

# Figures, Tables, and Listings

Preface	About This Book	xix
	<b>Figure P-1</b>	Roadmap to the QuickDraw GX suite of books xx
Chapter 1	QuickDraw GX and the Macintosh Environment	1-1
	<b>Listing 1-1</b>	Determining the presence and features of QuickDraw GX 1-5
	<b>Figure 1-1</b>	Converting from QuickDraw global to QuickDraw GX local and global coordinates 1-8
	<b>Table 1-1</b>	Translation scaling factors 1-11
	<b>Table 1-2</b>	Translation options settings 1-12
	<b>Listing 1-2</b>	QuickDraw commands to draw a simple line 1-14
	<b>Figure 1-2</b>	A QuickDraw line 1-14
	<b>Figure 1-3</b>	Translation of the QuickDraw line using <code>gxDefaultOptionsTranslation</code> 1-15
	<b>Figure 1-4</b>	Translation of the QuickDraw line using <code>gxSimpleGeometryTranslation</code> 1-15
	<b>Figure 1-5</b>	Translation of the QuickDraw line using <code>gxReplaceLineWidthTranslation</code> 1-16
	<b>Figure 1-6</b>	Conversion of standard QuickDraw fill patterns to QuickDraw GX shape fills 1-17
	<b>Listing 1-3</b>	QuickDraw picture data that includes a <code>picComment</code> 1-18
	<b>Figure 1-7</b>	Translating QuickDraw data containing a rotation <code>picComment</code> 1-19
	<b>Listing 1-4</b>	Translating QuickDraw picture data with <code>GXConvertPICTToShape</code> 1-20
	<b>Table 1-3</b>	Translation statistics options 1-20
	<b>Listing 1-5</b>	Installing and removing the translator 1-21
	<b>Listing 1-6</b>	Sample application-defined shape-spooling function 1-22
Chapter 2	QuickDraw GX Memory Management	2-1
	<b>Listing 2-1</b>	Creating a 'gasz' resource 2-6
	<b>Listing 2-2</b>	Explicitly creating a graphics client and its heap 2-7
	<b>Listing 2-3</b>	Disposing of graphics clients and graphics client heaps 2-10
	<b>Table 2-1</b>	QuickDraw GX functions that do not require a graphics client or heap 2-14
	<b>Listing 2-4</b>	Specifying the starting location and size for a graphics client and its heap 2-15
	<b>Figure 2-1</b>	Creating a graphics client by specifying the memory starting location 2-16

<b>Figure 3-1</b>	QuickDraw GX and application-defined error, warning, and notice management	3-4
<b>Table 3-1</b>	Non-debugging error number ranges	3-6
<b>Table 3-2</b>	Fatal errors	3-7
<b>Table 3-3</b>	Internal errors	3-7
<b>Table 3-4</b>	Recoverable errors	3-7
<b>Table 3-5</b>	Font management errors	3-8
<b>Table 3-6</b>	Font scaler errors	3-8
<b>Table 3-7</b>	Bad parameter errors	3-9
<b>Table 3-8</b>	Implementation limit errors	3-10
<b>Table 3-9</b>	Non-debugging warning number ranges	3-10
<b>Table 3-10</b>	Stack, heap, and object warnings	3-11
<b>Table 3-11</b>	Result out of range warnings	3-11
<b>Table 3-12</b>	Parameter out of range warnings	3-12
<b>Table 3-13</b>	Font scaler warnings	3-12
<b>Table 3-14</b>	Unexpected result warnings	3-13
<b>Table 3-15</b>	Storage warnings	3-13
<b>Table 3-16</b>	Debugging error number range	3-14
<b>Table 3-17</b>	Internal debugging errors	3-14
<b>Table 3-18</b>	Font parameter debugging errors	3-14
<b>Table 3-19</b>	Bad parameter debugging errors	3-15
<b>Table 3-20</b>	Restricted access debugging errors	3-16
<b>Table 3-21</b>	Wrong type and bad reference debugging errors	3-17
<b>Table 3-22</b>	Type validation debugging errors	3-18
<b>Table 3-23</b>	Cache validation debugging errors	3-18
<b>Table 3-24</b>	Shape cache validation shape debugging errors	3-19
<b>Table 3-25</b>	Memory block validation debugging errors	3-19
<b>Table 3-26</b>	Object validation debugging errors	3-20
<b>Table 3-27</b>	Path and polygon validation debugging errors	3-20
<b>Table 3-28</b>	Bitmap validation debugging errors	3-20
<b>Table 3-29</b>	Bitmap image validation debugging errors	3-21
<b>Table 3-30</b>	Text validation debugging errors	3-21
<b>Table 3-31</b>	Glyph validation debugging errors	3-21
<b>Table 3-32</b>	Layout validation debugging errors	3-22
<b>Table 3-33</b>	Picture validation debugging errors	3-22
<b>Table 3-34</b>	Text face validation debugging errors	3-22
<b>Table 3-35</b>	Transform validation debugging errors	3-23
<b>Table 3-36</b>	Font cache validation debugging errors	3-23
<b>Table 3-37</b>	View device validation debugging errors	3-24
<b>Table 3-38</b>	Color set validation debugging errors	3-24
<b>Table 3-39</b>	Color profile validation debugging errors	3-24
<b>Table 3-40</b>	Internal backing store validation debugging errors	3-25
<b>Table 3-41</b>	Debugging warning number range	3-25
<b>Table 3-42</b>	Invalid data debugging warnings	3-26
<b>Table 3-43</b>	Can't find debugging warnings	3-26
<b>Table 3-44</b>	Other debugging warnings	3-27
<b>Table 3-45</b>	Debugging version notice number summary	3-27
<b>Table 3-46</b>	Debugging notices	3-27
<b>Figure 3-2</b>	Polling for errors, warnings, and notices	3-31

<b>Figure 3-3</b>	Obtaining the first and last posted QuickDraw GX error	3-32
<b>Listing 3-1</b>	Obtaining the first posted error	3-33
<b>Listing 3-2</b>	Obtaining the first and last QuickDraw GX warning	3-34
<b>Listing 3-3</b>	Obtaining the first and last posted notices	3-34
<b>Listing 3-4</b>	Changing the error posted	3-36
<b>Figure 3-4</b>	Adding and removing warnings and notices from the ignore warning and ignore notice stacks	3-39
<b>Figure 3-5</b>	Enabling and disabling an error handler	3-41

## Chapter 4

### QuickDraw GX Debugging 4-1

---

<b>Figure 4-1</b>	The QuickDraw GX debugging environment	4-4
<b>Table 4-1</b>	QuickDraw GX drawing process sequence	4-9
<b>Table 4-2</b>	Shape type drawing errors	4-10
<b>Table 4-3</b>	Style drawing errors	4-11
<b>Table 4-4</b>	Ink drawing errors	4-12
<b>Table 4-5</b>	Transform drawing errors	4-13
<b>Table 4-6</b>	View port drawing errors	4-14
<b>Table 4-7</b>	View device drawing errors	4-15
<b>Table 4-8</b>	Validation modes	4-16
<b>Table 4-9</b>	Validation levels	4-16
<b>Table 4-10</b>	Memory validation options	4-19
<b>Listing 4-1</b>	Determining the function and parameter that caused the last validation error	4-22
<b>Table 4-11</b>	GraphicsBug commands and responses	4-23
<b>Listing 4-2</b>	Totaling the graphics client and its heap	4-25
<b>Listing 4-3</b>	Determining the memory locations of the shapes in the picture	4-26
<b>Listing 4-4</b>	Analyzing the rectangle shape in the picture	4-27
<b>Listing 4-5</b>	Analyzing the ink in the rectangle shape	4-28

## Chapter 5

### Collection Manager 5-1

---

<b>Figure 5-1</b>	The collection object	5-7
<b>Figure 5-2</b>	The collection item	5-8
<b>Figure 5-3</b>	Editing attributes in a collection item	5-10
<b>Figure 5-4</b>	Items in a collection	5-12
<b>Listing 5-1</b>	Determining whether the Collection Manager is available	5-13
<b>Listing 5-2</b>	Changing the default attributes of a collection	5-16
<b>Listing 5-3</b>	Adding items to a collection	5-17
<b>Listing 5-4</b>	Adding items with variable-length data to a collection	5-18
<b>Listing 5-5</b>	Determining the index of an item	5-20
<b>Listing 5-6</b>	Determining the tag and ID of an item given the item's index	5-21
<b>Listing 5-7</b>	Determining the size of an item's variable-length data	5-23
<b>Listing 5-8</b>	Examining the attributes of an item	5-25
<b>Listing 5-9</b>	Setting the lock and persistence bit attribute of an item	5-26
<b>Listing 5-10</b>	Replacing an item in a collection	5-28
<b>Listing 5-11</b>	Replacing an item using the item's index	5-29
<b>Listing 5-12</b>	Removing an item in a collection	5-30

<b>Listing 5-13</b>	Removing an item using the item's index	5-31
<b>Listing 5-14</b>	Removing multiple items with specific attributes	5-31
<b>Listing 5-15</b>	Retrieving the variable-length data from an item	5-33
<b>Listing 5-16</b>	Retrieving the variable-length data from an item using the item's index	5-34
<b>Listing 5-17</b>	Retrieving the variable-length data from an item using the tag and tag list position	5-35
<b>Listing 5-18</b>	Counting tags in a collection	5-36
<b>Listing 5-19</b>	Flattening procedure	5-38
<b>Listing 5-20</b>	The <code>FlattenCollectionToHdl</code> function	5-39
<b>Listing 5-21</b>	A possible implementation of the <code>UnflattenCollectionFromHdl</code> function	5-40
<b>Listing 5-22</b>	Flattening a collection to a disk file as a resource	5-42
<b>Listing 5-23</b>	Flattening a collection to a data fork of a disk file	5-42
<b>Listing 5-24</b>	Unflattening a collection from a disk file as a resource	5-43
<b>Listing 5-25</b>	Unflattening a collection from the data fork of a disk file	5-43
<b>Listing 5-26</b>	Reading a collection from a collection resource	5-44
<b>Listing 5-27</b>	A sample exception procedure	5-46
<b>Listing 5-28</b>	A Rez template for a 'cltn' resource	5-102

## Chapter 6

### Message Manager 6-1

---

<b>Figure 6-1</b>	Printing with the Macintosh Printing Manager	6-4
<b>Figure 6-2</b>	Printing with QuickDraw GX	6-5
<b>Listing 6-1</b>	Creating an A5 world for global data	6-9
<b>Listing 6-2</b>	Disposing of global data and deallocating memory	6-10
<b>Listing 6-3</b>	Storing global data for a single message handler instance	6-11
<b>Listing 6-4</b>	Getting and disposing of global data	6-12
<b>Listing 6-5</b>	Storing global data for multiple handler instances	6-13
<b>Listing 6-6</b>	Retrieving a message handler's class context	6-14

## Chapter 7

### QuickDraw GX Stream Format 7-1

---

<b>Figure 7-1</b>	A typical flattened shape data stream sequence	7-8
<b>Figure 7-2</b>	Basic components of a stream header or object	7-9
<b>Figure 7-3</b>	The format of the operation opcode byte	7-10
<b>Table 7-1</b>	Operation opcodes	7-10
<b>Figure 7-4</b>	Data format of the record size	7-12
<b>Figure 7-5</b>	The format of the data type opcode byte	7-13
<b>Table 7-2</b>	Compression values	7-13
<b>Figure 7-6</b>	Relationship of stream format components	7-14
<b>Table 7-3</b>	Data type opcodes for a new object	7-15
<b>Table 7-4</b>	Data type opcodes to modify a shape object	7-17
<b>Table 7-5</b>	Data type opcodes to modify a style object	7-18
<b>Table 7-6</b>	Data type opcodes to modify an ink object	7-20
<b>Table 7-7</b>	Data type opcodes to modify a color set object	7-20
<b>Table 7-8</b>	Data type opcodes to modify a color profile object	7-21
<b>Table 7-9</b>	Data type opcodes to modify a transform object	7-21
<b>Figure 7-7</b>	Omit byte relationship with the data that follows	7-23
<b>Figure 7-8</b>	Select the bits from the omit byte	7-24

<b>Table 7-10</b>	Constants from the <code>gxOmitTextMask</code> and the <code>gxOmitTextShift</code> enumerations 7-24
<b>Figure 7-9</b>	Compare the bits selected and shifted with the compression enumeration 7-25
<b>Table 7-11</b>	Correlation between <code>gxOmitTextMask</code> and the <code>GXNewText</code> function 7-26
<b>Listing 7-1</b>	Determining if <code>position(x)</code> is byte compressed 7-27
<b>Figure 7-10</b>	Mapping matrix elements 7-40
<b>Table 7-12</b>	Color space and words read 7-44
<b>Table 7-13</b>	Bit image compression opcodes 7-50
<b>Figure 7-11</b>	Print file format 7-52
<b>Listing 7-2</b>	A <code>GraphicsBug</code> annotation of the data stream of a flattened shape 7-55
<b>Listing 7-3</b>	A picture with seven shapes 7-56
<b>Figure 7-12</b>	A picture with seven shapes 7-59
<b>Figure 7-13</b>	The line shape drawn 7-60
<b>Listing 7-4</b>	<code>GraphicsBug</code> analysis of a flattened line 7-60
<b>Table 7-14</b>	Analysis of the data stream of a flattened line shape 7-61
<b>Figure 7-14</b>	The rectangle shape drawn 7-64
<b>Listing 7-5</b>	<code>GraphicsBug</code> analysis of a flattened rectangle shape 7-64
<b>Table 7-15</b>	Analysis of the data stream of a flattened rectangle shape 7-65
<b>Figure 7-15</b>	The curve shape drawn 7-67
<b>Listing 7-6</b>	<code>GraphicsBug</code> analysis of a flattened curve shape 7-67
<b>Table 7-16</b>	Analysis of the data stream of a flattened curve shape 7-68
<b>Figure 7-16</b>	The path shape drawn 7-69
<b>Listing 7-7</b>	<code>GraphicsBug</code> analysis of a flattened path shape 7-69
<b>Table 7-17</b>	Analysis of the data stream of a flattened path shape 7-70
<b>Figure 7-17</b>	The text shape drawn 7-72
<b>Listing 7-8</b>	<code>GraphicsBug</code> analysis of a flattened text shape 7-73
<b>Table 7-18</b>	Analysis of the data stream of a flattened text shape 7-74
<b>Figure 7-18</b>	The polygon shape drawn 7-79
<b>Listing 7-9</b>	<code>GraphicsBug</code> analysis of a flattened polygon shape 7-79
<b>Table 7-19</b>	Analysis of the data stream of a flattened polygon shape 7-80
<b>Figure 7-19</b>	The bitmap shape drawn 7-81
<b>Listing 7-10</b>	<code>GraphicsBug</code> analysis of a flattened bitmap shape 7-82
<b>Table 7-20</b>	Analysis of the data stream of a bitmap shape 7-83
<b>Listing 7-11</b>	Obtaining the page count from a portable digital document print file 7-89

## Chapter 8

### QuickDraw GX Mathematics 8-1

<b>Table 8-1</b>	Macro number-format conversions 8-8
<b>Figure 8-1</b>	Cartesian and polar coordinates 8-11
<b>Figure 8-2</b>	Transformation operations with a mapping matrix 8-14
<b>Figure 8-3</b>	Mapping matrix elements 8-15
<b>Figure 8-4</b>	Applying a mapping matrix to a point 8-15
<b>Figure 8-5</b>	The point (x, y) as transformed by the mapping matrix 8-16
<b>Figure 8-6</b>	The identity matrix 8-17
<b>Figure 8-7</b>	Changing the translation specified by a mapping 8-17
<b>Figure 8-8</b>	Translation by a relative amount with <code>MoveMapping</code> 8-18
<b>Figure 8-9</b>	Setting the origin specified by a mapping 8-19

<b>Figure 8-10</b>	Translation to a specific origin location	8-19
<b>Figure 8-11</b>	Changing the amount of scaling specified by a mapping	8-20
<b>Figure 8-12</b>	Scaling horizontally and vertically	8-21
<b>Figure 8-13</b>	Changing the degree of rotation specified by a mapping	8-22
<b>Figure 8-14</b>	Rotating about different center points	8-23
<b>Figure 8-15</b>	Changing the amount of skew specified by a mapping	8-24
<b>Figure 8-16</b>	Skewing a shape both horizontally and vertically	8-25
<b>Figure 8-17</b>	Changing the perspective specified by a mapping	8-26
<b>Table 8-2</b>	QuickDraw GX and Macintosh Toolbox fixed-point functions	8-27
<b>Figure 8-18</b>	Determining the length of a line with the <code>Magnitude</code> function	8-28
<b>Figure 8-19</b>	Converting between Cartesian and polar coordinates	8-29
<b>Listing 8-1</b>	Calculating a cross-product with <code>VectorMultiply</code>	8-30
<b>Listing 8-2</b>	Applying a mapping to one point	8-30
<b>Listing 8-3</b>	Using the <code>wideShift</code> function to create a fixed-point <code>VectorMultiply</code> function	8-31
<b>Listing 8-4</b>	Using the <code>wideShift</code> function in a fixed-point multiplication function	8-31
<b>Listing 8-5</b>	Using the <code>wideShift</code> function to create a fixed-point division function	8-31
<b>Listing 8-6</b>	Using the <code>wideShift</code> function to create a second fixed-point division function	8-32
<b>Listing 8-7</b>	Using the <code>wideScale</code> function to create a pseudo-floating-point function	8-32
<b>Listing 8-8</b>	A random number generator	8-33
<b>Listing 8-9</b>	Determining the lowest bit of a number	8-34
<b>Table 8-3</b>	<code>FixedMultiply</code> product bias	8-43
<b>Table 8-4</b>	<code>FixedDivide</code> quotient bias	8-44
<b>Table 8-5</b>	<code>FractMultiply</code> result bias	8-48
<b>Table 8-6</b>	<code>FractDivide</code> result bias	8-49