

Deutscher EDV-Gerichtstag 1995:

Aus den Protokollen (Teil 3)

Arbeitskreise D3 und D4: Anbieterübergreifende Software-Integration I und II

Moderation: *Prof. Dr. Helmut Rießmann*
Referent: *Matthias Kraft (C.H. Beck Verlag)*
Protokoll: *Kay Diedrich*

Den Ausgangspunkt des Arbeitskreises bildeten die Ausführungen des Referenten über die an ein Hypertextdokument zu stellenden Anforderungen und deren Realisierung in einem mehrere Anwendungen wie auch Anbieter integrierenden System. Als mögliche Lösung wurde sodann das Beck-Connectivity System vorgestellt und diskutiert. Als Alternativlösung stellte Herr Schweiger (Bund Deutscher Rechtspfleger, Gieseking Verlag) die Verknüpfung verschiedener Windows-Anwendungen mittels Schnittstellenprogrammierung vor. Ergänzend gab Frau Thierau (Carl Heymanns Verlag) Hinweise zu der elektronische Publikationen betreffenden Entwicklungsarbeit der Verlage (Arbeitskreis "Elektronisches Publizieren") sowie der kurz und mittelfristig zu erwartenden Entwicklungen auf dem Markt für Informationsanbieter.

Ausgangspunkt

Anforderungsprofil eines Hypertextdokuments

Um die Möglichkeiten eines elektronischen Textes auszuspähen, genügt es nicht, einen Text auf dem Bildschirm darzustellen. Vielmehr entwickelt sich der Vorteil elektronisch aufbereiteter Texte erst durch ein hinreichend individualisiertes Gesamtkonzept, welches sich dem Benutzer als einheitliches System präsentiert. Integrierende Funktionen wie Hypertextsprünge, anwendungsübergreifendes Back-Tracing sowie History-Funktionen, Anmerkungsmöglichkeiten, Volltextsuchen und individueller Hypertext schaffen Anreize zur Benutzung des neuen Mediums.

Anreize zur Benutzung des neuen Mediums

Betrachtet man die derzeit am Markt angebotenen Produkte, so stellt sich die Frage, ob der Benutzer einerseits gewonnene Vorteile eines elektronischen Dokuments nicht mit andererseits hinzunehmenden Nachteilen (zu) teuer bezahlt.

Das Preis-Leistungs-Verhältnis der Produkte

Ein Hypertextsystem sollte für folgende Problembereiche mit einer hinreichenden Lösung aufwarten können:

Zugriffsmöglichkeit auf gesamten Datenbestand

Ein Benutzer, der beispielsweise drei elektronische Titel erworben hat, wird wünschen, jederzeit auf Grundlage des Gesamtdatenbestandes aller drei Produkte arbeiten zu können. Hierfür wäre flexibler Hypertext zwischen den Anwendungen erforderlich. Ebenso müßten die Suchfunktionen die Datenbestände aller verfügbaren Titel gleichermaßen berücksichtigen.

Integrale Suche und Verknüpfungen

Individualisierte Produkte – Vermeidung von Doppeldatenbeständen

Ein Produkt sollte so individuell wie möglich auf die Bedürfnisse des Benutzers zugeschnitten sein. Andernfalls muß der Benutzer ein Kompaktprodukt erwerben – und vor allem auch bezahlen –, welches er nur zu einem geringen Teil nutzt. Der Unmut über diese Tatsache wird sich noch verstärken, wenn mehrere Kompaktprodukte notwendig sind, um auf die für ihn notwendigen Informationen zugreifen zu können. Da es bei mehreren Kompaktprodukten, insbesondere verschiedener Anbieter, fast unvermeidbar zu erheblichen Überschneidungen kommt, bezahlt der Kunde seine Informationen doppelt.

Individualprodukt vs. Kompaktprodukt

Daran anschließend sollte dem Benutzer die Möglichkeit offenstehen, nur diejenigen Daten zu aktualisieren, bei denen es sein persönlicher Bedarf erfordert.

Frau Thierau (Carl Heymanns Verlag) wies darauf hin, daß zukünftig die "Informationsvermeidung" im Sinne des Abbaus redundanter Informationen in immer stärkerem Maße eine Rolle spielen werde. Es sei damit zu rechnen, daß fortschreitend Information nicht in gesammelter Form vorgehalten werde. Vielmehr werden Benutzer zunehmend dazu übergehen, Daten nach Bedarf punktuell über On-Line-Dienste zu beziehen. Schon jetzt sollte die Entwicklung auf weitestgehende Vermeidung von Doppelinformationen abzielen.

*Herausforderung in der Zukunft:
"Informationsvermeidung"*



*Hinterlegung individueller
"Informationsobjekte"*

Erhaltung und Verfügbarkeit von Anmerkungen

Häufig liegt es im Interesse des Benutzers, einem elektronischen Dokument individuelle Zusatzinformation beizufügen. Die Software muß dem Benutzer bei Quantität und Form der Anmerkungen größtmögliche Freiheiten einräumen. Sinnvoll erscheint die Möglichkeit, im größeren Umfang Text abzuspeichern. Ebenso sollte der Benutzer Graphiken einbinden können.

Notwendige Voraussetzung einer sinnvollen Nutzung dieser erweiterten Funktion ist aber auch, daß dem Benutzer einmal erstellte Anmerkungen, sowohl während des Zugriffs auf vergleichbare Daten aus anderer Quelle, als auch nach einer Aktualisierung eines Hypertexttitels, erhalten bleiben.

Aktuell, preiswert oder schnell?

Benutzerbestimmter Zugriff auf Datenalternativen

Im Zuge des Ausbaus von Datennetzen wird der Benutzer vermehrt vor die Frage gestellt werden, aus welcher Datenquelle er seine Information schöpfen möchte. Die Entscheidung wird von seinen Bedürfnissen im Einzelfall abhängen. Kommt es auf Aktualität an, wird er auf On-Line Dienste (beispielsweise juris) zurückgreifen. Sollen Daten möglichst preiswert gesammelt werden, wird er das kostenfreie Angebot im Internet (insbesondere html-Dokumente) nutzen wollen. Möglichst schnellen Zugriff werden lokal vorhandene Datenquellen am besten leisten können. Derartige Auswahlmöglichkeiten sollte ein Hypertextsystem berücksichtigen können. Sinnvoll erscheint sowohl die nach Präferenzen automatisierte Auswahl durch das System oder eine im Zeitpunkt der Informationssuche dem Benutzer ermöglichte Auswahl aus den vorhandenen Alternativen.

Verweis-Schwierigkeiten

Gesicherte Verweisintegrität für den Entwickler

Der Entwickler einer Anwendung muß sich darauf verlassen können, daß ein von ihm angelegter Verweis dem Benutzer in dessen Individualsystem die richtige Information zeigt. Probleme sind zum Beispiel beim Verweis auf Vorschriften denkbar, die nach Auslieferung des Hypertexttitels geändert werden. Verwirrung, die entsteht, wenn der Titel sinngemäß auf den alten Wortlaut einer Vorschrift verweist, dem Benutzer aber der neue Wortlaut der Vorschrift gezeigt wird, sind auszuschließen. Entweder muß der Benutzer über die Rechtsänderung informiert werden, oder der Hypertextverweis muß ins Leere gehen. Steht die Norm noch in alter Fassung zur Verfügung, müßte diese abgefragt werden können.

Lösungsmöglichkeit:

DDE und OLE

Windows-Schnittstellen Programmierung

Die von Herrn Schweiger (Bund Deutscher Rechtspfleger, Giesecking Verlag) vorgestellte Lösung beruht auf der Verbindung verschiedener Windows-Anwendungen über DDE, Funktionsaufrufe und OLE. Dem Benutzer wird die Suche nach Begriffen auf Grundlage der in das System eingebundenen Datenquellen ermöglicht. Die Suche erfolgt anhand eines Begriffs, der sich zur Zeit des Suchaufrufs in der Windows-Markierung befindet.

Die Umsetzung

Die Suche wird durch Nutzung der anwendungs- bzw. betriebssystemimmanenten Programmiermöglichkeiten erreicht. Soweit die Anwendungsschnittstellen offengelegt sind, kann eine Anwendung anhand individuellen Codes in den funktionalen Verband der Anwendungen "einprogrammiert" werden.

Da dieses Konzept bezogen auf sein Leistungsangebot und dessen Erreichung eine andere Zielvorstellung verfolgt, wurde vorgeschlagen, die Diskussion für diesen Arbeitskreis auf das System der Beck-Connectivity zu konzentrieren.

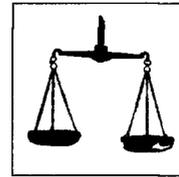
Lösungsmöglichkeit:

*Standardisiertes Format zur
Definition von Hypertextstellen*

System der Beck-Connectivity

Durch standardisierte Formate zur Definition von Hypertextstellen wird jedem Autor ermöglicht, im Rahmen der verwendeten Konvention "blind" fremde, auch erst zukünftig zu erstellende, Dokumente einzubeziehen. Entsprechend stellt das von ihm vorbereitete elektronische Dokument seine Dienstleistungen anderen Anwendungen zur Verfügung. Die Zitierform orientiert sich an juristischer Zitiertechnik, wobei die Normierung der juristischen Bezeichnungen – soweit möglich – an die üblicherweise gebrauchten Abkürzungen angelehnt ist.

Einigkeit bestand über die mittel- bis langfristige Notwendigkeit, einheitliche Normen zur Abkürzung von Gesetzesbezeichnungen unter Mitwirkung der betroffenen Anbieter zu erarbeiten. Der Zeitraum, bis dieser enorm aufwendige Vorgang abgeschlossen sei, müsse aber mit Behelfslösungen überbrückt werden (alias-Tabellen).



Im Gegensatz zu starrem Hypertext, in dem der Autor die Ansprungstelle im Code konkret bezeichnet, läuft eine Hypertextabfrage im Rahmen der Beck-Connectivity zweistufig ab.

Es wird ein anwendungsspezifischer Treiber benötigt, den der Autor des Hypertexttitels erstellt. Dieser Treiber übermittelt seine Daten an einen dem gesamten Windows-System zur Verfügung stehenden Programmteil, welcher die Verbindung zu sonst vorhandenen Titeln koordiniert, die sich bei ihrer Installation entsprechend angemeldet haben. Es wird geprüft, ob eine konkrete Information zur Verfügung steht. Können die gewünschten Daten von mehreren Datenquellen geliefert werden, trifft dieser Programmteil eine Kollisionsentscheidung zugunsten eines Titels.

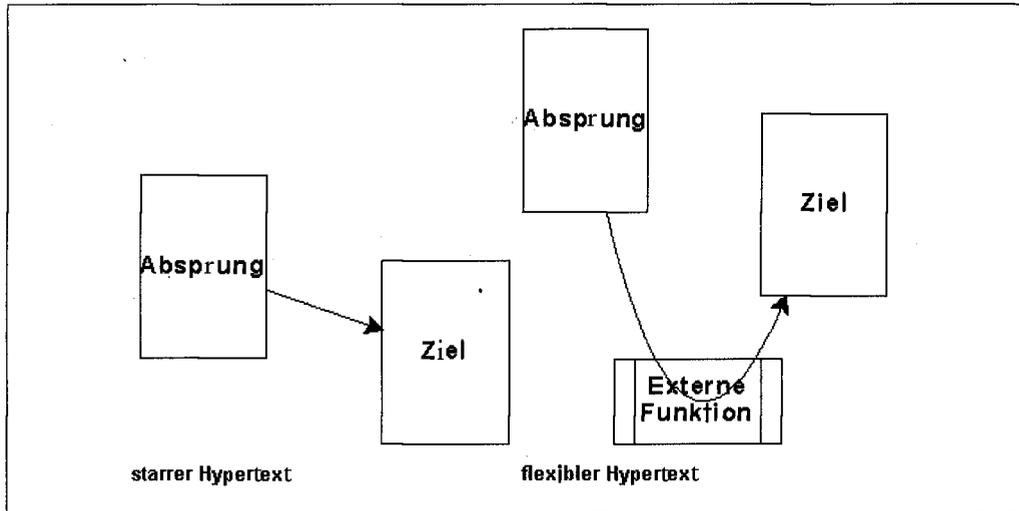


Abb. 1:
Hypertextarten

Da die Beck-Connectivity als offenes System programmiert wurde, können, soweit Bedarf besteht, Kollisionsentscheidungen zukünftig auch von einer breiteren Datenbasis – beispielsweise zusätzlich dem Datum oder der Auflage eines Dokuments auf das verwiesen werden soll – abhängig gemacht werden.

Zur technischen Umsetzung sei hier der einschlägige Passus aus der Tagungsbroschüre zitiert:

“Technisches Konzept

Zur Realisierung einer möglichst flexiblen Architektur wurden zwei Schichten eingerichtet:

- Der Connectivity-Kernel ist unabhängig vom Zielsystem.
- Einzelne Treiber setzen die Funktionalität auf das Zielsystem um.

Der Kernel ist für die Connectivity-Dienste verantwortlich. Er wird zentral entwickelt und gepflegt. Die Treiber werden von den Entwicklern der einzelnen Retrievalsoftware erstellt. Sie können so allgemein gehalten sein, daß sie für eine Vielzahl von Titeln anwendbar sind. Es ist jedoch auch eine individuelle Anpassung an spezifische Titel denkbar ...

1. Die Anwendung verwendet einen Funktionsaufruf des Kernels (z. B. BcExtern), um diesem die Beschreibung des angeforderten Zieldokuments zu übermitteln.
2. Der Kernel ermittelt den Titel, der das Dokument (vermutlich) enthält.
3. Der Kernel lädt den für den Zieltitel registrierten Treiber. Er übermittelt über die einheitliche Einstiegsfunktion (Entry Point Function) des Treibers an diesen die Dokumentbeschreibung einschließlich der Angabe des Zieltitels.
4. Der Treiber verwendet die optimalen Funktionen des Zielsystems, um das Dokument zu finden und anzuzeigen. (Hierzu können je nach Software API-Funktionen, DDE oder auch Windows-Botschaften dienen.)"

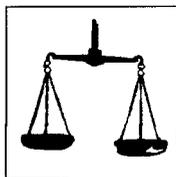
(Aus der Tagungsbroschüre zum 4. Deutschen EDV-Gerichtstag, Seite 45/46)

Geplant ist die Einbindung von OLE und Hypertextverknüpfungen in Anmerkungen sowie deren Durchsuchbarkeit. Auch die derzeitige Beschränkung des Anmerkungsumfangs auf maximal 64 kByte pro Domäne soll beseitigt werden.

Insgesamt stießen Idee und bisherige Entwicklung auf positive Resonanz. Kritik wurde mit Blick auf die beabsichtigte Lizenzierung des Beck-Connectivity-Systems angemeldet.

Offenes System

Entwicklungsziele



*Spracherkennungssystem SP
6000 von Philips*

Arbeitskreis "Spracherkennung"

Vorsitz und Protokoll: *Dr. Wolfgang Taichert, Richter am Bundespatentgericht*

Auch in diesem Jahr gab es wieder einen Arbeitskreis Spracherkennung. Der EDV-Gerichtstag hat sich mit diesem Thema – das erkennbar für die Arbeitswelt der Justiz an Bedeutung zunimmt – seit seiner Gründung befaßt. In diesem Jahr konnte die Fa. Philips zur Präsentation ihres Spracherkennungssystems SP 6000 gewonnen werden. Dieses stellt insoweit eine interessante Weiterentwicklung dar, als damit zum einen die Eingabe von Sprache ohne Kunstpausen zwischen den einzelnen Worten möglich ist und zum anderen auch die Einbindung in eine vernetzte Windows-Umgebung gut gelungen scheint. Leider gibt es derzeit dafür noch keinen juristischen Wortschatz, als erstes wurde vielmehr – wieder einmal – die Radiologie als Anwendung gewählt, so daß die Demonstration entsprechend fachfremd ausfallen mußte. Da nicht der Inhalt, sondern die Leistungsfähigkeit des Systems gezeigt werden sollte, erschien dies vertretbar. Im zweiten Teil des Seminars war dann Gelegenheit gegeben, Ergebnisse zum und Erfahrungen mit dem Umgang von im Einsatz befindlichen spracherkennenden Systemen zu erfahren. Hier hat H. Weihermüller von der GMD Untersuchungen zur Spracherkennung in der Justiz allgemein vorgetragen und Frau Hinkers, Richterin am Oberlandesgericht Düsseldorf, über eigene Erfahrungen mit dem Spracherkennungssystem Dragon-Dictate (vgl. hierzu auch Widmaier-Müller, jur-pc 8/94, S. 2780 ff.) berichtet.

*Das technische Prinzip der
Spracherkennung*

Die Umsetzung eines gesprochenen Textes in codierte Zeichen erfordert erheblich größeren Aufwand als etwa der umgekehrte, als "text to speech" bezeichnete Vorgang, also die Umsetzung der ASCII-Zeichen in – mehr oder weniger – deutliche Laute. Letzteres erfordert "nur" eine Nachbildung des menschlichen Stimmapparates, die Buchstaben werden in Lautelemente (Phoneme) umgesetzt, diesen wiederum bestimmte, empirisch zu ermittelnde Frequenzbänder zugeordnet. Dies gelingt in einfachen Fällen schon auf der Basis spezieller Mikroprozessoren, Beispiele hierfür sind Ihnen sicherlich aus der Verbrauchselektronik bekannt (Wecker, ...). Daß die Spracherkennung erheblich mehr Aufwand erfordert, erschließt sich unmittelbar aus dem ersichtlichen Aufwand; es werden UNIX-Risc-Rechner oder leistungsstarke PCs (ab 386er Prozessoren) unter Einbeziehung spezieller Erweiterungskarten eingesetzt. Dies liegt zum Teil am Umfang des aufzunehmenden Wortschatzes, zum Teil auch daran, daß bei der Sprachwiedergabe die Fähigkeiten des menschlichen Gehirns genutzt werden können, während diese bei der Spracherkennung nachgebildet werden müssen, was umfangreiche und rechenintensive Algorithmen erfordert. Wir – d. h. unser Gehirn – erkennen automatisch, daß eine bestimmte Lautfolge – nur minimal variiert – je nach ihrem Umfeld eine andere Bedeutung hat – beispielsweise Reeder/Räder, Haus/aus, Band/bahnt, Hund/und, fast/faßt ... usw. – die wir selbst bei individueller Aussprache ohne weiteres aufgrund des grammatikalischen und verbalen Umfeldes (Subjekt/Prädikat) identifizieren. Bei der Spracherkennung müssen diese Verbindungen und Mechanismen in schnelle Algorithmen umgesetzt werden, denn es soll ja nahezu in Echtzeit der Text erfaßt werden.

*"template matching"
kombiniert mit
Laut-/Wortstatistik*

Nach dem üblichen, auf einer Kombination von Mustervergleich ("template matching") und Laut- bzw. Wortstatistik beruhenden Verfahren wird dabei das analoge Sprachsignal zunächst digitalisiert und das sog. digitale Spektrum in Bruchstücke zerlegt. Alle 10-15 Millisekunden greift der Rechner ein Signal von 20 bis 30 Millisekunden heraus, so daß sich die Signale überlappen.

*Ermittlung akustischer Vektoren
und Zuordnung von Phonemen*

Danach folgt eine Fourier-Transformation des Signals; es werden die Lautstärken (also die Intensität bzw. Energie des Signals) in Abhängigkeit von einem Frequenzbereich über das erfaßte Frequenzspektrum festgehalten. Aus diesen Energiewerten in bestimmten Frequenzbereichen gewinnt man eine Folge sog. akustischer Merkmale oder auch akustischer Vektoren. Diesen werden auf der Sprachseite die Lautelemente, also die Phoneme zugeordnet. Diese Phoneme gelten den Sprachwissenschaftlern als kleinste Einheit eines gesprochenen Wortes und entsprechen weitgehend den Lautschrift-Zeichen. Ein deutscher Sprecher verwendet ca. 40-50 Phoneme. Die auf der Basis der akustischen Vektoren zugeordneten Phoneme werden mit einer im Sprechertraining aufgebauten, sprecher-spezifischen Phonem-Bibliothek verglichen und entsprechend zugeordnet. Sie sind für den Rechner nicht für sich allein, sondern nur aus dem Zusammenhang identifizierbar, wobei mit sprachspezifischen Wahrscheinlichkeiten auf der Basis der sog. Hidden-Markow-Methode gearbeitet wird. Beginnend mit dem ersten Phonem, das natürlich auch mehreren Buchstaben entsprechen kann, geht das Spracherkennungssystem alle Wörter durch, die damit beginnen und arbeitet sich zu den folgenden Phonemen vor. Das Programm enthält dabei Angaben

zur Wahrscheinlichkeit für die Abfolge bestimmter akustischer Vektoren innerhalb eines Phonems und bestimmter Phoneme innerhalb eines Wortes.

Zur Unterstützung der Eindeutigkeit der Zuordnung und zur Erhöhung der Verarbeitungsgeschwindigkeit muß zusätzlich auch noch ein Sprachmodell hinterlegt werden, bei dem die Wahrscheinlichkeit der Aufeinanderfolge von bestimmten Worten in einem Satz berücksichtigt wird.

Die Spracherkennung gelingt also zum einen durch die wahrscheinlichkeits-orientierte Zuordnung akustischer Vektoren zu einem Phonem, der Phoneme zu Worten und weiter durch Festlegung von Korrelationen zwischen den einzelnen Worten, d. h. von Wahrscheinlichkeiten, mit der ein bestimmtes Wort im Umfeld eines anderen Wortes auftritt. IBM arbeitet hier mit einer Wahrscheinlichkeit von 3 aufeinanderfolgenden Worten (Trigram-Statistik), Dragon mit der von zwei aufeinanderfolgenden Worten (Bigram-Statistik), Philips in der deutschen Version des Systems mit einer Bigram-Statistik, in der englischen Version mit einer Trigram-Statistik.

Die gegebene Beschränkung im Wortschatz und die Notwendigkeit eines Sprachmodells bedingen eine gewisse Optimierung und Spezialisierung in der Auswahl der Gebiete, bei denen Texte über Spracherkennung erschlossen werden sollen. Es müssen also für jedes Arbeitsgebiet mit entsprechender Anwendung spezialisierte Sprachkorpora geschaffen werden, auf die das System während des Prozesses permanent Zugriff hat.

Die Systeme, die den Erfahrungsberichten im Arbeitskreis zugrunde liegen, verfahren nach diesem System. Der Unterschied zwischen den einzelnen Systemen besteht darin, daß beim System Philips der Erkennungsvorgang nach der Digitalisierung der Signale im Hintergrund auf der Basis des gesamten, innerhalb einer Datei zusammengefaßten, gesprochenen und digitalisierten Textes stattfindet, während beim System nach IBM bzw. auch Dragon jedes Wort (fast) unmittelbar nach dem Sprechen auf dem Bildschirm dargestellt wird. Der Preis für diese Echtzeit-Verarbeitung ist eben, daß der mit dem System nach IBM oder Dragon arbeitende Sprecher eben – zur Unterstützung des Algorithmus – eben minimale Kunstpausen zwischen den Worten machen muß. Bei diesen Systemen, wo ja die Wortgrenzen als zusätzliche Information herangezogen werden, vereinfacht sich der Rechenaufwand, was sich auch in der Leistung des Systems (Echtzeit-Verarbeitung gegenüber Batchbetrieb beim Philips-System) auswirkt.

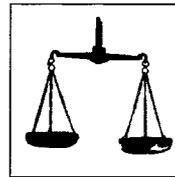
Welchem System dabei der Vorzug zu geben ist – also ob unmittelbare Anzeige des gesprochenen Textes auf dem Bildschirm oder erst nach Abschluß des gesamten gesprochenen Diktates – muß der Interessent in Abwägung seiner spezifischen Anforderungen und Präferenzen selbst entscheiden.

Da man in Echtzeit – oder auch in kurzer Zeit – nur eine bestimmte Menge von Information verarbeiten und vorhalten kann, ergibt sich auch eine Beschränkung für den zur Verarbeitung aktiv hinterlegten Wortschatz. Beim Philips-System wird ein Wortschatz von ca. 64.000 Wortformen, d. h. ca. 2.000 verschiedene Worte unter Einbeziehung ihrer Flexions- und konjugationsbedingten Abwandlungen hinterlegt.

Diese Sprachkorpora werden beim System Philips – wie auch bei IBM – anhand von Texten, die der Anwender liefert, maschinell erstellt, wobei jeweils mehrere tausend Wörter individuell hinzugefügt werden können.

Im Unterschied dazu muß beim System Dragon der Nutzer selbst die Anpassung an sein Wortschatz-Profil leisten. Aus einem allgemeiner gespeicherten Wortschatz von 80.000 Worten in der DOS- bzw. 120.000 Worten in der Windows-Version können jeweils bis zu 30.000 bzw. 60.000 aktiv im Erkennungsprozeß geladen und individuell spezifiziert werden. Wird ein Wort nicht erkannt, schlägt das System ähnliche Worte vor. Ist das Wort auch darunter nicht vorhanden, kann es in das System als neues Wort eingetippt oder einbuchstabiert werden. Bei Erreichen der Grenze des aktiven Speichers wird das am wenigsten benutzte Wort "ausgelagert". Die Anpassung an den individuellen Wortschatz erfolgt also interaktiv durch den Benutzer selbst.

Unabhängig davon erscheint es sinnvoll, ein Spracherkennungs-System nicht nur vor Ort, sondern auch in einem sog. "remote"-Betrieb bedienen zu können. Dies ermöglicht die Nutzung der auf der Anwendung von Diktiergeräten ausgerichteten Organisation des Schreibdienstes. Die Überführung in vorhandene Strukturen in der juristische Organisation erscheint für einen erfolgreichen Einsatz unabdingbar. Nach dem oben Gesagten könnte das beim System Philips zugrundeliegende Konzept (nachträgliche "offline"-Erkennung unter Zugrundelegung eines digitalisierten Sprachfiles) dem bisherigen Ablauf eines Diktates in der Büro-Organisation entgegenkommen. Entsprechende Anwendungen bei den anderen genannten Systemen erscheinen prinzipiell nicht ausgeschlossen, da das digitalisierte Sprachsignal ja vorliegt.



Sprachmodell

Bigram- und Trigram-Statistik

Arbeitsgebietsspezifische Sprachkorpora

*Die Unterschiede der Systeme:
Echtzeit-Verarbeitung vs.
Batch-Betrieb*



Die Systemvoraussetzungen der verschiedenen Lösungen

Die Studie der GMD

Fehlerquote: 5-10 %

Die Voraussetzungen zur Installation eines Systems sind ähnlich: PC mit mindestens 486er Prozessor, 12-16 MB RAM, 50 bis 100 MB (je nach der Anzahl der Nutzer und dem Umfang des hinterlegten Sprachmodells) Speicher auf der Festplatte. Die Demonstration der Philips-Spracherkennung auf dem EDV-Gerichtstag 1995 durch Herrn van der Linde entsprach den Ankündigungen. Ein medizinischer Befund wurde vollständig und flüssig, d. h. ohne Kunstpausen, diktiert, danach im Hintergrund verarbeitet und nach einer Verarbeitungszeit, die in etwa der Diktierzeit entsprach, auf dem Bildschirm zur Korrektur dargestellt. Parallel dazu konnte der diktierte Text abgehört werden (mit Steuerung der Geschwindigkeit, wie bei einem Diktiersystem), so daß der Text entweder durch den Diktierenden selbst oder durch eine Schreibkraft mit Hilfe des Gehörten korrigiert werden kann. Der verbesserte Text konnte sofort in eine Windows-Textverarbeitung übernommen werden.

Die GMD (H. Weihermüller und H. Blankenburg) hat schon im vergangenen Jahr unter Zugrundelegung von anonymisierten Urteilen aus dem Bereich der Verwaltungsjustiz des Landes Nordrhein-Westfalen untersucht, inwieweit sich die in den Spracherkennungssystemen zur Verfügung stehenden Wortmengen den praktischen Anforderungen der Justiz, speziell der Verwaltungsjustiz genügen. Nach diesen Untersuchungen kann mit ca. 20.000 Wortformen tatsächlich eine Abdeckung von 90 bis 95% erreicht werden. Aus dem Verlauf der ermittelten Abhängigkeit ergibt sich, daß eine 100prozentige Abdeckung nicht zu erreichen ist. Gegen eine Anreicherung des Wortschatzes wesentlich über den angegebenen Umfang hinaus spricht – nach den Untersuchungen von H. Weihermüller und Herrn Blankenburg nicht nur die anwachsende Verarbeitungszeit, sondern auch eine zunehmende Verwechslung ähnlich klingender Worte.

Wie sich aus den Ausführungen erkennen läßt, ist der Vorgang der Spracherkennung mit einer gewissen Fehlerquote (5-10 %) behaftet. Daß trotzdem gerade im Bereich der Justiz ein Bedürfnis nach derartigen Systemen besteht, ergibt sich unmittelbar aus der in der täglichen Arbeit zu erstellenden Menge von Dokumenten. Die Automatisierung der Texterstellung wird auch dadurch unterstützt, daß im Bereich der Schreibkanzleien ausreichend Kräfte in manchen Regionen nicht mehr bereitgestellt werden können. Schätzungen der Fa. Philips haben ergeben, daß durch den Einsatz dieses Spracherkennungssystems, das ja auf der Eingabeseite ein reines Diktiersystem ist, die Anzahl der Schreibkräfte in etwa halbiert werden kann. Information am Rande: Das IBM-System wird mittlerweile auch als PC-Version unter OS/2 angeboten, Preis ca. DM 2.500,-, wobei die Kosten für die Aufbereitung des anwenderspezifischen Wortschatzes nicht eingeschlossen sind (IBM: Ca. DM 3.000,- für den allgemein juristischen Wortschatz). Eine Version unter DOS/Windows ist nach letzten Informationen in Vorbereitung.

Auch das System Dragon-Dictate – bisher unter DOS – wird jetzt in einer Windows-Version mit erweitertem Wortschatz ausgeliefert, Preis des Gesamtsystems ca. DM 4.500,-.