

SIGGRAPH 1990

*17th International Conference
On Computer Graphics and
Interactive Techniques*

*Dallas Convention Center
August 6th—10th*

COURSE NOTES

16

INTRODUCTION TO WINDOW MANAGEMENT

Chair

Jonathan E. Steinhart
Independent Consultant

Lecturers

Mark Callow
Silicon Graphics Computer Systems

Richard J. Greco
Intel Scientific Computers

Robert Reed
Independent Consultant

Jonathan E. Steinhart
Independent Consultant

CONTENTS

Forward	1
Preface	3
Chapter 1 - What Is A Window Management System?	5
1.1 Introduction	5
1.2 Window Management System - Operating System Relationship	6
1.3 The Application Program Interface	6
1.4 Kernel Topics	9
1.4.1 Device Drivers	9
1.4.2 The Area Management Unit	11
1.4.3 The Input Management Unit	13
1.4.4 Task Scheduling	14
1.4.5 The Control Store	15
1.5 User Topics	16
1.5.1 Toolkits	16
1.5.2 Window Managers	16
1.5.3 Virtual Terminals	17
1.5.4 Application Programs	17
1.6 Conclusions	18
1.7 Problems	18
Chapter 2 - Considerations in Selecting an Application Program Interface	19
2.1 Introduction	19
2.2 Exposure versus Abstraction	19
2.3 The Transport Mechanism	20
2.3.1 Choice of Transport Mechanism	20
2.3.2 Transport Bandwidth and Latency	21
2.3.3 Synchronous versus Asynchronous Transports	22
2.4 Abstracting Output	22
2.4.1 Output Abstraction using Enumeration	22
2.4.2 Complete Output Abstraction	23
2.4.3 Level of Output Functions	24
2.5 Abstracting Input	25
2.5.1 Input Abstraction using Enumeration	25
2.5.2 Complete Input Abstraction	25
2.6 Coupling	25
2.7 Event Mapping	26
2.7.1 Loosely Coupled Event Mapping	26
2.7.2 Filtering Loosely Coupled Event Mapping	26
2.7.3 Tightly Coupled Event Mapping	27
2.8 Feedback	27
2.8.1 Coupling Feedback to Event Generation	28
2.8.2 Coupling Feedback to Event Mapping	29
2.8.3 Feedback that Occurs After Event Delivery	30
2.9 Policy Considerations at the Application Program Interface	31
2.10 Process Model Considerations at the Application Program Interface	31
2.10.1 Single Processor Environments	32
2.10.2 Multiprocessor Environments	32
2.11 Conclusions	32
Chapter 3 - Task Management and Scheduling	35

3.1	Introduction	35
3.2	The Connection	35
3.3	Scheduling 101	35
3.4	Dispatching	37
3.5	Scheduling Algorithms	37
3.5.1	Tasks	37
3.5.2	Task Context	38
3.5.3	Heavyweight and Lightweight Tasks	38
3.5.4	Critical Regions	38
3.5.5	Round-Robin Scheduling	39
3.5.6	Priority-Based Scheduling	39
3.5.7	Preemptive Scheduling	39
3.6	Task Classification	39
3.7	Thread Scheduling	40
3.8	Process Scheduling	41
3.9	Scheduling in X and NeWS™	41
3.10	Multiprocessing Considerations	41
3.10.1	Separate Graphics Processor	41
3.10.2	Multiple Processors	41
3.11	Connection Implementation	42
3.12	Conclusions	44
3.13	Problems	44
Chapter 4 - Database Concepts		45
4.1	Introduction	45
4.2	What is a Database?	45
4.3	Database Operations	45
4.3.1	Create Table	45
4.3.2	Add Field	45
4.3.3	Delete Field	46
4.3.4	Delete Table	46
4.3.5	Insert Record	46
4.3.6	Query Record	46
4.3.7	Delete Record	46
4.3.8	Change Record	46
4.3.9	Lock Table	46
4.3.10	Unlock Table	46
4.3.11	Notification	46
4.4	Database Usage	46
4.4.1	Scheduling	46
4.4.2	Window Attributes	47
4.4.3	Graphics Attributes	47
4.4.4	Inter-Application Communication	47
4.4.5	Control Store	48
4.4.5.1	Writable Control Store Systems	48
4.4.5.1.1	The Blit	48
4.4.5.1.2	NeWS™	48
4.4.6	Writable Control Store Uses	48
4.4.6.1	Application-Specific Interface Tailoring	48
4.4.6.2	Downloadable Policy	48
4.4.6.3	User Interface Support	48
4.5	Conclusion	49
4.6	Problems	49

Chapter 5 - Area Management	51
5.1 Introduction	51
5.2 Policy versus Mechanism	51
5.2.1 Tiled Windows	51
5.2.2 Overlapping Windows	51
5.2.3 Hierarchical Overlapping Windows	52
5.3 The Area Management Database	52
5.3.1 Window Creation	53
5.3.2 Window Destruction	53
5.3.3 Window Relationship Manipulation	53
5.3.4 Window Attribute Manipulation	53
5.4 Coordinate Systems	53
5.5 Rectangular Area Management	54
5.5.1 Intersection Calculation	55
5.6 Batching	56
5.7 Hierarchical Area Management	56
5.8 Non-Rectangular Area Management	56
5.8.1 Quadrees	56
5.8.2 Y-Sorted Rectangle Lists	57
5.8.3 Y-Sorted Lists of X-Sorted Rectangles	57
5.8.4 An Implementation of Non-Rectangular Windows	58
5.8.4.1 Shape Algebra	58
5.8.4.1.1 Data Structures	58
5.8.4.1.2 Union	60
5.8.4.1.3 Intersection	60
5.8.4.1.4 Difference	61
5.8.4.1.5 Utility Functions	61
5.8.4.2 Area Management	62
5.8.4.2.1 Data Structures	62
5.8.4.2.2 Create Window	63
5.8.4.2.3 Locate Window	63
5.8.4.2.4 Install Window	63
5.8.4.2.5 Uninstall Window	64
5.8.4.2.6 Delete Window	64
5.8.4.2.7 Process Hierarchy	64
5.9 Transparent Windows	65
5.10 Other Topics	66
5.10.1 Area Based Input Distribution	66
5.10.2 Area Management Hardware	66
5.10.3 Distributed Graphics	66
5.11 Conclusions	67
Chapter 6 - Device Drivers for Monochrome Frame Buffers	69
6.1 Architecture of a Monochrome Frame Buffer	69
6.2 Characteristics of a Frame Buffer	69
6.3 Simple Graphics in a Frame Buffer	69
6.3.1 Scan Converting Lines	70
6.3.2 Special Requirements	70
6.4 Mathematical Clipping	71
6.5 Scissoring to a Single Rectangle	71
6.6 Scissoring to Multiple Rectangles	72
6.7 Scan Converting Occluded Areas	73
6.8 Clipping to Arbitrary Regions	73
6.9 Drawing Using Shape Algebra	75

6.10	Damage from Raster Operations	77
6.11	Drawing Cursors	77
6.12	Conclusions	77
Chapter 7 - Device Drivers for Frame Buffers with Color Maps or Gamma Correctors		79
7.1	Architecture of a Color Frame Buffer	79
7.1.1	Changes to Frame Buffer Organization	79
7.1.2	Changes to the Video Display System	79
7.1.2.1	What is a Color Map	80
7.1.2.2	What is a Gamma Corrector	80
7.2	Dynamic Versus Static Attributes	80
7.2.1	Private Copies	80
7.2.2	Sharing One Physical Color Map	81
7.2.3	Color Maps in the X Window System™	81
7.2.4	Fixed Color Map Entries/Dithering	81
7.3	Cursors	82
7.4	Conclusions	82
Chapter 8 - Device Drivers for Graphics Processors		83
8.1	Introduction and Overview	83
8.2	Using a Graphics Processor Chip	83
8.3	Context Management	83
8.4	Clipping	86
8.5	Why use a Graphics Processor	86
8.6	Hardware Cursors	87
8.7	Comparison of Graphics Processor	87
8.8	3-D Processors	88
8.9	Related Issues	88
8.9.1	Display Modes	88
8.9.2	Double Buffering	89
8.9.3	Overlay Planes	89
8.9.4	Video Mixers	89
8.9.4.1	Characteristics of a Video Mixer	89
8.9.4.2	Graphics in a Video Mixer Architecture	90
8.9.4.3	Limitations of a Video Mixer Architecture	90
8.9.4.3.1	Cost	90
8.9.4.3.2	Physical Memory Usage	91
8.9.4.3.3	Other Approaches are Catching Up	91
8.9.4.3.4	Conclusions	91
8.9.5	Live Video	91
8.10	Conclusions	92
Chapter 9 - Input		93
9.1	Introduction	93
9.2	What Is Input?	93
9.3	Input Generation	93
9.3.1	Timers and Time Stamps	93
9.3.2	Input Event Ordering	94
9.3.3	Input Events	94
9.4	Input Distribution	95
9.4.1	Real-Estate Based Input Distribution	95
9.4.2	Listener Based Input Distribution	95
9.5	Application Based Input Distribution	95

9.5.1 Race Conditions	96
9.5.2 Performance	97
9.5.3 Improved Application-Based Distribution	97
9.5.3.1 A Sample Programmable Policy Engine	97
9.5.3.1.1 Event Masks	98
9.5.3.1.2 Event Duplication	98
9.5.3.1.3 The Window Search Path	98
9.5.3.1.4 Traps	100
9.6 X Windows Input Distribution	100
9.7 NeWS™ Input Distribution	100
9.8 A More Detailed Look At Input	101
9.8.1 Synthetic Input	101
9.8.2 Window Manager	101
9.8.3 Keyboard Focus	102
9.8.4 Input Compression	102
9.9 Implementation Details	102
9.9.1 Event Masks and Traps	102
9.9.2 The Pointer Window	103
9.9.3 Distribution	103
9.10 Conclusions	103
Chapter 10 - User Space Overview	105
10.1 Introduction	105
10.2 Job Control Programs	106
10.3 Programmer Toolkits	107
10.4 Libraries	108
10.5 Compatibility Packages	108
10.6 User Interface Standards	109
10.7 User Interface Builders	109
10.8 User Toolkits	109
10.9 Conclusions	109
Chapter 11 - Writing Applications for Window Managed Systems	111
11.1 Introduction	111
11.2 The Structure of a Traditional Interactive Application	111
11.3 The Structure of Window System Applications	112
11.4 Controller-Model-View Abstraction	114
11.4.1 Model	114
11.4.2 View	114
11.4.3 Controller	114
11.5 Application Scheduling	114
11.5.1 Optimizing to Decrease Granularity	115
11.5.2 Breaking Up Long Granularity Events	115
11.5.3 Using Multiple Processes to Decrease Granularity	116
11.5.3.1 Heavyweight Processes	116
11.5.3.2 Lightweight Processes	116
11.6 Applications in a Multiprocessing World	116
11.7 Conclusions	117
Chapter 12 - Window Managers and Other Control Programs	119
12.1 Introduction	119
12.2 Window Managers	119
12.2.1 Architectural Role of Window Managers	119
12.2.2 Resource Management	120

12.2.2.1	Area Management	121
12.2.2.2	Input Focus	121
12.2.2.3	Colormaps	121
12.2.2.4	Impacts on Applications	122
12.2.3	User Interface Component	123
12.2.3.1	Basic Window Manager Functions	123
12.2.3.2	Customizing the Interface	125
12.3	Selection Controllers	125
12.3.1	Text Selection	126
12.4	Session Management	127
12.5	Summary	127
Chapter 13	Foundations for Graphical Applications and User Interfaces	129
13.1	The Problem of Programming	129
13.2	Window System API	129
13.3	Imaging Model	130
13.4	User Interface Toolkits	130
13.4.1	What is a Toolkit?	130
13.4.2	Toolkit Components	130
13.4.2.1	Panes	130
13.4.2.2	A Button	130
13.4.2.3	A Slider	131
13.4.3	Adding the Look	131
13.4.4	Building a new gadget	132
13.4.5	Input	132
13.4.6	Toolkit Interfaces	133
13.4.7	Packaging the Gadgets	135
13.5	Object Oriented Programming for Toolkits	135
13.5.1	An Objective Solution	135
13.5.2	Building an Object-Oriented House	135
13.5.3	Applying the Paradigm to Window Programming	136
13.5.4	Class Application	136
13.5.5	Gadgets as Objects	136
13.6	Selections	137
13.7	User Interface Management Systems	137
13.8	Conclusion	137
Chapter 14	User Interface Standards	139
14.1	The Desire for UI Standards	139
14.2	The Scope of the Standards	139
14.3	Quantum Effects of Standards	139
14.4	Components: Look and Feel Disected	140
14.4.1	Philosophy	140
14.4.2	Containers	141
14.4.3	Controls	141
14.4.4	Interactions	142
14.5	The Drawbacks to Standards	142
14.6	The Myths	142
14.7	Visual Surveys	142
14.8	Macintosh Human Interface Guidelines	142
14.8.1	Philosophy	142
14.8.2	Containers	143
14.8.3	Controls	144
14.8.4	Interactions	144

14.9 OPENLOOK	144
14.9.1 Philosophy	144
14.9.2 Containers	144
14.9.3 Controls	145
14.9.4 Interactions	145
14.10 That's Not All	145
Chapter 15 - User Interface Builders	147
15.1 Introduction	147
15.2 UIB Structure	147
15.2.1 Components	147
15.3 Front Ends	148
15.3.1 Graphical Front Ends	148
15.3.2 Prototyper	148
15.3.3 Sun Guide	149
15.4 Back Ends	149
15.5 Summary	150
Chapter 16 - User Toolkits	151
16.1 Window System Components for Users	151
16.2 Relationship of Application to Data	151
16.3 What Constitutes the Common Set	151
16.3.1 Current Limitations	151
16.4 A Tool to Create Tools	152
16.5 Fledgling Environments	152
Chapter 17 - Survey of Window Systems	153
17.1 Introduction	153
17.2 Smalltalk	153
17.3 DLisp	153
17.4 Tajo, Docs, Star, Viewers	153
17.5 NU	154
17.6 Oriel	154
17.7 SunView™	154
17.8 Andrew™	155
17.9 The Macintosh™	155
17.10 The W Window System	156
17.11 VGTS	156
17.12 The X Window System™	157
17.13 NeWS™	158
Chapter 18 - Conclusions	161
18.1 Introduction	161
18.2 Operating System Support	161
18.3 Networking	161
18.4 Creeping Features	161
18.5 User Interface	161
18.6 Input Device Support	162
18.7 Graphics Hardware Advances	162
18.8 Standards	162
18.9 Blind Alleys	162
Bibliography	165

Appendix A - Course Slides	171
A.0 Slides for Preface	171
A.1 Slides for Chapter 1	175
A.2 Slides for Chapter 2	183
A.3 Slides for Chapter 3	195
A.4 Slides for Chapter 4	203
A.5 Slides for Chapter 5	207
A.6 Slides for Chapter 6	221
A.7 Slides for Chapter 7	231
A.8 Slides for Chapter 8	237
A.9 Slides for Chapter 9	249
A.10 Slides for Chapter 10	259
A.11 Slides for Chapter 11	265
A.12 Slides for Chapter 12	275
A.13 Slides for Chapter 13	287
A.14 Slides for Chapter 14	301
A.15 Slides for Chapter 15	307
A.16 Slides for Chapter 16	311
A.17 Slides for Chapter 17	313
A.18 Slides for Chapter 18	315

Forward

Introduction

This course is an introduction to the field of window management. Window management is treated as a relatively mature area in which the problems are understood, if not all solved in a single package. A model is presented that partitions window management into a number of distinct areas. Each area is discussed, its problems examined, and a number of solutions are offered. Existing window systems serve as examples during the discussion.

This course is not a tutorial for the X window system™, NeWS™, or any other window management system. Instead, it is an introduction to the basic principles behind window management.

Prerequisites

This course assumes that the student is familiar with graphics, particularly the raster variety. Some of the material requires a cursory knowledge of graphics hardware. A knowledge of operating systems and operating system principles is essential. A familiarity with the UNIX® is particularly helpful.

The Course Notes

There are two sections to the course notes. The first section contains the papers from which the presentation is extracted. The second section contains a copy of the slides that are used in the presentation.

The course notes contain much more information that we will be presenting. The authors hope that these notes will serve as a valuable reference outside of the course. Everything that we will be talking about is in the notes except for the live demos, so you can pay attention instead of furiously scribbling down everything that we say.

History

Some of the reviewers of the course proposal felt that the course should include a history of window systems. After much deliberation I concluded that the history is no longer of great importance. That isn't to say that the mistakes and successes are not important; just that it is no longer significant to spend time telling you that window management originated at Xerox.¹ After all, how many people in the computer graphics field know who invented the cathode ray tube?

¹ There, I did it!

Note

The papers in the first section are a draft of a book that will be published by Van Nostrand Reinhold in late 1990 or early 1991. Any comments and/or suggestions would be appreciated. Send them to:

Jonathan E. Steinhart
10886 Northridge Square
Cupertino, California 95014
...apple!4-winds!jon

(408) 446-3770

Trademarks

Andrew is a trademark of Carnegie-Mellon University.

Apple is a trademark of Apple Computer, Incorporated.

MS-DOS is a trademark of Microsoft Corporation.

MacApp is a trademark of Apple Computer, Incorporated.

MacDraw is a trademark of Apple Computer, Incorporated.

MacPaint is a trademark of Apple Computer, Incorporated.

MacRocket is a trademark of the Safe X Foundation.

Macintosh is a trademark of Apple Computer, Incorporated.

Motif is a trademark of the Open Software Foundation.

NeWS is a trademark of Sun Microsystems, Incorporated.

OPEN LOOK is a trademark of UNIX International.

POSTSCRIPT is a registered trademark of Adobe Systems, Incorporated.

QuickDraw is a trademark of Apple Computer, Incorporated.

SunView is a trademark of Sun Microsystems, Incorporated.

UNIX is a registered trademark of AT&T.

xtool is a trademark of Xgraph, Incorporated.