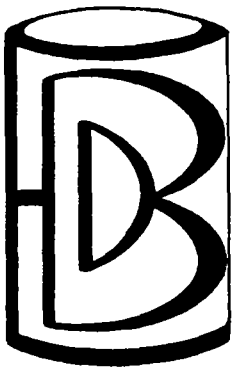

Data Base Engineering

Vol. 2 No. 3 Contents September 1978

A Quarterly Bulletin published by the IEEE Computer Society Technical Committee on Data Base Engineering

Data Base Engineering Bulletin

A Quarterly Publication of the IEEE Computer Society
Technical Committee on Data Base Engineering

Chairman:
Stuart E. Madnick
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02139
(617) 253-6671

Data Base Engineering Bulletin is a quarterly publication of the IEEE Computer Society Technical Committee on Data Base Engineering. Its scope of interest includes: data structures and models, access strategies, access control techniques, data base architecture, data base machines, intelligent front ends, mass storage for very large data bases, distributed data base problems and techniques, data base software design and implementation, data base utilities, etc.

Contribution to the Bulletin is hereby solicited. News items, letters, technical papers, book reviews, meeting previews and summaries, etc., should be sent to the Editor. All letters to the Editor will be considered for publication unless accompanied by a request to the contrary. Technical papers are unrefereed.

Opinions expressed in contributions are those of the individual author rather than the official position of the TC on Data Base Engineering, the IEEE Computer Society, or organizations with which the author may be affiliated.

Membership in Data Base Engineering Technical Committee is open to IEEE Computer Society members, student members, and associate members. (Application form in this issue.)

Message from the Past Chairman

First of all I would like to take this opportunity to introduce our new chairman, Stuart Madnick of MIT Sloan School. As many of you have already known, Stuart has been very active in research in DB area and in professional affairs, including as American Chairman for the 3rd VLDB held in Tokyo last year. Under his leadership, our TC will surely reach another level of success. We are very fortunate indeed to have Stuart accepting the role of our chairman.

While we are looking forward to a new height of success in the forthcoming year, our TC has accomplished much in the past year. In cooperation with other professional groups we have sponsored the VLDB and Non-numeric Processing Conferences and organized the Data Base Design Workshop. In behalf of our TC, Glen Langdon of IBM will be editing a special issue on DB Machines in the IEEE Transaction, and Jane Liu of the University of Illinois will be writing a tutorial-survey article for the Computer Society. In addition, Jane will organize technical sessions in the forthcoming Spring COMPCON. Thus our TC has accepted a full responsibility within the parent society of the IEEE Computer Society.

To make our TC stronger and more effective, as I reported before, I have requested Lorraine Duvall of IIT to organize membership committee for membership drive. It is gratifying to know that our membership has more than doubled within the past year. I like to thank Lorraine and her committee as well as those who helped the drive for having done such a good job. Further, as our TC matures, we wish to move to a more formal structure. Thanks are due to Bing Yao of Purdue University and his Roles Committee for an excellent job in writing the TC Bylaws that has been reported in our Newsletter before.

On the subject of Newsletter, I wish to extend my most sincere gratitude to Jane Liu for continuing to produce TC Newsletters that form the standard of excellence. I believe her Newsletter contributed greatly in our membership drive.

Last, but not least, I wish to thank all those who assisted me in various capacities. I have enjoyed working with you all.

FROM THE INCOMING CHAIRMAN

I look forward to the opportunity to continue the excellent accomplishments and directions started by David Hsiao and Vincent Lum, the first two chairmen of the TC.

As I prepare to take over responsibilities as chairman, I would like to solicit your suggestions and your active participation in the TC. There are many opportunities to help continue and improve upon our existing activities as well as to initiate new activities. Please call me at (617) 253-6671 or write to me at M.I.T. Sloan School, Room E53-317, 50 Memorial Drive, Cambridge, MA 01960.

With your help we will have another successful year. A more detailed message will be included in the next issue of the Data Base Engineering Bulletin.

SOME OBSERVATIONS ON THE FOURTH WORKSHOP ON COMPUTER ARCHITECTURE FOR NON-NUMERIC PROCESSING

Lee A. Hollaar
Computing Services Office
University of Illinois
Urbana-Champaign, Illinois 61801

The Fourth Workshop of the series was held on August 1 through 4, 1978, at Syracuse University's Minnowbrook Conference Center. (These workshops have been held approximately every 16 months, with the previous ones in Dallas, Gainesville, and Syracuse.) Minnowbrook, situated on Blue Mountain Lake in the Adirondacks, was an ideal site. With few other distractions from the scheduled sessions, there were opportunities for interaction between participants, recreation, and excellent meals. The next workshop will certainly have a difficult act to follow!

The papers presented at the workshop will appear shortly in a special combined issue of the ACM SIGARCH, SIGIR, and SIGMOD newsletters. What follows is some general, non-impartial (since I was program chairman and selected the various papers) observations. The workshop format was one allowing questions to be asked at any time, and with equal time allotted for presentations and questions. The kick-

off paper at the first session, "Data Physics - An Unorthodox View of Data and Its Implications in Data Processors," presented by Bob Korfhage of SMU, was selected to promote a lively initial discussion and workshop, and it certainly did. A number of times the discussion shifted from simple questions of the author of a paper to spirited arguments among the participants.

As in the past, the majority of the approximately fifty attendees came from universities, but an increasing number of participants from industry were present (including people from Intel, Bell Labs, Tektronix, IBM, and a group from Siemens). The workshop has always been an ideal setting for graduate students to present their preliminary research, and receive comments, suggestions, and criticism from others in the field, and a number of students took advantage of this.

The primary emphasis of the workshop continued to be database machines, primarily systems which implement some form of the relational model. Means of exploiting the parallelism for multi-user operation of RAP, the use of David Hsiao's DBC for relational databases, and new incarnations of CASSM (for artificial intelligence in Jack Lipovski's paper and as INDY in two papers by George Copeland from Tektronix) provided both a good review of past work and the current thinking of the architects of the best known logic-per-track machines.

Other papers described the work of the Central Intelligence Agency and Operating System, Inc., on the development of full-text search processors as part of their High Speed Text Search system. A different approach to the same goals was presented by Jean Rohmer of France's IRIA. The problems of concurrent access control and the resulting network architectures for distributed databases were discussed in papers by Mohamed Gouda of Honeywell and M. J. Stucki of Washington University.

A variety of non-database non-numeric systems were discussed, ranging from dataflow and state-saving computers, to the Bell System's Electronic Switching Systems (ESS, the reliability aspects of which were discussed by Wing Toy of Bell Labs in Naperville), to a multiprocessor system which directly executes LISP (by Rhon Williams of Illinois).

Plans are currently being made for the next workshop, with David Hsiao tentatively acting as Program Chairman. It's certainly hoped that the Fifth Workshop will be as successful for its participants as were its predecessors.

SOME OBSERVATIONS REGARDING VLDB-4 IN BERLIN

by Lee A. Hollaar

From September 13 to 15 of this year, the Fourth International Conference on Very Large Data Bases (VLDB) was held in West Berlin's

Congress Hall conference center. (Previous conferences have been held in Boston, Brussels, and Tokyo, and next year's is scheduled for Rio de Janeiro, Brazil.) This report is not a complete summary of the technical program, but only an overview of the conference and some personal observations. For the actual papers, the impressive (over 500 pages) proceedings, which are available from either ACM or IEEE, should be consulted.

The formal part of the Conference consisted of about 40 papers presented in 14 different sessions. A wide range of topics were covered, including DBMS architecture and design, user interfaces, distributed databases, formal database theory, and specialized database machines. Surprisingly, since the theme of the conference was very large databases, the first session was entitled Very Small Databases. However, the systems presented could easily form the basic components of a very large distributed database system -- The first paper described a database system which runs under LSX, the LSI-11/floppy disk version of UNIX, while in the second Hsu Chang described how bubble memories can be configured to perform relational database operations.

In addition to the formal papers, a number of tutorial/panel sessions were scheduled. A number of these were held in conjunction with the International Congress for Data Processing (IKD), which was also meeting at the Berlin Congress Hall. These provided a chance for interaction between the university and industrial database system researchers and the ultimate users of such research. Topics discussed in these sessions included database design, software engineering, distributed databases, and security mechanisms, with the format generally consisting of a tutorial paper, followed by discussion by a distinguished panel and questions from the floor. Perhaps one of the most interesting sessions was on the impact of new technology on database system, with IBM's T. C. Chen providing an amusing, but thought-provoking view of the future of LSI technology. Since he feels that the classic von Neumann architecture will remain the predominate form for digital computers (perhaps to the exclusion of many other forms of specialized processors) due to economies of production, and a number of panelists (including David Hsiao, Stu Schuster, Stanley Su and myself) are closely associated with specialized database machines of various sorts, a lively discussion followed.

It is hard to imagine a better setting for a conference, since the various chairmen and conference personnel did their best to accomodate all the participants' needs. This included such luxuries as simultaneous translations in English and German during the panel sessions. Even the weather was well suited for the conference, with periods of rain when the participants should be in the sessions, and cool breaks in the rain to allow sightseeing when the conference was not in session.

RISS: A Relational Data Base Management System for Minicomputers
Meldman, McLeod, Pellicore, and Squire
Van Nostrand Reinhold Company, New York, 1978.

Review by: D. R. Deutsch
Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234

The introduction to this 113 page hard-bound publication states "This book provides a case study of the design and implementation of a relational data base management system: the Relational Inquiry and Storage System (RISS)". The text includes an adequate description of RISS, developed for the PDP 11/40 under the RSTS/E operation system at the Forest Hospital in Des Plaines, Illinois. It also provides some tutorial material about data base and relational data model concepts. It is not a study of data base software design, however. It is more precisely described as an extended documentation package rather than as a case study. Specifically, the book includes the following:

1. Explanation of the general structure of RISS
2. Description of the nature and use of the two RISS interface levels: Naive user and Application program
3. Discussion of the data storage strategy used for implementing RISS
4. Extended BASIC source code for the RISS naive level interface programs and for the functions that comprise the applications level interface
5. Discussion of and code for an application program demonstrating the use of RISS.

The code included in the book is sufficient for implementation of RISS on any PDP 11 with a compatible operating system and BASIC. A cursory review of the potential for installing the code in other hardware/software environments was made. Specifically, transporting RISS to a PDP 10 running TOPS and to an intelligent terminal with BASIC capability was considered by the reviewer. Both installations could not be made without difficulty. On the PDP 10, some of the functions of the advanced BASIC in which RISS was implemented are not available. Also, file manipulation capabilities are different for the two DEC compilers. Installation on the intelligent terminal would require almost a complete rewrite because of the lack of a multiline function capability in its BASIC language.

RISS and the supporting document should be evaluated as a software product. Viewed in this light, two criticisms become apparent. The code would be easier to understand if it contained descriptive comments; there are none in the source presented in the book. Second, the authors' apparent confusion about the purpose of the document causes the reader some difficulty. The introduction and initial chapters do not clearly state the purpose of the book nor do they tell the reader precisely what is contained in the subsequent chapters. Once the nature of the book is understood, however, it is interesting and potentially useful.

"Data Base Organization for Data Management"
Sakti P. Ghosh
Academic Press, Inc., New York, 1977

Reviewed by: Steven R. Deller
Computer Science Corporation
6565 Arlingbri Boulevard
Falls Church, Virginia 22046

This well organized text presents a graduate level description of mathematical concepts that form the basis for research in data base organization. The first five chapters build from basic data structure and query concepts to searching, hashing, and algebraic filing schemes. The last three chapters cover the consecutive retrieval property, organization on drum storage, and access path retrieval. The consecutive retrieval property introduced by the author, investigates filing schemes with minimum access time but without redundancy. Organization on drum storage investigates filing schemes efficient for fixed-head rotational storage devices. Access path retrieval investigates methods for constructing and searching data paths.

All subjects are treated formally, frequently with detailed mathematical proofs. Unfortunately, the descriptive text between proofs is often minimal, making it hard to follow the lines of development. Contrary to the author's stated intention, I found several sections no more readable than many research papers. One difference, which should not be underrated, is that the subjects are brought together into a structured volume.

To the detriment of the book's subject, no description of current research in formal data base modeling is presented. Formal approaches to data base reliability and security are not discussed, even though these are topics of considerable current and future interest. No mention is made of approaches to data base organization which support multiprocessors, paged virtual systems, intelligent memory, or serial shift-registers (as used in bubble memories).

Used as a graduate school text, this book will provide an orderly sequence of subjects for teaching well-established, formal data base analysis concepts. With the addition of some research papers on the more current topics mentioned, a good course in data base analysis skills could be developed.

CALL FOR ABSTRACTS
OF REPORTS, WORKING PAPERS AND THESES ON
DATA BASE ENGINEERING

Interesting research results and important contributions are often documented in the form of reports, working papers, theses, technical memoranda and similar documents. Many of these contributions remain unpublished or appear in journals or in conference proceedings at a much later date. For the purpose of fostering a prompt dissemination of technical information, Data Base Engineering will publish the abstracts of unpublished documents dealing with the various aspects of data base technology. The purpose of this service is to publicize the availability of these technical documents not to distribute them. Interested readers must address their request for copies directly to the author(s) or to the issuing organization.

We thereby solicit submission of abstracts for documents conforming to these guidelines:

- The document topic must fall within the scope of interest of Data Base Engineering (see inside cover).
- The document must not have previously appeared in books, journals, or conference proceedings
- A reasonable number of copies of the document are available and will be mailed on request.

A photo-ready copy containing title, author's name and affiliation, abstract and complete ordering information should be fit into the upper half of a 6.5" × 10." page and mailed to the Editor of DBE, Dr. Jane Liu, Department of Computer Science, University of Illinois, Urbana, IL, 61801. Publication of the abstracts is contingent upon meeting the previous guidelines and upon the available page space.

★PLEASE DUPLICATE AND POST★

ERRATA for DATABASE DESIGN

Gio Wiederhold and Ramez El Masri
Stanford University; Computer Science Department
August 1978

This is a list of known errors; and their corrections; for the book Database Design by Gio Wiederhold; McGraw-Hill 1977;

There are two types of entries in the list:
1; Known errors with known corrections; Correspond to unmarked enries;
2; Suggested changes in writing style; increased explanation; etc;
Correspond to lines marked {;

The paragraph number in identifying an error is counted from the top of the page; with a paragraph continued from the previous page designated as par;0; Examples in the text are ignored in numbering paragraphs;

The symbol '-->' is used to indicate that the Correction entry should replace the Error entry;
The symbol '++>' is used to indicate that the Correction entry should be added to the Error entry;
Other instructions are placed in angle-brackets as <meta-text>;

Also;
T stands for Table
E stands for Example
Exc stands for Exercise
F stands for Figure
floor[;;;] is the floor function
ceil[;;;] is the ceiling function
<greek> is the named greek letter; or special symbol; All these symbols are defined in Appendix C

Preface

	Page	Par;	Line	Error		Correction
	xii	2	4	analysis of ;;;	-->	analysis or ;;;
{	xiii	1	1	I wish to ;;;	-->	I ;;;

Chapter 1

{	1	2	2	database concerns;	-->	databases;
{	3	3	all	++> A very large database presents particular problems in its management; since it cannot be turned off without affecting the operation and well-being of the enterprise into which it is integrated; Routine maintenance; reorganization to clean out old data; structural changes to update the database design; and even copying a consistent snapshot of the database require additional effort in the design and operation; and makes some simple methods used in databases infeasible;		
	10	T1-1	1	information data ;;;	-->	information: data ;;;
{	10	T1-1	35	C: Required ;;;	-->	C: Requirements ;;;
	12	E1-1	6	s2:DECLARE workspace(20);	-->	s2:DECLARE workspace(20); title CHAR(8);pct CHAR(2);
	12	E1-1	14	UPDATE:	-->	UPDATE;
{	12	E1-1	17-22	<indent three spaces>		

	Page	Par;	Line	Error		Correction
	13	1	1	record named ;;;	-->	record titled ;;;
{	16	1	3	in this manner ;;;	-->	with this dynamic approach ;;;
{	17	4	3	departments;	-->	departments; we write:
{	18	2	4	cause again	-->	also cause
{	18	3	6	not yet uncommon;	-->	still common;
	20	F1-8		Trans; ;;;ssing time	-->	Trans; ;;;ssing ------> time
{	21	1	5	execution and ;;;	-->	execution; but ;;;
{	22	4	4;5	applications <to end>	-->	support systems; tend not to use structured databases to the same extent as the applications listed in the table above; so that this book is less relevant for databases containing primarily text;
	24	2	9	for successful ;;;	-->	for a successful ;;;
	24	4	3	Shannon; ;;;	-->	Shannon ;;;
	25	1	6	Salton62	-->	Salton68

Chapter 2

{	28	T2-1	12	33;7M	-->	33;7M[i]
{	28	T2-1	27	++> i The capacity is based on a practical blocksize and is hence less than the maximum blocksize of a single huge block on tape		
	29	1	9	foldout page	-->	inside cover
	31	3	10	single-platter disk unit which has two heads --> one platter of a disk unit with its two heads		
	31	3	11	disk;	-->	disks;
{	31	3	12	IBM 2314	-->	IBM 2314 type
	32	F2-3		single-platter disk drive --> two-platter disk drive (only one platter is shown)		
	34	2	5	with align ;;;	-->	which align ;;;
{	34	5	4	microfilmlike ;;;	-->	microfilm-like ;;;
{	35	F2-5		< delete the line in switch to drum >		
	37	3	2	undertermined	-->	undetermined
{	37	F2-6		IBM 2314 Mod 1	-->	IBM 2314 model 1
	38	2	12	j(j-1)	-->	jxj
	38	1st formula		j(j-1)	-->	j^2
	38	2nd formula		$j - j^2$	-->	j^2
	38	F2-7		< switch arrow identifiers (r;1) for cylinders 99;100;101;198;199 >		
{	39	0	1	by inspection;	-->	by inspection of Fig 2-7
	39	F2-8		0;0101	-->	0;00995
				0;00005076	-->	0;00005
	40	4	3;4	The time required can be subtracted from --> The apparent seek time then includes		
	40	5	8	class 2	-->	class 5
{	43	4	3	the table ;;;	-->	Table 2-1 ;;;
	45	F2-10	fixed	< records R1 to R8 should have the same size >		
	47	1	7	space or unspanned	-->	space for unspanned
	48	0	2	bulk transfer rate	-->	bulk transfer rate t'
	52	T2-3		< move table before section 2-3-3 >		
{	59	2	6	of data;	++>	so that all blocks can be read as they appear;

Page	Par; Line	Error	Correction
{ 70	0 3	criteria; ++>	We ignore the complexity of programming the alternatives;
70	4 2	foldout page of App; -->	inside cover or App;
75	F3-1	148 g -->	148;
77	5 3	which defeat ;;; -->	which defy ;;;
78	0 4	two -->	some
78	0 5	= and #; ;;; -->	= ; and ; ;;;
78	0 5	also have to be stored -->	are stored to mark
81	4 7	using n log;;; -->	using order n log;;;
84	2 7	; ++> which specify field lengths for data items;	
84	E3-4 14	WRITE ;;; -->	WRITE FILE ;;;
84	4 3	consequential -->	cosequential
86	F3-5 right	block = 1/2*n*R/B ++>	; delta = block/2
	line from	left <add block with> ++>	delta = delta/2
		record > 'V' -->	First record > 'V'
		block = block/2 -->	block = block-delta
		block = block+block/2 -->	block = block+delta
	F3-5 left	i = 1/2*B/R ++>	; d = i/2
	line from	left <add block with> ++>	d = d/2
		i = 1/2*i -->	i = i-d
		i = i+1/2*j -->	i = i+d
97	F3-9	151- (2 times)-->	156-
		676- (4 times)-->	704-
100	4 4	<formula> -->	i1 = ceil[n/floor[B/R]]
100	4 9	<formula> -->	$i_{level} = ceil[i_{level-1}/y]$
101	E3-8 10	ceil[;;;]=1099 -->	floor[;;;]=1099
102	1 1	3000/1009 -->	3000/1090
103	0 16	i-1 -->	i=1
105	4 2	the probabilities -->	the appropriate combination
116	4 4	half as much -->	half as much space
123	E3-10 7	-6678 giving 339 -->	-6678 giving 178
	8	-6784 392 -->	-6392 392
	9	-0373 186 -->	-0373 373
	10	-4785 392 -->	-4892 392
125	2 6	arg1|arg2 -->	arg1 <or> arg2 <use V symbol>
		& (2 times)-->	<and> <use inv; V symbol>
		ARG1 -->	arg1
{ 127	0 2	; ++> The third method uses a separate overflow file;	
128	0 2	in the block -->	in the bucket
	0 3	B/R -->	floor[B/R]
{ 129-130		< rearrange to keep Fig; 3-22 with the subsection " Prob; of Success " by placing both section and Figure prior to section " Overflow Areas " >	
129	4 12	m/m_n -->	m/m^n
130	3 3	a(V+P) (2 times)-->	(aV+P)
131	0 1	a(V+P) -->	(aV+P)
133	2 7	Fig; 3-18 -->	Fig; 3-10
134	3 2	slot ;;; computed -->	record space at the computed address
	4 3	T_F -->	T_I
137	4 4;5	estimate can be based on-->	estimate of the effective file density is
	4 5	This --> Loading a direct file record by record	
142	1 2	a choice among making-->	making a choice among
146	0 3	files ++> ; see however Sections 4;2;5 and 4;4;1;	
148	3 5	y = ;;; = ;;; -->	$y = ;;; = ;;; = n^{1/a_F}$

Page	Par; Line	Error	Correction
161	T4-1 Direct	;;; F <last entry>-->	;;; E
168	loop-2 3	base out -->	base_out
	a-loop 6	; -->	;
170	1 3	below -->	on page 168
{ 173;174	E 4-1;4-2;4-3	< should be reset to align characters >	
173	3 3	three -->	two
174	0 2	; ++> The amount of low-order abbreviation varies and is given after the minus-sign;	
	E4-3	< should read >	118-2=p;1-1=q;9-2=r;213-1=s;201=t;5=u;9=v;1-1=w;384=x;
{ 175	2 1	Example 4-3 -->	Figure 4-6
{	F4-8	< add example > ++>	Find arg = 12205 Begin at entry 1: 1: arg>key -> go to entry 6 6: arg<key -> go to entry 8 8: arg=key -> found; it's u!
176	0 3	generally be than a -->	often be greater than for a
{	F4-9	< add example> ++>	Find arg = 12205 Begin at entry 1: 1: 5th digit is 5>0 -> go to entry 6 6: 5th digit is 5>4 -> go to entry 8 8: 5th digit is 5 -> found; it's u!
177	2 1-4	<replace to> fewer entries ;;; --> The space for a binary trie entry is composed of the bit position for the key and two internal references; For instance a 5 character key (40 bits) and up to 500 entries per block will require only $CEIL[\log_2 40] + 2\ CEIL[\log_2 500] = 24$ bits or 3 bytes; Trie entries are hence apt to be smaller than index entries ;;;	
180	F4-11	<delete right border of figure; keep line between curves>	
		T Fix -->	T_{Fix}
{	0 5	++> The effect of the waste W due to packing blocks into tracks has been ignored in the above analysis;	
	1 1	Multiway -->	Multi-attribute
181	F4-12	Multiway -->	Multi-attribute
	2 1	2^{a-1} -->	2^a-1
{	F4-13 1	St -> St||Pr -> St||Pr||Be ++>	-> St||Pr||Be||Lo
{	5	St||Lo -> St||Pr||Be||Lo -->	St||Lo
{	6	1 4 1 -->	2 3 1
{ 182	0 1	triangle) -->	triangle for a)
{	3 8	Trains are then no longer in sequence; --> Serial trains are then no longer in physical sequence;	
{ 187	5 4	index is (twice)-->	index record is
	6 1	Index Block Structure-->	Index Entry Blocking
	7 4	In order read -->	In order to read

Page	Par;	Line	Error		Correction
{ 311	0	3	Arrival-time	-->	Inter-arrival-time
	0	2	unit;	++>	Requests will depart from the system at a throughput-rate <rho> = uf<mu>;
	2	3	and <mu> are	-->	and <rho> are
312	F6-13		<add formula on side>++>		$E_m = \frac{(\lambda m)(\lambda m\, t)^{m-1}\, e^{-\lambda m\, t}}{(m-1)!}$
			<to bottom add>	++>	t=inter-arrival time=1/<lambda>
313	1	5	queue	-->	average queue
314	0	9	3;162(80;7)=325s	-->	sqrt(10)(42;8)=205ms
	1	0	<move Figure 6-15 above section "Mix of Distributions" >		
	1	F6-15	<explain on the side>++>		The waiting time is less than w(p)/wbar times the average waiting time in p % of the cases;
315	0	3	pc_h(:::	-->	pc_h^2(:::
317	1	37-39	END;	-->	END;
			END;		END;
			END;		END;
319	F6-18		C_0;<sigma>	-->	C_o;o
320	0	10	Then	++>	the cost coefficient associated with fetching overflows each day is
	0	11	$C_o = (r + btt) / n$	++>	L_F cost
{	0	11	<insert line>	++>	and with L_I overflows
328	Exc15	1	6-1-2	-->	6-1-5
		1;2	compute the median and the mode -->		use the mean and the mode to estimate the median

Chapter 7

330	3	8	All three files	-->	Two of the files
332	1	6	interesting	-->	interesting;
334	1	7	tuples have	-->	tuples represent values for
335	0	3	<subset>	-->	<Set or subset>
	6	3	ordered and complete	-->	appear in the same order
342	F7-9		Phillip	-->	Phillip 5
345	1	5	reference	-->	referenced
347	F7-15		relation;	-->	relation of order 2;
348	0	2	of the owning	-->	of an owning
349	2	4	An entity relation type	++>	(primary or referenced)
353	0	1;4;11	that T (three times)-->		that each tuple T
{	1	14	$n_5 = n_3 - n_2$	++>	$= n_1 - n_4$
355	2	12	suppliers	-->	supply
357	3	4	$d_Q = n_R + 1$	-->	$d_Q = d_R + 1$
{ 360	1	6	through all tuples	-->	through all the tuples
{	2	9	as ruling part	-->	as the ruling part
{ 363	2	3	<move seg- to next line for readability >		
364	1	5	added as ruling part to the relation -->		used as ruling part for the all_children relation
	1	10	<J>(children;;;;	-->	all_children = <J>(children;;;;
374	1	6	PACKED Array	-->	PACKED ARRAY
376	0	1	extendible	-->	extensible
378	2	2	command is shown	-->	command on a terminal is shown
	2	2	Example 8-4 ++>		The PL/1 compiler used here has VARYING as the default for characterstrings;

Chapter 8

Page	Par;	Line	Error		Correction
379	E8-4	36	data_record	-->	datarecord
		51	cp+1	-->	cp+2
{		55	;'RECORDS	-->	;' RECORDS
{		55	(I(4);A(18));	-->	(I(4);A);
{ 383	1	2	an element	-->	a data element
383	1	13	0 <letter>	-->	0 <zero>
384	E8-6	11	FROM 1 TO 25	-->	FROM 0 TO 25
{ 385	0	1	use an entity	-->	use instead of a nest a referenced entity
387	E8-9	11	DO WHILE DEFINED ;;;	-->	DO WHILE DEFINED ;;; GET (employee);
		12	= department	-->	= department;name
391	3	5	record definition	-->	record definition at runtime
395	F8-14		number table: 15 | 1 |	-->	15 | 0 |
{ 397	F8-15		interfaces;(Based on Steel)	-->	interfaces;

Chapter 9

413	2	4	employee;name ;;;	-->	(employee;name ;;;)
	2	11	employee;birth ;;;	-->	(employee;birth ;;;)
	2	17	current_year ;;; job	-->	(current_year ;;; job)
415	0	2;3;4	<subset>(three times)	-->	<member of> <add bar in center>
	E9-2	11;12	<subset> (two times)	-->	<member of> <add bar in center>
	E9-2	16	7500381	-->	750381
416	4	1	example can	-->	example (Figure 7-6) can
	E9-3	1	(e1;e2)	-->	e1;e2
417	2	6	wlist	-->	w_list
421	1	3	age_c > 6	-->	age_c => 6
	3	10	operand type (;;;	-->	operand type selected from (;;;
422	0	4	=_(m_attr_1)	-->	=<not>m_attr_1
423	E9-5	4;7	(type OF p_id) (2x)	-->	(type;p_id)
424	F9-4		Sedan (9 times)	-->	Body
			Conv; (13 times)	-->	Fender
426	5;6		<Start paragraph 6 at "Retrieval of records"		instead of at "The results" >
427	E9-6	3;7	employees;name (2x)	-->	employee;name
		4	employees;children	-->	employee;children
		8	employee IN	-->	employees IN
428	0	8;9	employee IN (2x)	-->	employees IN
{ 3;8;13;14;20;28			& (6 times)	-->	<and> <inv; V symbol>
429	F9-6	children	< Move "B" to the right from "Hare" to "Hart" >		
431	E9-8	12	children OF employee	-->	children OF employees
431	E9-8	21	superv; OF employee	-->	superv; OF employees
435	5	5	Fig; 9-11;	-->	Fig; 9-10;
436	F9-11		Political sceence	-->	Political science
438	F9-13		<Indent "BY DATA-BASE-KEY" >		
440	E9-9	5	CALC-KEY in	-->	CALC-KEY IN
441	E9-10		FIND employee;	-->	FIND employee; GET employee;
447	2	2	PHILLIPS	-->	PHILIPS
451	F9-19	arrow	skill--*auto_section	-->	parts--*auto_section
{ 452	2	3	sequence	-->	sequence of a specified type
456	F9-21	HIDAM	$Tree_j$ has has	-->	$Tree_j$ has had
459	1	10	regis	-->	aegis
	Exc7	4	&	-->	<and> <inv; V symbol>

Page	Par;	Line	Error		Correction
{ 188	3	9	"9"	++>	is the last printable character
189	1	5	greater	-->	smaller
{ 190	E4-2	4	End value first train	-->	first train BAA to BEZ
		5	End value second train	-->	second train BFA to BIZ
		6	End value third train	-->	third train BJA to BRESZ
		7	End value fourth train	-->	fourth train BRETA to BRETZ
		8	End value fifth train	-->	fifth train BREUA to BROZ
		9	End value sixth train	-->	sixth train BRPA to BRZ
		10	End value seventh train	-->	seventh train BSA to BUZ
		11	End value eighth train	-->	eighth train BVA to CALHZ
		12	End value ninth train	-->	ninth train CALIA to CALZ
		13	End value tenth train	-->	tenth train CAMA to CZ
{	0	1	no entry	-->	no further entry
192	9	1	ENDRQ	-->	ENDREQ
	10	3	<add note>	++>	VSAM now provides support for secondary indexes; approximating a multi-indexed file;
{ 193	3	all	<replace algorithm>	-->	proceed: go to the left of the block below; loop: take the next field; if it is a subtree then "proceed" ; if it is a record then do; process it; "loop" ; end; if there is no field then do; if there is a superior block then do; go there; "loop" ; end; else "done" ; end; done: ;;;
{ 194	F4-19		<add on side of fig>	++>	level number x=3 2 1
195	1	2	(B-P) / (R+P)	-->	floor[(B-P) / (R+P)]
	2	2	ratio	-->	ratio - 1
	3	3	skillcode;	-->	6 character skillcode as key;
196	0	2	y = ceil[;;;] +	-->	y = floor[;;;] +
	1	3	y = ceil[;;;]	-->	y = floor[;;;]
	2	3	ceil[B/(R+P)]	-->	floor[B/(R+P)]
	2	5	200;000/5	-->	20;000/5
203	0	11-13	<indent level 6 entries in DECLARE>		
	0	14	n1 patients	-->	n1 patients with 2 datafields and a subtree
	0	15	drugs	++>	with 3 datafields;
	2	4	500(2) ;;; = 31;501;001	-->	500(3) ;;; = 31;051;501
	2	6	problems is 2;3	-->	problems per day is 2;3
	2	7	450(2) ;;; = 42;589	-->	450(3) ;;; = 43;039
{ 204	4	1	We are using here as key	-->	As shown in Fig;4-28; the key is
{ 208	2	1	Global names	-->	Names of global variables
217	F4-34		idenification	-->	identification
219	5	1;11	Multiway (two times)	-->	Multi-attribute
220	1	1;6	Multiway (two times)	-->	Multi-attribute
	2	4	$2^{a'-1}$	-->	$2^{a'}-1$
221	F4-37		Multiway	-->	Multi-attribute
	F4-38		22 buckets	-->	23 buckets
			queries;	++>	One query (Pr) has two values;
222	F4-39		Multiway	-->	Multi-attribute
229	1	2	structures);	-->	structures)
235	3	3	Reitnam	-->	Reitman

Page	Par;	Line	Error		Correction
251	3	5	deliver 50 ladders	-->	deliver 501 adders
258	2	1	operation:	-->	operation four component types:
259	F5-8	5 CPU	btt box	++>	sparse hashmarks (// // //)
261	3	10	fus = ;;; + nc	-->	fus = ;;; + np
263	2	15	poss ch	-->	$poss_{ch}$
	.2	16	poss = #devices	-->	poss = $\#devices^{\#pr}$
272	3	8	problems	-->	problem
275	1	11	Migamoto	-->	Miyamoto
276	0	1	Kerr	-->	King
Chapter 6					
286	1	10	ifr = CEIL	-->	ifr = 1 + FLOOR
290	4	2	standard deviation	-->	variance (<sigma>^{2})
291	0	1	532	-->	550
	0	3	532/400=1;33	-->	550/400=1;38
	P-2	3	n(EXP(mean)	-->	n(EXP(-mean)
		4	left=n;	-->	left=n-expfr(0);
		6	;;;);	-->	;;;);/{ next expfr --> 1 {/
		8	expf(ov)	-->	expfr(ov)
292	2	10	samples; with	-->	samples with
{	2				<Move Table 6-3 in front of section "Collision Probability" >
	F6-6		80%	-->	90%
293	0	3	n^q/m	-->	$(n/m)^q$
	0	9;11	P(j) (two times)	-->	$P_b(j)$
{	0	10	AL;	-->	AL
{ 294	0	4	so that	++>	using the Poisson distribution for P_b
	0	9	cumulative freq;dist;	-->	sum of possible record counts times their probabilities
	3	2	distribution	-->	distribution so that
{ 295	0	4	sequence	-->	sequence (page 289)
	0	4	<sigma>	-->	<sigma>^{2}
{	0	5	;	++>	; we expect a standard deviation of o' of
	0	6	o =;;;	-->	<sigma> = sqrt(n)sqrt(n/m) =;;;
{	0	6	<insert line>	++>	This <sigma> applies of course also to the complementary primary file;
	0	7	the example used	-->	Example 3-11
{ 296	6	3	and hence	++>	the probability that x exceeds the maximum of the range; i;e; mean + c <sigma>; is
297	0	2	0;99 of the records	-->	in 0;99 of the cases the records
{ 304	2	1	delay	-->	mean delay
{	2	2	as	-->	and approximates
{ 305	0	1	a total	-->	an average total
	0	2	equals	++>	depends on their delays w' as
	1	3	Rq = <lambda> w	-->	Rq = <lambda> w'
{	1	7	distribution	-->	distribution of arrivals
{ 307	2	2	for a SCAN	-->	using SCAN
309	2	4	average	-->	maximum
	2	6	1;5(200) = 300	-->	1;5(200)/2 = 150
	3	3	+ 2 E f	-->	+ 2 r E

Chapter 10

	Page	Par;	Line	Error		Correction
	462	3	3	structure	-->	structures
{	476	E10-1	year_hired<in scale colomn>	++>	1	
	480	F10-9	arrow from "INF;Retr;Sys;" to 'Users" should have two half heads			

Chapter 11

	491	0	4	r=4	-->	r=5
	505	7	3	Petersen	-->	Peterson

Chapter 12

	528	1	3	loc._name	-->	lock_name

Chapter 13

{	551	6	all	++> Blocking of processes and resources will not only affect performance; but unfortunate interaction sequences can delay the system forever; creating a deadlock; These problems are the subject of the next section;		

Chapter 14

	576	1	8	The vowels (;;;) are	-->	The vowels(;;;) and H;W are
	577	T14-4		Test(tokens)	-->	Text(tokens)
	580	4	8	ADCOMSUBORDCOMPHIS;;;	-->	ADCOMSUBORDCOAMPHIS;;;
{	581	3	8	compression	-->	encoding
	582	1	14	fr(JZC)	-->	fr(JZQ)
	585	1	11	<add a box with:>	++>	80-20 rule '80% of all accesses are made to 20% of the records in a file;'
{	588	2	7	points out	-->	also points out

Appendix A

	603	0	42;43	multiattribute: multiway	-->	multiway: multi-attribute

Appendix B

	610		38	DBMS	-->	FMS
			47;49	Phillips	-->	Philips
	612	0	6	trf	-->	tnf
		1	8	<insert>	++>	Dev Developmental

Bibliography

	626	Floyd 74	1974;	-->	1972; pp; 103-109;	
	632	Knuth 74	1974	-->	1973	

Index

	647		35	size;41;;;;615	-->	size;41;;;;616
{	651			<insert>	++>	graphics;25;268;279;282;287;466
{	653			<insert>	++>	mark;46;78
				<move> entries for Multiway to Multi-attribute		
	656		17	Seek time;;;;615	-->	Seek time;;;;616
{	658			<insert>	++>	80-20 rule;584;588

;;;; END ;;;;

TC/DBE MEMBERSHIP
APPLICATION/RENEWAL FORM

To become a member of the TC/DBE and be on the mailing list for the Data Base Engineering Bulletin, please return this form or a copy of it to:

IEEE TC/DBE
Department of Computer Science
University of Illinois, Urbana, IL 61801

NAME _____
(please print)

INSTITUTION _____

ADDRESS _____

_____

Call for Papers

Very Large Data Bases

Rio de Janeiro, Brazil
October 3-5, 1979

Fifth International Conference on Very Large Data Bases

This fifth conference in the series is intended to identify and encourage research, development and applications of data base systems. Its objectives are the promotion of an understanding of current research, the furthering of exchanges of information and experience gained in the design, construction and use of data bases; and the provision of a forum for the discussion of future research and development. In addition, a special focus this year will be aimed at applications of data bases in developing countries, and on mini- and microcomputer data base management systems.

Papers of up to 5,000 words are sought on topics which include but are not limited to the following subjects:

Data Base Design
System Analysis
Requirement Specification
Logical Data Base Design
Data and Program Conversion

Data Base Software Engineering
Formal Specification
Design Methodology
Development Tools
Verification

Data Base Machines
Microprocessors
Minicomputer Backend Machines
Memory Organization
Communications Protocols

Distributed Data Bases
Network Architecture
Resource Management
Concurrency Control
Data Migration
Data and File Location

Data Semantics and Modeling
Concepts
Formalism
Consistency and Equivalency

User Interface
Natural Languages
Graphical Interfaces
Design Concepts and Goals

Implementation Considerations
Query Evaluation
Integrity and Recovery
Concurrent Access
Measurement and Monitoring

Data Base Applications
Office Automation
Personal Computing
Process Control
Decision Support Systems
Technology in Developing Nations

In addition, extended abstracts (two pages) that discuss applications of data base technologies in developing nations are sought. These abstracts will be presented at special sessions addressed to the use of technology in this environment.

Since one of the goals of the conference is to further the use of data base technology in developing nations, practically-oriented survey and tutorial papers are especially welcome. It is expected that many practitioners from all over South America and the rest of the world will attend.

Where to submit papers
Send five copies of each full paper by March 5, 1979 to one of the following persons:

U.S. Program Committee Chairman
Prof. Howard L. Morgan
Dept. of Decision Sciences
The Wharton School
University of Pennsylvania
Philadelphia, PA 19104
U.S.A.

Latin American Program Chairman
Prof. Antonio L. Furtado
Departamento de Informatica
Pontificia Universidade Catolica–RJ
Ru Marques de Sao Vicente, 209–Gavea
Rio de Janeiro, RJ
BRASIL

Co-sponsors
Association for Computing Machinery (ACM) –SIGMOD, SIGBDP, SIGIR
International Federation for Information Processing
Institute of Electrical and Electronics Engineering, Inc.
SUCESU–Share Users Association, Brazil
CNPQ–National Council of Research, Brazil
CAPRE– Government Agency for Data Processing, Brazil
MEC–Ministry of Education, Brazil
FINEP–Government Agency for Project Development and Finance, Brazil

Important Dates

March 5, 1979	Papers due
May 14, 1979	Authors notified of acceptance of the papers
June 11, 1979	Final revisions of papers due
October 3-5, 1979	Conference meets in Rio de Janeiro

Publication
A Conference Proceedings will be published and selected conference papers will be published in the ACM Transactions on Database Systems (TODS) and the IEEE Transactions on Software Engineering (TOSE). All papers accepted for presentation will be available to the participants at the conference.

More Information and Travel Grants
Suggestions for panel or tutorial sessions should be directed to one of the two program chairmen. Requests for other information should be directed to one of the conference chairmen. It is hoped that some partial travel grants will become available to help support the travel of technical program participants.

General Conference Chairman
Mr. R.J. Líbero
IBM do Brasil
Caixa Postal, 1830–ZC–00
Rio de Janeiro–RJ–20.000
Brasil

Latin American Conference Chairman
Prof. V.W. Setzer
Departamento de Matemática Applicada
Instituto de Matemática e Estatística
Universidade de São Paulo
01000 São Paulo–SP
Brasil

U.S. Conference Chairman
Prof. Stanley Y.W. Su
Dept. of Computer and Information Sciences
University of Florida
512 Weil Hall
Gainesville, FL 32611
U.S.A.