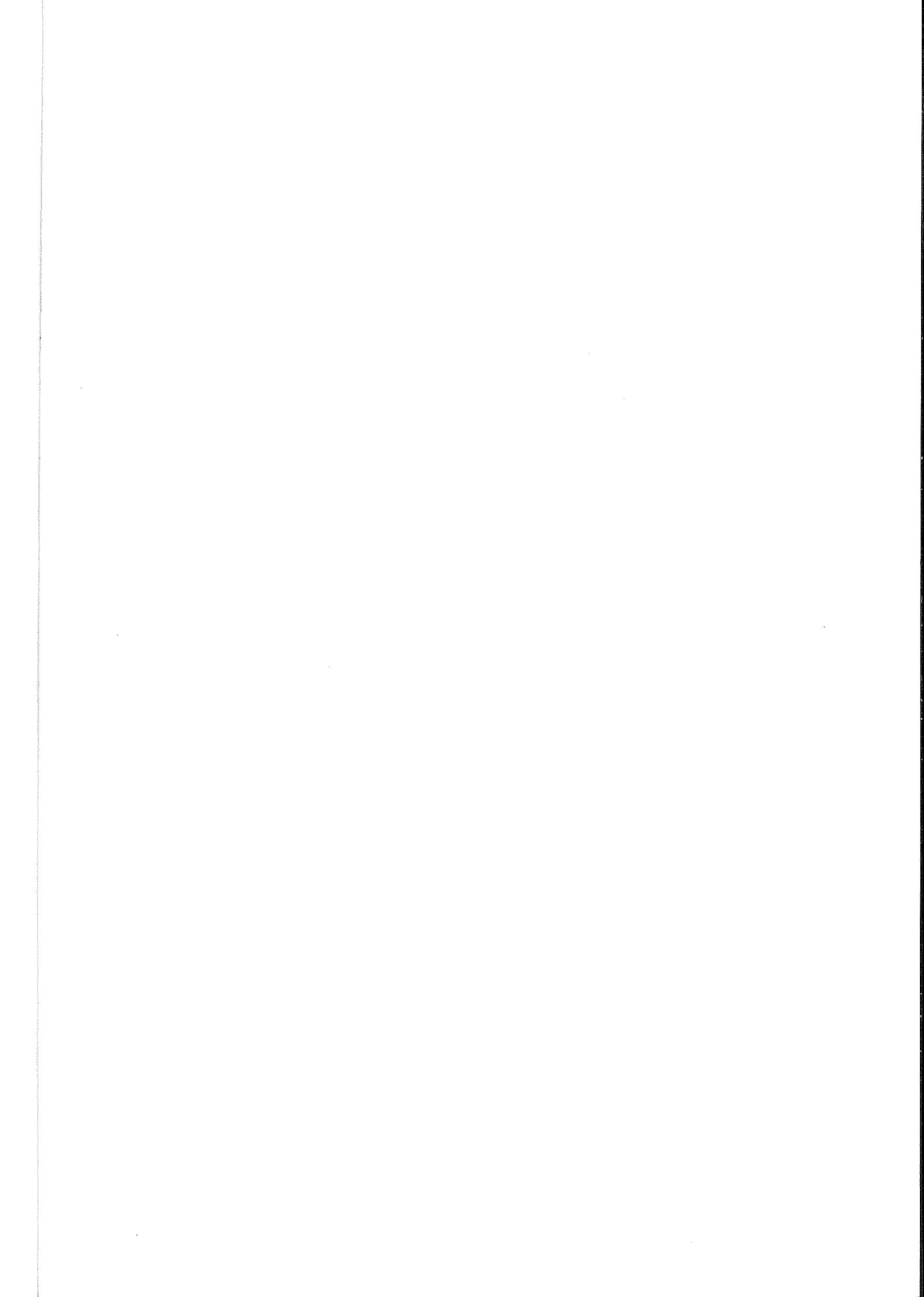
Robert Bosch Stiftung (Hrsg.)
Kunstförderung - Steuerstaat und Ökonomie
Beiträge zu einem Kolloquium der Robert Bosch Stiftung und der
Stiftung Preußischer Kulturbesitz
Gerlingen 1987
ISBN 3-88350-580-3

Zu beziehen durch Deutsches Museum, München:

B. Graf und G. Knerr (Hrsg.)
Museumsausstellungen • Planung • Design • Evaluation.
Deutsches Museum München in Zusammenarbeit mit dem Institut für
Museumskunde, Berlin und der Robert Bosch Stiftung, Stuttgart

Zu beziehen über das Institut für Museumskunde, Berlin:

Landesstelle für Museumsbetreuung in Zusammenarbeit mit dem
Museumsverband Baden-Württemberg e.V. (Hrsg.)
Christof Wolters
Vorschläge zur Planung zentraler Dienstleistungen für Museen
in der Bundesrepublik Deutschland und Berlin (West)
in: Museumsblatt 2, 1990, S. 6-26.



S M
B Institut für Museumskunde

ISSN 0931-7961 Heft 33

Materialien aus dem **Institut für Museumskunde**