Programming Java 2
Micro Edition
on Symbian OS
Contents

About This Book ix
Author Biographies xiii
Author’s Acknowledgements xvii
Symbian Press Acknowledgements xix
Foreword xxii
Innovation Through Openness xxiii

Section 1: J2ME and MIDP 1

1 Introduction to J2ME 3
   1.1 Configurations and Profiles 3
   1.2 CLDC and MIDP 7
   1.3 CDC and Personal Profile 16
   1.4 J2ME on Symbian OS 21
   1.5 Summary 22

2 Getting Started 23
   2.1 Introduction to MIDP 23
   2.2 Helloworld, Turbo Edition 46
   2.3 Introduction to Tools for MIDP 54
   2.4 Installing and Running a MIDlet 82
   2.5 MIDP on Symbian OS Phones 89
   2.6 Summary 89
3 MIDP 2.0 and the JTWI 91
  3.1 Introduction to the JTWI 91
  3.2 The CLDC on Symbian OS 94
  3.3 MIDP 2.0 95
  3.4 Optional J2ME APIs in the JTWI 155
  3.5 MIDP 2.0 and Symbian OS Phones 201
  3.6 Summary 202

4 Java APIs for Bluetooth Wireless Technology 205
  4.1 Introduction to Bluetooth 205
  4.2 Introduction to the Bluetooth APIs 206
  4.3 Programming the Bluetooth APIs 208
  4.4 L2CAP Protocol 224
  4.5 Security 227
  4.6 Java Bluetooth API and the MIDP 2.0 Security Model 229
  4.7 Sample Code 230
  4.8 Development Tools 241
  4.9 Java Bluetooth APIs and Symbian OS 244
  4.10 Summary 244

5 MIDP 2.0 Case Studies 247
  5.1 Introduction 247
  5.2 The Expense Application 248
  5.3 The Demo Racer Game 282
  5.4 The Picture Puzzle 294

Section 2: Writing Quality Code for Smartphones 317

6 Making Java Code Portable 319
  6.1 Introduction 319
  6.2 Design Patterns 320
  6.3 Portability Issues 326
  6.4 Summary 333

7 Writing Optimized Code 335
  7.1 Introduction 335
  7.2 What Are We Starting With? 336
  7.3 Benchmarking 336
  7.4 General Guidelines for Optimization 337
  7.5 Feedback and Responsiveness 338
  7.6 Object Creation 338
  7.7 Method Modifiers and Inlining 340
  7.8 Strings 343
CONTENTS vii

7.9 Using Containers 348
7.10 How Not To Do It 349
7.11 Copying an Array 351
7.12 Thoughts on Looping 352
7.13 Graphics 358
7.14 LifeTime Case Study 366
7.15 Arithmetic Operations 385
7.16 Design Patterns 386
7.17 Memory Management 388
7.18 JIT and DAC Compilers 390
7.19 Obfuscators 391
7.20 Summary 392

Section 3: The Evolution of the Wireless Java Market 393

8 The Market, the Opportunities and Symbian’s Plans 395
8.1 Introduction 395
8.2 The Wireless Java Market 395
8.3 Meeting Market Needs 400
8.4 Providing Advanced Services 402
8.5 Why Java? 406
8.6 Symbian and Java 409
8.7 Java and Digital Rights Management 418
8.8 The Java Verified Program 420
8.9 Beyond Advanced Consumer Services 421
8.10 Trends in Technology 421

Appendix 1: CLDC Core Libraries 423

Appendix 2: MIDP Libraries 429

Appendix 3: Using the Wireless Toolkit Tools at the Command Line 437

Appendix 4: Developer Resources and Bibliography 439

Appendix 5: Specifications of Symbian OS Phones 445

Index 461
About This Book

In 2001, Symbian’s first book devoted to Java on Symbian OS was published. Jonathan Allin’s *Wireless Java for Symbian Devices* (WJSD) provided an in-depth exposition targeted at programming PersonalJava on Symbian OS. The embedded Java story has moved on a lot in two years and so has Symbian’s implementation, so once again we decided to put pen to paper to produce a new book aimed at helping developers program Java on the latest generation of Symbian OS phones.

This book is not intended to supersede Jonathan Allin’s WJSD, which dealt very thoroughly with Symbian’s PersonalJava implementation and still remains the definitive guide for developers programming PersonalJava on Symbian devices such as the Nokia 9200 or Sony Ericsson P800 and P900. Instead, this new book covers very different territory, focusing on programming MIDP, particularly MIDP 2.0, on Symbian OS.

Symbian’s Java implementation has evolved over the years from a JDK 1.1.4-based implementation in Symbian OS Version 5.0, through PersonalJava on Symbian OS Version 6.0 and is now moving, with Symbian OS Version 7.0 and subsequent releases, to a single Java 2 Micro Edition (J2ME) CLDC/MIDP-based implementation. The latest generation of Symbian OS phones support MIDP 2.0 plus a range of additional, optional APIs, all conforming to Java Specification Requests (JSRs) arrived at through the Java Community Process.

Phones based on the latest releases of Symbian OS, such as the Nokia 6600 and Sony Ericsson P900, support MIDP 2.0 as well as implementations of the Wireless Messaging API (JSR 120), Java Bluetooth API (JSR 82) and also, in the case of the Nokia 6600, the Mobile Media API (JSR 135).

This book is not just about MIDP 2.0. Instead we will show developers how to get the best out of the latest generation of Symbian OS phones, by providing a practical, in-depth, guide to programming J2ME on these devices. In addition to a thorough discussion of MIDP we have also included an in-depth exposition of all the optional J2ME APIs that can be found on phones such as the Sony Ericsson P900 and Nokia 6600.
Our approach has been to illustrate the new MIDP 2.0 features and optional APIs by way of concrete examples tested on real devices. In addition to extensive sample code we include a chapter of case studies that develop more or less complete applications. By adopting this approach developers will be equipped with code that they know will run on real devices. Where specifications allow optional functionality we indicate whether this is supported on Symbian phones. We also point out known bugs and possible work-arounds. In addition we aim to use the considerable experience available within Symbian to show the reader how to write efficient and effective code for constrained devices. To complete the picture we also discuss what Java has to offer in the wireless space and how it may enrich the wireless value chain. We also provide an insight into how Java is likely to evolve on Symbian OS in the future.

In writing this book, our desire has been to give enough information in one volume for developers to make the most of the Java 2 Micro Edition on Symbian OS, enabling them to provide the compelling content that will enrich the wireless ecosystem.

The book is divided three sections:

1. Section 1: J2ME and MIDP
2. Section 2: Writing Quality Code for Smartphones
3. Section 3: The Evolution of the Wireless Java Market

In Section 1 we introduce the Java 2 Micro Edition and the ideas behind configurations and profiles. We then concentrate on programming MIDP and the additional APIs that make up the Java platform on the latest generation of Symbian OS phones.

Section 2 investigates design and implementation considerations involved in writing high-quality code, focusing on the issues of portability and efficiency.

The final section looks at the strategic importance of Java to the wireless ecosystem and provides a glimpse as to how Wireless Java may evolve on Symbian OS.

Who Is This Book For?

The book is aimed at Java developers already programming in the wireless space or interested in moving into the wireless space and who wish to know what can be achieved with J2ME on the latest Symbian OS phones. Enough introductory information and examples are provided for newcomers to J2ME to get going with MIDP programming, while the thorough treatment of the new MIDP 2.0 and optional APIs provides more weighty fare for the experienced MIDP programmer.
Conventions

To help you get the most from the text and keep track of what’s happening, we’ve used a number of simple conventions throughout this book.

When we refer to words you use in your code, such as classes, attributes and methods, or to the name of a file, we use this style:

Person class: we obtain the name attribute by invoking the getName method on our Person instance

When we list code, or the contents of files, we use the following convention:

```
SocketConnection conn = (SocketConnection)Connector.open(url);
DataOutputStream out = conn.openDataOutputStream();
byte[] buf= request.getBytes();
out.write(buf);
out.flush();
out.close();
```

We show commands typed at the command line like this:

```
C:\WTK20\apps\Example\src>javac -d tmpclasses -bootclasspath %MIDPAPI% -classpath %J2MECLASSPATH% *.java
```

URLs are written: www.symbian.com/developer
Author Biographies

Martin de Jode

Martin graduated from the University of York with a BSc in Physics and, after a brief spell in industry, returned to academia to undertake research in the field of non-linear optics at Essex University. Graduating with a PhD, Martin spent eight years working in research at the London Hospital Medical College, studying the use of lasers to treat cancer. During this time he developed a particular interest in using Monte Carlo simulation to model the interaction of light with biological tissue using Fortran.

Martin joined Symbian in 2000, after completing an MSc in Object Oriented Software Systems from City University. As a Java Developer Consultant in Symbian’s Developer Network he spends his time providing support to Symbian’s Java developer community. In addition to writing technical papers on PersonalJava and J2ME for Symbian’s website, Martin has developed numerous utility and sample applications showing how to make the most of Symbian’s Java platform. Other activities include delivering training, evangelizing Java on Symbian OS and trying to keep up with the proliferation of J2ME JSRs.

Outside of work, Martin is passionate about cricket, having spent what seems like a lifetime playing competitive club cricket in and around the home counties.

Jonathan Allin

Jonathan is Symbian’s Product Manager for Java Technology. His role is to ensure that Symbian OS provides a first class Java platform for mobile phones, and covers Symbian’s Java strategy and implementation roadmap, partnerships, and, of course, how Java relates to other development environments. Jonathan was the lead author of Wireless Java for Symbian Devices, authored the “Developing with Java” chapter in