Contents

Preface v

1 Overview 1
  1.1 Historical notes 1
  1.2 The Oberon user interface 4
  1.3 The Oberon system architecture 7
  1.4 Summary of innovations 11
  1.5 System version, implementations and applications 13

Part I User's guide 17

2 The Oberon user interface and the standard system editor 19
  2.1 Tiled display 19
  2.2 The mouse and its use 20
  2.3 The keyboard 26
  2.4 The viewer 27
  2.5 Commands 34
  2.6 Tool viewers 37
  2.7 Naming 42
  2.8 Design for user satisfaction 43

3 Using the standard editor 46
  3.1 Mouse and keyboard 47
  3.2 Multiple views and the large selection 48
  3.3 Closing viewers and saving to disk 49
  3.4 Command module Edit 51

4 File administration and system commands 53
  4.1 Commands dealing with files and the file directory 53
  4.2 Other commands exported by command module System 55
### Part I  
**Command module** Miscellaneous

### Part II  
**Reference**

#### 9 System overview  
- Programs in the traditional sense
- Oberon commands, module loading
- The event loop
- System architecture for up-calls: active objects
- The module hierarchy
- Guide to the notation used to describe the modules of the outer core

#### 10 Keyboard, mouse, network and printer  
- Module Input
- Module V24
- Module SCC
- Module Printer

#### 11 Module Files  
- Files and the file directory
- Read/write access: the rider

#### 12 Module Display  
- Bitmapped display, the display area
- Raster operations
- Display control
- The definition of Frame, FrameMsg and Handler

#### 13 Module Viewers  
- The logical display
- The viewer
13.3 Viewer messages 125
13.4 Tracks 128
13.5 The viewer manager 129
13.6 The viewer data structure 133

14 Module Texts 135
14.1 Module Fonts 138
14.2 Text and buffer 139
14.3 Reading from texts, writing to buffers 143
14.4 Text files 151

15 Module Oberon 154
15.1 Tasks and the event loop 157
15.2 Cursors 163
15.3 Command activation 165
15.4 Focus, mark and selection 166
15.5 Display management 169
15.6 Miscellaneous procedures 170
15.7 Exported system-wide resources 172

16 Module MenuViewers 174
16.1 The menu viewer 175
16.2 The modify message 177
16.3 The handler 180

17 Module TextFrames 183
17.1 The frame 185
17.2 The display manager 191
17.3 The handler and its components 195
17.4 Facilities dealing with texts 197
17.5 Opening and creating frames 199

Part III Programming guide

18 Programming commands 203
18.1 General programming rules 204
18.2 Modules and commands 206
18.3 Working with texts 207
18.4 Accessing parameters 213
18.5 Working with text viewers 218
18.6 Working with text frames 225
18.7 Working with files 227
18.8 Long running commands, background tasks 237
18.9 Rules for well-behaved commands 242
# Contents

19 Programming viewers and frames  |  244  
| 19.1 The design of a viewer class  |  246  
| 19.2 Working with the display  |  254  
| 19.3 Handler for a viewer  |  264  
| 19.4 Handler for a frame to be installed in a menu viewer  |  268  
| 19.5 Handling mouse events, the track message  |  277  
| 19.6 Example: handler for a text frame  |  283  
| 19.7 Rules for well-behaved handlers  |  291  

Appendix A Viewer class note board: an extended example  |  293  
| A.1 Introduction  |  293  
| A.2 Module Boards  |  295  
| A.3 Module BoardFrames  |  300  
| A.4 Module PostIt (command module)  |  321  

Appendix B Keyboard and ASCII characters  |  324  

Appendix C MS/DOS files  |  329  

Bibliography  |  331  

Glossary  |  333  

Index  |  339  

Trademark notice  
Apple™ and Macintosh™ are registered trademarks of Apple Computer, Inc.  
DECsystem™ is a trademark of Digital Equipment Corporation  
IBM PS/2™ is a trademark of International Business Machines Corporation  
PC-DOS™ and MS-DOS™ are trademarks of Microsoft Corporation  
SUN Sparcstation™ is a trademark of Sun Microsystems Incorporated