Copyright Statement

The digital copy of this thesis is protected by the Copyright Act 1994 (New Zealand). This thesis may be consulted by you, provided you comply with the provisions of the Act and the following conditions of use:

- Any use you make of these documents or images must be for research or private study purposes only, and you may not make them available to any other person.

- Authors control the copyright of their thesis. You will recognise the author's right to be identified as the author of this thesis, and due acknowledgement will be made to the author where appropriate.

- You will obtain the author's permission before publishing any material from their thesis.

To request permissions please use the Feedback form on our webpage. http://researchspace.auckland.ac.nz/feedback

General copyright and disclaimer

In addition to the above conditions, authors give their consent for the digital copy of their work to be used subject to the conditions specified on the Library Thesis Consent Form
Investigations in Graphical Statistics

Paul R. Murrell
A thesis submitted in partial fulfilment of the requirements for
the degree of Doctor of Philosophy in Statistics,
The University of Auckland, 1998

July 31, 1998
This thesis is concerned with the design and development of statistical graphics software—programs to help draw graphs. Graphs serve two major functions in statistics. Firstly, graphs are used for exploratory data analysis—for detecting the message in a set of data—and secondly, graphs are used for data display—for presenting the message in a set of data.

The most important feature of software for exploratory data analysis is extensibility. This is the ability to quickly and easily develop new graphical images and is vital for being able to explore a data set in many different ways. The most important feature of software for data display is customisation. This is the ability to finetune a graphical image in great detail and is vital for the production of presentation-quality graphics. In both cases it is important that a graphical image should be constructed to best explore or show-off the peculiarities of a specific data set.

A pervading theme of this thesis is that statistical graphics software should be flexible. The software tools described herein allow graphical images to be modified in arbitrary ways; the structure of graphical images is also arbitrary and not restricted to standard graphic formats; a simple, coherent method, based on a general constraint system, for developing novel graphical images is explored; and a mechanism for specifying the arrangement of the components of a graphical image is introduced. Some of these ideas are incorporated within an existing statistical analysis package.
I would like to thank the following for their contributions to this thesis:

**The University of Auckland** for the provision of a University of Auckland Doctoral Scholarship.

**The Department of Statistics** for the provision of generous space, computing, printing, and photocopying resources.

**Ross Ihaka** for his supervision. In particular, thanks for always being available, always having an answer, and always seeming to place more importance on my career than your own.

**Julia Murrell née Clark** for proofreading, advice, support, and for teaching me the importance of “time off.”
Contents

1 Introduction ........................................... 1
   1.1 What is Graphical Statistics? ..................... 1
   1.2 Issues in Graphical Statistics .................... 2
      1.2.1 Customisation .................................. 2
      1.2.2 Extensibility .................................. 2
      1.2.3 Presentation-Quality Output vs. Flexibility and Interaction 3
   1.3 Modern Graphical Statistics Software ............. 4
      1.3.1 S .............................................. 4
      1.3.2 R .............................................. 4
      1.3.3 Quali .......................................... 5
      1.3.4 Pictor ......................................... 6
      1.3.5 XLispStat ...................................... 7
      1.3.6 XGobi .......................................... 8
   1.4 The Structure of this Thesis ...................... 9
   1.5 References ........................................ 10

2 Simplisp ............................................. 12
   2.1 Aims .............................................. 15
      2.1.1 Fully incorporating 3D graphs ................ 15
      2.1.2 Accessing every detail of a graph ............ 15
      2.1.3 Sharing elements between graphs ............... 16
      2.1.4 3D selection .................................. 17
      2.1.5 Arbitrary Plot Structures ...................... 19
   2.2 Design ............................................ 20
      2.2.1 Constructing a Scene .......................... 20
      2.2.2 Viewing a Scene ................................ 20
      2.2.3 Automatic updating ............................ 21
      2.2.4 Selecting with a mouse ......................... 21
   2.3 Implementation ................................... 23
      2.3.1 Simplisp Classes ............................... 23
   2.4 Simplisp Reference ................................ 29
   2.5 User’s Guide ...................................... 37
      2.5.1 Getting Started ................................ 37
      2.5.2 Creating a plot ................................ 37
      2.5.3 Selecting Elements of a Plot ................... 37
      2.5.4 Sharing elements between graphs ............... 42
      2.5.5 Creating Arbitrary Plots ....................... 43
      2.5.6 Getting stopped ................................ 46
      2.5.7 An Advanced Simplisp Example ................... 46
## Contents

2.6 Results and Conclusions .................................................. 48  
2.6.1 3D-based graphics .................................................. 48  
2.6.2 Accessing every detail of a graph .................................. 49  
2.6.3 Sharing elements between graphs .................................. 50  
2.6.4 3D selection .......................................................... 51  
2.6.5 Arbitrary graph structure .......................................... 52  
2.6.6 General problems .................................................... 52  
2.7 References ............................................................. 55  

3 Xtend ................................................................. 56  
3.1 Aims ................................................................. 62  
3.1.1 Extensibility ....................................................... 62  
3.1.2 Incremental Graphics ............................................. 62  
3.1.3 A Stand-Alone Graphics System .................................. 63  
3.2 Design ............................................................... 65  
3.2.1 Constructing a graph ............................................. 65  
3.2.2 Developing new plotting elements .............................. 66  
3.2.3 Combining plotting elements .................................... 66  
3.2.4 Constraints between plotting elements ........................ 66  
3.2.5 Communication with R ............................................ 67  
3.3 Implementation ....................................................... 68  
3.3.1 Xtend Prototypes .................................................. 70  
3.3.2 Inheritance ....................................................... 74  
3.3.3 Edit-slot .......................................................... 74  
3.3.4 Rslave ........................................................... 75  
3.4 Xtend Reference ...................................................... 77  
3.4.1 Functions .......................................................... 77  
3.4.2 Prototypes ........................................................ 78  
3.5 User's Guide .......................................................... 90  
3.5.1 Getting started .................................................... 90  
3.5.2 Creating a plot ................................................... 90  
3.5.3 The edit-slot function ............................................ 90  
3.5.4 Controlling plot layout .......................................... 90  
3.5.5 Adding plotting elements to each other ........................ 91  
3.5.6 Communicating with R .......................................... 91  
3.5.7 Constraints between plotting elements ........................ 92  
3.5.8 Creating new plotting elements ................................. 93  
3.5.9 The Motif interface .............................................. 95  
3.5.10 An Advanced Xtend Example ................................... 96  
3.6 Results and Conclusions .............................................. 101  
3.6.1 Extensibility ..................................................... 101  
3.6.2 Incremental Graphics ............................................ 102  
3.6.3 Stand-Alone Graphics .......................................... 106  
3.6.4 Problems ........................................................ 106  
3.7 References .......................................................... 108
6.9 References ...................................................... 178

A An introduction to
Object-Oriented Programming
in Common Lisp 179
A.1 Programming in Lisp ........................................... 179
A.1.1 Lisp Syntax .................................................. 181
A.1.2 Lisp and Lists ............................................... 181
A.1.3 Symbols .................................................... 182
A.1.4 Special forms ............................................... 182
A.1.5 Keyword Arguments ...................................... 182
A.2 CLOS: Object-Oriented Programming in
Common Lisp .................................................... 183
A.2.1 Classes and Instances .................................. 183
A.2.2 Generic Functions and Methods ...................... 184
List of Figures

1 The common components of a graph ........................................... ix

1.1 A hierarchy of Pictor grobs ....................................... 7

2.1 A simple scatterplot .............................................. 12
2.2 A scatterplot and a 3D plot together on the same page .......... 13
2.3 Modifying a shared y-axis ........................................ 13
2.4 Modifying individual data symbols and tick-mark labels ......... 13
2.5 Relocating an axis label in 3D .................................... 14
2.6 Customising a “temperature” axis ................................... 15
2.7 Overlapping graphical objects ..................................... 18
2.8 Back-to-front ordering of graphical objects ................. 18
2.9 The viewing elements of the Simplisp system ............... 22
2.10 The resource classes in Simplisp ................................. 24
2.11 The viewing classes in Simplisp ................................. 25
2.12 The input/output classes in Simplisp ................... 26
2.13 The graphical classes in Simplisp ................................. 26
2.14 The collection classes in Simplisp ................................. 27
2.15 A Simplisp plot viewed in an X11 window ................. 38
2.16 A Simplisp object hierarchy ..................................... 39
2.17 Customisation and 3D views in Simplisp .................... 40
2.18 Customisation in 3D ............................................ 41
2.19 An object hierarchy showing shared elements ............... 42
2.20 Two plots with shared elements ................................ 43
2.21 Two plots after modifying a shared element ................ 44
2.22 A scatterplot of height vs. age ................................ 44
2.23 Adding new elements to a plot .................................. 45
2.24 The Simplisp 3D picking algorithm ......................... 51

3.1 A graph with a dot-plot per group ................................. 57
3.2 A graph with dot-plots and boxes ................................ 58
3.3 A graph for performing an analysis of variance by eye .......... 59
3.4 Another graph for performing an analysis of variance by eye .... 60
3.5 The components of the viewing system in Xtend ................ 65
3.6 An example of inheritance between graphic objects ............ 80
3.7 A Motif-style dialog ............................................. 96
3.8 Two possible implementations of a boxplot plotting element .... 102
3.9 A novel data symbol plotting element ........................... 105

4.1 Specifying the location of a plot on a page ...................... 110
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2</td>
<td>Coordinate systems in $S$</td>
<td>111</td>
</tr>
<tr>
<td>4.3</td>
<td>The $S$ margin coordinate systems.</td>
<td>111</td>
</tr>
<tr>
<td>4.4</td>
<td>Allocating rows and columns from a layout specification</td>
<td>120</td>
</tr>
<tr>
<td>4.5</td>
<td>Allocating a region to a plot</td>
<td>120</td>
</tr>
<tr>
<td>4.6</td>
<td>Allocating absolute rows and columns</td>
<td>121</td>
</tr>
<tr>
<td>4.7</td>
<td>Allocating rows and columns with full respect</td>
<td>121</td>
</tr>
<tr>
<td>4.8</td>
<td>Allocating rows and columns with partial respect</td>
<td>122</td>
</tr>
<tr>
<td>4.9</td>
<td>Allocating a mixture of relative and absolute rows and columns</td>
<td>123</td>
</tr>
<tr>
<td>4.10</td>
<td>A simple layout example</td>
<td>129</td>
</tr>
<tr>
<td>4.11</td>
<td>A layout with full respect</td>
<td>130</td>
</tr>
<tr>
<td>4.12</td>
<td>A layout with row-heights and column-widths specified</td>
<td>131</td>
</tr>
<tr>
<td>4.13</td>
<td>A layout with absolute row-heights and column-widths</td>
<td>132</td>
</tr>
<tr>
<td>4.14</td>
<td>A layout with margins</td>
<td>133</td>
</tr>
<tr>
<td>4.15</td>
<td>Arranging components within a plot with layouts</td>
<td>134</td>
</tr>
<tr>
<td>5.1</td>
<td>A diagram of the bounding box of a character</td>
<td>139</td>
</tr>
<tr>
<td>5.2</td>
<td>Typesetting for the text &quot;hey there&quot;.</td>
<td>139</td>
</tr>
<tr>
<td>5.3</td>
<td>Typesetting for the expression $\text{over}(x[i], y + 2)$</td>
<td>140</td>
</tr>
<tr>
<td>5.4</td>
<td>A plot of the Normal probability density function</td>
<td>143</td>
</tr>
<tr>
<td>5.5</td>
<td>A periodogram with mathematical annotation</td>
<td>143</td>
</tr>
<tr>
<td>5.6</td>
<td>Specifying figure regions with $\text{par(mfrow=...)}$</td>
<td>147</td>
</tr>
<tr>
<td>5.7</td>
<td>Specifying the figure region with $\text{par(fin=...)}$</td>
<td>148</td>
</tr>
<tr>
<td>5.8</td>
<td>Specifying the plot region with $\text{par(mar=...)}$</td>
<td>149</td>
</tr>
<tr>
<td>5.9</td>
<td>Specifying the plot region with $\text{par(pin=...)}$</td>
<td>149</td>
</tr>
</tbody>
</table>
Terminology

In this thesis, a **graph** or **plot** is simply considered to be a collection of graphical elements. The graphical elements may be anything from basic **graphical primitives** such as lines, rectangles, and text to complex statistical objects such as axes, boxplots, and scatterplots. The latter are often referred to as **plotting elements**. Plotting elements are considered to be made up of a number of **components**. For example, the components of an axis are a major line, a number of tick-marks, and a label. Figure 1 shows the terminology used in this thesis for the standard components of a graph.

![Diagram of graph components](image)

**Figure 1**: The common components of a graph.