

Inhalt

Vorwort

1	Einführung	1
1.1	Programmiersprachen in Generationen	1
1.1.1	Sprachen der 1. und 2. Generation	1
1.1.2	Sprachen der 3. Generation	2
1.1.3	Wichtige Merkmale von Sprachen der 3. Generation	4
1.2	Die Wurzeln der 4. und 5. Generation	8
1.3	Vom Rechner zum Medium	11
2	Sprachen als Werkzeuge der 4. Generation	13
2.1	Das Relationenmodell stellt neue Weichen	18
2.1.1	Die neue Idee des E.F. Codd	18
2.1.2	Das relationale Datenmodell auf einen Blick	26
2.1.2.1	Der strukturelle Aspekt des Datenmodells	26
2.1.2.2	Der manipulative Aspekt des Datenmodells	27
2.1.2.3	Der Integritätsaspekt des Datenmodells	32
2.1.3	Das Relationenmodell als natürliche Art der Informationsdarstellung	35
2.1.3.1	Die fünf Normalformen	35
2.1.3.2	Zur Datenmodellierung – konzeptionelles und logisches Datenmodell	48
2.1.4	Von der Abfragesprache zur Datenbanksprache	61
2.1.4.1	Structured Query Language (SQL) als Standard?	62
2.1.4.2	Die Mängel von SQL	76
2.1.4.3	NATURAL	88
2.1.4.4	Formuliersprachen	99
2.2	Das Relationenmodell und die PC-Welt	100
2.2.1	Tabulare PC-Datenbanksysteme und ihre Sprachen	101
2.2.2	Von der Kalkulation zu den integrierten Paketen	111
2.3	Industrielle Werkzeuge der 4. Generation	116
2.3.1	Kriterien	116
2.3.2	Neue Werkzeuge ermöglichen neue Vorgehensweisen	119
2.3.3	Ein Fall aus der Praxis	122
2.3.4	Ausblicke	126
3	Sprachen als Werkzeuge einer 5. Generation	129
3.1	Programmieren in Logik	132
3.1.1	Einführung	132
3.1.2	Die formale Sprache der Prädikatenlogik (1. Stufe)	134
3.1.3	Problemlösen mit logischer Programmierung	138
3.1.3.1	Einführung	138

3.1.3.2	Algorithmus = Logik + Steuerung	142
3.1.4	PROLOG	162
3.1.4.1	Grundbegriffe von PROLOG	162
3.1.4.2	Auch in PROLOG gilt: Algorithmus = Logik + Steuerung ...	168
3.1.5	Eine Expertensystem-Shell auf PROLOG-Basis: TWAICE	186
3.2	„Lisp-Sprachen“	189
3.2.1	Universalität durch Einfachheit	190
3.2.2	Programmieren in LISP	199
3.2.3	Der Appell des John Backus	215
3.3	Objektorientierte Sprachen	228
3.3.1	Der objektorientierte Ansatz	228
3.3.2	Das Beispiel SMALLTALK	234
3.3.3	Entwurf objektorientierter Softwaresysteme	242
4	Ausblicke auf mögliche Entwicklungen	253
4.1	Die Synthese von prozeduralen und nichtprozeduralen Sprach- konstrukten	253
4.1.1	Von MODULA-2 zu MODULA/R	254
4.1.2	GAMBIT – das LIDAS-Datenbank-Definitionssystem	263
4.2	Sprachen als Werkzeug des Software-Ingenieurs	270
4.2.1	Evolutionäres Entwickeln nach Tom Gilb	271
4.2.2	Evolutionäres Entwickeln mit Sprachen der fünften Generation .	279
4.3	Petris Fazit: Nur Verbesserung von schon Bestehendem	287
4.3.1	Einführung in die Sprache OCCAM	287
4.3.2	OCCAM als Sprache für den Systementwurf	295
4.4	Gibt es Ansätze in der Informatik, die grundsätzlich Neues bieten?	297
4.4.1	Die Netztheorie	297
4.4.2	Der Rechner als Medium	302
	Anhang 1: Erläuterungen	311
	Anhang 2: Personenverzeichnis	319
	Anhang 3: Literaturverzeichnis	323
	Sachindex	337
	Personenindex	339