1

For my US-friends, the *hardhats*
in admiration of VistA and their achievements

Principles of an *active Electronic Health Record (aEHR)* Lessons Learned from BAIK

Wolfgang Giere

Version 0.3
Nov, 17, 2012

*I am grateful to my daughter Katharina Nimmo in Inverness, Scotland, for her valuable help to revise my English.

Dedicated gratefully
to my wife, children and grandchildren
especially to my staff, friends and teachers
last but not least to the MUMPS community
in short to all who supported me on the long way

Contents

List of Figures

1 Abstract

Computer verändern die Medizin

(Manfred W. Gall, 1971 [2])

VistA is an outstanding electronic Health Record and patient management system, unparalleled and exemplary. It was developed in a decentralized, evolutionary effort together with or rather by the users (thanks to its outstanding database and development environment FileMan). It is widely used, not only in the Veterans Administration where it originated.

BAIK is an (older) German electronic medical record system which has some unique features. BAIKweb is the most recent attempt to reach the goal of an active Electronic Health Record (aEHR). It is a prototype, but some of its features could be desirable as future additions to VistA.

This publication is geared towards the VistA developers and responsible decision makers. It describes the results of more than forty years of development and modelling of medical documentation and linguistics, information processing, classification, retrieval, selection, and presentation.

It contains many details and examples that were used in teaching. I will try to make the essentials clear, to reevaluate them in the light of modern technology and the example of VistA, and to concentrate the *lessons learned* in chapter 7 *VistA compared to BAIK* at the end. The chapters cover the development of BAIK from my first programming attempt 1966 until 2003, when I retired as professor emeritus. My work was guided by the models which are presented here, the Zipf Model and especially by my BAIK information model[1].

The very last chapter, 8 *Dream: WorldVistA + BAIK features* tries to outline further development of VistA using the lessons learned from BAIK and some features like e.g. Thesaurus-use for content-analysis, techniques to guide searches in the web and to filter the results to help the physician with appropriate support in the moment of decision etc. Thus VistA would become an *active Electronic Health Record (aEHR)*, giving the user much more[2] than he invested using the international community and cutting edge technology.

[1] This *Giere Model* has become canonical in Medical Informatics[3, pp 569 and 579].

[2] MuchMore was the working title of a transatlantic research cooperation financed by the US-National Science Foundation (NSF) and the European Union (EU) based on my BAIK information model[4]

2 Introduction

Often I was asked why – as a physician – I started to work with computers. I cannot answer that question. But I know for sure that very early, already in 1957 as a first year student of medicine, I experienced the need for better information for the physician in the moment of decision. Had the doctor who maltreated me after a traffic accident used an *active Electronic Health Record*, I would not have spent many months (September to April) helpless in bed, lying flat on my back and missing a whole year of studies. The physician responsible for that disaster was – at least nominally – a specialist for both surgery and orthopedics. And some years later I had discussions with my *Doktorvater* (supervisor of my MD-thesis) about the use of computers in medicine to overcome the rapid growth of medical knowledge[1] (see fig. 2.1)

In 1968 he phoned me and asked whether I was still engaged in computer work. By then I had already brought *programmed reporting* into daily routine. So the answer was yes. And he told me of a position in Stuttgart at the well known *Robert-Bosch-Hospital* for a physician who knew how to program an IBM 360/30 computer. I got the job and switched to full time programming. In 1969 I got the responsibilty for the data processing center and organisation of the German Clinic for Diagnostic in Wiesbaden. From there I was called to the J.W.Goethe University in Frankfurt in 1976. During all that time I continued with my attempts to bring better information to the physician. I always knew that this required the computer to *understand* what was documented in the medical record. Only this knowledge would enable the computer to help the physician with his decision problem. The data of the patient has to be known and understood to search for relevant information for the actual decision. But how to get valid electronic data about the patient from the physician? The handwritten records were no help . . .

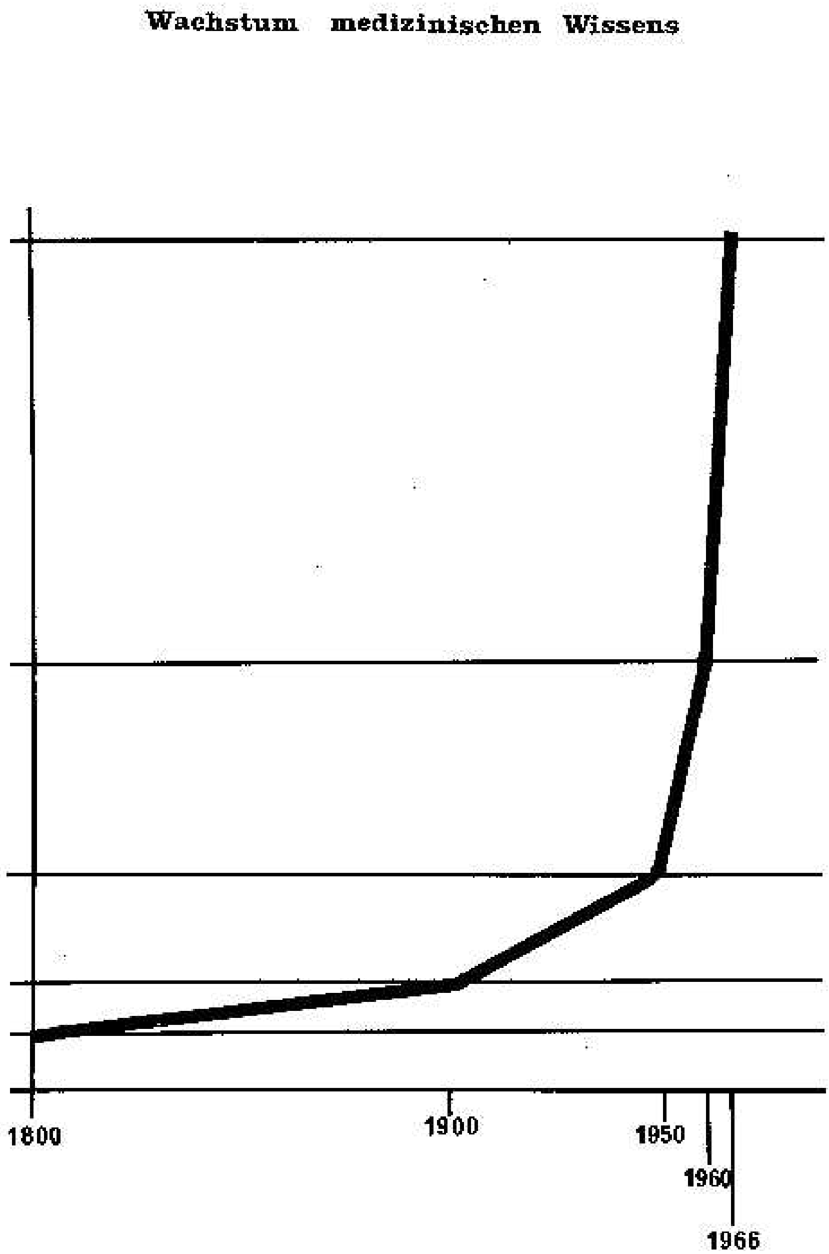

Figure 2.1: Medical knowledge explosion until 1966

3 Practizing Physician and Programmer: Birth of a system

Early in 1967 my new boss hired me to see patients and to organize the new clinic for nuclear medicine[1] (the first one in Germany outside the universities) including documentation. When I presented him with documentation sheets for punched cards – I have been very poud of them – he asked me what we needed (a key punch machine and a sorter) and who should do it: The physician had to fill in the forms and the secretary to keypunch them. Additional work for both, not to speak of the costs. He threw me out. The secretary saw my sad face and asked if it could not be done otherwise: She had realized that we were always dictating the same phrases in the same order with only a few variations. She triggered the idea of computerizing it. Without her, *Frau Wenz*, I would not have dared to engage on programmed reporting.

A typewriter with a paper tape punch unit was borrowed. The data processing center of the city of Duisburg was interested. It worked with an IBM 360/30 computer. The system programmer *Horst Baumann* helped me and taught me /360-Assembly language. January 1, 1968 the system went into daily routine[5]. And I had learned the first lesson: If you wish to get good data from the physician you have to relieve him of work, you have to reward him. The system must save time. It did: The doctor needed much less time, the secretary had to type much less and the reports were printed daily by the computer. Since there were no restrictions to add free text wherever needed, without any restrictions for the physician to express himself, the programmed reporting was well received.

The system had positive side effects:

- The data was severely controlled for formal correctness and plausibility. Only if all entries were correct, the letter came out ready to be signed. If not, a huge error message would show up in the middle of the text. (The data had to be corrected, the letter rerun.)
- The production of the letters immediately followed the end of the daily routine. The analysis of the test results and the nuklid scan (scintigram) was entered into the forms, it was typed and the paper punch tape was transported to the data processing center (via *bicycle line* connection). The next day when the patient came back to learn the results, he could take the printed report with him. It was

[1] Abteilung für Nuklearmedizin am Ev. Krankenhaus Bethesda in Duisburg, director Dr.med. H.A.E. Schmidt

checked and signed in his presence. (Compared to the normal waiting times for referral letters this was sensational.)

- The content of the computer letters was well structured and allowed the receiving physician rapid orientation. Consequently they liked it, too, inspite of the capitals only script of the so called rapid line printers of that period.
- All data was stored electronically and could be analyzed.

The press was enthusiastic about the innovation[2].

The program for this *Arztbrief* (referral letter) was huge, monolithic, written in IBM /360 Assembly language, several thousand commands, one punched card for each one[3]. However, it was well structured and had already subroutines for different data types. What is nowadays called *Natural Language Generation (NLG)* required to work with non fixed length variables throughout. That was unusual in those days.

3.1 Reprogramming and Generalization

When I started to work in Stuttgart[4] later that year I had to learn the Fortran language but continued to use an IBM /360 computer. My job was, to rewrite and enhance the MEDIUC automated diagnosis program for intoxications[6]. I did it successfully and learned a lot, including teamwork.

However, during those days I continued my electronic medical record work and generalized the principles of programmed reporting for both, data input and text output, DUSP[7][5] and DUTAP[8][6]. Both became modular assembly programs, the basis for the DKD-system[7]. I redesigned the text generating and invented the language DUTAP. My resident Assembly kernel could interpret DUTAP statements fast and without prior linking! The DUTAP natural language generating language remained the same over the years[8]. It understood the ternary logic[9] for branching, could interpret the different data types and was recursive using its own stack.

[2]See picture 3.1 on page 7

[3]The compile and link run took a long time on the IBM 360/30. When we started the routine Jan. 1st 1968 there was still one error, a BNE (branch if not equal) instead of a BE (branch if equal). We corrected that error *after* loading the program(!). We stopped its execution by pulling the magnetic switch at the disc unit IBM 2314 and then altered one bit at a certain address in the RAM using the controls of the console. Only after some weeks we reran the time consuming compiling and linking with a corrected statement.

[4]Medizinisch-Biologische Forschungsstelle am Robert-Bosch-Krankenhaus, Stuttgart, director Dr. R. Pirtkien

[5]DUSP is an acronym for **D***atenerfassungs-* **U***nd* **S***peicherungs*-**P***rogramm* (data acquisition and storage program)

[6]DUTAP = **D**ekodierungs **U**nd **T**ext-**A**usgabe **P**rogramm (decoding and text output program)

[7]See the folowing chapter 4 on page 9

[8]It was later implemented in Fortran and Mumps, see below 4.8 DIADEM and 5 BAIK

[9]Will be explained below in chapter 4.3.1 on page 12

Donnerstag, 25. Januar 1968 — Nr. 21 DUISBURGER NACHRICHTEN NRZ an RHEIN und RUHR

Bethesda und Rechenzentrum leisten Forschungsarbeit

Ein Elektronengehirn schreibt den Duisburger Ärzten Briefe

ph. Duisburg. Die teuerste Schreibmaschine der Welt wird zur Zeit vom Krankenhaus Bethesda benutzt: eine elektronische Datenverarbeitungsanlage vom Typ IBM 360/30. Aber nicht Großmannssucht steht hinter diesem Beginnen, sondern der Versuch, in organisatorisches und medizinisches Neuland vorzudringen. Das Rechenzentrum der Stadt Duisburg unterstützt dieses von dem jungen Arzt Dr. Wolfgang Giere inspirierte Vorhaben.

Vier Minuten braucht der Großcomputer der Stadt Duisburg, 70 mehrseitige Briefe an praktizierende Ärzte zu schreiben. Er vergißt dabei nicht die höfliche Anrede „Sehr verehrter Herr Kollege" und die Schlußfloskel „Mit kollegialen Grüßen, Ihr sehr ergebener . . ." Die Schreiben enthalten komplizierte Befunde und Therapievorschläge für Patienten, die der nuklearmedizinischen Abteilung des Krankenhauses zu Spezialuntersuchungen überwiesen worden sind. Diese „programmierten Briefe" ersparen Krankenhausärzten rund die Hälfte, Sekretärinnen oder Assistentinnen sogar dreiviertel der üblichen Schreib- und Diktatarbeit. Das Bethesda ist das erste Krankenhaus, das ein solches Verfahren anwendet.

Freie Hand gelassen

Eineinhalb Jahre hat Dr. Giere von der Idee bis zur Verwirklichung seiner Pläne gebraucht — und aufgeschlossene Helfer. Er brauchte einen Chef, der ihm freie Hand ließ. Er fand ihn im Leiter der nuklearmedizinischen Abteilung des Bethesda-Krankenhauses, Dr. Schmidt. Und er brauchte eine hochgezüchtete elektronische Datenverarbeitungsanlage und Praktiker, die seine Vorstellungen in Computer-Programme umzusetzen verstanden. Er fand sie im Rechenzentrum der Stadt Duisburg.

Nächtelang hat Dr. Giere mit dem 23jährigen Stadtinspektor Horst Baumann beraten und geplant, bis das Ziel erreicht war. Seitdem ist der junge Beamte mit einigen hundert medizinischen Fachausdrücken vertraut, und der Arzt wurde halbwegs zum Elektroniker. „Ohne die Unterstützung des Bethesda-Krankenhauses und des Rechenzentrums hätte ich mich an eine solche Aufgabe nie heranwagen können", sagt Dr. Giere

Mit Selbstkontrolle

Die mit der nuklearmedizinischen Abteilung des Bethesda-Krankenhauses zusammenarbeitenden Ärzte haben sich inzwischen daran gewöhnt, daß die meisten Briefe an sie von einem Computer geschrieben werden. Sie stoßen sich auch nicht daran, daß die Maschine nur über Großbuchstaben verfügt. Die Großschreibung ist praktisch der einzige „Fehler", den das Elektronengehirn macht. Es ist gewissermaßen eine ideale Sekretärin, denn es kontrolliert sich selbst und kann sogar sinnwidrige Angaben in Schriftsätzen aufspüren. Trotzdem werden die Briefe von den Ärzten noch einmal durchgelesen, bevor sie abgeschickt werden. Auch der perfekten Technik gegenüber bleibt ein Rest von Skepsis.

Voraussetzung zur Verwirklichung programmierter Ärztebriefe war die Volldokumentation von Krankengeschichten. Dr. Giere hat nicht auf die bereits von einigen Universitätskliniken erarbeiteten Methoden zurückgegriffen, sondern ein eigenes, erweitertes System entwickelt, das, wie er sagt, die Möglichkeiten eines Computers der dritten Generation besser nutzt. Das beinahe unbegrenzte „Gedächtnis" solcher Maschinen speichert alle dem Arzt wesentlich erscheinenden Informationen über einen Patienten und erlaubt eine spätere wissenschaftliche Auswertung. In der modernen Medizin brauchen nicht mehr Tausende verstaubter Krankengeschichten nachgelesen und mühsam ausgewertet zu werden, wenn man statistisches Material der Forschung dienstbar machen will. Große Computer leisten diese Arbeit, die früher Monate und Jahre gekostet hat, in Minuten, — vorausgesetzt, daß man zur elektronischen Datenerfassung übergeht.

Ein Arzt, Dr. Wolfgang Giere (links), und ein Beamter, Stadtinspektor Horst Baumann, leisteten Teamarbeit. Sie entwickelten den „programmierten Ärztebrief". NRZ-Foto: Paetzold

Universelles Programm

Dr. Giere hat sein Programm für ein medizinisches Teilgebiet, die Erkrankungen der Schilddrüse, zusammengestellt. Das von ihm entwickelte System ist jedoch universell und kann, wie er sagt, ohne Schwierigkeiten auf andere Bereiche ausgedehnt werden. Die Computer sind nicht zimperlich. Ihnen ist es gleichgültig, ob sie Befunde über Schilddrüsen- oder Gallenleiden speichern und zu einem Brief verarbeiten. Es kommt nur auf die Programmierung an.

Ein Computer-Programm setzt sich aus einer Vielzahl von Bausteinen zusammen. Einzelinformationen für diese Bausteine werden aus Fragebogen gewonnen, auf denen die gebräuchlichsten Diagnosen und Befunde, Therapievorschläge und Angaben des Patienten erfaßt sind. Der Arzt wählt unter den wie in einem Quizspiel vorgesehenen Möglichkeiten des Fragebogens aus und unterstreicht bestimmte Worte. Jedes dieser Worte enthält einen Code-Buchstaben, der zur Information für den Computer wird. Wenn eine Sekretärin bei der Auswertung des Fragebogens auf ihrer mit einem Lochstreifenstanzer gekoppelten Schreibmaschine beispielsweise die Buchstaben f-g-i-n tippt, macht der Computer für den Ärztebrief daraus den Satz „Bei der Untersuchung konnte man die feste, symmetrische, indolente Schilddrüse gut abgrenzen".

Erheblicher Effekt

Der Rationalisierungseffekt des neuen Verfahrens ist beträchtlich. Es wäre deshalb durchaus denkbar, daß sich in Zukunft Krankenhäuser zusammenschließen, um gemeinsam elektronische Datenverarbeitungsanlagen zu nutzen, zumal sich die Computer nicht nur im medizinischen, sondern auch im Verwaltungsbereich kostensparend verwenden lassen.

Figure 3.1: Newspaper, Jan 25, 1968: Author (left) and Horst Baumann, the programmer, control the printed reports

4 The DKD-system

1969 I started to work for the *Deutsche Klinik für Diagnostik (DKD)*[1].

When starting the computer Center in the DKD – it opened April 1970[2] – the physicians wished to use *programmed reporting*, too. I transported my system to the Siemens 4004/35, first the two programs DUSP for data input and DUTAP for text output[3]. We designed many data acquistion forms and reports for different specialities from anesthesia to zytology. Due to the reduction of needed key strokes (codes for frequent findings and automated Ientification of the patient!) a secretary produced more than 300 pages of reports in one morning, a huge time saving!

4.1 Data Input and Structure: The DUSP System

DUSP was an input generator controlled by *document parameter sets*. For each form these parameter sets described its name, structure, the set of fields and their data types, formal and plausibility checks[9]. I programmed and tested DUSP in its new form in Stuttgart, it was used in the DKD and later in DIPAS for many years. At the Goethe University Klinikum[4] we reprogrammed it for BAIK using Mumps.

Let us look into some details which remain unchanged until today:

4.2 Identification of a Form

A form is the semantical unit. Its Identification requires the following IDs:

4.2.1 Institution ID

The reporting and documentation of medical facts can differ from institute to institute, from clinic to clinic, from school to school – as can the sets of Patients. Consequently the highest ID in BAIK is the *GKZ*[5].

[1] German Clinic for Diagnostic at Wiesbaden, modeled after the *Mayo Clinic*
[2] one of the very first clinical computer centers
[3] See chapter 3.1 and the footnotes 5 and 6
[4] Klinikum der J.W.Goethe-Universität Frankfurt. *Klinikum* means Medical School as well as Univerity Hospital. I use the term *Klinikum* throuout.
[5] GKZ = **G**ruppen**k**enn**z**eichen (Group ID, Institution ID)

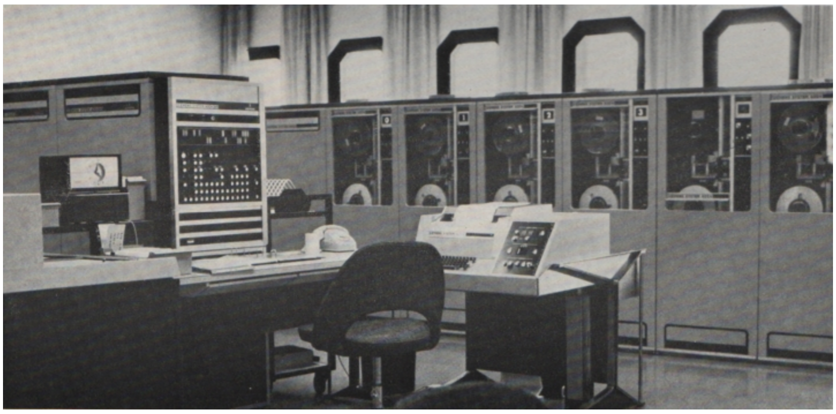

Figure 4.1: DKD-Computer Center with the Siemens 4004/35

4.2.2 Forms: Principle of Semantical Frames

To be able to ‹understand› automatically (electronically) the content of an entry we used semantical frames. Within a frame *EKG*[6] a *78* in a specific field means pulse 78; in another context the same figure has a different meaning. Always the meaning is determined by the semantical frame and inside the frame by the position (field). The semantical frame in BAIK is determinded by the so called *AWZ*[7] and the Version number. The latter changed with every change in the acquisition form.

4.2.3 Identification: Principle of Double Sequence

Whereas the AWZ for a semantical unit, e.g. EKG, remains the same, the version number changes according to the evolving structure of the data. The EKG report might first be dictated, then acquired using a filled in form, then a dataset from the Pipberger EKG analysis program, then the one from a more recent apparatus and so on. It is always an EKG. But the Version Number changes with the evolution of medicine. Whether one can use and combine certain data for research purposes has to be analyzed from case to case.

Consequently we get one sequence per semantical Entity, e.g. EKG

The other sequence, of course, is the number of EKG reports, a specific patient has, called *LNR*[8].

It may well happen that a patient has reports with Version number 1, none with 2, several with three and so on. The two sequences are different. The one is bound to the biography of the patient, the other to a method, the development of medicine or reporting in a specific institution.

[6] EKG Elektrokardiogram – we use this abbreviation (not ECG) throughout.
[7] AWZ = **A**us**w**ahl**z**eichen (Retrieval-ID)
[8] LNR = **L**aufende **N**umme**r** (running number)

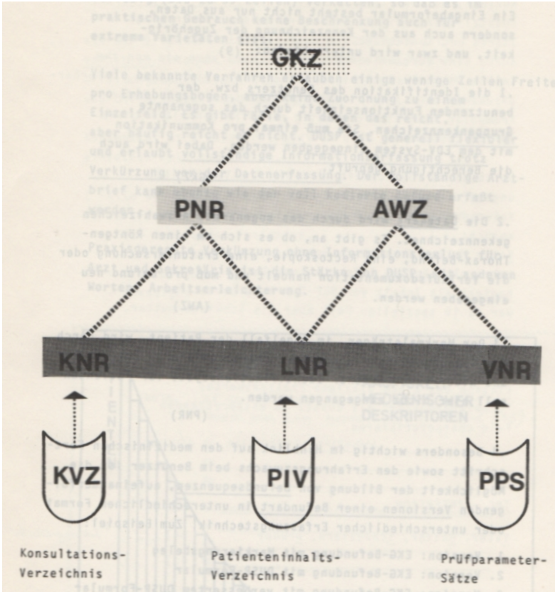

Figure 4.2: Identification of a BAIK-Form: Principle of double sequence, forms per patient, form-versions per group[1]

4.2.4 Episode-ID

Many forms can be attributed to one medical episode, e.g. a chemotherapy or a specific study. This *KVZ*[9] is optional.

4.2.5 Structure: Chapters within a Form

Within a form (GKZ,AWZ,VNR) there may be one or more chapters. Each chapter allows for repetitive structures – very much like subfiles in FileMan.

4.3 General Field Structure: Principle of Zipf's Law

The smallest adressable unit of a semantical unit is called a field. In BAIK the general structure of a field is different from FileMan in order to comply with Zipf's Law. This is illustrated in picture 4.3 on page 13. It shows an early study counting different words in Diagnoses:

- HÄUFIGKEIT JEDES WORTES frequency of each word (of 429,665)
- ANZAHL VERSCHIEDENER WÖRTER count of different words (24,462)

[9]KVZ = **K**onsulations**v**er**z**eichnis (Consultation ID)

- SUMMENKURVE summation: 0.75% of different words result in 52.8% of all words[1]

We reflected this law in the definition of a field[10]: In every field after the typed or coded part a free text *Zusatz*[10] is allowed (see Abb. 4.4). It is separated from the first (coded or typed) part by an asterisc. Thus a BAIK field with its two parts resembles a complex number. The semantical structure of both parts is different:

4.3.1 Typed part: Ternary semantics

The first, the typed part is similar to FileMan fields with a few differences: Every typed part has a ternary structure:

1. " = Nullstring: implicit negation
2. 0 = Zero: explicit negation
3. type = one of the many datatypes: positive content, to be decoded if coded

The ternary logic is needed for the natural language generation and classification. Statistically it is not the same to have no answer, i.e. undefined, or an explicit zero (without pathological finding).

4.3.2 *Zusatz*: Binary logic

The second, the *Zusatz*-part, is binary: Either there is a free text or not. It is optional: There may be a text (of up to 99 lines) but it is not needed.

The Zusatz-capability allows the physician to mark any value. E.g. the height in cm might be given as *172*kyphoskoliosis*. Then it depends on the type of study whether the value will be included or not.

Another example of free text Zusatz is *S*after sphincterotomy*. *S* stands for *sphincter* **s***tarr*, in this case after an operative procedure. The binding of free text to a specific semantic frame allows short texts with well defined meanings.

It also allows to add personal remarks, unusual findings etc. In principle every field can have a *Zusatz*. Its meaning is defined by the semantical frame of the field.

[10] Zusatz = second part, optional

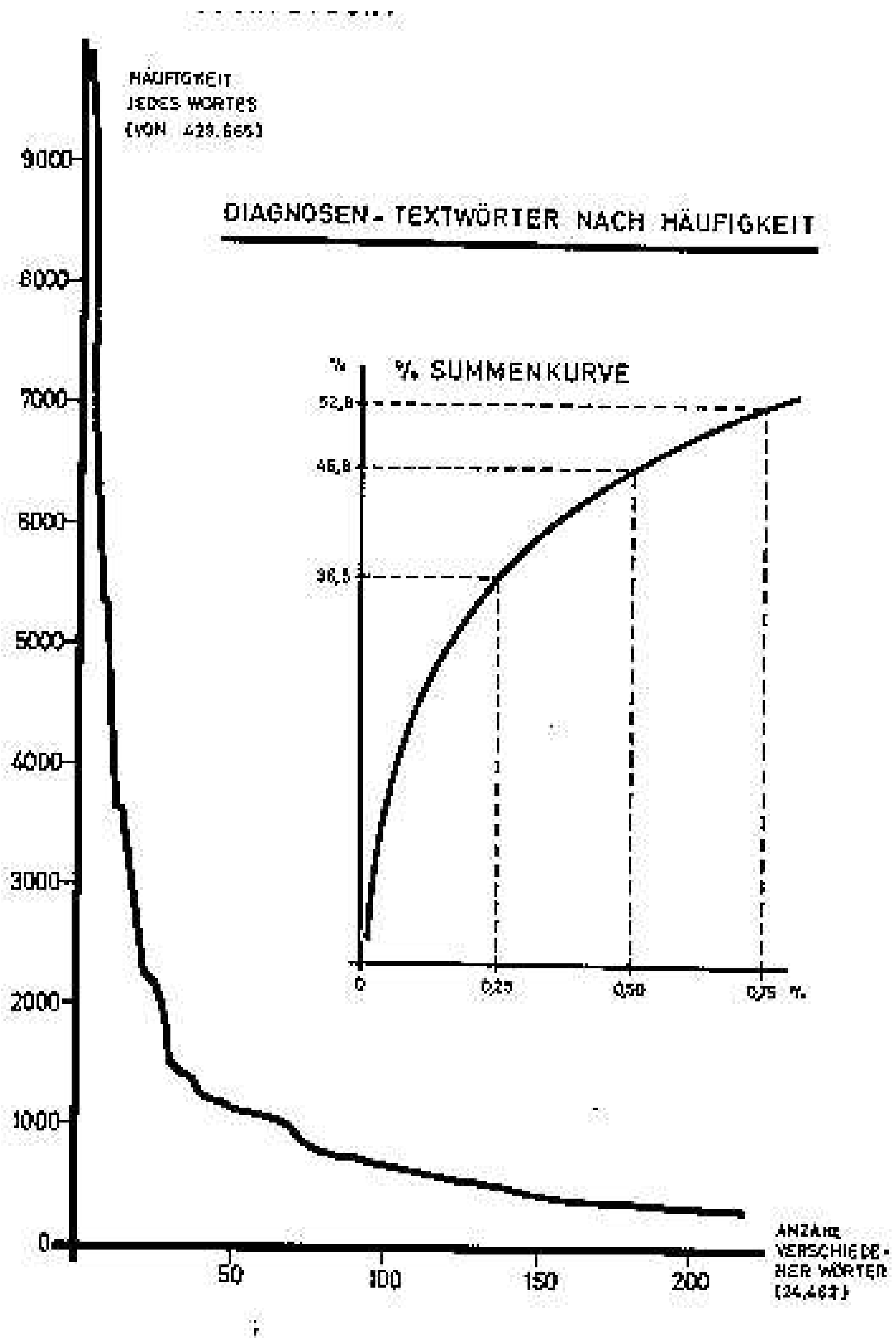

Figure 4.3: Illustration of Zipf's law counting the frequency of words in DKD-Diagnoses, for explanation see text on page 11

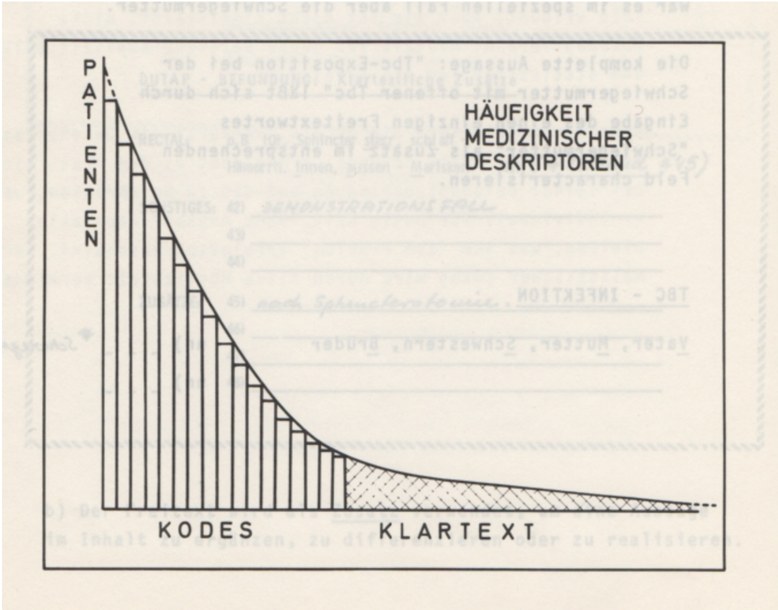

Figure 4.4: Kodes for frequent, free text for rare findings – according to Zipf's Law[1]

4.4 Data Types: Principle of Pragmatism

The datatypes were chosen to minimize the necessary key strokes at input time. All data types are well defined in Backus Naur formalism[10].

In this publication I only list the most frequent BAIK-datatypes. Details may be found in the literature[1]. Only interesting differences between BAIK and FileMan are pointed out.

IKn Multiple choice, one answer per position

IKV Multiple choice, multiple answers

IKZ Time encoding (3w = 3 weeks, 12t = 12 Tage (days), 1j = 1 Jahr (year)

IKR *Radiomat* Radiological dictation with code hierarchies, a very effective dictation tool[11]

ITK Thesaurus Control: For this field a thesaurus could be indicated, against which the words were matched. In case the word did not match, the typist had the choice of correcting the spelling or hit the terminology mailbox button. In that case the word was recognized in future. (The terminology mailbox war revised periodically)

INV Numerical with *V*ariable boundaries: Allowed the typist to override the realistic lower and upper limits if needed for unusual values. Think e.g. of an extreme liver weight with a big echinococcus cyst. Otherwise in medicine you would have to keep widening the window of allowed values resulting in less and less plausibility control!

Like in FileMan so called document parameter sets contained the data types of every field including possible controls and boundaries.

4.5 Text Generation System – DUTAP

The output system was called DUTAP. Generation of quasi dictated reports from the kind of short hand entries – codes for frequent findings, free text for rare and atypical ones – was attractive for physicians and secretaries. It saved work for both of them. The idea was to reward the physician for good data. The computer program would proofread the data and print a big error message if there were implausibilities. Only if it was complete and ok, it would generate a nice looking report, ready for the physician to sign.

4.6 Online Service for Physicians: DIPAS

This service for physicians was well accepted not only within the DKD but also by outside specialist physicians. Thanks to government grant money we could offer them online services in 1973. Twelve physicians with different specialities had online access to our DUSP/DUTAP system via the use of 300 baud lines and rapid online typewriters (kind of teletypes)[11] In the project report[1] I described two main principles:

1. The principle of *qualifizierende Informationsaufbereitung* (qualifying information preparation) as model and longterm goal. (That is, what I call nowadays an active electronic medical record based on the information model.) The first step towards this longterm goal was the retrieval system IATROS (Information Analyzing Text Retrieval Oriented System).

2. Interactive development as introduction strategy (very much like the VA-strategy later).

Both goals were illustrated in the DIPAS-brochure[1] (see fig. 4.5 and 4.6)

Within this project the problem of classification and retrieval of the code/text-mix was solved:

[11] The project was named «Einführung der Datenverarbeitung in die ärztliche Praxis – Dokumentation und Informationsverbesserung in der Praxis des niedergelassenen Arztes mittels EDV-Service (DIPAS)»

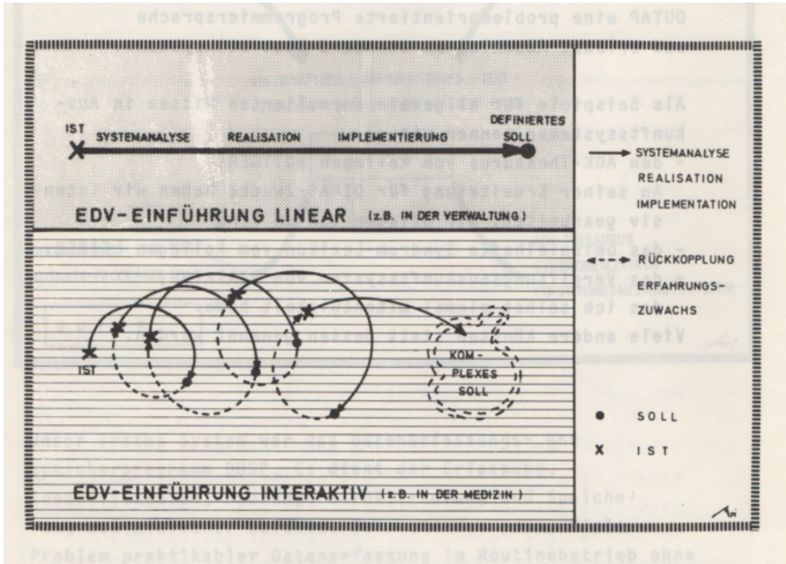

Figure 4.5: Interactive development with feed back and an evolving complex target instead of a linear approach[1]

4.7 Data Evaluation: The System IATROS

The data controlled and stored by the system DUSP was well structured in forms (see 4.2) and Fields (see 4.3). Remember, as pointed out above, due to Zipf's Law the stored data consist principally of a mixture of coded (or typed) data and free text. To evaluate coded data and numericals is not difficult. To evaluate text is more difficult, to analyse fields with a mix of coded and textual data requires special efforts. The original data have to be transformed into a metastructure that allows the combined evaluation. The transformation depends on the question to be solved. Every study has its own needs. They may differ from each other. This is the reason for our strict distinction between primary patient data in the form/field-structure, as described above and the derived secondary data[12] (Details below in chapter 5.1 on page 19).

Within DIPAS all the principles were developed. We designed a methodology to build the secondary file for evaluation and an interpretive language IATINT to question it. This included the use of thesauri and semantical hierarchies for the searches[13].

The so called ‹Dokumentations Parametersatz› (Documentation Parameterset, see fig. 4.7 on page 17) indicated how each field had to be transformed. It contained 24 bits[12]

- ENDANZEIGE Indicator of the last field to be evaluated
- ZEILENNUMMER Line number

[12]Our computers in those days had max. 64K byte of memory and disks with 7 MB storage capacity. So we had to save bits!

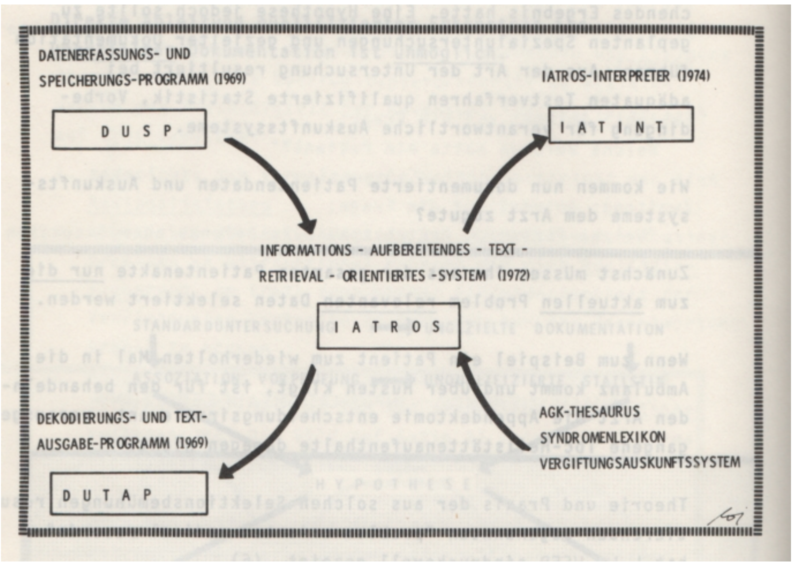

Figure 4.6: IATROS Information Analyzing Text Retrieval Oriented System with IATINT Query language for DUSP data

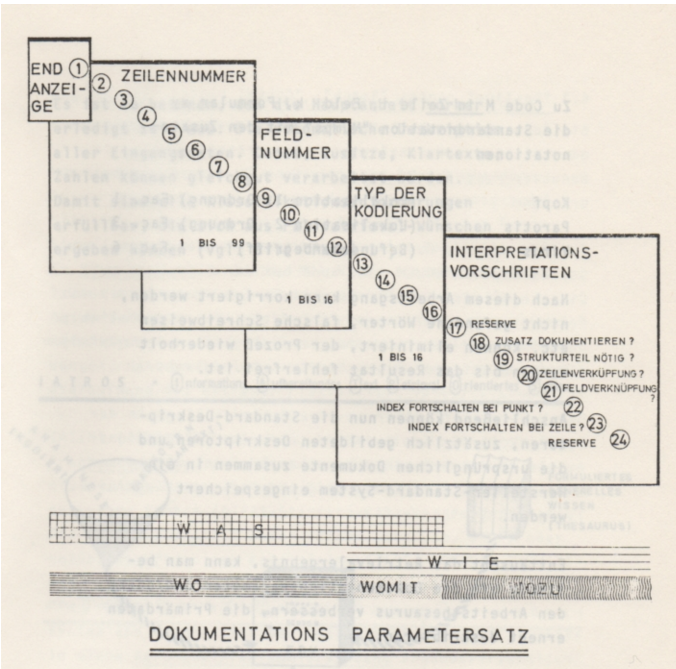

Figure 4.7: Documentation Parameter Set[1], for explanation see text on page 16

- FELDNUMMER Field number
- TYP DER KODIERUNG data type of coding
- INTERPRETATIONSVORSCHRIFTEN Interpretation instructions
  - bit 17 Reserve
  - bit 18 ZUSATZ DOKUMENTIEREN? With or without Zusatz[13]
  - bit 19 STRUKTURTEIL NÖTIG? Is the identification of the semantical frame needed?
  - bit 20 ZEILENVERKNÜPFUNG? Valid for more than one Line
  - bit 21 FELDVERKNÜPFUNG? Valid for more than one Field
  - bit 22 INDEX FORTSCHALTEN BEI PUNKT? Augment index at period[14]
  - bit 23 INDEX FORTSCHALTEN BEI ZEILE? Augment index for new line[15]
  - bit 24 Reserve
- WAS = What?
- WIE = How?
- WO = Where?
- WOMIT = Which Method?
- WOZU = What for? (Purpose)

4.8 Minicomputers in Hospitals: DIADEM

The DUSP/DUTAP/IATROS system, heavily used in the DKD and as online service for outside physicians (DIPAS), was attractive for other hospitals, too. In a follow-up grant DIADEM[16] it was transported to minicomputers in different hospitals[17].

Personally I am convinced if we were allowed to use Mumps (which we were not), DIADEM would have been more successful.

[13] For ZUSATZ see 4.3.2 on page 12, it is the uncoded free text addition allowed in every field.

[14] Within free text a special index counted the phrases. A special algorithm was used to find out the end of a phrase.

[15] Used if arrays had the same semantics

[16] Dokumentations- und Informationsverbesserung für den Arzt mit Dezentralem EDV-Modul (enhancement of documentation and information for the physician using decentralized EDP modules) (DIADEM)

[17] Interesting that we were not allowed to use Mumps, but were forced to use Fortran instead. We suffered greatly from the incompatibility of the existing Fortran compilers[14]. This Fortran version was later marketed as the first Doctors Office System in Germany. Neither system was successful, alas!

5 Mumps System: BAIK

In 1976 I was called to the professoral chair for Documentation and Data Processing at the J.W.Goethe-Unversity Klinikum in Frankfurt/Main. This gave me the freedom to redesign the system and reimplement it using Mumps. We did not change the data structure[1] according to Zipf's law, nor the semantical identification with the double sequence[2].

Our redesign was guided by the Information Model:

5.1 The BAIK Information Model

The BAIK information model is described extensively in all its aspects in the BAIK-book[15] and in some detail in the BAIK-Chapter of Open Systems in Medicine[16]. For the convennience of the reader I quote the latter:

> «**The Information Model – Cybernetics of Collecting and Ordering**
>
> The BAIK information model has guided the development for more than 20 years[3]. It still is the guideline for future development. ...
>
> The backbone of the information model is a cybernetical information flow linking care, teaching and research together:
>
> (1) A patient comes with a problem (?) to a physician. He examines the patient and notes symptoms, signs and tests in the medical record The medical record – next time – gives him the information he needs to treat (!) the patient. This is the classical, care oriented cycle.
>
> (2) From the medical record data is selected to be classified and put into a register. The register in turn allows comparison of cases, adds comparative information of similar cases to the individual information of a single patient. This is the teaching oriented cycle, describing a multitude of similar cases. Teaching means to teach others or to gain insight oneself comparing similar cases.
>
> (3) From the register statistical information can be drawn which allows the researcher to formulate a hypothesis. This in an experiment can be verified

[1]See 4.1 DKD system on page 9

[2]See chapter 4.2.3 *Identification* on page 10 and figure 4.2 on page 11

[3]The quotetd text was published in 1995

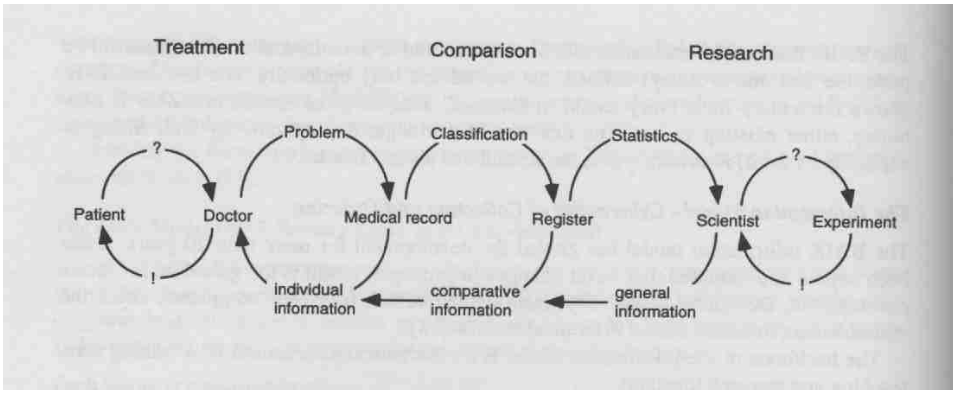

Figure 5.1: BAIK Information Model. Explanation in chapter 5.1 on page 19

or falsified. The resulting knowledge again adds general information to the comparative and individual one.

Thus a complete cybernetical loop is formed between daily practice and scientific research. This ‹BAIK-Byke› allows to find the appropriate place for different constituents of a physicians workstation:

The acquisition of symptoms, signs and tests by the physician depends on his *experience.* The classification depends on a question which allows the correct establishments of differentiation *Criteria.* Tere is no general classification, classification alsways depends on a goal, never exists per se. Withour a specific question there is noch answer by classification. Classification means selction and appropriate transformation, always concentration and hence loss of details, gain on information about groups.

The statistical information of course is rendered by the interaction with *Methods.* this is the place for Systems like SAS, SPSS, BMDP and the like.

The general information is stored in electronic *Libraries.* The National Library of Medicine is a good example. It renders additional information from the books.

The comparative information can be augmented by *Expert Systems*, computer aided instruction and rule based quality assurance. HELP[4] is a very good example for the data driven analysis which I had in mind when I designed the BAIK model.

The individual *Information Profile* is meant as a filter depending on the skills and interests of the physician, his previous knowledge an what was presented to him earlier.

What we wished to achieve was a mechanism by which the physician using BAIK would get – in addition to and triggered by the data of a patient he

[4]Warner, R, JD Morgan, TA Priyor, S Clark, W Miller: HELP – A Self-Improving System for Medical Decision Making. In: Anderson J, Forsyte JM (eds) MEDINFO, Stockholm, 1974

entered himself – automatically relevant information to his specific case. that could be a recently published article, help from a decision support system, information of other patients he had seen earlier and so on. The selection should take into account his speciality, interest, a specific selection of journals, in short: a physician-specific profile. We designed the system to use the world of electronically available information to help a physician. Again: A thesaurus was needed to translate medical record information into MeSH or MEDIUC or DXplain.»

5.2 Thesaurus

A thesaurus is needed for every cycle of the *BAIK-Byke* described above.

- It controls the input vocabulary at dictation or description. As a result every entered word is known to the thesaurus. If a string is new to the thesaurus the entering person is warned and has two choices: Correction of a misspelling or acceptance of a new word. In that case the word is entered in a *nomenclature letterbox* which is periodically reviewed and emptied. The terminologist adds the new word to the thesaurus with the appropriate semantical descriptions either as new synonym or as a completely new concept.

- It governs the classification. The information contained in the semantical frame (its context and location) has to be used to correctly bind items (numbers, codes, or words) to the semantical net, the thesaurus nodes.

- It is used for retrieval. A physician may ask summary questions, e.g. *smoker*, or *viral infection* and gets answers regardless of the primary terms used in the relevant information. The thesaurus helps to interpret and translate the terminolgy of the question to the terminology of the entered patient data.

As mentioned already the thesaurus is also used as interface to the terminology of information systems like MEDLINE, rendering the appropriate MeSH-codes. The same is true for expert systems. Wherever different terminologies of systems have to be matched, the thesaurus is helpful.[5]

[5]Building of the thesaurus was started in the late 60s using punched paper tape[17]. In the 70s we entered every dictated word of the referral letters of the German Clinc for Diagnostic in Wiesbaden[13]. At the end of the 70s we installed BAIK in the J.W.Goethe-University-Klinikum for all dissection reports and other medical dictations[18]. By then the maintenance proved to be quite managable: one terminologist for half a day per month to work through the new entries in the terminology letter-box[19]. Our thesaurus was the basis of the ICD-10 diagnosis thesaurus which nowadays is maintained by the governmental agency DIMDI (the German analogue of NLM) and used in all German speaking countries (Germany, Austria, Switzerland) On the occasion of my 75th birthday DIMDI surprised me with a dedication, «Widmung»(see figure 5.2 on page 22)

Widmung

Habent sua fata libelli – Büchlein haben ihre Schicksale. Dieser Satz des Terentianus Maurus, vielleicht etwas aus dem Zusammenhang gerissen, gilt natürlich auch für das Alphabetische Verzeichnis zur ICD-10-GM.

Alles begann vor 15 Jahren in einem Projekt des Zentralinstituts für die kassenärztliche Versorgung. Unter der wissenschaftlichen Leitung von Prof. Dr. med. Wolfgang Giere, damals Leiter des Zentrums für medizinische Informatik der Universität zu Frankfurt/Main, entstand der ICD-10-Diagnosenthesaurus, ein schmales Bändchen mit vielleicht 10.000 Einträgen. Vorbild war Martin Luther: „Dem Volk auf's Maul schauend" sollte statt einer klassifikatorischen Kunstsprache die ärztliche Alltagssprache gesammelt und für die Kodierpraxis genutzt werden.

Über die Jahre hat der Diagnosenthesaurus sich gemausert zu einem umfangreichen Standardwerk mit über 100.000 Einträgen, das wohl jede Ärztin und jeder Arzt im deutschen Sprachraum kennt und – freiwillig oder unfreiwillig – immer wieder in die Hand nimmt.

Wir hatten das große Vergnügen, an diesen Arbeiten teilzunehmen. Wie kein anderer konnte Prof. Giere eine konstruktive Arbeitsatmosphäre schaffen, Probleme dank seiner langen Erfahrung pragmatisch aus dem Weg räumen und jederzeit sein Vorgehen theoretisch untermauern.

Für uns – in diesem Geschäft schon damals nicht ganz unerfahren – war das faszinierend, lehrreich, anregend und unterhaltsam.

Vor einigen Jahren hat das Zentralinstitut die Verantwortung für das Alphabetische Verzeichnis auf das DIMDI übertragen. Auch nach seiner Emeritierung stand uns Prof. Giere mit wissenschaftlichem Rat zur Seite.

Weshalb wir ihm die Ausgabe für das Jahr 2011 widmen? Nun, wer ihn gut kennt, wird um den Anlass wissen. Verraten wollen wir ihn hier nicht. Aber ein herzliches Dankeschön für die gemeinsame Zeit und die besten Wünsche für seine Zukunft sind uns wichtig. Ad multos annos!

Deutsches Institut für Medizinische Dokumentation und Information
Köln

ICD-10-GM Version 2011 3

Figure 5.2: Dedication of DIMDI in the 2011-edition of the ICD-10-Thesaurus honoring my role in the development. (See footnote 5 in chapter 5.2 on page 21)

5.3 The Difference between Collecting and Ordering

Many more aspects of the BAIK-information model are explained in the BAIK monography[15], alas, in German only. But they are less relevant for the aim of this publication, for the lessons learned from BAIK for VistA.

This one, however, is essential in my opinion and I quote again the above cited publication of 1995[16]:

> «Since we found that it is quite often overlooked, another aspect of the BAIK model shall be briefly mentioned: The difference between the data collection (medical record) and data order (register). The collection is unique, data is entered once. The form of the data may vary over time in spite of the fact that semantical frames in principle are rather stable. However, there can be and usually there are many registers per system depending on the ongoing research and appropriate classification criteria. A register is not just another *view* of the primary data. Selection and classification can imply complex transformations, including the normalization of different versions of data (see EKG example above[6]). A register contains metadata. The difference between data collection (medical record) and order (register) may be highlighted by these pairs of terms:

Data Collection (Medical Record)	**Data Order (Register)**
patient	case
individual	group
characterizing	typing, classifying
communicative	distributive
open for news terms	closed, predefined, standardized
lifelong	episode
primary	secondary

> This list should provide a feeling for the fundamental differences of the two databases: The medical record with the primary cata and the register(s) with secondary, transformed, standardized and normalized data. They are by no means identical!»

5.4 IATROS

The Mumps system BAIK enhanced the functionality of the old IATROS system:

- The classification used identical Document Parameter Sets (DPS)[7], but allowed for Mumps code to control the transformations. Like FileMan it became a programming environment.

[6]on chapter 4.2.3 on page 10
[7]See figure 4.7 on page 17

- We constructed a powerful parser for the search commands. It replaced IATINT[8]. Especially the use of the thesaurus for semantical questions using the semantical net was greatly enhanced, including an interface to the widely used statistical package BIAS[20], developed by Ackermann in our Center of Medical Informatics.

5.5 Adaptation of FileMan for BAIK

All databases in BAIK used FileMan with one exception: The patient record file itself had a different design, was not transferred to the FileMan database. The main reasons for this were:

- different identifications (problem of double sequence)[9]
- different structure of a field (complex Field to comply with Zipf-law, ternary/binary, see chapter Field on page 11)
- different datatypes of fields[10]

We had many discussions with hardhats (and George Timson especially) about means to reflect our different data structure to FileMan. It would have been feasable but clumsy. Consequently we did not change the well established format for the primary data, but secondary (derived, standardized and possibly transformed) data and all other files used FileMan. To reflect the double sequence identification structure – so important for semantical analysis – we invented for the thesaurus the *Kunstwort – artificial word* consisting of an identification praefix combined with the content of the field. These artificial words were used as synonyms to the appropriate terms in the semantical net.

5.6 Graphical User Interfaces for BAIK

There were some early attempts to adapt grafical user interfaces for BAIK, generators and others. Finally BAIK-web was the answer to that problem. Nowadays I would use EWD, of course.

5.7 Achievements and Wishlist in the Eighties

In the eighties BAIK was used by several institutions and hospitals. In certain states (Bavaria, Bremen, Hessen, Niedersachsen, Rheinland-Pfalz) it was available free of charge. These states supported the centralized maintenance (sufficient to finance one person).

[8] See chapter 4.7 on page 16
[9] See chapter 4.2.3 on page 10
[10] See chapter 4.4 on page 14

New development required additional capacities: Students who wrote their doctoral theses in medicine, computer science or medical informatics. At my institute I could promote physicians to *Dr. med.*, scientists to *Dr. rer. med.* and in cooperation with the Technical University Berlin Engineers to *Dr. Ing.*. This capability provided me with «slave labour». Step by step these doctorands worked on the wish list.

So far the requirements of model had been fulfilled only by half. The automated feedback to the physician was not yet available. For instance the use of expert systems was not automatized, in spite of the fact that we had already designed successful expert systems in the sixties. Also we translated DXplain using the language translation software *TRANSOFT*[21], that Bill Moore invented for the translation of German medical texts into English. Together we developed the English to German version.

And in the meanwhile the world-wide web was invented and allowed for new functionality.

On the other hand software companies began to feed the growing market with «Hospital Information Systems». (And hospitals preferred to buy software from a company rather than get it from a university!)

At the J.Goethe-University Klinikum Frankfurt the routine operation of the data processing center was separated from the scientific institute, the *Zentrum der Medizinischen Informatik (ZInfo)* (Center of Medical Informatics) freeing ressources for research and development. All of it was geared toward the fulfilment of the BAIK-model: Better information for the physician.

6 WorldWideWeb Services: BAIKweb

In the nineties all open parts were identified and solved one by one by doctorands producing prototypes.

6.1 Xmed

The main disadvantage of IATROS was the fact that it answered only to direct questions. We were able to answer whether there were autopsies of HIV-patients before the illness had been named due to the semantical capabilities of the thesaurus-based search. But we did not easily get statistics of the morbidity of all patients. For this we had to classify the patient information and used ICD-9 and (later) ICD-10.

Automated classification of patient data containing a lot of dictated free text (e.g. dissection reports) is a challenge. Of course the thesaurus, which knows every dictated word[1], is of great help. But you need to analyse rules. E.G. an aortic valve defect has an ICD-Code, a mitral valve defect another, but if both occur in the same patient at the same time, there is a third code for the combined vitium.

Xmed is a powerful system developed by two talented computer scientists[22][23]. Xmed has many capabilites:

1. It translates dictated German text into standardized German text. Standardized means
   - use of preferred terms only (thesaurus based)
   - isolation of medical facts (using conjunction-bound rules)
   - normalisation of the grammar of the isolated facts (using grammar and POS[2] rules)
2. Analysis of the relations of medical facts to each other identifying
   - local relations (above, under, frontal, dorsal, . . .)
   - time relations (same time, before, after)
   - causal relations (due to, causing)

[1]see above chapter 5.2 on page 21
[2]Part Of Speech

Arztgruppen	Satzanzahl	Verschlüsselt	%
Allgemeinärzte	293444	264.240,00	90,0
Anästhesie	31151	26.676,00	85,6
Augenärzte	327406	310.520,00	94,8
Chirurgen	45034	36.133,00	80,2
Gynäkologen	156864	144.892,00	92,4
HNO-Ärzte	110150	102.976,00	93,5
Hautärzte	114262	109.964,00	96,2
Internisten	244564	216.630,00	88,6
Kinderärzte	153411	137.382,00	89,6
Nervenärzte	53254	44.995,00	84,5
Orthopäden	120555	98.362,00	81,6
Radiologen	150158	105.802,00	70,5
Urologen	59737	55.097,00	92,2
Gesamt	1859990	1.653.669,00	88,9

Figure 6.1: Xmed provided the best results in a comparative study classifiying huge amounts of real life data.

- intentional relations (to care for, to prevent)

(time, cause, intention)

3. encoding of facts into ICD (9 or 10) using facts, relations and ICD-rules
4. encoding of facts into German procedure codes (OPS) using OPS-rules

As one can see, in addition to the thesaurus many rule collections have to be taken into account:

Part-of-Speech (POS) rules describing the structure of narrative German sentences and the resulting normalized sentence

Relation rules describing trigger-words and the resulting relations. This quite frequently includes disambiguation of conjunctions with more than one meaning.

ICD-rules for the choice of the correct code for any combination or certain circumstances (see example of vitia above)

OPS rules for the choice of the correct OPS code.

The power of Xmed has been tested officially in a large blind study by the German *Bundesärztekammer* (Federal Association of Physicians) in comparison to two commercial systems. The original data was not controlled or filtered and drawn directly from doctors' office computers. The evaluating Person (gold standard) did not know which provider gave the classification of the texts, which system provided the results. According to the tester Xmed delivered the best results[3][24].

6.2 Documentation Control for DRG-Classification

To use the system in todays Germany, there would have to be added DRG rules as all reimbursement for hospitals is based on DRG. We designed a system to control whether all relevant facts to differentiate between possible DRG are documented and to remind the physician if decision criteria are missing[25].

6.3 Dr. Antonius

Dr. Antonius was a web search machine. Originally it contained a web robot searching German medical webpages[26][4]. They were analysed, descriptors put into a dadabase and searches enabled using the thesaurus. If you looked for *Wochendippel* you got only a few documents. But if you ticked the box *using the thesaurus* the search included the preferred term *Mumps* and the other synonyms and you got many hits.

[3]See figure 6.1 on page 28

[4]A major problem we had to solve to identify medical content was the neighbourhood to red light terminology.

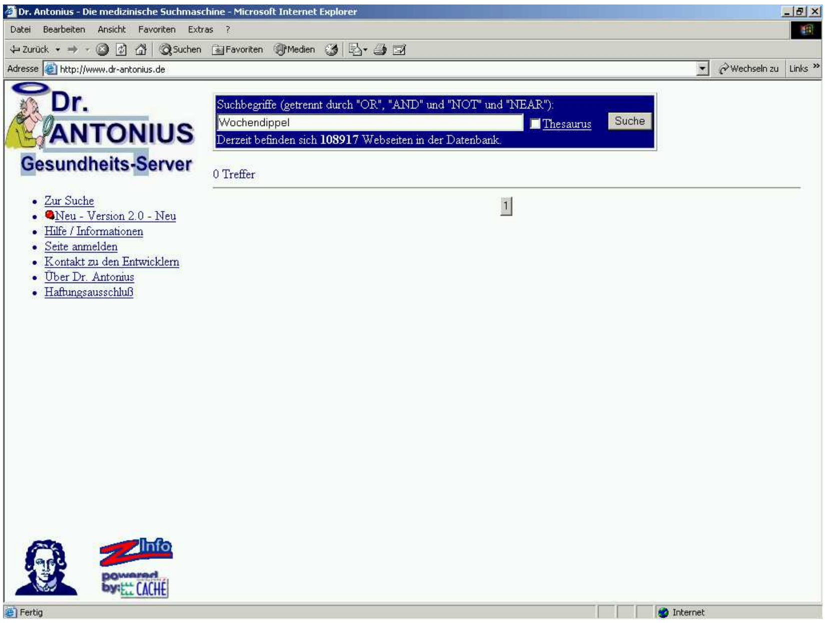

Figure 6.2: Search engine Dr. Antonius: No Results for *Wochendippel*

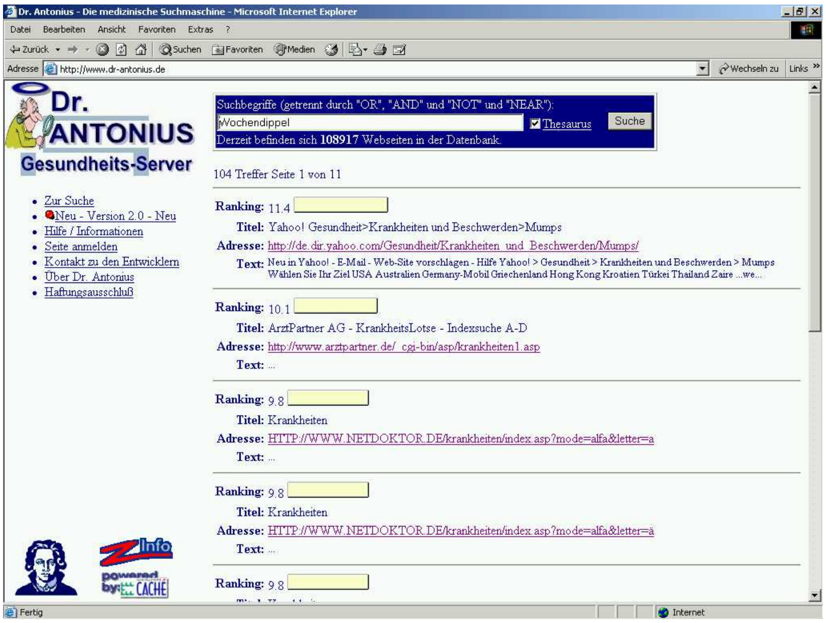

Figure 6.3: Search engine Dr. Antonius: Result for *Wochendippel* with *Thesaurus* ticked showing 104 *Mumps* results, *Wochendippel* being synonym for *Mumps*

The system was heavily used. A later version was designed to use google instead of our own web robot[27]. It was a fast system using GT.M and its bitmap-capability. The idea behind the system was to provide the physician with actual information.

But ZInfo was dissolved[5] and consequently the maintenance stopped. It could (and in my opinion should) be reanimated ...

6.4 MedIAS

To help the decision making of a physician in view of a patient problem using computer capabilities has motivated me from the very beginning of my career[6]. The BAIK-model [7] requires feedback to the physician. It had to be triggered by the classified patient data[8] and would select appropriate information from the output of Dr. Antonius[9], of expert systems and of similar cases.

To achieve this goal we first had to analyse and formalize the physicians information needs[28], then to construct a prototype using the actual web capabilities[29]. The Medical Information Agent System (MedIAS)[30] is the result. Of course, it uses the thesaurus, but it also uses profiles of the information needs of specific physicians. It knows whether he wishes to see original articles in *New England Journal of Medicine* or surveys and recommandations in the *Deutsches Ärzteblatt* only. It also counts how often the source had been presented to the physician already.

[5] When I retired, became emeritus profeesor, my position was not filled again with a medical informatics professor but with a clinician. The Center of Medical Informatics was closed. Some staffmembers changed to the computer center.

[6] After I had become hospitalized for many months due to maltreatment, wrong decisions of colleagues.

[7] See above chapter 5.1 on page 19

[8] Output of Xmed, see above chapter 6.1 on page 27

[9] See previous chapter 6.3 on page 29

7 VistA compared to BAIK

First of all: VistA is successful and used by unnumbered people, maintained by the government and supported by *hardhats*, it is younger and up to date. BAIK is history, only a few installations still survive but will die due to lack of maintenance and active experts. The center of its development, ZInfo, alas, is dissolved.

But some differences are of interest:

Double Sequence: The sequence of changes in the structure of data (EKG interpretations e.g.) and the sequence of semantically identical reports delivered in different formats (EKG-interpretive-systems) reflected in the BAIK identification[1] is needed for lifelong patient records in my opinion. Only with this distinction of camparable data can it be dicided whether the data can be included in a study or must be excluded. VistA miraculously is still able to draw up the EKGs of Pipberger. EKGs ever since were stored in different formats. But as far as I know something like the principle of the *double sequence* is not available.

Medical record and register distinction: The distinction between collecting and ordering, primary and secondary data[2] is strict in BAIK. For each scientific study an appropriate register can be generated keeping in mind that there is no correct classification for every use in medicine. Classification is a function of the question to be answered. And classification depends on time and scientific development. Who would have classified a stomach ulcus as an infection 30 years ago. BAIK allows for this, VistA doesn't.

Use of thesauri: VistA's data dictionary capabilities are great. BAIK was developed around thesauri[3] and linguistic capabilities including automated translation (e.g. in *Xmed*[4]). That is not the same, but the one does not exclude the other. Thesauri, Xmed and the like could enhance VistA. Especially the BAIK data-type ITK indicating thesaurus control[5] for the input would be helpful.

Augmented feedback to the physician: An active medical record delivers much more to the physician than what he himself has put into it. It analyses the patient data and searches automatically for relevant information which could help the specific physician keeping track of what he has already seen before[6]. This requires

[1] See chapter 4.2.3 on page 10
[2] See above chapter 5.3 on page 23
[3] See chapter 5.2 on page 21
[4] See above chapter 6.1 on page 27
[5] See chapter 4.4 on page 14
[6] How often a *new* relevant information is presented to the physician is controlled by the physician's interest profile

- normalized search data (descriptors) as delivered by Xmed[7]
- a thesaurus based web search technique like Dr. Antonius[8]
- an agent like MedIAS[9] to keep track of the physician's interest profile and its use in the past, including an interface to decision support systems like DXplain[31].
- An agent to control the completeness of the documentation for automated classification[10]

[7]See above chapter 6.1 on page 27
[8]See chapter 6.3 on page 29
[9]See chpater 6.4 on page 31
[10]See cahapter 6.2 on page 29

8 Dream: WorldVista + BAIK features

Already as a joung physician and pioneer of medical computer use I had a dream. As retired professor of medical informatics I still have that dream: To provide the physician with actual, highly useful and well selected information for his actual decision problem to best help a patient. VistA is the best electronic medical record. With some of the BAIK features the dream could become true. All the components are available.

It could be feasable – by the way: It's all Mumps.

Bibliography

[1] Wolfgang Giere. *Einführung der Datenverarbeitung in die ärztliche Praxis: Dokumentation und Informationsverbesserung in der Praxis des niedergelassenen Arztes mittels EDV-Service (DIPAS).* gsf Gesellschaft für Strahlen- und Umweltforschung, Projekt Datenverarbeitung in der Medizin, DVM-Bericht 3, München, 1975. http://www.zinfo.de/DIPAS.

[2] Manfred W. Gall. *Computer verändern die Medizin.* Fischer Taschenbuch, Frankfurt, 3. edition, 1971.

[3] Thomas M. Lehmann and Erdmuthe Meyer zu Bexten, editors. *Handbuch der Medizinischen Informatik.* Hanser, München, Wien, 2002.

[4] MuchMore. Web, 2001. http://muchmore.dfki.de.

[5] Wolfgang Giere, Horst Baumann, and Helmut A.E. Schmidt. Der programmierte Arztbrief; ein Weg zur klinischen Volldokumentation. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 41–56. epsilon, Darmstadt, Hochheim, 1996. Reprint IBM-Nachrichten 193, 505-511, 1969.

[6] Rudolf Pirtkien and Wolfgang Giere. *Computereinsatz in der Medizin, Diagnostik mit Datenverarbeitung.* Georg Thieme, Stuttgart, 1971.

[7] Wolfgang Giere and Horst Baumann. Zur Erfassung und Verarbeitung medizinischer Daten mittels Computer; 1. Mitteilung; ein Datenerfassungs- und Speicherprogramm (DUSP) zur Dokumentation von Krankengeschichten. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 57–72. epsilon, Darmstadt, Hochheim, 1996. Reprint Meth.Inform.Med. 8: 11-18, 1969.

[8] Wolfgang Giere. Zur Erfassung und Verarbeitung medizinischer Daten mittels Computer; 3. Mitteilung; das Dekodierungs- und Text-Ausgabe-Programm (DUTAP). In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 57–72. epsilon, Darmstadt, Hochheim, 1996. Reprint Meth.Inform.Med. 10: 19-25, 1971.

[9] Wolfgang Giere. Zur Erfassung und Verarbeitung medizinischer Daten mittels Computer; 2. Mitteilung; die Fehlerprüfung der durch das Datenerfassungs- und Speicherprogramm (DUSP) gespeicherten Daten. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 73–78. epsilon, Darmstadt, Hochheim, 1996. Reprint Meth.Inform.Med. 8: 197-200, 1969.

[10] Wolfgang Giere, Joseph P. Heger, and Norbert Krier. DOC-Programmierung. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 371–380. epsilon, Darmstadt, Hochheim, 1996. Reprint Datenverarbeitung im Gesundheitswesen, eds. B.Schneider, E. Schönenberger, Springer 1976.

[11] Hans-Peter Gockel, Wolfgang Giere, Jürgen Krause, W Nagel, Ulrich Traunecker, and Lotte Windrath. Das automatische Befundverarbeitungssystem RADIOMAT. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 99–110. epsilon, Darmstadt, Hochheim, 1996. Reprint Radiologe 14: 327-333, 1974.

[12] Brigitte Zips and Wolfgang Giere. Klassifikation, befundorientierte Speicherung und Informationsgewinnung mit IATROS. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 137–144. epsilon, Darmstadt, Hochheim, 1996. Reprint Proc. 28. Jahrestagung GMDS Heidelberg 1983.

[13] Detlev Schalck, Franz-Juergen Arndt, and Wolfgang Giere. Erfahrungen bei der Anwendung des AGK-Thesaurus im Bereich der Inneren Medizin. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 91–98. epsilon, Darmstadt, Hochheim, 1996. Reprint Proc. Klartextanalyse in der Medizin, Wien 1973.

[14] Wolfgang Giere, Joseph P. Heger, and Norbert Krier. Practical aspects of the portability of a FORTRAN-program system. In Wolfgang Kirsten and Rüdiger Klar, editors, *Dokumentation und Informationsaufbereitung für den Arzt*, pages 381–388. epsilon, Darmstadt, Hochheim, 1996. Reprint Proc. IRIA, Toulouse 1977, Taylor& Francis.

[15] Wolfgang Giere. *BAIK, Befunddokumentation und Arztbriefschreibung Im Krankenhaus*. Media, Taunusstein, 1986.

[16] Wolfgang Giere. Medical Information Processing - the BAIK Model. In E. Fleck, editor, *Open Systems in Medicine*, pages 24–34. IOS Press, Amsterdam, Oxford, Washington DC, 1995.

[17] P Röttger, H Reul, H Sunkel, and I Klein. Neue Auswertungsmöglichkeiten pathologisch-anatomischer Befundberichte, Klartextanalyse durch Elektronenrechner. *Methods of Information in Medicine*, 9:35–44, 1970.

[18] P Röttger. *Theoretische Grundlagen, empirische Generierung und Anwendungsstruktur eines Textverarbeitungssystems für die Pathologie*. Habilitationsschrift, J.W.Goethe-Universität, Frankfurt, 1979.

[19] Hubert Dettmer. *Klartextverarbeitung von Sektionsberichten in der Praxis*. Dissertation, J.W.Goethe-Universität, Frankfurt, 1979.

[20] Hanns Ackermann. *BIAS*. epsilon, Darmstadt, sixth edition, 2010. includes programs.

[21] G W Moore, U N Riede, R A Polacsek, R E Miller, and G M Hutchins. Automated Translation of German to English Medical Text. *American Journal of Medicine*, 81(1):103–111, 1986.

[22] Christoph Luz. *Xmed, vom Freitext zum Kode, Praxis der EDV-gestützten Klassifikation medizinscher Texte nach ICD und IKPM*. epsilon, Darmstadt, 1997.

[23] Andreas Gregori. *Klartextverarbeitung als integraler Bestandteil medizinischer Dokumentation*. epsilon, Darmstadt, 1995.

[24] H.-P. Siedhoff. Qualitätsvergleich Klartextdiagnosenverschlüsselung. Brief v. 26.9., 2001.

[25] Björn Koschel. *Entwicklung einer regelbasierten Validierung der Dokumentationsqualität zur Unterstützung der automatischen ICD-10 und OPS-301-Kodierung im Rahmen der G-DRG*. Dissertation, J.W.Goethe-Universität, Frankfurt, 2007.

[26] Wolfgang Kirsten, Michael Schlarmann, and Wolfgang Giere. Ein Prototyp eines deutschen medizinischen Webrobots als Teil des Frankfurter Index. In R Muche, G Büchele, D Harder, and W Gaus, editors, *Tagungsband, GMDS 42. Jahrestagung*, pages 316–320, Ulm, München, Sep 1997. MMV Medizin.

[27] Winfried Bantel. *Ein neuer Ansatz im Information-Retrieval – Entwicklung, Implementierung und Test eines neues Algorithmus in der medizinischen Suchmaschine Dr. Antonius 3.0*. Dissertation, J.W.Goethe-Universität, Frankfurt, 2003.

[28] D Paparoditis. *Ein Modell für Informationssysteme im medizinischen Umfeld*. Dissertation, Technische Universität Berlin, 1996.

[29] C Winkler. *Profilgesteuerte automatische Informationsrecherche für den Arzt – Konzeption und Implementierung eines Informationssuchagenten für heterogene medizinische Datenquellen*. Dissertation, J.W.Goethe-Universität, Frankfurt, 2003.

[30] A Thier. *MedIAS Webservice, Medizinisches Informations- und Auskunfts-System, ein arzt- und behandlungsspezifischer Web-Agent zur automatischen Informationsrecherche und -aufbereitung im Internet/Intranet*. Dissertation, Technische Universität Darmstadt, 1999.

[31] G. Octo Barnett. DXplain. Web, Nov 2006. http://dxplain.org.

Index