

Eberhard Sturm

The New PL/I

From Enterprise Architecture to IT Governance

by Klaus D. Niemann

Understanding MP3

by Martin Ruckert

Process Modeling with ARIS

by Heinrich Seidlmeier

Microsoft Dynamics NAV

by Paul M. Diffenderfer and Samir El-Assal

Eberhard Sturm

The New PL/I

... for PC, Workstation and Mainframe

With 80 illustrations



Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie;
detailed bibliographic data are available in the Internet at <http://dnb.d-nb.de>.

This book is the expanded and improved 7th edition of the German book "Das neue PL/I"
(© Vieweg+Teubner | GWV Fachverlage GmbH, Wiesbaden 2008) now offered to the English speaking
audience by the original author.

1st Edition 2009

All rights reserved

© Vieweg+Teubner | GWV Fachverlage GmbH, Wiesbaden 2009

Editorial Office: Sybille Thelen | Walburga Himmel

Vieweg+Teubner is part of the specialist publishing group Springer Science+Business Media.
www.viewegteubner.de



No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the copyright holder.

Registered and/or industrial names, trade names, trade descriptions etc. cited in this publication are part of the law for trade-mark protection and may not be used free in any form or by any means even if this is not specifically marked.

Cover design: Künkellopka Medienentwicklung, Heidelberg

Printing company: Krips b.v., Meppel

Printed on acid-free paper

Printed in the Netherlands

ISBN 978-3-8348-0726-7

Preface

The new Workstation Compiler from the IBM Corporation, originally available only for the OS/2 Operating System, has since made its way from Windows and AIX (IBM's version of Unix) to the mainframe z/OS Operating System, including the version Unix System Services.

This book claims to be a basis for certification as an “IBM PL/I Certified Programmer/Developer”. On IBM computers, there is now decimal floating-point hardware available, which PL/I supports via the well-known DECIMAL FLOAT attribute and a new compiler option. So in the 7th edition, the section on floating-point arithmetic has been substantially rewritten.

IBM has worked on a set of questions which enables one to become certified as a PL/I Certified Programmer as well as a PL/I Certified Developer. Since the author was honored to help in developing the questionnaire, I have modified the text to reflect this test.

A Certified Programmer is expected to have only two to three years of experience with IBM PL/I. The expectation of the PL/I Certified Developer is that he or she will have had five to six years experience with the programming language. I would go so far as to imply that anyone who has read and understood this book should not have any problems answering the PL/I related questions of the certification test.

This book offers a modern introduction to “Programming Language Number One”. Its aim is to provide beginners with material for self-study as well as to provide professionals with new ideas, particularly on the basis of its comprehensive presentation of the language. If an example should at first appear unintelligible to an experienced PL/I programmer, I hope that from its explanation he or she can pick up something new in the language.

You should expect no theoretical discussions of algorithms or structure diagrams, but instead a practical introduction which should allow you to solve real problems with the help of PL/I in a straightforward way. In contrast to a reference manual, I will always advise you which usage of the language is “good” and which is “bad”.

This book would not exist without the Internet. It's obvious that I could not inflict an English reader to read a book written by a non-native speaker. So I wrote a posting to the PL/I mailing list asking for proofreaders. It's a great honor for me to thank Richard Barrow, Francis Byrne, Peter Elderon, Peter Flass, John Gilmore, Tom Linden, Ray Mullins and Robin Vowels for undertaking the task to correct and improve the English created by a German who never lived more than three weeks in succession in an English speaking country.

An unexpected but inevitable side effect happened: many suggestions helped even improve the German version of this book. If you nevertheless find errors, then this is probably my fault when incorporating the proofreaders' comments into the final text.

All examples and the PARSE macro mentioned in the next to last section (similar to the REXX statement) can be found on the web under:

<http://www.uni-muenster.de/ZIV/Mitarbeiter/EberhardSturm.html>

Contents

Preface.....	V
Introduction.....	1
1. Elementary PL/I.....	3
1.1 The programming environment.....	3
1.1.1 Highest instance – the operating system.....	3
1.1.2 How are things – program and compiler.....	3
1.2 Data attributes.....	5
1.2.1 In central place – main storage	5
1.2.2 A whole thing – fixed-point numbers.....	6
1.2.3 Going to pieces – floating-point numbers	9
1.2.4 A language of character – character strings	11
1.2.5 It doesn't get any smaller – bits.....	13
1.2.6 Everything with everything – operators	15
1.3 Loops	17
1.3.1 Ask first – the WHILE loop	17
1.3.2 Shoot first – the UNTIL loop	18
1.3.3 Upward and downward – the counting loop	19
1.4 Input and output.....	20
1.4.1 There is something to get – the GET statement	20
1.4.2 Nothing there any more – the ON statement	22
1.5 Distinction of cases.....	24
1.5.1 Either, or – the IF statement	24
1.5.2 For each individual case – the SELECT group	26
2. Extending the basics.....	31
2.1 Input and output of a character stream	31
2.1.1 Not to count with – the FILE attribute.....	31
2.1.2 Self-determination – EDIT-directed input/output.....	33
2.1.3 A language in itself – data formats.....	36
2.2 The general loop	40
2.2.1 Endlessly – LOOP and accessories	40
2.2.2 The whole truth – DO generally	42
2.3 Arrays	45
2.3.1 A thousand or one variable – working with arrays	45
2.3.2 In convoy – array operations	48
2.3.3 Not only for mathematicians – several dimensions.....	49
2.3.4 The last run fastest – the INITIAL attribute	51
2.4 Structures.....	52
2.4.1 Mind the hierarchy – working with structures	52
2.4.2 Why not – array of structures	56
2.4.3 That's the limit – multiple declarations	57
2.5 Manipulation of character strings	59
2.5.1 Two would do – SUBSTR and LENGTH	59
2.5.2 Where and how many times – INDEX and TALLY	61
2.5.3 Hocus-pocus – TRANSLATE	62
2.5.4 Forward and backwards – VERIFY(R) and SEARCH(R).....	64
2.5.5 What you will – more functions	67
2.5.6 Self-made – PICTURE character strings.....	68

2.5.7 Without detour – STRING instead of FILE.....	69
2.6 Arithmetic	69
2.6.1 Having a different base – the FIXED attribute	70
2.6.2 Disappearingly small – the FLOAT attribute	74
2.6.2.1 Floating-point since time immemorial	75
2.6.2.2 Floating-point binary.....	77
2.6.2.3 Floating-point decimal.....	78
2.6.3 Arithmetic means – rules and pitfalls.....	81
2.6.3.1 Mixed operations.....	81
2.6.3.2 FLOAT operations.....	82
2.6.3.3 FIXED operations in the ANSI standard	82
2.6.3.4 FIXED operations in the IBM standard.....	83
2.6.3.5 Built-in functions.....	84
2.6.3.6 The default concept	86
2.6.4 Janus-faced – PICTURE numbers	86
2.6.5 Character weakness – calculating with character strings.....	91
2.6.6 Nothing real – complex numbers.....	92
2.7 Manipulation of bit strings	94
2.7.1 A sister of character – bit operations	94
2.7.2 Set theory – working with bit strings	96
2.7.3 Orienting the machine – UNSPEC and others.....	97
2.8 Abstract data types.....	100
2.8.1 Types having aliases – DEFINES ALIAS	100
2.8.2 Showing colors – enumerating types.....	101
2.8.3 Strong types – DEFINE STRUCTURE	104
2.9 Time calculations	105
2.9.1 The fright of the turn of the millenium – date and time.....	105
2.9.2 A language with SECS – the Lilian format.....	106
2.9.3 Revenge of the inherited – conversion of years	108
3. Block and program structure.....	111
3.1 Scope and lifetime of variables.....	111
3.1.1 Useful overhead – the BEGIN block	111
3.1.2 More than once – the PROCEDURE block.....	112
3.1.3 Taking care – nesting of blocks.....	113
3.2 Structure of a PL/I program	115
3.2.1 A matter of order – parameters	115
3.2.2 One-way street – dummy arguments	118
3.2.3 (Not) lasting long – AUTOMATIC and STATIC	119
3.2.4 Home-made – functions	121
3.2.5 As in the Munchausen story – recursive procedures	122
3.2.6 Compile separately, execute united – external procedures.....	125
3.2.7 Packed procedures – PACKAGE	129
3.2.8 A dynamic load – FETCH, RELEASE and DLLs.....	135
3.3 Exceptional conditions.....	138
3.3.1 As a precaution – handling of conditions.....	138
3.3.2 Also Roman numerals – computational conditions	144
3.3.3 Close encounters – program testing.....	147
3.3.4 Red alert – remaining conditions	154

4. Dynamic storage management.....	159
4.1 The CONTROLLED attribute.....	159
4.1.1 Only when desired – ALLOCATE and FREE	159
4.1.2 A new construction – the stack	161
4.1.3 As general as could be – the INITIAL CALL attribute	164
4.2 The BASED attribute	165
4.2.1 Change of address – dynamic storage interpretation	165
4.2.2 Using paper and pencil – linear lists	169
4.2.3 Into botany – general lists	174
4.3 The AREA attribute.....	178
4.3.1 Good neighbourhood – use of areas	179
4.3.2 Closing holes – garbage collection.....	181
4.4 Dynamics with structure types.....	185
4.4.1 For a thorough grasp – the HANDLE attribute	185
4.4.2 New – further type functions.....	187
5. Use of files.....	191
5.1 PL/I files.....	191
5.1.1 Generalized – file values.....	191
5.1.2 Alternative and additive – file attributes	192
5.1.3 Works also automatically – opening and closing	193
5.2 Input and output of records	196
5.2.1 Variously – data sets	197
5.2.2 One after the other – CONSECUTIVE data sets	198
5.2.3 Numbered – REGIONAL (1) data sets	200
5.2.4 At discretion – VSAM data sets	203
5.2.4.1 organization (consecutive) – ESDS.....	204
5.2.4.2 organization (relative) – RRDS.....	206
5.2.4.3 organization (indexed) – KSDS.....	207
5.3 Special possibilities of input and output.....	209
5.3.1 Directly – LOCATE mode.....	209
5.3.2 Unformatted – FILEREAD and FILEWRITE.....	212
5.3.3 One after the other – PLISRTx.....	213
6. Special PL/I techniques.....	217
6.1 Array expressions.....	217
6.1.1 A straight guess – built-in array functions	217
6.1.2 Generalized – array function values.....	220
6.2 Definition of variables	221
6.2.1 We're in the cell – the UNION attribute	222
6.2.2 New names – correnspondence definition	223
6.2.3 A question of position – overlay definition	224
6.2.4 Overwhelming – iSUB definition	224
6.3 Parallel processing.....	226
6.3.1 For re-entry – the task attribute.....	227
6.3.2 Move by move – synchronization of threads	229
6.4 Program generation at compile time	234
6.4.1 As usual – basics of the macro language.....	234
6.4.2 As called for – the preprocessor procedure.....	238
6.4.3 Self-made – definition of statements of your own.....	241

7. Interfaces to the world.....	245
7.1 Low-level programming.....	245
7.1.1 C-bits – bit manipulations on numbers	245
7.1.2 Anonymous – storage manipulations.....	246
7.1.3 Internals – foreign data formats.....	249
7.1.4 Systematically – API programming.....	251
7.1.5 For long runners – checkpoint/restart.....	254
7.2 Manipulation of Wide Characters	255
7.2.1 The first attempt – the GRAPHIC attribute	256
7.2.2 The second attempt – the WIDECHAR attribute	256
7.3 Using REXX Components	258
7.3.1 Relationship – REXX calling conventions.....	259
7.3.2 Simply huge – REXX programs in PL/I variables.....	260
7.4 Utilizing Java components.....	262
7.4.1 Using the front end – PL/I sub-programs for Java.....	263
7.4.2 Without Java – Java classes for PL/I.....	266
7.5 CGI and XML.....	269
7.5.1 Classical – CGI in PL/I.....	269
7.5.2 Working to rule – interpreting XML.....	274
Appendix A: Solution ideas.....	281
Appendix B: Built-in functions/subroutines.....	285
Arithmetic.....	285
Array-handling.....	285
Buffer-management.....	285
Condition-handling.....	286
Date/time.....	286
Floating-point inquiry (constants).....	287
Floating-point manipulation.....	288
Input/Output.....	288
Integer manipulation.....	288
Mathematical.....	289
Miscellaneous.....	289
Ordinal-handling.....	290
Precision-handling.....	290
Pseudovariables.....	291
Storage control.....	291
String-handling.....	292
Subroutines.....	293
Type functions.....	294
Preprocessor.....	294
Index.....	297