

Contents

PART ONE: BACKGROUND TO CBL

1	A Short History of the Future	
1.1	Pre-History	2
	A Tale of Two Giants	3
	Educational CBL - A Case of Retarded Development	10
1.2	What is CBL?	12
1.3	Why Use CBL?	14
1.4	Some Cost Benefits	16
1.5	A Word About the Future	16
2.	Success Stories	
2.1	CBL in Schools	18
	Forest Ranger (Sinclair Spectrum)	18
	IMAGES (BBC Computer)	20
	LOGO (Many Different Computers)	21
	Basic Skills in Math for Florida High Schools (PLATO)	23
	SMILE - Mathematics and CBL (Research Machines 380-Z)	24
	SLICK! - A pollution Game (BBC Computer, Apple, 380-Z)	26
	Natural Language and the Computer (BBC Computer)	28
	MicroQUERY - Information Retrieval (Research Machines 380-Z)	30
2.2	CBL in Colleges and Universities	32
	Distillation (Illinois University, PLATO)	33
	Organic Analysis (in BASIC and FORTRAN)	35
	Discovering Physics (Open University, TERA, OASIS & Pascal)	35
2.3	CBL in Training	38
	Keyboard Trainers (Apple)	38
	Ice Making at ICI Organics Division (Apple)	39
	Plastic Film Manufacture at Mobil Chemical (Regency Systems)	41
	Processing Travellers' Cheque at American Express (Wicat)	43
	Flight Simulation at British Airways (PLATO)	47

PART TWO: PLANNING - THE KEY TO SUCCESSFUL LESSON DESIGN

3. Choosing What to do and Where to Start	
3.1 Choosing a Style	55
Slide Show	55
Testing	55
Drill and Practice	56
Tutorial	56
Command Mode	58
Numerical Simulation	60
Decision Tree	60
Games	61
Adventure Games	61
3.2 Choosing a Topic	62
3.3 Choosing Hardware and Software - First Thoughts	65
4. Drawing Up a Plan	
4.1 Why Did Henry Fail?	67
4.2 "I Want that Course ready by Tomorrow"	68
Five Good Reasons Why It Won't Be	69
4.3 A Systematic Approach	70
Analyse Requirements	71
Specify Objectives	71
Design Test Questions	72
Define Target Audience	75
Specify Resources	75
Design the Course	76
Implement the Course	77
Evaluation	77
4.4 Designing for Real - Some Practical Methods	77
Who do you Believe?	78
5. Stimulus - Response Models	79
5.1 Learning Points	81
5.2 Matrices	81
5.3 Optimisation of Learning Sequences	84
6. Component Display Theory	
6.1 The Performance - Content Matrix	86
6.2 Using the P-C Matrix	87
6.3 Task Analysis and the P-C Matrix	90
6.4 Writing Objectives the Easy Way	91

6.5	How the Theory Helps in Correct Sequencing of Lessons	92
6.6	Primary Displays	96
6.7	Secondary Displays	99
6.8	An Example	102
7.	It Works for Pigeons but...	
7.1	Arguments against the Systems Approach	104
7.2	The Deadly Combination	105
7.3	All is not Lost	105
8.	Preparing Your Plan For The Computer	
8.1	Hierarchies and CBL	108
8.2	Planning an Instructional Unit	112
	Using Design Sheets	112
	The Four Types of Frame	114
	Presentation Frames	116
	Menu Frames	116
	Question Frames	116
	Calculation Frames	120
8.3	On to the Computer!	120
8.4	A Simple Program	122

PART THREE: PILOT - As An Example Authoring Language and Apple SuperPILOT As An Authoring System

9.	Let's Start Programming!	
9.1	General Purpose Languages and CBL	126
9.2	Why Choose PILOT (and Why Not to do)?	129
	Requirements	130
	Possible Contenders	131
9.3	Which Version of PILOT?	132
	Check-List	133
	Apple SuperPILOT	133
	SuperPILOT, Pascal and UCSD	136
	Other Versions of PILOT	139
9.4	Making a Start in PILOT	139

10. PILOT Text and Graphics	
10.1 Displaying Text	142
SuperPILOT Variations	144
Viewport,ts:vl,r,t,p	145
Goto (column,row),ts:gx,y	146
Size of Type, ts:sn	146
Thickness of Type,ts:tn	147
Modes of Type,ts:mn	147
Transmit ASCII,ts:xn	147
Line Spacing,ts:ln	147
Printer Output,ts:p	148
Animation,ts:an\$,ts:wl,r,u,d and ts:dn	148
Colour Commands	149
10.2 Variables and the Type Statement	149
10.3 Remarks	151
10.4 Graphics Commands	151
Standard PILOT Graphics Commands	151
Enhancements in SuperPILOT	154
Text Viewport (g:vl,r,t,b)	154
Type on the Graphics Screen (g:t)	155
Turtle Graphics	155
Graphics Exercises	159
10.5 Graphics the Easy Way - With the SuperPILOT Graphics Editor	160
11. Principles of Answer Processing	
11.1 Getting a Response	163
Using the PR: Code for Automatic Editing	164
Over-riding Automatic Editing	164
Single Key Input	165
Extensions in SuperPILOT	165
11.2 Matching the Response (Non-Numeric)	166
Sequences (&)	167
Alternatives (!)	168
Combining & and !	168
Approximate Spellings	169
Delimiters	169
A Test Program	170
Match Jumping	171
11.3 Extracting Information from Responses	171
11.4 Matching Numeric Responses	173
11.5 Feedback Messages	174
11.6 Tricky Matching	174

12.	Keeping in Control	
12.1	Labels and the Jump Command	176
	Conditional and Unconditional Jumps	177
	Special Jump Conditions - Look, no Labels!	179
	How Many Labels?	180
12.2	Answer Counting	180
	An Example	181
12.3	Subroutines	182
	An Example 'Menu' Program	183
13.	Calculations and Logical Operations	
13.1	Numerical Calculations Using Constants and Variables	185
	Using C: In Recording Scores and in Matching Sequences	187
	Combining Compute and Graphics Statements	189
13.2	Numerical Calculations with Arrays	190
	One Dimension	190
	Two Dimension	191
	Student Scores	192
13.3	Computing With Strings	193
	Concatenation	193
	String Slicing	193
	Simulating String Arrays	194
	Editing Strings	195
	Miscellaneous String Functions	195
13.4	Logical Relations and Operations	196
13.5	An Example Program	197
14.	PILOT Files and Computer Managed Learning	
14.1	The SuperPILOT K: (Keep) Instruction	200
14.2	Do-It-Yourself CML	201
14.3	What Sort of Data Should be Stored in Files?	204
14.4	An Example: Student Registration	204
15.	Advanced Features	
15.1	The GOTO and ESCAPE Options	209
15.2	Execute Indirect (xi:)	211
	Menu Programs with xi:	211
15.3	The Link Commands (SuperPILOT)	214
	Link to a PILOT Lesson (l: and lx:)	214
	Link to a Pascal Program (lp:)	214
	Pascal Notes	214
15.4	Sound Output and the Sound Editor (SuperPILOT)	216

15.5	Video Output (SuperPILOT)	217
15.6	The Character Editor (SuperPILOT)	218
15.7	In Conclusion.....	219
16.	An Example Application: Training a Librarian	
16.1	The Training Problem	221
16.2	The Plan	222
16.3	The Program	223
16.4	The Documentation	233
	PILOT Summary	237
	SuperPILOT Summary	240

PART FOUR: BEYOND PILOT

17.	Authoring Without Coding: A Prototype Program Generator	
17.1	What Does a Program Generator Do?	245
17.2	What Does This One Do?	246
17.3	Some Deficiencies	247
17.4	A Session With The Generator	248
17.5	How It Works	251
17.6	A Pascal Data File Conversion Program	259
17.7	Conclusion - And What Next?	262
18.	Other Authoring Systems	
18.1	The PLATO System	263
	The PLATO Editors	265
	Microcomputer Versions of PLATO	268
18.2	COMBAT	269
18.3	WISE and SMART	270
	WISE	271
	SMART	271
18.4	Other Systems	272
	STAF	272
	MENTOR-2	273
	TICCIT	275
	MICROTEXT	275
	USE	276
	PASS	276
	DIALOG	277

18.5 Counting the Cost: WHICH SYSTEM IS BEST?	277
The Financial Model	278
The Program	280
The Results	282