

Das Juristische Literatur Archiv JULIA

Herbert Göttler, Uwe Günzel, Bernd Himmelreich*

1. Hintergrund / Motivation

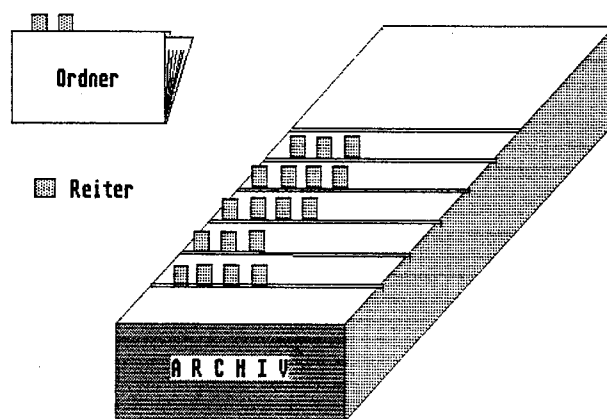
Die Computerunterstützung für Rechtsanwaltskanzleien schien zumindest bis Mai 1982 unbefriedigend gewesen zu sein, denn der Deutsche Anwaltsverein (DAV) beschloß auf einer Vorstandssitzung zu dieser Zeit, verstärkte Anstrengungen zu unternehmen, um den Anwälten bei ihren Rationalisierungsmaßnahmen helfen zu können. Im August 1983 gründete der DAV das „Institut der Anwaltschaft für Büroautomatisierung und Bürotechnik GmbH“, das einem der Ausschüsse des DAV zugeordnet wurde. Das Institut und der Ausschuß fördern Softwareprojekte, die anwaltliche Arbeit unterstützen sollen.

Unabhängig davon trug zu Beginn des Jahres 1983 ein Nürnberger Rechtsanwalt an den Lehrstuhl für Programmiersprachen die Bitte heran, eine Studie für ihn zu erstellen, welche Arbeitsvorgänge in seiner Kanzlei durch EDV-Einsatz unterstützt werden könnten. Wir übernahmen die Aufgabe als Projekt und konzipierten nach Abschluß der Problemanalyse ein System, das um eine Stammdatenverwaltung herum einen Aktenspiegel, eine Buchhaltung, einen Termin/Fristenkalender sowie ein cleveres Textbausteinverwaltungssystem als optimale Konstellation für eine Kanzlei zum Ziel hatte. Die Ergebnisse der damaligen Aktivitäten können in [4], [5] und [6] nachgelesen werden.

Der DAV wurde auf die Arbeit des Lehrstuhls für Programmiersprachen der Universität Erlangen-Nürnberg aufmerksam und hat über seinen Ausschuß ein weiteres Projekt angeregt, das von dem, mit der (bundeseigenen) Gesellschaft für Mathematik und Datenverarbeitung (GMD) assoziierten „Verein Recht und Information e. V.“ im zweiten Halbjahr 1988 finanziert worden ist: Es sollte ein computergestützter Arbeitsplatz für Literatursuche entwickelt werden, der für Laien einfach zu bedienen ist und auf ihre (juristische) Verständnisebene zugeschnitten sein sollte. Diesem Projekt JULIA stand noch das Institut für Wirtschafts- und Arbeitsrecht unserer Universität beratend zur Seite.

Zwei Gründe waren ausschlaggebend, sich mit dem JULIA-Projekt zu befassen:

Erstens ist es auf einem so anwendungsorientierten Gebiet, wie es die Software-Entwicklung darstellt, im Hochschulbereich besonders wichtig, Projekterfahrung zu vermitteln, um nicht bei der Diskussion über Methodeneinsatz auf diesem Feld passen zu müssen. Freilich gibt es auch im universitären Bereich eine Reihe z.T. sehr schwieriger Aufgaben, an denen man Software-Entwicklungsmethoden studieren kann. Meist sind diese Probleme aber selbstgestellt, d.h. Auftraggeber und Ausführende(r) sind weitgehend identisch. Für das Sammeln von Erfahrungen in der Projektarbeit kann es aber schon von Nachteil sein, wenn Auftraggeber und -nehmer zwar verschiedene Personen sind, aber gleiche oder ähnliche Vorbildung haben. Dann fällt aber zumindest in der Problemanalysephase die Erschwernis weg, daß Auftraggeber und -nehmer i.a. „verschiedene Sprachen sprechen“. Der Reiz war also auszuprobieren, ob wir das beherrschen, was einen guten Informatiker vom schlechten zu unterscheiden gestattet: die Kunst der „partizipativen Phänomenologie“, was nichts weiter bedeuten soll, als die streng objektive Aufzeichnung und Beschreibung des Gegebenen und des Beabsichtigten zusammen mit den Auf-



Figur 1

traggebern. Ziel der „Ist- und Sollanalyse“ ist die Entwicklung eines geeigneten „Modells“, das die Gesamtheit aller notwendigen Informationen zur Beschreibung des vollständig oder teilweise zu automatisierenden Systems repräsentiert. Das Modell muß einerseits präzise genug sein, um Mißverständnisse oder Fehldeutungen auszuschließen, es muß aber anschaulich genug sein, damit der Auftraggeber sich zurechtfinden kann.

Der zweite Grund ist das große Interesse der Rechtsanwälte an einem Software-Produkt, wie es JULIA realisieren soll. Das Finden von einschlägigen Paragraphen und Urteilen zu einem Fall ist der Schlüssel zum Erfolg einer Kanzlei. Es war zu erwarten, daß die Anforderungen, die aus dem juristischen Bereich stammen, zu einem Produkt führen würden, das auch auf anderen Gebieten gut einsetzbar wäre.

2. Das Modell

Ein Informatiker bekommt - wie oben bereits angedeutet - das Wissen eines Anwendungsgebietes nur dann weitgehend unverfälscht, wenn er ein Modell des Sachverhaltes mit den Denk-, Sprech- und Arbeitsweisen des Anwendungsbereiches entwickelt. Wir mußten also die Frage untersuchen, wie ein Rechtsanwalt seine Informationen speichert und wiederfindet.

Anwaltliches Wissen wird aus verschiedenen Quellen hergeleitet: Eigene Bibliotheken, Zeitschriften, Vertrags- und Rechtsprechungsarchive, oder die entsprechenden Einrichtungen Dritter, externe Datenbanken, usw. Der Informationsträger ist fast ausschließlich Papier, das in irgendwelchen Sammelvorrichtungen abgelegt ist. Ein Jurist könnte ideal arbeiten, wenn er vom Zeitungsausschnitt bis zum voluminösen Gerichtsurteil dieses Papier in einen Hängeordner („Jurismappe“, „Aktenordner“ wären ebenfalls zulässige aber nicht ganz so brauchbare Modellvorstellungen) stecken und diesen dann mit beliebig vielen beliebigen Reitern (äquivalent dazu wären „Stichwörter“, „Indices“, usw.) versehen könnte, die ihm den Inhalt erschließen. Will er eine abgelegte Information wiederfinden, wäre es nur nötig, auf die gewünschten, frei kombinierbaren Reiter zuzugreifen.

Herbert Göttler, Uwe Günzel, Bernd Himmelreich sind Mitarbeiter am Lehrstuhl für Programmiersprachen, Friedrich-Alexander Universität Erlangen-Nürnberg, Martensstr. 3, 8520 Erlangen

Alle Funktionen der Informationsspeicherung- und -wiedergewinnung lassen sich mit diesem „Hängeordner/Reitermodell“ ausdrücken, s. Fig. 1. Dieses mechanistische, der Erfahrungswelt von Rechtsanwälten entlehnte Modell mag im nachhinein simpel erscheinen, es dauerte aber eine geraume Zeit, bis es klar formuliert war und sowohl von den Entwicklern als auch von den Auftraggebern als einziges zulässiges Kommunikationsmittel akzeptiert wurde. (Es ist immer bequemer, sich im eigenen Jargon auszudrücken.)

Neben der Hängeordnerregistratur gibt es in unserem Modell noch zwei „Listen“ (äquivalent dazu: „Karteikästen“). In der einen sind die Reiter alphabetisch, in der anderen hierarchisch gegliedert zusammengefaßt. Auf diese Listen, künftig auch „Thesauri“ genannt, wird im folgenden noch eingegangen.

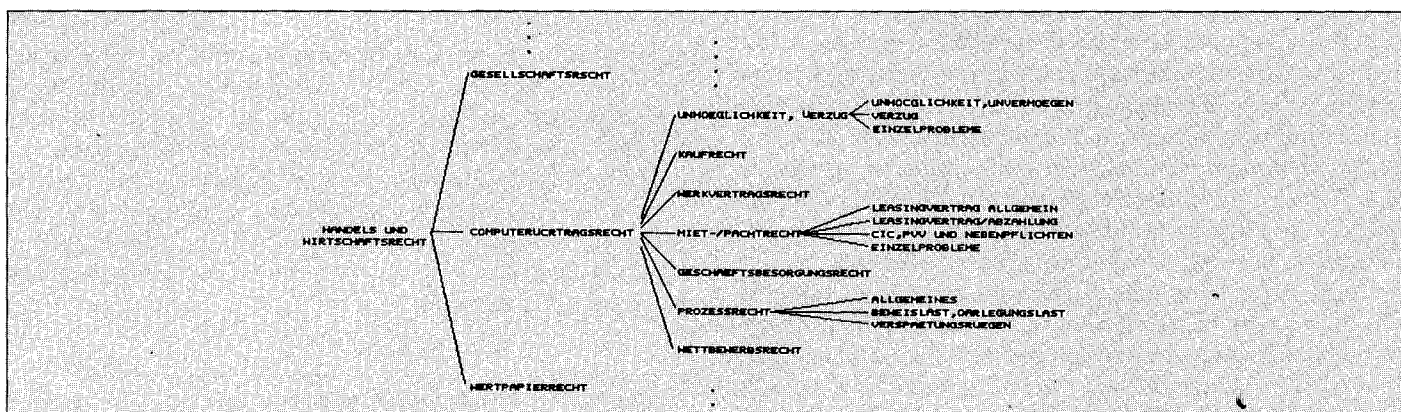
Durch Disziplin bei der Handhabung des geschilderten Modells – z.B. war verboten, von einer „Datei“ zu sprechen – wurde die Funktionalität, d.h. die gewünschten Benutzerfunktionen, präzise festgelegt. Dabei zeigte sich auch, daß das Konzept von JULIA für die gesamte Registratur geeignet wäre, denn auch der Name eines Mandanten oder Gegners könnte Reiter eines Hängeordners sein.

Im Verlauf der Problemanalyse schälte sich ein von den existierenden Literaturarchivsystemen nicht oder nur unvollständig berücksichtigtes Problem als zentral heraus: das „Redakteur-Benutzer-Problem“. Dieses hat seinen Ursprung in der Tatsache, daß bei der Erstellung umfangreicher Textarchive der Verwalter („Redakteur“) des Archivs sich bei der Vergabe von Reitern für die abzulegenden Texte oftmals nicht mit dem späteren Benutzer über die zu verwendenden Reiterbegriffe abstimmen kann. (Das Problem tritt auch dann auf, wenn Redakteur und Benutzer identisch sind.) Eine Schwierigkeit liegt z.B. darin, daß es verschiedene Stichwörter gleicher Bedeutung gibt (Synonyme), wie z.B. „Jura“ und „Rechtswissenschaft“ oder „illegal“ und „ungesetzlich“. In den gleichen Problemkreis fallen auch die Verwendung von gebräuchlichen Abkürzungen, wie „KFZ“, „BGB“, sowie die verschiedenen Schreibweisen eines Wortes, z.B. „Graphik“ und „Grafik“. Die andere Schwierigkeit ergibt sich aus der Existenz von Homonymen, also gleichlautenden Begriffen, deren Bedeutung kontextabhängig ist. So kann z. B. der Begriff „Steuer“ sowohl eine finanzielle Abgabe als auch das Lenkrad eines Autos bezeichnen.

Das Setzen unterschiedlicher Schwerpunkte bei der Reiterverwendung oder die Verwendung unterschiedlicher Stichwörter für gleichbedeutende Begriffe kann im schlechtesten Fall dazu

führen, daß der Benutzer die für ihn relevanten Texte im Archiv nicht findet ([2] spricht hierbei von einem zu geringen „Wiedergewinnungsmaß“), oder daß zu viele irrelevante Texte gefunden werden (nach [2] ein zu geringes „Präzisionsmaß“). In unserer Arbeit, s. [7], sollte daher als wichtigste Sollvorgabe ein Weg gefunden werden, das „Redakteur-Benutzer-Problem“ zu entschärfen und ein hohes Präzisions- und Wiedergewinnungsmaß zu garantieren. Als ersten Schritt zur Lösung des Problems erkannten wir die Organisation der Reiter in Form der bereits zitierten Thesauri. Ein Thesaurus ist hier eine (gut) verwaltete Kartei von Begriffen (Reitern), in der der Bezug der Begriffe zueinander sowie alle Synonyme und Homonyme eines Begriffs aufgenommen und als solche gekennzeichnet werden. Der Benutzer und der Redakteur erhalten dabei jederzeit die Möglichkeit, die Thesauri am Bildschirm abzurufen, um sich über die verwendbaren Reiterbegriffe zu informieren. Der Redakteur kann die Thesauri auch verändern.

Für die Organisation der Reiterthesauri bieten sich zwei Formen an, alphabetisch oder hierarchisch geordnet. (Eine eingehende Diskussion der Problematik findet man in [9].) Die alphabetische Ordnung bedarf keiner weiteren Erläuterung. Die Fig. 2 ist ein Beispiel für eine hierarchische Ordnung. Das Bild zeigt einen Ausschnitt aus einer möglichen gegliederten Organisation des Handels- und Wirtschaftsrechtes, das sich unterteilen läßt in Gesellschaftsrecht, Computervertragsrecht, Wertpapierrecht, usw. Diese Begriffe lassen sich noch weiter unterteilen. An diesem Bild sieht man auch, daß eine hierarchische Ordnung der Begriffe eine elegante Homonymverwaltung gestattet. Die Bedeutung eines Homonyms wird durch seine Position in der Hierarchie festgelegt. Wie man sehen kann, kommt der Begriff „EINZELPROBLEME“ an zwei Stellen im Hierarchiebaum vor. „EINZELPROBLEME“ ist also ein Homonym, dessen Bedeutung durch den Kontext präzisiert wird, einmal als „EINZELPROBLEME (bei UNMOEG- LICHKEIT, VERZUG)“ und „EINZELPROBLEME (bei MIET-/PACHTRECHT)“. (Anmerkung: JULIA erlaubt den Gebrauch von Umlauten. Daß hier und in den anderen Bildern die Ersatzdarstellungen oe-, ae- bzw. ue benutzt werden, liegt daran, daß zur Erstellung der Bilder der sogen. DOS-Screen-Print verwendet wurde, der für Umlaute andere Sonderzeichen einsetzt. Reiter werden von JULIA aber stets in Großbuchstaben umgewandelt.) Eine solche Begriffs-Hierarchie kann dem Redakteur bei der Reitervergabe wirksame Hilfe leisten, indem sie zu jedem Begriff selbständig die dazugehörigen Ober- und Unterbegriffe anbietet. Auch der Benutzer kann davon profitieren. Wenn er aus der alphabetischen Kartei einen Begriff auswählen will, wird ihm ggf. angezeigt,



Figur 2

daß der Begriff mehrfach, also als Homonym, in der Hierarchie vorkommt. Für diesen Fall bietet das System als Hilfe an, sich die verschiedenen Positionen des Homonyms im Hierarchiebaum anzeigen zu lassen und so durch die Angabe von Ober- bzw. Unterbegriffen den Suchbegriff zu präzisieren.

3. Funktionalität

Aufgrund der umfangreichen Vorarbeiten bei der Modellierung und der Erstellung des Pflichtenhefts sowie mit der Vorgabe, ein relationales Datenbanksystem für die spätere Implementierung zu verwenden, erwies sich die Spezifikation der Programm-Modulen und der Entwurf des Datenmodells als relativ einfache Aufgabe: Die Modularisierung konnte sehr eng anhand der klar aufgetrennten Funktionsbereiche „Textablage“, „Textsuche“ und „Verwaltungsfunktionen“ erfolgen. Die Fig. 3 gibt einen Überblick über die Funktionen von JULIA.

<JULIA>		HAUPTMENU		<VERSION 1.1>	
SYSTEMZUSTANDS-ÜBERSICHT					
Benutzer-Identifikation	:	USER			
Anzahl der Ordner im Textarchiv	:	45 Ordner			
Anzahl der Ordner im Suchergebnis	:	45 Ordner			
Anzahl der Suchbegriffe im Suchprofil	:	0 Stichworte			
Anzahl der Reiter-Karteikarten...	:	333 Stichworte			
...in der alph. Reiterkartei	:	51 Stichworte			
...in der hier. Reiterkartei	:	51 Stichworte			
Wahl der Suchworte		Suche durchführen		Ergebnisausgabe	
Neuen Text ablegen		Verarbeitung		Ordner-Verwaltung	
Alph. Kartei-Verw.		Hier. Kartei-Verw.		ZURÜCK	--><--

Figur 3

Schwieriger war die Umsetzung unseres konzeptuellen Anspruchs, eine (prinzipiell) unbegrenzte Anzahl von Reitern auf einer (prinzipiell) unbegrenzten Anzahl von Textordnern zu verwalten und außerdem sowohl einen alphabetisch als auch einen hierarchisch gegliederten Thesaurus von a priori unbegrenzter Größe zu ermöglichen. Dieses Ziel wurde erreicht. Eine weitere Schwierigkeit war die optimale Gestaltung der Benutzerschnittstelle, die über Erfolg und Akzeptanz eines Software-Pakets entscheidet, insbesondere, wenn das Software-Produkt von DV-Laien angewendet werden soll. Daher legten

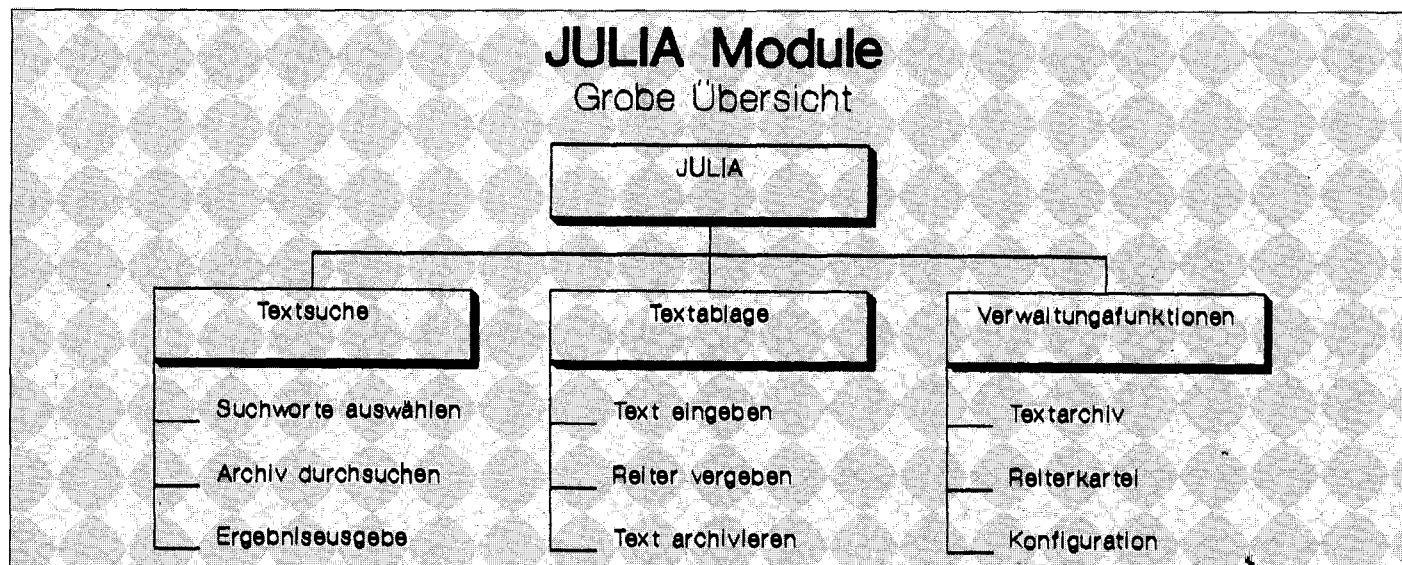
wir auch großen Wert auf eine angemessene Konzeption und eine verständliche Benutzerführung. Dabei mußten wir die Erfahrung machen, daß z.B. das Finden von griffigen, plastischen und prägnanten Bezeichnungen für die einzelnen Systemfunktionen beileibe keine triviale Aufgabe ist und einiges an Abstimmung mit den beteiligten Rechtsanwältinnen erforderte. Auch für eine sinnvolle Bildschirmmasken-Gestaltung und -Aufteilung erwiesen sich die Hinweise und Anregungen der späteren Benutzer als hilfreich.

JULIA ist ein menügesteuertes System, wobei sich die Funktionen jeweils durch einen Tastendruck aktivieren lassen. Bei der Gestaltung eines Benutzerdialogs sollte dem Benutzer ein möglichst geschlossenes, einheitliches Konzept geboten werden. JULIA präsentiert sich dem Anwender daher in allen Programmteilen in einer festgelegten Form der Bildschirmaufteilung. Fig. 4 zeigt exemplarisch den Aufbau des JULIA-Hauptmenüs.

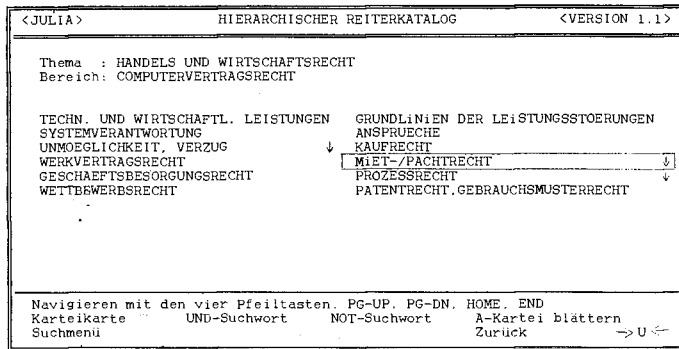
Am oberen Bildschirmrand befindet sich die Titelzeile, wo der Name der aktivierten Programmfunktion ausgegeben wird. Dadurch kann der Benutzer auf den ersten Blick feststellen, in welchem Programmteil er sich gerade befindet. Der mittlere Teil des Bildschirms dient vollständig der Eingabe oder Ausgabe von Informationen aller Art. So sieht man in Fig. 4 eine Systemzustandsübersicht, in der die Anzahl der abgelegten Texte, die Anzahl der dem System bekannten Stichwörter und weitere Informationen aufgeführt sind. Durch diese Art der Bildschirmaufteilung in feste Funktionsbereiche muß der Benutzer die für ihn relevanten Informationen nicht erst aus dem meist umfangreichen Bildschirmangebot herausuchen, sondern findet zusammengehörige Informationen immer an gewohnter Stelle. Dies erleichtert die Einarbeitung in die Programmbedienung.

3.1 Darstellung der Karteikästen

Wie bereits angedeutet, werden die beiden Ordnungsstrukturen (alphabetisch bzw. hierarchisch) anschaulich als Karteikästen modelliert, auf deren Karteikarten jeweils ein Reiterbegriff steht. Zunächst erklären wir den Aufbau der hierarchischen Kartei, von der ein Ausschnitt in Fig. 2 graphisch als Baum dargestellt wird. JULIA setzt diese Begriffs-Hierarchie auf dem Bildschirm um, wie es Fig. 5 zeigt.



Figur 4



Figur 5

Zentral sieht man die graphische Darstellung eines Teils des hierarchischen Karteikastens. Die aktuelle Karteikarte (= Reiter) ist durch ein Kästchen hervorgehoben. (Dies ist wegen der Verwendung des bereits genannten „DOS-Screen-Prints“ hier nur eine Hilfsdarstellung. In Wirklichkeit bekommt der Benutzer auf dem Bildschirm die aktuelle Karteikarte in inverser Schrift dargestellt. Die Anfangsbuchstaben der Menüaktionen werden fett angegeben.) Die in den Menüzeilen am unteren Rand des Bildschirms angebotenen Aktionen beziehen sich jeweils auf den Reiterbegriff der aktuellen Karteikarte. Der Überbegriff des aktuellen wird durch den Indikator „Bereich:“ gekennzeichnet, dessen Überbegriff, also der Überüberbegriff des aktuellen, durch „Thema:“.

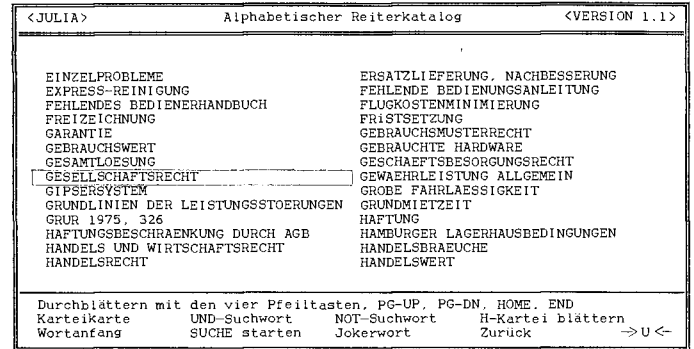
In Fig. 5 ist „MIET-/PACHTRECHT“ der aktuelle Reiterbegriff. Als direkten Überbegriff dazu findet man „COMPUTERVERTRAGSRECHT“, dessen Überbegriff „HANDELS-UND WIRTSCHAFTSRECHT“ ist. Alle anderen sichtbaren Begriffe sind Reiter, die den gleichen Überbegriff wie der aktuelle Begriff besitzen, sich also auf der gleichen Hierarchiestufe wie dieser befinden. In Fig. 5 sind dies z. B. „WERKVERTRAGSRECHT“ oder „GESCHAFTSBESORGUNGSRECHT“. Ein Pfeil rechts neben dem Begriff zeigt an, daß zu diesem Begriff noch Unterbegriffe vorhanden sind.

Möchte man den aktuellen Begriff wechseln, kann man in der hierarchischen Kartei blättern. Die Steuerung geschieht in unserer Programmversion durch Sondertasten, die sich in der Tastatur auf dem sogen. 'Cursor-Tasten-Block' befinden. Ginge man z. B. vom „MIET-/PACHTRECHT“ eine Ebene nach unten, z. B. zu „LEASINGVERTRAG ALLGEMEIN“, so würde als dessen Bereich-Begriff „MIET-/PACHTRECHT“ und als Thema-Begriff „COMPUTERVERTRAGSRECHT“ angezeigt werden.

Die genannten Pfeiltasten werden in der alphabetisch geordneten Kartei ebenfalls zum Blättern benutzt. In dieser Kartei werden alle Reiter, die in JULIA vorhanden sind, alphabetisch sortiert auf dem Bildschirm angezeigt. Da bei einem größeren Wortschatz nicht alle Stichwörter gleichzeitig auf dem Bildschirm erscheinen können, wird nur eine Seite der Liste abgebildet. Ein typischer Bildschirmaufbau beim Arbeiten mit der alphabetischen Kartei ist in Fig. 6 dargestellt.

3.2 Die Textarchivierung

Wie wird bei konventionellen Archiven Text abgelegt? Nach dem Ausschneiden, Kopieren oder Schreiben eines Textes wählt man Beschriftungen für die Reiter auf dem für den Text vorgesehenen Hängeordner aus, steckt sie auf und legt das Dokument in dem Ordner ab. Um die Karteikästen (alphabetisch



Figur 6

oder hierarchisch) stets auf dem neuesten Stand zu halten, müssen die Karteikarten mit den neuen Stichwörtern hinzugefügt bzw. in die alphabetische sowie hierarchische Kartei eingeordnet werden.

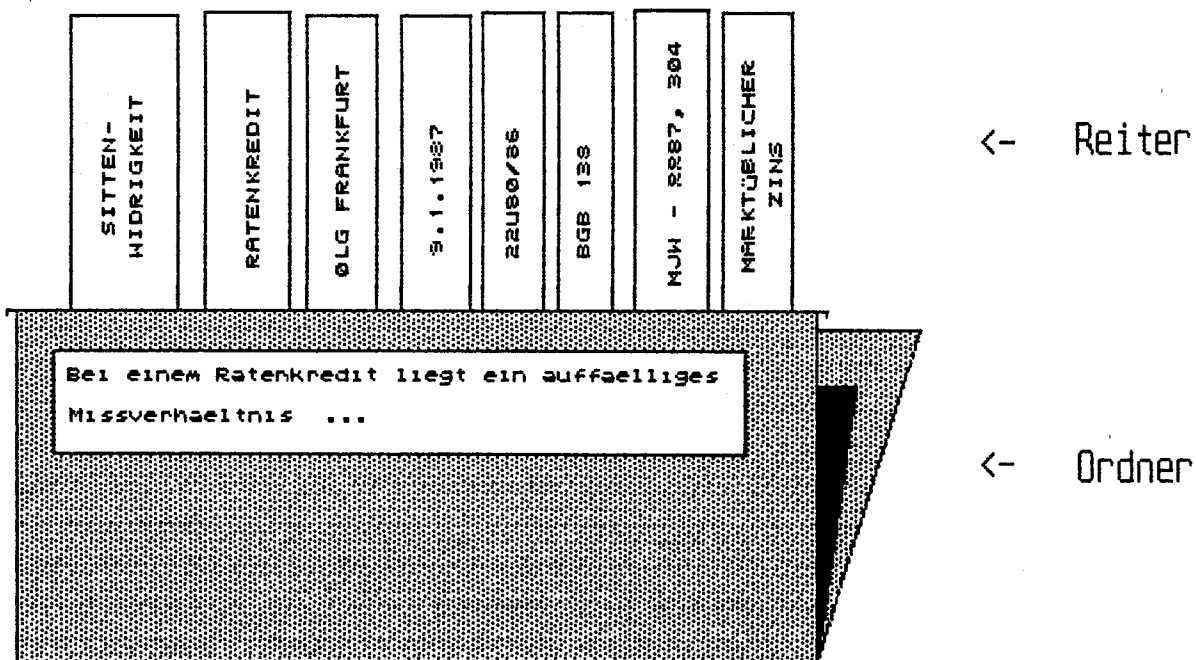
Diese Vorgänge werden durch JULIA genau nachgebildet. Zur Texteingabe stehen dem Benutzer zwei Möglichkeiten zur Verfügung: Zum einen kann er einen Text übernehmen, der bereits mit Hilfe eines (beliebigen) Textverarbeitungsprogramms erstellt wurde; zum anderen hat er die Möglichkeit, mit Hilfe eines Text-Editors neue Texte zu erstellen. Die fachlich anspruchsvolle Aufgabe der Reitervergabe durch den Redakteur kann somit getrennt von der eigentlichen Textablage durchgeführt werden, so daß die Routine-Arbeit auch von einer Schreibkraft übernommen werden kann. Nachdem der Text eingegeben ist, muß der Redakteur mit charakteristischen Stichwörtern den Text beschreiben. Bei der Reitervergabe stehen dem Redakteur sowohl der alphabetische als auch der hierarchische Reiterthesaurus zur Verfügung, aus denen er Reiter per Knopfdruck auswählen und vergeben kann. Werden als Reiter solche Stichwörter „aufgesetzt“, die in keinem Thesaurus vorkommen, werden sie automatisch in das alphabetische Verzeichnis aufgenommen. Der Redakteur kann diese neuen Reiter auch in die hierarchische Kartei an die passenden Positionen einordnen.

Mit der Frage der Texterstellung und der Einordnung der Reiter in die hierarchische Kartei wollen wir uns in diesem Bericht aus Platzgründen nicht beschäftigen, sondern im folgenden die Textarchivierung in der Modellsprache anhand eines Beispiels beschreiben. Stellen wir uns also die Aufgabe, folgenden Leitsatz aus der Neuen Juristischen Wochenschrift (NJW) in unserem Archiv abzulegen:

„Bei einem Ratenkredit liegt ein auffälliges Mißverhältnis zwischen Leistung und Gegenleistung auch dann vor, wenn der vereinbarte Zins den marktüblichen Zins um mehr als 12%-Punkte übersteigt. Zur Berechnung des marktüblichen Zinses bei Krediten von über 60 Monaten Laufzeit. OLG Frankfurt, 9.1.1987, 22 U 80/86, NJW-RR 87, 304“.

Mit welchen Reitern könnten wir diesen Leitsatz beschreiben? Naheliegenderweise wird der Redakteur des Archivs Reiter vergeben, auf denen der Gerichtsort „OLG FRANKFURT“, das Datum „9.1.1987“ und das Aktenzeichen „22 U 80/86“, sowie die Fundstelle „NJW-RR 87, 304“ vermerkt sind. Diese Kombination von Reitern kommt nur auf dem Ordner dieses Leitsatzes vor. Gibt der Benutzer später die entsprechenden Reiter in irgendeiner Kombination (oder nur einige davon)

dem System als Suchschlüssel bei einer Textabfrage gezielt vor, wird als Ergebnis der Suche dieser Ordner gefunden. Um aber den Ordner auch bei einer Textrecherche wiederfinden zu können, bei der diese Reiterbegriffe nicht bekannt sind oder einfach nicht an sie gedacht wurde, muß der Redakteur den Ordner auch mit Reitern bestücken, die auf den Inhalt des Leitsatzes verweisen. Die Reiter können zunächst aus dem Wortlaut des Leitsatzes direkt entnommen werden, wie z.B. „RATENKREDIT“ und „MARKTÜBLICHER ZINS“. Auch ein Reiter mit dem einschlägigen Paragraphen sollte auf dem Ordner unseres Beispieltexes nicht fehlen. Dennoch würde der Ordner dann noch nicht gefunden, wenn dem Benutzer bei einer Textrecherche etwa nur der Begriff „SITTENWIDRIGKEIT“ einfällt. Es ist also Aufgabe des Redakteurs, die Reiter des Ordners auch mit solchen Begriffen zu beschriften, die den Themenkreis des Textes charakterisieren, obwohl sie nicht direkt im Text vorkommen. An dieser Stelle wird die Bedeutung der hierarchischen Kartei besonders deutlich, weil sie dem Redakteur gestattet, sich an die einschlägigen Stellen des Hierarchiebaums zu manövrieren. Nachdem der Redakteur dem System mitgeteilt hat, daß er die Reitervergabe beenden möchte, übernimmt JULIA die restlichen Arbeiten vollständig und ordnet den Text mit seinen Reitern ins Archiv ein. Zusammenfassend ergibt sich jetzt das in Fig. 7 dargestellte Bild unseres Ordners:



3.3 Die Textrecherche

Nun kommen wir zur zweiten zentralen Funktion, der Textsuche. Ihr Ablauf läßt sich in verschiedene Phasen aufteilen, die der Reihe nach (unter Umständen mehrmals) durchlaufen werden. Zunächst wird nach der Auswahl des Hauptmenüpunktes „Wahl der Suchworte“ das „Suchprofil“ aufgebaut. JULIA muß natürlich mitgeteilt werden, nach welchen Reitern auf den Ordnern gesucht werden soll. Die gewünschten Begriffe kann der Benutzer schrittweise und mit Unterstützung der zwei Reiterkarteien eingeben. Ohne vorherige Eingabe eines Ausgangsstichwortes bietet JULIA dem Benutzer die Begriffe der obersten Ebene der Reiterhierarchie zur Auswahl an. Ein gewählter Reiter kann dann in das Suchprofil aufgenommen bzw. von ihm ausgehend kann weiter in der Hierarchie „herumgestiegen“ werden.

Im allgemeinen wird nach denjenigen Ordnern gesucht, auf denen alle im Suchprofil enthaltenen Reiterbegriffe aufgesteckt sind. JULIA erlaubt auch die Suche nach Ordnern, auf dem angegebene Reiter nicht aufgesteckt sind. Eine weitere Variante ist die „Jokersuche“, die es erlaubt, Reiterbegriffe mit „*“ und „?“ zu maskieren. Die Angabe von „OLG*“ wäre äquivalent mit der Suche nach Texten, auf denen Reiter wie „OLG NÜRNBERG“, „OLG HAMM“, usw., also ein beliebiger Anhang zum Präfix OLG aufgesteckt sind.

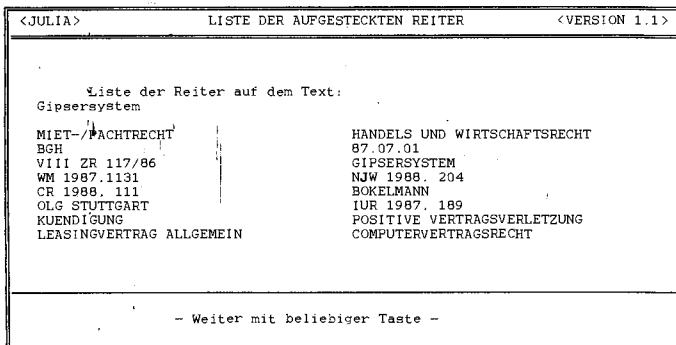
Nach dem Erstellen des Suchprofils kann der Benutzer die Suche starten. Dabei werden – anschaulich gesprochen – die Ordner „abgesenkt“, die den angegebenen Suchkriterien nicht genügen, so daß sie für den Benutzer des Archivs nicht mehr sichtbar sind und die noch oben hängenden Ordner das Ergebnis der Textsuche darstellen. Dem Benutzer wird daraufhin eine Liste der noch oben hängenden Texte angeboten, aus der er sich Texte zur weiteren Bearbeitung auswählen kann.

Gibt der Benutzer etwa manuell oder mit Hilfe der hierarchischen Kartei (s. Fig. 5) das Stichwort „MIET/PACHTRECHT“ in das Suchprofil ein, und veranlaßt er eine Textsuche, so erfährt er von JULIA, daß z.B. 45 Ordner diesen Reiter tragen und somit oben hängen, s. Fig. 8.

<JULIA>		TEXTSUCHE	<VERSION 1.1>
Aktueller Inhalt der Suchwortliste:			
1. MIET-/PACHTRECHT			
Bisheriges Suchergebnis: 45 Ordner			
Suche mit allen Texten	Fortsetzen der Suche	Initialisieren	
Ergebnisausgabe	Joker-Suche	ZURÜCK	
Wahl der Suchworte	Masken-Suche	->z <	

Figur 8

Angenommen, dem Benutzer wären dies noch zu viele Ordner, und er wäre besonders an Urteilen des OLG Stuttgart interessiert, dann würde er „OLG STUTTGART“ als weiteres Stichwort wählen. Von den verbliebenen Ordnern würde auch der Ordner mit dem Kurztitel ‚Gipersystem‘ diese Reiter tragen. Der Benutzer kann sich nun alle Reiter dieses Ordners ansehen, s. Fig. 9, oder den Inhalt des Ordners, sowie das Suchprofil für eine Fortsetzung der Textsuche verändern.



Figur 9

3.4 Verwaltungsfunktionen

Um die Arbeit des Redakteurs zu unterstützen, stellt JULIA weitere Funktionen zur Verwaltung des Textarchivs zur Verfügung. Dazu gehören u.a. das Entfernen nicht mehr benötigter Texte aus dem Archiv, das Verändern der auf einem Textordner aufgesteckten Reiter sowie das Edieren der Texte selbst. Bei der Auswahl eines zu bearbeitenden Textes stehen dem Verwalter die gleichen Mittel wie bei der Textsuche zur Verfügung. Auch die Pflege der Reiterverzeichnisse wird unterstützt, wobei besonderes Gewicht auf die Handhabung des hierarchischen Verzeichnisses gelegt wird. Die Reiterkarteikarten können verändert und nicht mehr benötigte Reiterbegriffe aus dem Verzeichnis gestrichen werden.

Der Redakteur (und natürlich auch der Benutzer) kann sich zu einem Reiter die zugehörige Karteikarte in ihrer gesamten Informationsbreite ausgeben lassen. Dazu gehören evtl. vorhandene Unter- bzw. Oberbegriffe in der Reiterhierarchie sowie Synonyme und Homonyme. Zu einem Reiter kann der Redakteur beliebig viele Synonyme eingeben, damit der Benutzer z. B. bei der Suche nach „KFZ“ automatisch auch die Reiter „PKW“ oder „AUTO“ mit erreicht. Dies gelingt durch eine Rückführung von Synonymen auf einen gemeinsamen „Vorzugsbegriff“. (Näheres in [7])

4 Implementierung

Wie bei vielen größeren Software-Projekten stellte sich auch bei der Entwicklung von JULIA die Frage, ob das System von Grund auf neu und eigenständig programmiert oder ob auf bestehende Software (in unserem Fall auf ein Datenbank-System) zurückgegriffen werden soll. Verschiedene Überlegungen führten uns zur Entscheidung, das Datenbanksystem DBASE als Grundlage für die Implementierung zu verwenden. Gewisse Nachteile in der Effizienz, Portabilität und Gestaltung der Benutzerschnittstelle gegenüber einer maßgeschneiderten Lösung in einer problemorientierten Programmiersprache wie PASCAL oder C nahmen wir in Kauf, da sich durch das Aufsetzen auf einer fertigen Datenbankverwaltung der Programmieraufwand doch erheblich verringerte. So konnten wir in relativ

kurzer Zeit ein erste JULIA-Version erstellen, die sich vom Funktionsumfang durchaus bereits mit angebotenen Textarchiv-Programmen messen kann.

Genaueren Aufschluß über die Implementation ist in Ä7Ü zu finden.

5 Vergleich mit anderen Programmen

Für Rechtsanwälte werden die verschiedensten Computerprogramme zur Literaturrecherche angeboten. Sie lassen sich in zwei Klassen einteilen.

Zur einen gehören die Systeme wie LEXINFORM, JURIS oder ALEXIS, die von Dienstleistungsunternehmen angeboten und verwaltet werden. Der Hauptvorteil einer solchen Datenbank ist die für den Benutzer entfallende Texteingabe und Pflege. Nachteile sind eine wesentlich größere Einarbeitungszeit und ein geringeres Präzisionsmaß von relevanten Texten im Vergleich zu gefundenen. Außerdem hat ein Anwalt keine Möglichkeit, den Datenbestand dieser Auskunftssysteme zu beeinflussen.

Zur zweiten Klasse zählen wir die individuellen Datenbanken wie JULIA. Ein typisches Merkmal ist der auf den Benutzer zugeschnittene Datenbestand. Der Benutzer ist hier oft gleichzeitig Redakteur und selbst verantwortlich für die Auswahl, Eingabe und Pflege des Datenbestandes. In letzter Zeit finden sich in der einschlägigen Literatur immer häufiger Berichte über das in die Klasse der individuellen Datenbanken gehörende Programm BGH-DAT (s. z. B. [3] und [8]), das von Dr. Günther Krohn (Vorsitzender Richter des 4. Zivil-Senats am Bundesgerichtshof) konzipiert und seit 1986 ständig weiterentwickelt wurde. Ähnlich wie JULIA baut BGH-DAT auf einem vorhandenen Datenbanksystem („ADIMENS“) auf. Die Datenbank umfaßt Fundstellen für alle seit 1986 ergangenen Zivilrechtsurteile des BGH. Die Datenbestände werden laufend aktualisiert und haben bereits einen beachtlichen Umfang erreicht.

Eine der Stärken von BGH-DAT ist die Verwendung einer fensterorientierten, mausgesteuerten graphischen Benutzeroberfläche, die vom zugrundeliegenden Datenbanksystem zur Verfügung gestellt wird. Dies trägt sicher zur Akzeptanz des Systems bei, obwohl im Gegensatz zu JULIA keine Auskunftsfunktionen bei Fehlbedienung verwirklicht wurden.

Bei BGH-DAT ist, ebenfalls anders als bei JULIA, die Anzahl der Reiter pro Text beschränkt. Außerdem wurde bei BGH-DAT (wie bei den meisten anderen Systemen auch, s. z.B. TITAN (s. [1]) und JURINFO (s. [10])) das Redakteur-Benutzer-Problem unserer Meinung nach nicht genügend beachtet: Neben einer „Reiter-Abkürzungsdatei“, die in etwa mit der alphabetischen Reiterkartei JULIAS vergleichbar ist, existiert keine Hierarchie unter den Reiterbegriffen. Ebenso wird das mögliche Auftreten von Synonymen oder Homonymen unter den Reitern nicht berücksichtigt. Auch gegenüber BGH-DAT weist sich die klare Modellbildung von JULIA überlegen.

Eine Kombination beider Datenbanksklassen könnte als der Idealzustand bezeichnet werden: Für den „Hausgebrauch“ steht dem Benutzer die individuelle Datenbank zur Verfügung, mit der er schnell und problemlos die gewünschten (Spezial)Texte finden und verarbeiten kann, während für allgemeinere Informationen, die nicht aus seinem Arbeitsgebiet stammen, eine zentrale Datenbank verwendet wird. Wünschens-

wert wäre auch, Informationen aus der zentralen Datenbank, mit selbstgewählten Reitern versehen, in die eigene Datenbank übernehmen zu können (sog. „down-loading“).

In näherer Zukunft jedoch werden die individuellen Datenbanken stark an Bedeutung gewinnen. Beispielsweise wird der Inhalt der JURIS-Datenbank neuerdings auch auf sogenannten CD-ROMs (mit Disketten vergleichbare Speichermedien von sehr hoher Kapazität) zur Verfügung gestellt. Verschiedene juristische Fachverlage (wie schon jetzt der C.H. Beck-Verlag seit dem 2. Quartal 1989) werden die Benutzer von individuellen Datenbanken durch die Bereitstellung von Textmaterial in computergerechter Form unterstützen. Wer diesen Trend rechtzeitig erkennt, dem eröffnen sich große Vorteile gegenüber seinen Mitbewerbern, sowohl auf der Anbieterseite in der Gewinnung von Anteilen auf einem stark expandierenden Markt, als auch dem Datenbank-Benutzer, der sich durch konsequente Nutzung moderner Bürotechnik einen Informationsvorsprung und damit Wettbewerbsvorteile aufbauen kann.

Für Interessenten

Ab September 1989 ist es möglich, eine Demo-Version von JULIA zum Diskettenpreis plus Porto zu beziehen. Wenden Sie sich für weitere Informationen an die Autoren.

Danksagung

Unser Dank gebührt Peter Freud, Nürnberg, Dr. Benno Heussen und Herrn Peter Wahl (beide München) für ihre Hilfe beim Eindringen in die Gedankenwelt von Juristen, und Christel Riedel (Bonn) vom Verein Recht und Information e.V. für die finanzielle Unterstützung.

Rechtsinformatik in Frankreich

Teil 3: Informatik im Gericht – Fortgeschrittene Anwendungen

Louis Barbet

Es hat sich eingebürgert, von „l'informatique judiciaire“ zu sprechen, wenn man die EDV-Anwendungen im Bereich des Justizministeriums unter verschiedenen Gerichtsbarkeiten meint. Tatsächlich handelt es sich um einen Zweig der Verwaltungsinformatik.

Die Ausgangsidee ist einfach zu formulieren. Die Zahl der Prozesse nimmt ständig zu. Jährlich ergehen über 2 Millionen Urteile, wobei die Verurteilungen wegen Ordnungswidrigkeiten in der Größenordnung von 8 Millionen Entscheidungen noch nicht einmal mitgezählt sind. Man kommt so im Ganzen auf circa 10 Millionen gerichtliche Entscheidungen. Die „Cour de Cassation“ war im letzten Jahr mit 26.000 Revisionsverfahren befaßt. Ergänzend ist zu berücksichtigen, daß sich die Prozeßdauer verlängert hat (1986 betrug sie im Durchschnitt 20 Monate für die „Cours d'Appel“ und 12 Monate für die „tribunaux de Grande Instance“).

6. Literaturverzeichnis

- [1] W. Andrae: „TITAN – ein mächtiger Helfer“; Neue Juristische Wochenschrift – Computer Report, 1/1988.
- [2] D. Blair / M.E. Maron: „An Evaluation of Retrieval Effectiveness for a Full-Text Document Retrieval System“; Communications of ACM, Bd. 28, 1985. Vgl. auch Berring, Volltext-Datenbanken und juristische Informationssuche, Informatik und Recht, 1/1987, 2/1987, 3/1987
- [3] K. Birkigt: „BGH-DAT: Offline, leicht und pfiffig“; Neue Juristische Wochenschrift – Computer Report, 1/1988.
- [4] H. Göttler: „Automatisierung einer Rechtsanwaltskanzlei (Konzepte, Methoden, Erfahrungen)“, GI-Jahrestagung 1985, Informatikfachberichte Nr. 108, Springer Verlag, Heidelberg, 1985.
- [5] H. Göttler / H. Schwarz / R. Straubmeier: „Ein Textsystem für eine Rechtsanwaltskanzlei“, GI-Jahrestagung 1985, Informatikfachberichte Nr. 108, Springer Verlag, Heidelberg, 1985.
- [6] H. Göttler: „Die rationelle Rechtsanwaltskanzlei“, DSWR, Verlag C.H. Beck, München, Aug. 1986.
- [7] U. Günzel / B. Himmelreich: „Entwurf und Implementierung des Juristischen Literaturarchivs JULIA“, Studienarbeit, Institut für Mathematische Maschinen und Datenverarbeitung, Univ. Erlangen-Nürnberg, Jan. 1989.
- [8] A. Krämer: „BGH-DAT“, Anwaltsblatt, 8+9/1988.
- [9] P. Mertens / J. Griese: „Industrielle Datenverarbeitung 2 (Informations- Planungs- und Kontrollsysteme“), Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, 5. Aufl., Wiesbaden, 1988
- [10] R. Neugebauer: „jurinfo – ein Dokumentenverwaltungsprogramm für Juristen“, Informatik und Recht, 7+8/1988.

Gleichzeitig stagniert die Zahl der Richter bei 6.000 – das ist dieselbe Zahl wie 1914.

Ansichts dieser Probleme ist die Einführung der EDV, sprechen wir es aus, als eine Art Wunderheilmittel gepriesen worden. Minister Badinter zog 1980 einige Tage vor dem Ausscheiden aus dem Amt die Bilanz seines Programms der Einführung von Informatik bei Gericht und bemerkte, daß „Ende 1988 ... der wesentliche Teil der Gerichtsbarkeit informatisiert“ sein werde. Das Amtsblatt des Ministeriums (Nr. 62, Januar/Februar 1988) sprach jedoch von einer neuen Informatikpolitik in den Gerichtsbarkeiten und kündigte als Projekte die Maßnahmen an, die bereits hätten realisiert sein müssen.

Trotzdem muß man nicht alles Grau in Grau sehen. Es haben Realisierungen stattgefunden. Dabei sind drei Etappen zu unterscheiden.