Для начинающих

Создание игр

Для мобильных телефонов

Создайте четыре игры:

- Henway
- High Seas
- NetConnect4
- Space Out

На CD:

- Полный список кодов и готовые программы всех игр, описанных в книге
- Примеры разработки игр на языке Java

М. Моррисон

ДМК

Майкл Моррисон
Создание игр для мобильных телефонов

Текст предоставлен издательством
http://www.litres.ru/pages/biblio_book/?art=631885
Создание игр для мобильных телефонов: Издательский дом ДМК-пресс; Москва; 2006
ISBN 5-9706-0007-5

Аннотация

Книга «Создание игр для мобильных телефонов» – это практическое руководство, которое поможет разработать и реализовать игру для мобильного телефона.

Книга написана простым языком, не содержит сложной и скучной теории программирования и шаг за шагом знакомит с методикой создания технологии «plug-and-play» применительно к созданию огромного количества игр.

В издание включены подробные описания и примеры кодов для четырех игр, а также информация, необходимая для реализации вашей собственной задумки.

Если вы любите играть в игры и занимаетесь программированием, то эта книга – для вас!

Компакт-диск прилагается только к печатному изданию.

Содержание

Об авторе 5
Посвящение 6
Благодарность 7
Мы хотим услышать вас 8
Поддержка книги 10
Введение 11
Как построена эта книга 19
Что вам потребуется 21
Часть I 23
Глава 1 23
Основы программирования игр для мобильных телефонов 24
Первая игра для мобильного телефона 27
Рынок мобильных игр 28
Культура мобильных игр 30
Сильная сторона мобильных игр 32
Знакомство с мобильными платформами 35
Java 2 Micro Edition (J2ME) 37
Binary Runtime Environment for Wireless (BREW) 38
Symbian 39
Windows Mobile Smartphone 41
Java как платформа для мобильных игр 42

Что такое Java? 43
Почему Java? 43
Java и программирование мобильных игр 45
Небольшой пример на J2ME 49
Конфигурация и ограниченная конфигурация мобильного устройства 51
Профили и MIDP 52
Резюме 57
Экскурсия 57
Глава 2 59
Основы разработки игр 60
Основная идея 61
Разработка сюжетной линии 63
Режимы игры 63
Пример разработки игры на J2ME 64
Знакомство с J2ME Wireless Toolkit 70
Использование KToolbar 72
Управление проектами J2ME 74
Сборка мидлета 78
Тестирование игрового мидлета 80
Эмулятор J2ME и реальные устройства 86
Резюме 88
Конец ознакомительного фрагмента. 90

Майкл Моррисон Создание игр для мобильных телефонов

Об авторе

Майкл Моррисон (Michael Morrison) – разработчик и создатель игр, а также автор различных книг по компьютерным технологиям и интерактивных курсов в Internet. Кроме своей основной профессии писателя и фрилансера, Майкл является креативным директором компании Stalefish Labs, развлекательной компании, которую он основал вместе со своей женой Машид (Masheed). Коммерческим дебютом этой компании была игра Tall Tales: The Game of Legend and Creative One-Upmanship (http://www.talltalesgames.com/). Когда Майкл не сидит за компьютером, не играет в хоккей, не катается на скейтборде и не смотрит фильмы со своей женой, он любит гулять у пруда. Вы можете посетить сайт Майкла в Internet (http://www.michaelmorrison.com/).

Посвящение

Моему давнему другу Рэнди Вимсу (Randy Weems), который научил меня почти всему, что я знаю о программировании игр, и помог разработать мою первую игру для мобильного телефона. Мы потратили на это целую ночь почти 15 лет назад.

Благодарность

Спасибо Майку Стефенсу (Mike Stephens), Лоретте Ятс (Loretta Yates), Марку Ренфроу (Mark Renfrow) и другим замечательным людям из компании Sams Publishing House за то, что они превратили работу над этой книгой в интересный опыт. Также хочу выразить огромную благодарность моему другу и жене Машид за поддержку.

Мы хотим услышать вас

Как читатель этой книги вы – самый важный критик и рецензент. Мы ценим ваше мнение и хотим знать – что делаем правильно, а что, с вашей точки зрения, можно улучшить, какие разделы мы должны еще осветить или рассмотреть более полно.

Как издатель Sams Publishing я жду ваших комментариев. Вы можете отправить мне электронное или обычное письмо, рассказать, что вам понравилось в книге, а что нет, а также поделиться мнением о том, как мы можем сделать наши книги лучше.

Пожалуйста, обратите внимание, что я не смогу вам помочь в решении технических проблем, связанных с темой этой книги. Для этого у нас существует специальная служба поддержки, куда я перенаправлю все технические вопросы.

Пожалуйста, не забудьте указать автора и название книги, а также свое имя, адрес электронной почты и телефон. Я тщательно изучу ваши комментарии и передам их автору и редакторами, работавшими над изданием.

Электронная почта: feedback@samspublishing.com

Почтовый адрес: Michael Stephens,
Associate Publisher,
Sams Publishing,
800 East 96th street,

Indianapolis, IN 46240 USA

Поддержка книги

Если вы хотите узнать больше об этом и других изданиях Sams Publishing, посетите наш Web-сайт (www.samspublishing.com). В поле Search (Поиск) введите ISBN (0672326655) или название книги, которую вы ищете.

Введение

Итак, я сижу в кресле стоматолога, жду, пока подействует укол новокаина и мне запломбируют коренной канал. Внезапно ко мне приходит мысль, что я могу потратить это время с куда большей пользой, чем просто ждать. Хотя стоматологическое кресло и слюнявчик ограничивают свободу действий, они не смогут помешать мне сразиться с несколькими сотнями моих близких друзей и заклятых врагов. Быстрым движением руки я достаю из кармана мобильный телефон, поддерживающий Java, и запускаю игру, которая поможет мне не думать об устрашающих инструментах дантиста.

Некоторое время назад я и сам скептически относился к тому, что можно на самом деле сделать на устройстве, которое я часто поднимаю с земли и молю Бога о том, чтобы оно работало. Давайте рассмотрим проблему: мобильные телефоны – это совсем не то, что мы представляли себе, когда думали о будущем интерактивных развлечений. Однако теперь мобильные телефоны превратились в игровые машины, и если вы читаете эту книгу, очевидно, вам хочется проникнуть в мир игр.

Хотя логично объединить в одну группу мобильные телефоны и другие типы переносных устройств (Palm Plot, Pocket PC и Game Boy), в этой книге рассматривается вопрос программирования игр именно для мобильных телефонов. Я не

против других устройств, однако ни одно из них не получило столь широкое распространение. Вспомните хотя бы пятерых ваших знакомых и посчитайте, сколько у них сотовых телефонов. Если ответ меньше, чем 4, то дайте им еще пару лет. Вне зависимости от того, смогла ли убедить вас моя речь, мобильные телефоны завоевывают мир с огромной скоростью, и она постоянно увеличивается!

Итак, обладатели персональных коммуникаторов (мобильных телефонов) используют их главным образом для общения друг с другом. Но в наше время появляются новые технологии, которые помогут расширить границы их применения. Технологии, подобные Java, наделяют мобильные телефоны практически теми же возможностями, которыми обладают настольные компьютеры. Совместите это со способностью всех мобильных телефонов поддерживать работу с беспроводными сетями, и вы получите уникальное устройство: широко распространенное, компактное, переносное, работающее с сетями, программируемое – просто мечта для создателя игр!

Я бы солгал, если сказал бы, что программирование игр для мобильных телефонов уже хорошо развито. Технологии новы, а модели телефонов, поддерживающие их, еще более новы. Но, как вы, вероятно, знаете, чтобы быть в курсе новых технологий, нужно быть на шаг впереди них. Учась разрабатывать и создавать игры для мобильных телефонов по мере развития этой сферы, вы сможете в большей степени пользо-

ваться открывающимися возможностями. Игра ли это, которой вы хотите поделиться с друзьями и родственниками, новый хит в среде коммерческих игр, который обеспечит вам место в зале славы создателей игр, – в любом случае в этой книге вы найдете все, что нужно для начала работы. На рис. 1 показан пример игры, которую вы создадите, пользуясь этой книгой.

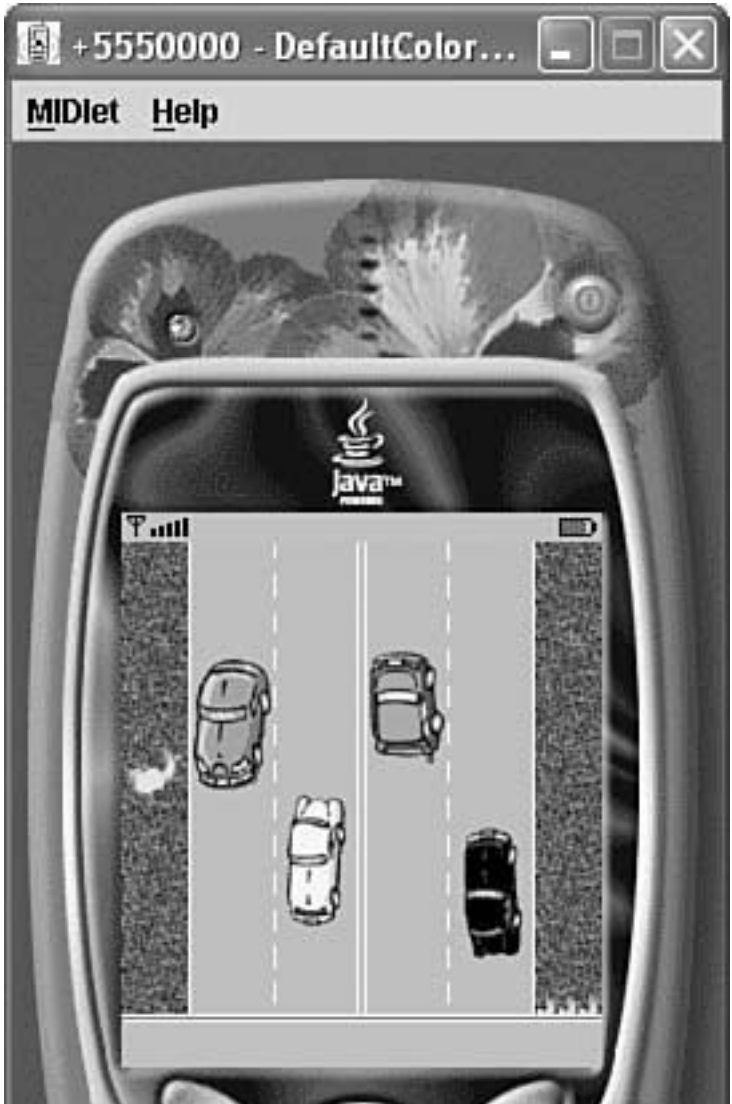
+5550000 - DefaultColor...
MIDlet
Help
Java

Рис. 1. Игра Henway очень похожа на классическую аркаду Frogger

Игра Henway, показанная на рис. 1, очень похожа на классическую аркаду Frogger. Если вам не по душе переводить цыпленка через дорогу, вас, вероятно, привлечет другая игра – High Seas (рис. 2).

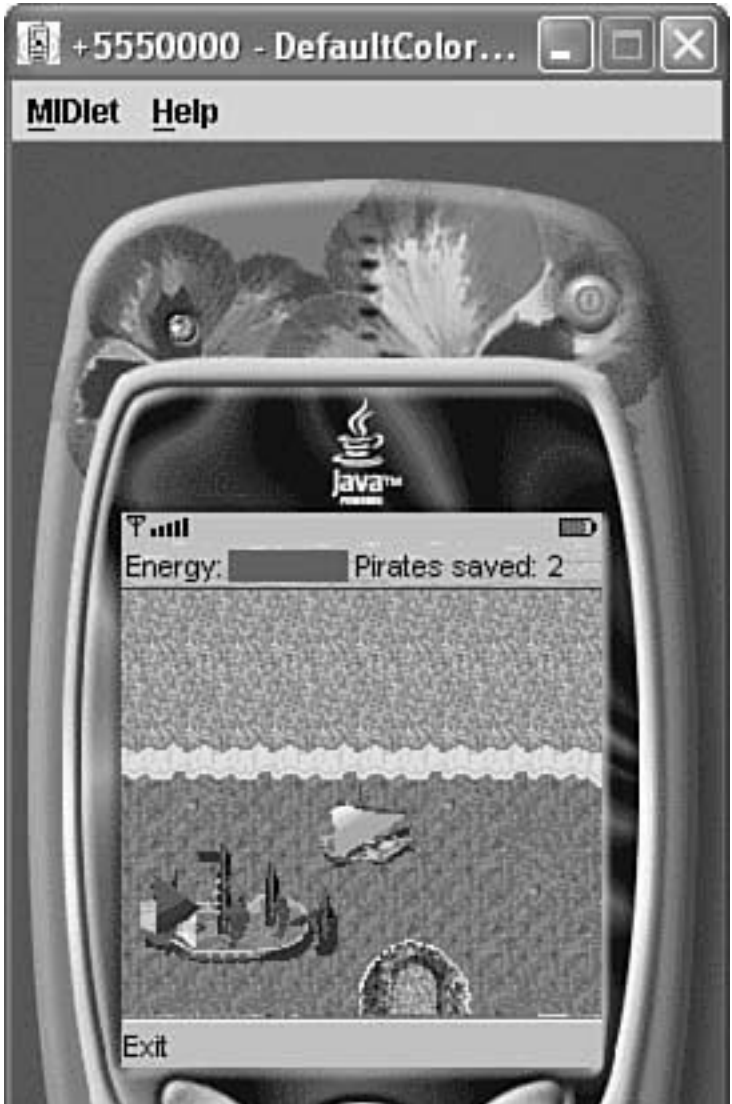
+5550000 - DefaultColor...
MIDlet
Help
Energy:
Pirates saved: 2
Exit

Рис. 2. Игра High Seas впитала все преимущества перемещающегося фона и «интеллектуальных» компьютерных врагов

High Seas – это игра, в которой вы бороздите морские просторы и спасаетесь от пиратов, избегаете спрутов, плавающих мин и большого корабля пиратов. И это только две игры из пяти, которые вы создадите, работая с книгой. Пусть остальные игры станут для вас сюрпризом!

Для написания игр для мобильных телефонов в этой книге отдано предпочтение языку программирования Java, и для этого есть причина. Я более подробно рассмотрю этот вопрос в главе 1, но если говорить коротко, Java – это доминирующий инструмент разработки игр в настоящее время и в обозримом будущем. Если вы не знаток Java, то на прилагаемом компакт-диске можно найти программу обучения этому языку программирования, «Java Programming Primer». Вне зависимости от того, как вы относитесь к Java в настоящее время, думаю, что к концу прочтения книги, вы согласитесь, что это идеальная технология для разработки игр для мобильных телефонов.

Меня часто спрашивают, какой мобильный телефон я использую для отладки игр. Я отвечаю, что огромный, который лежит на моем столе. Я говорю о своем настольном компьютере, который значительную часть времени я использую в качестве эмулятора мобильного телефона. На сегодняш-

ний день на рынке так много телефонных аппаратов, а новые модели появляются настолько часто, что было бы невозможно порекомендовать какую-то определенную модель, не перенося сроки издания этой книги. Поэтому я советую использовать эмулятор Java, входящий в состав J2ME Wireless Toolkit, который вы можете найти на сопроводительном CD. Конечно, вы наверняка захотите протестировать созданные игры и на реальном сотовом телефоне, но вы оцените, насколько удобно применять эмулятор.

Так же, как и программирование игр для настольных компьютеров и консольных систем, создание игр для мобильных телефонов – дело непростое. Вам придется использовать и комбинировать различные приемы и методы программирования, не забывая и про хорошую порцию креативного мышления. Именно сочетание креативного мышления и технических навыков делает программирование игр столь притягательным. Добавьте к этому еще необходимость создания игры на миниатюрном устройстве, имеющем беспроводное соединение с сетью, и вы получите рецепт «техновеселья»!

Как построена эта книга

Эта книга разделена на пять частей, каждая из которых затрагивает различные вопросы программирования игр:

 # Часть I: «Основы программирования игр для мобильных телефонов». В этой части вы познакомитесь с основами разработки мобильных игр на основе языка Java и познакомитесь с J2ME Wireless Toolkit. Вы создадите «скелет» игры для мобильного телефона, который будете впоследствии использовать при работе с книгой для написания остальных игр, протестируете созданную игру с помощью эмулятора мобильного телефона Java.

► **Часть II: «Специфика создания мобильных игр».** В этой части книги вы научитесь использовать графику в мобильных играх, создавать изображения. Вы также узнаете об анимации с использованием спрайтов (спрайт – небольшое изображение, переносимое по экрану независимо от других. – Прим. перев.). Этот метод создания анимации является основным приемом программирования Вы также разработаете две игры: Henway и Cat Catcher. Примечательно, что в этой части книги рассматриваются игры, героями которых являются животные. Не волнуйтесь, чуть позже мы доберемся до пиратов и пришельцев.

► **Часть III:** «Виртуальные миры и интеллектуальные мобильные игры».

Эта часть познакомит вас с применением слоев и их использованием для создания игр с перекрывающимися объектами. Вы познакомитесь с основами создания Искусственного Интеллекта (ИИ), узнаете, почему он столь важен для игр. Тема ИИ очень сложна, поэтому я обращу ваше внимание лишь на основные и несложные приемы, которые вы сможете воплотить в своих играх. Также в эту части книги включено создание еще одной игры, High Seas, в которой вы будете бороздить морские просторы и сражаться с пиратами и морскими чудовищами.

► **Часть IV: «Преимущества беспроводной сети».** В этой главе вы узнаете, как использовать главную возможность мобильного телефона – беспроводную сеть. Научившись основам сетевого программирования игр, вы создадите игру NetConnect4 – сетевой аналог популярной игры Connect4. А затем вы разработаете и создадите игру Mad Plumber, в которой вам придется соревноваться с противником в скоростной прокладке водопроводных труб.

► **Часть V: «Совершенствование игр».** Из этой главы вы почерпнете ряд интересных приемов программирования игр, которые помогут вам сделать игры как можно более красивыми. Также вы узнаете, как создавать и сохранять в телефоне список лучших игроков. В этой части книги вы создадите еще одну игру, Space Out, – космическую «стрелялку», для разработки которой вам потребуется применить все знания, полученные при работе с предыдущими частями.

Что вам потребуется

Предполагается, что вы знаете и понимаете Java. В сущности, я не полагаюсь на сложные программные конструкции этого языка, поэтому вам достаточно знать его основы. Если ваши знания языка Java уже запылились, воспользуйтесь самоучителем Java, который находится на прилагаемом компакт-диске. Книга не предполагает навыков использования Java для создания мобильных игр, не волнуйтесь, если вы никогда не держали в руках телефон, поддерживающий Java.

Все примеры, рассматриваемые в книге, вы можете найти на прилагаемом CD, включая командные файлы, используемые в командной строке J2ME Wireless Toolkit для построения и запуска игр. Пакет J2ME Wireless Toolkit вы также можете найти на сопроводительном компакт-диске. Как я уже упоминал, в составе пакета J2ME Wireless Toolkit вы также найдете инструмент KToolbar, который чрезвычайно полезен для создания и тестирования примеров. Все примеры в книге разработаны так, что их можно легко открыть, откомпилировать и эмулировать с помощью KToolbar.

Помимо некоторых знаний Java, вам потребуется ясный ум и немного творческого подхода, чтобы достичь лучшего результата от работы с книгой. Эта книга послужит вам отправной точкой в увлекательном путешествии в мир создания игр для мобильных телефонов. И если

вдруг вы собьетесь с пути, причальте на мой сайт, http://www.michaelmorrison.com/! Здесь, в форуме, посвященном книге, вы, вероятно, сможете найти ответы на мои вопросы. Веселитесь!

Часть I
Основы программирования игр для мобильных телефонов

Глава 1
История электронных игр

Архив Аркад

Выпущенная в 1971 компанией Nutting Associates игра Computer Space заслуженно называется первой аркадой. Эта игра была создана Ноланом Бушнеллом (Nolan Bushnell). Годом позже он основал собственную компанию Antari и продолжил разжигать революцию в мире компьютерных игр, создав Pong. Computer Space – очень простая игра, но ее физическая и электрическая реализации послужили основой для всех последующих аркад. В автомате монеты складывались в покрашенную консервную банку – этот элемент конструкции не был повторен в дальнейшем! Игру Computer Space можно увидеть в классическом фильме 1975 года «Челюсти», в сцене, где эта аркада находится рядом с пляжем.

Обращаясь к прошлому, по крайней мере, к ранним 80-м, когда были популярны спортивные электронные игры, мо-

бильные игры – не новинка. Наиболее распространенная современная электронная игровая система – Nintendo Game Boy – претерпела ряд модификаций, но осталась популярной по сей день. Мобильные компьютерные игры всегда уступали своим «полноценным» вариантам, главным образом потому, что сложно уместить большую производительность в маленьком приборе. Но положение стремительно изменяется, последние модели переносных компьютеров и мобильных телефонов показали, что теперь настоящая вычислительная мощь может поместиться и в ваш карман. С ростом производительности карманных систем увеличились возможности разработчиков игр для мобильных телефонов. Как создателю мобильных игр вам очень важно понимать инструменты и технологии, смежные с процессом разработки игр.

В этой главе вы узнаете:

► о широком рынке мобильных игр;

► об обозримых перспективах игр для мобильных телефонов;

► почему Java является идеальной платформой для разработки мобильных игр;

► что такое J2ME, и каково его место среди языков Java.

Основы программирования игр для мобильных телефонов

По определению, мобильная игра – это игра, в которую

вы можете играть в движении. Хотя вы можете не согласиться и сказать, что согласно этому определению игра на ноутбуке – это тоже мобильная игра. Мы ограничим это определение, сказав, что это игра, в которую можно играть на мобильном устройстве, помещающемся на ладони. Но тогда в эту категорию попадут мобильные телефоны, пейджеры, «наладонники», персональные цифровые помощники и карманные игровые устройства. Поэтому для целей этой книги мы будем рассматривать только те игры, которые можно запускать на мобильном телефоне. Хотя это может показаться весьма деспотичным решением, вы увидите, что мобильные телефоны представляют собой уникальный и очень важный тип устройств для создания мобильных игр.

Почему в книге уделяется внимание лишь мобильным телефонам, а не рассматриваются такие устройства, как PDA? Дело в том, что в отличие от ноутбуков, PDA и даже переносных игровых устройств, мобильные телефоны широко распространены во многих социальных и экономических категориях. Поскольку всем близка идея максимально гибкой коммуникации, мобильные телефоны могут не угодить только компьютерщикам, любителям технических новинок или геймерам. Мобильные телефоны – одни из самых мощных компактных устройств, подходящих для программирования игр. Такое сочетание огромной аудитории и быстро растущих технических возможностей делает мобильные телефоны новым рубежом для разработчиков игр.

В копилку Игрока

Поставщики беспроводных услуг связи уже создали инфраструктуру продажи и доставки мобильных игр пользователям. В отличие от традиционных игр для персональных компьютеров и игровых консолей, для которых необходима упаковка, CD-ROM и место на полке в магазине, мобильные игры можно доставлять на рынок прямо по воздуху. Это значительное подспорье для тех, кто хочет ворваться на рынок мобильных игр.

Хорошим аргументом может послужить то, что мобильные телефоны открывают значительные возможности для разработчиков игр. Это приводит нас к выводам: что необходимо для разработки мобильных игр, что, в свою очередь, раскрывает технологические причины, почему эта книга рассматривает вопросы создания игр только для мобильных телефонов. Имея ограниченные аппаратные возможности и уникальные требования операционных систем мобильных телефонов, игры существенно отличаются от тех, что создаются для других устройств. Разработка игр для мобильных телефонов требует учета специфических особенностей и требований.

Поэтому, хотя вы можете перенести игру, написанную для настольного или переносного компьютера, в реальность, мобильные игры требуют особых приемов разработки, чтобы они могли удовлетворять особым ограничениям, накладываемым их размером и технологической спецификой. Кро-

ме того, мобильные телефоны поддерживают работу с беспроводными сетями, поэтому многие мобильные игры разрабатываются с учетом возможности одновременной игры нескольких участников посредством беспроводной сети.

Первая игра для мобильного телефона

Чтобы понять, что представляют собой игры для мобильных телефонов, полезно обратиться к истории. Я хочу вернуться в 1997 год, когда была создана первая мобильная игра, поставлявшаяся на телефонах производства фирмы Nokia. Эта игра называлась Snake и была очень простой как с точки зрения графики, так и с точки зрения самого процесса игры: вы управляете бегающей по экрану змеей, которая должна съедать шарики и расти (рис. 1.1).

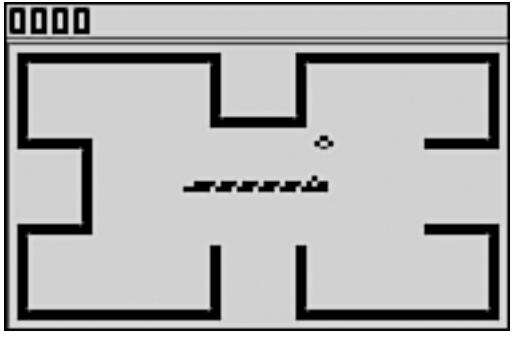

Рис. 1.1. Классическая игра Snake – пример одной из первых игр для мобильных телефонов

Несмотря на то что Snake очень простая игра, она навсегда изменила образ мобильного телефона – он стал не только средством общения. Вскоре после появления Snake на свет появилось огромное количество других игр, и люди начали больше думать о телефоне как о средстве развлечения.

В копилку Игрока

Игра Snake – компьютерная игра, существовавшая задолго до появления мобильных телефонов. Некоторые предыдущие версии этой игры появились на персональных компьютерах Commodore VIC-20 и Commodore 64, хотя я не удивлюсь, если узнаю, что она появилась еще раньше.

Важно понять, что игра Snake – это быстрая игра, длящаяся всего несколько минут или даже секунд. Даже без дополнительных пояснений вы можете быстро разобраться в игре и веселиться! Не страшно, если игру прервет телефонный звонок, потому как вы с легкостью сможете к ней вернуться. В этом смысле Snake заключает истинный дух мобильный игры – она простая, интуитивно понятная и веселая. Несомненно, вы увидите и сложные игры для мобильных телефонов, но я думаю, что пользователи мобильных телефонов предпочитают игры, подобные Snake.

Рынок мобильных игр

Если логика людей, ожидающих авиарейса или находя-

щихся в кресле пациента, вас не убедила о перспективах рынка мобильных игр, то, вероятно, это смогут сделать некоторые цифры. По оценкам одной нью-йоркской исследовательской фирмы, в 2005 году 200 миллионов человек будут играть в игры на своих мобильных телефонах, тем самым создав рынок стоимостью около 6 миллиардов долларов. Это не долгосрочный прогноз – это реальность! Ключ к пониманию цифр заложен в том, чтобы осознать, что несмотря на общий спад продаж на рынке мобильных телефонов число пользователей этих устройств будет постоянно расти и значительно увеличится в течение следующих нескольких лет. Это и является причиной появления новых моделей телефонов, обладающих более обширными возможностями поддержки игр.

По некоторым оценкам, 200 миллионов в 2005 году – это весьма скромная цифра. Некоторые эксперты в сфере беспроводных технологий предсказывают, что в 2006 году число людей, играющих в мобильные игры, достигнет 850 миллионов. Это очень много! Если вы сравните эти цифры с числом обладателей традиционных игровых приставок или персональных компьютеров, то увидите, что коммерческий потенциал рынка мобильных телефонов чрезвычайно высок.

Но, может быть, я перегибаю палку. Может быть, вы хотите создавать игры лишь для собственного удовольствия и не стремитесь завоевать львиную долю мирового рынка мобильных игр. Может быть, вы смотрите на мобильные игры

как на средство общения с другими людьми, средство дружеских соревнований. Поэтому давайте оставим капитализм в стороне, и взглянем на мобильные игры с социальной точки зрения.

Культура мобильных игр

Как и в случае программ для обмена короткими сообщениями, которые помогли нам быстро и удобно общаться, намного лучше, чем мы, вероятно, могли предположить, мобильные игры открывают новые горизонты общения, что заставляет пересмотреть всю концепцию создания игр. Вы думаете, это прозвучало слишком театрально? Возможно. Но представьте сцену: вы, прогуливаясь на природе, принимаете участие в массовой игре по сети с сотнями людей из других стран со всего мира! Конечно, сразу возникает вопрос, почему вы играете, а не наслаждаетесь видами окружающей природы, но смысл заключается в том, что мобильные игры всегда и везде соединяют людей. Это не просто фантазия, такие игры существуют уже сегодня.

Я знаю: идея глобальной сетевой игры не нова, люди играют в подобные игры на настольных компьютерах постоянно. Но для этого требуется компьютер и сетевое соединение, что не всегда возможно и доступно. Даже самым тонким ноутбукам требуется плоская поверхность, на которую их можно поставить, а также беспроводная сеть, к которой можно под-

ключиться. В то же время мобильный телефон может легко поместиться в вашем кармане, он по определению подключен к беспроводной сети. В результате вы с легкостью можете подключиться к сетевой игре или выйти из нее.

Создавая среду, в которой игроки могут подключаться к игре и отключаться без проблем, социальный аспект мобильных игр, вероятно, будет наиболее привлекательным. Мобильные коммуникаторы уже сделали мир меньше, а мобильные игры переводят вещи на новый уровень, позволяя людям всей планеты играть вместе вне зависимости от их физического местоположения. Игры могут преодолеть не только пространственные барьеры, но также языковые и культурные. Вам не нужно знать другого языка, чтобы играть в Pong или Snake. И даже более сложные игры сделаны так, чтобы преодолевать межкультурные барьеры.

В копилку Игрока

Вероятно, вы не сразу сможете представить игру как средство культурной коммуникации, однако возьмите простую детскую (даже некомпьютерную) игру, подобные игры прошли не одно поколение и даже культуру. Подобно преданиям и легендам, игры, в которые играют люди, говорят многое об их культуре. Распространение игр по всему миру – это реальная возможность узнать других людей и поведать о себе.

Сильная сторона мобильных игр

Пожалуй, самое интересное в мобильных играх – это то, что их рынок до сегодняшнего дня не сегментирован. Новые жанры игр до сих пор не придуманы. Подумайте, как, например, можно использовать возможности GPS (Global Positioning System – Глобальная Система Позиционирования) в играх. Технически возможно играть в рамках реальной географии мира. Иначе говоря, чтобы перемещать своего героя, вы сами должны перемещаться по миру, технология GPS делает это возможным.

Если вы думаете, что идея мобильной игры, взаимодействующей с GPS, – это фантастика, позвольте мне познакомить вас с игрой Pac-Manhattan. Pac-Manhattan – это оригинальная «крупномасштабная городская игра», которая использует карту города Нью-Йорк как игровое поле для игры Pac-Man.

Идея игры заключается в перемещении классической игры из виртуального пространства в реальный мир. Более того, героями игры являются непосредственно люди, бегающие по улицам Манхэттена. Игрок, Pac-Man, прокладывает свой путь через Парк Вашингтон Сквер, а его преследуют привидения Inky, Pinky и Clyde. На рис. 1.2 показана игровая карта Pac-Manhattan, которая превращает улицы Нью-Йорка в игровой лабиринт.

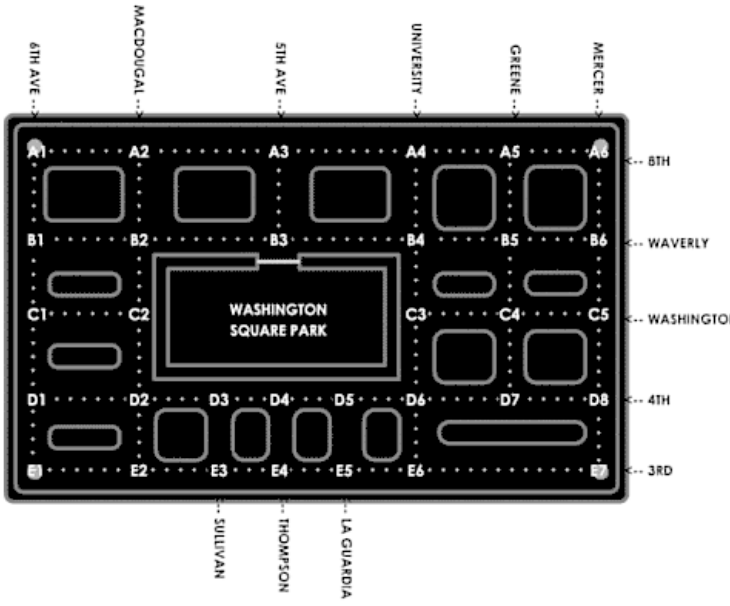

Рис. 1.2. Игровая карта Pac-Manhattan превращает улицы Нью-Йорка в игровую карту

Используя мобильные телефоны и сеть WiFi для передачи данных, центр управления обновляет данные о положении игроков и отображает их на сайте Pac-Manhattan (http://www.pacmanhattan.com/). На рис. 1.3 показан пример картинки из игры Pac-Manhattan в самом разгаре действий. Помните, что все действия игры разворачиваются на улицах реального города.

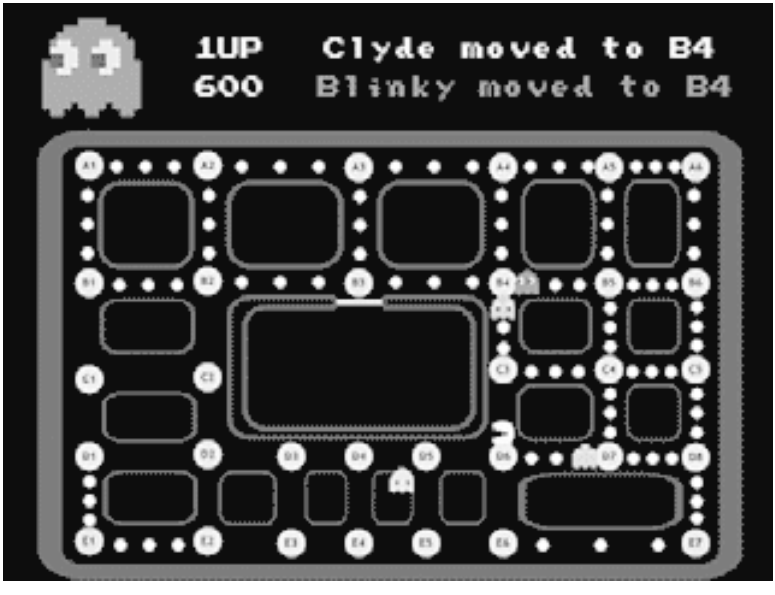

Рис. 1.3. Игра Pac-Manhattan очень похожа на оригинальную Pac-Man, однако в ней перемещаются реальные люди по улицам Нью-Йорка

Хотя для реализации Pac-Manhattan требуются нестандартные коммуникативные возможности, в сущности, эта игра не попадает под данное нами определение мобильной игры, потому что мобильные телефоны в ней используются исключительно как голосовые коммуникаторы. Другими словами, на мобильном телефоне нет как таковой игры, вы с такой же легкостью можете использовать walky-talky (пере-

носная рация). Более интересная с технической точки игра Pac-Manhattan основывалась бы на использовании мобильных клиентов GPS, определяющих местоположение игрока и отправляющих эти данные на центральный сервер. По слухам, ребята из Pac-Manhattan сейчас работают над этим, поэтому следите за новостями на сайте!

В копилку Игрока

Если вы хотите развернуть действия Pac-Manhattan на улицах вашего города, то на сайте этой игры можно загрузить пакет «In Your City Kit», который содержит всю информацию, необходимую для постановки игры в вашем городе. Предупреждаю, что эта игра намного опаснее своего компьютерного прототипа, поэтому вы играете на свой страх и риск.

GPS в игре Pac-Manhattan – это лишь один из примеров того, как мобильные телефоны могут объединять не совместимые ранее технологии и открывать новые горизонты для создания мобильных игр. Мы живем в чрезвычайно удивительное и динамичное время мобильных игр!

Знакомство с мобильными платформами

Прежде чем я расскажу о специфике существующих мобильных платформ, важно отметить, что по сей день программирование мобильных игр находится в зародышевом

состоянии. Это очень важно, потому что это означает, что средства и технологии меняются очень быстро. Очень важно, чтобы вы держали руку на пульсе и отслеживали новые технологии и тенденции, включая беспроводные технологии, предлагаемые и поддерживаемые провайдерами.

Хотя «ландшафт» мобильных игр изменяется быстро, приходит день, когда та или иная платформа становится доминирующей. Можно составить список платформ, подходящих для написания мобильных игр:

- Java 2 Micro Edition (J2ME);
- Binary Runtime Environment for Wireless (BREW);
- Symbian;
- Windows Mobile Smartphone.

В копилку Игрока

Еще одна платформа для мобильных игр – это SMS или Short Message Service (Служба коротких сообщений). SMS – это технология, которая позволяет отправлять и получать короткие сообщения от игрового сервера. SMS – это предыдущий этап развития мобильных игр, который можно использовать для текстовых игр и чатов, но эту платформу сложно применять для создания чего-то другого. Кроме того, за отправку SMS необходимо платить, а это может оказаться весьма накладным при большом объеме получаемых/отправляемых сообщений.

Все эти четыре платформы поддерживаются современными телефонами. Все они предлагают разработчику широкие

возможности, обеспечивая бесплатными инструментами и документацией. В последующих разделах дается более подробное описание каждой из платформ, что поможет вам понять их различия и особенности.

В копилку Игрока

Одним из самых сложных моментов при написании этой книги был момент принятия решения, привязываться ли к конкретной платформе. В конечном счете было решено, что невозможно рассказать об основах программирования мобильных игр и охватить две или три платформы. Поэтому мы выбрали наиболее широко поддерживаемую платформу, имеющую наиболее ясное будущее, – J2ME. К счастью, большинство методов программирования, о которых пойдет речь далее, можно применить при создании игр для других платформ.

Java 2 Micro Edition (J2ME)

J2ME – это компактная версия популярного языка программирования Java, созданного Sun Microsystems. Многие и не подозревают, что изначально Java создавался как язык программирования мобильных устройств, поэтому он вернулся к истокам и воплотился в J2ME. J2ME содержит широкий спектр инструментов для разработки и богатый программный интерфейс приложения (application programming interface, API) для разработки приложений для мобильных

телефонов, известных как MIDlets (о значении этого термина речь пойдет позже).

J2ME также включает виртуальную машину, которая отвечает за эмуляцию выполнения кода Java на конкретном телефоне. Создавая общий код приложения вместо собственного, J2ME позволяет без труда создавать игры, совместимые с широким кругом мобильных телефонов. На самом деле, если бы телефоны не отличались размерами экранов и графическими возможностями, то не требовалось бы прикладывать и малейшего усилия, чтобы перенести игру с одного телефона на другой.

Эта платформа наиболее широко поддерживается производителями мобильных телефонов в США. Такие гиганты, как Motorola, Nokia, Research in Motion (RIM) и Samsung, поставляют телефоны, поддерживающие J2ME.

Binary Runtime Environment for Wireless (BREW)

В отличие от J2ME, поддерживаемого широким спектром телефонных аппаратов, BREW – это платформа, ориентированная на телефоны с технологией Qualcomm's CDMA (Code Division Multiple Access – множественный доступ с кодовым разделением каналов). Но это не говорит о том, что BREW имеет весьма ограниченное распространение. Для создания игр для платформы BREW разработчики часто используют

язык программирования C или C++ в сочетании с BREW API. BREW также поддерживает такие языки разработки приложений, как XML и даже Java.

Подобно J2ME, BREW может выступать промежуточным звеном между игрой и операционной системой телефона. В отличие от J2ME, платформа BREW также поддерживает и собственный код, это значит, что игра может быть скомпилирована под конкретный процессор телефона. Собственный код обрабатывается намного быстрее, чем его интерпретированный аналог, однако при его использовании могут возникать проблемы с переносом приложения с одного телефона на другой.

Платформа BREW нашла наибольшее распространение в Азии (особенно в Японии и Южной Корее). В США на сегодняшний день Alltel и Wireless – это лидирующие поставщики беспроводных решений, предлагающие телефоны с поддержкой BREW.

Symbian

Symbian – это мобильная операционная система, которая имеет существенное отличие от BREW. Оно заключается в том, что это открытая операционная система, доступная по лицензии любому производителю мобильных телефонов. Операционная система Symbian была разработана компанией Symbian Ltd., которая является консорциумом компаний

Motorola, Nokia, Panasonic и Sony/Ericsson. Благодаря простоте лицензирования, эта ОС поддерживается многими моделями телефонов.

Существует масса возможностей при разработке приложений для Symbian, поскольку эта операционная система поддерживает такие языки программирования, как C++, Java и Visual Basic. На сегодняшний день большинство мобильных игр и приложений для Symbian написаны на C++, поскольку это обеспечивает большое быстродействие и интеграцию с операционной системой, нежели аналоги, созданные с использованием Java. Несомненно, Java начинает компенсировать недостаток в производительности между приложениями, написанными на этом языке, и собственным кодом, но все-таки в большинстве случаев собственные программы работают эффективнее, чем Java-аналоги. Это особенно важно, если дело касается игр, где каждая доля мощности процессора на счету.

Так почему бы не остановиться на Symbian, а не на Java? Простой ответ заключается в том, что Java распространен широко, в то время как Symbian – это лишь одна из операционных систем.

Мобильные телефоны значительно отличаются от персональных компьютеров тем, что имеют весьма разнообразное программное и аппаратное обеспечение. Java – это унифицирующая технология, позволяющая использовать созданное приложение на различных типах телефонов.

Windows Mobile Smartphone

Вы ведь не думали, что Microsoft будет сидеть в стороне и спокойно следить за развитием мобильных игр без собственной операционной системы, не так ли? Конечно, нет! Компания Microsoft немного преуспела, выпустив операционную систему Windows Mobile, устанавливаемую на карманных компьютерах и усовершенствованных мобильных телефонах, известных как смартфоны (Smartphone). Хотя некоторые Pocket PC могут выполнять функции мобильных телефонов, все-таки они ближе к PDA, нежели к телефону, хотя бы по своим габаритным характеристикам: Pocket PC имеет достаточно широкий экран (240x320), а для ввода используется перо (stylus).

Хотя Pocket PC как устройства не очень похожи на мобильные телефоны, операционная система Windows Mobile – это совершенно иное дело. Компания Microsoft смиренно ждет времени, когда технологии мобильных телефонов смогут соответствовать требованиям недавно выпущенной операционной системы Smartphone, которая является разновидностью Windows Mobile для мобильных телефонов. Эта операционная система привлекает прежде всего тем, что она не является «урезанным» вариантом Windows Mobile. Главное ее отличие заключается в измененном интерфейсе (ввиду меньших размеров экрана и отсутствия электронного пера).

В итоге вы получаете полноценную систему Windows Mobile в мобильном телефоне с логотипом Smartphone.

Но что это значит с точки зрения перспективы разработки мобильных игр? Это значит, что вы можете использовать те же самые инструменты API, которые сегодня применяются для создания игр для Pocket PC, например, C, C++ или Microsoft C# в сочетании с Windows Mobile API. Разработка игр для Pocket PC активно ведется в течение нескольких последних лет, поэтому в некотором смысле Smartphone изначально получает значительный толчок, несмотря на то, что эта операционная система является сравнительно новой для рынка.

Компании Motorola и Samsung производят смартфоны в США на основе AT&T Wireless и Verizon Wireless. Однако ввиду силы компании Microsoft, я вижу в скором будущем быстрое увеличение количества предлагаемых на рынке устройств, снабженных этой операционной системой.

Java как платформа для мобильных игр

Если вы умудрились прочитать все примечания «В копилку Игрока», то вы уже знаете, что в этой книге внимание будет уделено именно J2ME. Я объясню такое решение чуть позже, а пока рассмотрим, что же представляет собой эта технология, и какие возможности она предоставляет с точки зрения программирования мобильных игр.

Что такое Java?

Ранее я упоминал, что изначально Java был языком программирования, который позволял сетевым устройствам связываться друг с другом. Если быть более точным, то Java зарождался как проект в Sun, целью которого было научиться внедрять компьютеры в повседневную жизнь. Одной из основных задач проекта было заставить все компьютеризованные устройства взаимодействовать друг с другом. Как выяснилось, Sun опередил свое время в попытке использовать Java для соединения бытовых приборов. Однако компания быстро отреагировала и завершила работу над проектом, сделав Java успешным Web-ориентированным языком программирования.

Как только технология и потребность на рынке совпали с исходным назначением Java, и Sun подняла свои предыдущие наработки и приспособила Java для мобильных телефонов. J2ME разработан не только с учетом ограничений мобильных телефонов, он также подходит для программирования беспроводных соединений. J2ME – это лишь подмножество более глобального инструмента Java, который состоит из языка программирования, API и среды выполнения.

Почему Java?

Даже если бы Java был идеально приспособлен для раз-

работки приложений для мобильных телефонов, он бы не нашел широкого применения без поддержки промышленности. На сегодняшний день Java – доминирующая технология разработки программного обеспечения для мобильных телефонов. Все говорит о том, что Java укрепится на рынке и, вероятно, расширит свой сегмент. По оценкам ряда специалистов к 2007 году будет продано 450 миллионов телефонов, поддерживающих Java, что составит 75 % рынка сотовых телефонов.

Разработчики активно используют Java, поскольку эта платформа является открытой. Это значит, что если вы разработали код, то его можно использовать в различных мобильных устройствах. К сожалению, такая «открытость» Java была несколько осложнена наличием API других производителей и различиями аппаратного обеспечения различных устройств. Тем не менее вы можете написать код и, не вносяся никаких изменений, использовать его на разнообразных мобильных телефонах. Сравните это с платформой BREW, созданной специально для телефонов, работающих в сетях Qualcomm CDMA.

В копилку Игрока

С технической точки зрения, несмотря на то что Java и J2ME «более открыты», чем BREW, тем не менее они до сих пор не являются языками программирования с открытым кодом. Все разновидности Java-технологии, включая J2ME, принадлежат Sun Microsystems. К

счастью, Sun была очень любезна и дала возможность свободного формирования стандартов Java однако многие разработчики до сих пор лоббируют их, чтобы перевести Java в сообщество Открытого Кода.

Интересный виток взаимоотношений между Java и BREW случился в конце 2002 года, когда в свет вышла виртуальная машина Java для устройств, работающих на платформе BREW. Это означает, что BREW-устройства могут эффективно работать с Java-приложениями, как будто они изначально поддерживали Java. Но при этом Java не конкурировала с BREW как платформа для разработки игр. Однако поскольку телефоны, поддерживающие только Java, не поддерживают BREW, появление виртуальной машины означает увеличение доли на рынке, если вы используете Java.

В копилку Игрока

Вероятно, вы обратили внимание, что я попеременно использую термины Java и J2ME. Хотя технически J2ME – это часть более обширной технологии Java в рамках этой книги эти термины имеют один и тот же смысл, поскольку я употребляю их в контексте мобильных телефонов.

Java и программирование мобильных игр

Вы знаете «что» и «почему», а теперь важно рассмотреть вопрос «как». Иначе говоря, как программировать мобильные игры с помощью Java? Прежде всего при программи-

ровании игр вызывают интерес следующие области технологии:

- ► графика и анимация;
- ► звук;
- ► пользовательский ввод;
- ► работа в сети.

В следующих нескольких разделах рассматривается каждый из этих вопросов с точки зрения J2ME.

Графика и анимация

Стандартный API включает поддержку всевозможных графических элементов, таких как, например, изображения, двухмерные графические примитивы (линии, прямоугольники, эллипсы и т. д.) и анимация. В терминах анимации J2ME поддерживает спрайты – изображения, свободно перемещаемые по экрану вне зависимости от других. API среды J2ME также поддерживает детектирование столкновений спрайтов, что позволяет определять, столкнулись ли два спрайта. Это очень важное свойство, необходимое для создания практически любой игры в стиле «экшн». Вы познакомитесь со спрайтами в главе 5.

Другая интересная особенность J2ME – это замощенные слои. Вы можете выбрать небольшое изображение и замостить им фоновый слой. Такие слои используются в играх для создания больших перестраиваемых карт, что позволя-

ет сэкономить память. Благодаря удобному менеджеру слоев в J2ME можно с легкостью управлять несколькими слоями. Таким образом, можно создать один слой – полностью фоновый, который можно использовать как декорацию, и еще один слой – для создания преград на пути героя игры. В главе 10 рассказывается о замощенных слоях, а в главе 11 речь идет о менеджере.

Обработка ввода пользователя

Пользовательский ввод очень важен для игр: он определяет, как удобно игроку взаимодействие с игрой. Также ввод очень важен потому, что определяет первичный интерфейс между игроком и игрой. J2ME поддерживает клавишный ввод, который на сегодняшний день является единственным способом ввода на мобильном телефоне. Существует возможность непосредственно считывать состояние клавиш на телефоне, это очень важно, если вы хотите обеспечить высокую степень взаимодействия игрока и игры. Особенности обеспечения ввода через J2ME API описаны в главе 6.

В копилку Игрока

Помните, что клавиши на мобильных телефонах значительно отличаются в зависимости от модели, но всегда есть похожие клавиши, выполняющие сходные функции. Если говорить более подробно, то на телефонах, поддерживающих Java, всегда есть клавиши,

отвечающие за перемещения влево, вправо, вверх, вниз и стрельбу, а также ряд прочих «полезных клавиш». Для телефонов, оснащенных джойстиками, каждое из отклонений соответствует направлению: влево, вправо и т. д.

Использование звука в играх

«Большую тройку» самых важных элементов мобильных игр завершает звук. J2ME поддерживает воспроизведение цифрового звука в форматах PCM или WAV, а также музыку в формате MIDI. Поддержка звука основана на Java Media API – API, предназначенного для записи и воспроизведения звука и видео на мобильных устройствах. Для разработки игр все, что вам понадобится узнать, – это как в нужный момент воспроизвести звук и, может быть, видео.

В копилку Игрока

Чтобы не уходить далеко от разработки игр, в этой книге будут рассмотрены лишь вопросы воспроизведения аудио средствами J2ME.

В главе 8 вы познакомитесь с программированием звуков, а также звуковым форматом WAV и музыкальным MIDI.

Мобильные сети

Самой привлекательной возможностью мобильных игр будет, вероятно, возможность работы в сети. Помня это, вы можете понять, что ориентированный на сети Java чрезвычайно удобен как платформа для мобильных игр. Сетевые возможности Java являются неотъемлемой частью его среды выполнения. В отличие от других языков программирования игр (C или C++), язык Java был ориентирован на поддержку сетей.

Объедините сетевую ориентированность Java и независимость этой платформы, и вы получите игровую платформу, которая преодолевает все преграды на пути доступности пользователю. Это очень важно, особенно когда вы поймете, что пользователи захотят играть в игры на различных устройствах и в различных беспроводных сетях. Игрок не должен вникать в проблемы, связанные с отличиями мобильных телефонов разных производителей. Благодаря поддержке сетей языком Java разработчикам теперь не нужно заботиться о различиях аппаратного обеспечения.

О сетевых мобильных играх речь пойдет в главе 14, а в главе 15 будет рассмотрен пример создания такой игры.

Небольшой пример на J2ME

Основной набор инструментов и API, необходимых для создания полноценных Java-приложений, известен как J2SE (Java 2 Standard Edition). J2SE используется как для создания самостоятельных приложений, так и для программирования Web-апплетов. Другая разновидность Java – это J2EE (Java 2 Enterprise Edition), предназначенный для создания корпоративных приложений. J2EE отличается от J2SE, поскольку для него существенна функциональная поддержка корпоративных приложений. Представьте большое сетевое приложение, обеспечивающее работу eBay или Amazon.com, и вы поймете, для чего предназначен J2EE.

Учитывая, что мобильные беспроводные устройства имеют меньшие вычислительные мощности и меньшие экраны (по сравнению с настольными аналогами), становится очевидным, что J2ME – это упрощенная версия J2SE с уменьшенным набором функций. На самом деле J2ME – это часть J2SE, которая поддерживает минимальный набор инструментов, необходимый для программирования мобильных устройств как проводных, так и беспроводных.

Также J2ME обладает рядом особенностей, уникальных для мобильных устройств. Эти три пакета (J2ME, J2SE и J2EE) образуют технологию Java 2.

Вы можете спросить, зачем я затрагиваю прочие разновидности Java в книге, посвященной программированию мобильных игр. Дело в том, что это обязательный минимум, который вы должны знать в отношении J2ME. Не волнуйтесь

– после того как вы немного больше познакомитесь с J2ME, мы приступим к созданию игр!

Конфигурация и ограниченная конфигурация мобильного устройства

Если вы используете J2ME, то вы столкнетесь с новыми терминами и аббревиатурами. Во-первых, вы встретите термин «конфигурация» (configuration). Конфигурация – это минимальный набор API, необходимый для написания приложения и его запуска на ряде мобильных устройств. Стандартная конфигурация мобильных устройств известна как Ограниченная Конфигурация Мобильного Устройства (Connected Limited Device Configuration, CLDC). CLDC – это минимальный набор функций, которым должно обладать любое беспроводное устройство. В CLDC учитываются такие факторы, как объем свободной памяти устройства, а также мощность процессора.

Если рассмотреть более подробно, CLDC включает в себя следующие параметры мобильного устройства:

- множество используемых Java-функций;
- функциональность виртуальной машины Java;
- набор API, необходимый для разработки приложения;
- аппаратные средства мобильного устройства.

Вы, вероятно, можете подумать, что при программировании для мобильных устройств в вашем распоряжении нахо-

дятся все возможности Java, однако это не так вследствие ограничений CLDC, связанных с пониженной вычислительной мощностью таких устройств. Кроме ограничений API, CLDC также накладывает ограничения и на аппаратную часть устройств, поддерживающих Java:

► объем памяти, необходимый Java, составляет 160 Кб;

► 16-битный процессор;

► низкое потребление энергии (обычно низкий расход батареи);

► соединение с сетью (часто беспроводное со скоростью 9600 бит/с или меньше).

К CLDC устройствам относятся (но не только указанные устройства) мобильные телефоны, пейджеры, PDA, карманные компьютеры и бытовые приборы. Конечно, нас прежде всего интересуют мобильные телефоны.

В копилку Игрока

Кроме CLDC J2ME определяет еще одну конфигурацию, известную как CDC (Connected Device Configuration – Конфигурация сетевого устройства), которая накладывает ограничения на более мощные и габаритные устройства по сравнению с мобильными. Следовательно, CDC имеет больший набор возможностей, чем CLDC.

Профили и MIDP

Во главе конфигурации находится профиль (profile), ко-

торый представляет собой особый набор API, предназначенный для конкретного типа устройства. Конфигурация в общих чертах описывает семейство устройств, в то время как профиль дает более детальное описание, выделяющее тип устройства внутри семейства. MIDP (Mobile Information Device Profile – информационный профиль мобильного устройства) – это профиль, построенный на основе CLDC, который описывает беспроводные мобильные устройства, такие как телефон или пейджер.

Кроме указания API, используемых для разработки приложений для конкретного типа устройств, MIDP также описывает минимальные требования к аппаратному и программному обеспечению. Это очень важно, поскольку вы всегда знаете, каковы наихудшие условия работы созданного вами приложения.

В копилку Игрока

Существует две версии профиля MIDP: 1.0 и 2.0. Хотя телефоны MIDP 1.0 имеют достаточно большую функциональность, MIDP 2.0 предоставляет более широкие возможности J2ME для программирования игр. Эта книга целиком посвящена MIDP 2.0, поскольку такие мобильные телефоны с огромной скоростью замещают телефоны, поддерживающие более ранний профиль MIDP 1.0, если уже не полностью вытеснят их к моменту, когда вы закончите прочтение книги.

Оценка требований MIDP к аппаратному обеспечению

Важной частью стандарта MIDP являются требования к аппаратному обеспечению устройств MIDP 2.0. Эти требования накладываются на следующие параметры:

► память;

► экран;

► ввод;

► сеть.

Требования к памяти согласно MIDP 2.0 следующие:

► 256 Кб недоступной памяти для библиотек MIDP API;

► 128 Кб памяти для системы работы Java;

► 8 Кб недоступной памяти для постоянных данных приложений.

Требования к вводу MIDP-устройств оговаривают, что устройство должно иметь клавиатуру или сенсорный экран. Обратите внимание, что мышь не является устройством ввода, поскольку сложно представить мобильное устройство, работа с которым осуществляется с помощью мыши. Однако такое устройство вполне может иметь сенсорный экран и электронное перо.

В копилку Игрока

Если вам интересно, то в MIDP постепенно начинают входить джойстики. Компании Sony/Ericsson и Samsung

предлагают мобильные телефоны с маленькими джойстиками, встроенными в клавиатуру. Поскольку на самом деле MIPD непосредственно не поддерживает джойстики, то джойстик можно использовать, если его положения ассоциированы с соответствующими клавишами клавиатуры.

Требования MIDP к дисплею представляют особый интерес, поскольку для мобильных устройств экран – это один из самых ограниченных параметров. Устройство MIDP должно иметь экран размером 96x54 пикселя с глубиной цвета 1 бит. Это означает, что экран должен быть как минимум 96 пикселей в высоту и 54 пикселя в ширину и, по крайней мере, должен быть черно-белым. Кроме того, форматное соотношение экрана должно быть 1:1, это означает, что пиксели должны иметь форму прямоугольника.

В копилку Игрока

В реальности большинство телефонов MIDP 2.0 превосходят минимальные требования за счет цветного дисплея и более широкого экрана.

Последнее требование к аппаратному обеспечению – это работа в сети, которое оговаривает минимальные требования к поддержке сетей. MIDP-устройство должно иметь двунаправленное беспроводное сетевое соединение любого вида. Такое соединение может быть прерывным (например, dial-up) и иметь ограниченную скорость передачи данных (9600 бит/с). Это очень важно, поскольку при разработке мо-

бильных игр вы должны быть очень внимательны при определении скорости передачи данных, особенно в играх, для которых быстродействие необходимо (игры в стиле «экшн»).

Оценка требований MIDP к программному обеспечению

Кросс-платформенная природа Java позволяет преодолевать различия между множеством операционных систем. Но несмотря на это, спецификация MIDP устанавливает ряд ограничений, касающихся операционной системы мобильного устройства. Ниже перечислены основные требования к программному обеспечению MIDP-устройств:

► минимальное ядро, необходимое для выполнения низкоуровневых функций, таких как, например, прерывания, исключения и очередь;

► механизм чтения и записи в постоянную память;

► механизм для установки таймеров и отметки времени данных;

► доступ на запись/чтение к сетевому соединению устройства;

► механизм перехвата ввода с клавиатуры или сенсорного экрана;

► минимальная поддержка битовых изображений;

► механизм распределения жизненного цикла приложений.

Эти требования, пусть даже и минимальные, предоставляют достаточно широкий набор средств, который можно использовать для создания MIDP-игр.

Резюме

Я знаю, что вам, вероятно, уже не терпится написать какой-нибудь код и окунуться с головой в программирование мобильных игр, однако в этой главе речь шла об основах. Вы узнали не только о мобильных играх в общих чертах, но также и о различных опциях разработки мобильных игр. Если говорить более подробно, вы узнали, что Java – это лидирующий пакет, и почему эта платформа будет пользоваться значительным успехом в будущем. В конце главы вы познакомились с J2ME, версией Java, предназначенной для программирования мобильных телефонов. Я не люблю выдавать много «фоновой информации», но будет весьма полезно, если вы как можно быстрее узнаете о тонкостях создания игр для Java-совместимых мобильных телефонов.

Экскурсия

Я не могу закончить эту главу, не порадовав вас. Если вы – счастливый обладатель мобильного телефона с поддержкой Java, посетите Handango (http://handango.com/) и найдите какую-нибудь игру. Да, именно Handango, а не Fandango

– сайт для заказа билетов в кинотеатр. Большинство игр на этом сайте имеют демо-версии, которые вы можете бесплатно загрузить, перед тем как купить ту или иную игру. Поэтому эта экскурсия не будет вам ничего стоить. Пролистывая списки игр, обратите внимание, может быть, что-то упущено, и у вас, вероятно, появятся идеи создания собственной игры

Глава 2
Основы разработки мобильных игр на Java

Архив Аркад

В 1976 году компания Midway выпустила игру Sea Wolf. В ней вы смотрите в перископ и выпускаете торпеды в проплывающие в верхней части экрана корабли. При этом раздается звук, очень похожий на звук подводной лодки. Игра Sea Wolf – это аналог популярной в конце 60-х годов механической игры Periscope компании Sega Интересно, что Periscope была первой игрой, в которой за каждый из боев приходилось платить 25 центов. Такой стандарт цены перекочевал и в другие аркады.

Поняв, почему в обозримом будущем именно Java станет платформой для разработки мобильных игр, вы будете готовы к тому, чтобы научиться создавать мобильные игры. К счастью, Sun Microsystems абсолютно бесплатно предлагает пакет J2ME Wireless Kit для разработки мидлетов (MIDlet) на J2ME. В этой главе вы познакомитесь с J2ME Wireless Toolkit и узнаете, как его использовать для создания мобильных игр для Java-устройств. Вы также научитесь использовать эмулятор J2ME для проверки работы и запуска приложений, что позволит вам играть в созданные игры, не загру-

жая их на мобильный телефон. Вы узнаете, что эмуляция – это очень важный этап при разработке мобильных игр.

В этой главе вы узнаете:

► об основах разработки игр;

► как использовать J2ME для создания мобильных игр;

► как применять инструмент KToolbar для построения и тестирования мобильных игр;

► как с помощью эмулятора J2ME можно имитировать реальные мобильные телефоны.

Основы разработки игр

Перед тем как мы начнем изучать J2ME Wireless Toolkit и его использование для построения и тестирования мобильных Java-игр, полезно узнать об основах разработки игр. У вас есть идеи о том, какую игру написать? Если так, то вы, вероятно, уже осознали, что идея – это самая простая часть процесса создания игры. Разработка концепции игры и ее реализация – для многих из нас очень сложное занятие. Но ничего страшного, всему свое время, прежде надо продумать процесс создания игры.

Первый шаг на пути от концепции игры к ее реализации – четко ответить на вопрос, что вы хотите получить в результате. При этом не нужно все детализировать с точностью до отдельных сцен, существ и действий. Необходимо сформулировать несколько основополагающих принципов – цель и

финал игры. Ниже представлены основные моменты, которые вы должны продумать, приступая к разработке игры:

- ► основная идея;
- ► режимы игры.
- ► сюжетная линия;

Основная идея

Первое, что вы должны сделать, – это определиться с основной идеей вашей игры. Вы хотите создать «стрелялку», лабиринт, ролевую игру или что-то среднее? Или вы придумали игру, которую нельзя отнести ни к одной из этих категорий? Цель игры – уничтожить плохих парней, спасти хороших или просто исследовать неизвестные миры? Как долго будет длиться ваша игра, или, может быть, она будет бесконечной? В любом случае, фиксируйте все мысли, поскольку идеи приходят и уходят, было бы плохо упустить что-либо. Фиксируя идеи, вы больше начинаете думать об игре и представлять ее более детально.

Если вы испытываете трудности с идеей игры, вспомните существующие компьютерные игры. Многие из них основаны на фильмах, исторических событиях или спорте. Очевидно, что компьютерные игры – это модели окружающего нас мира, неважно – фантастического или реального, поэтому старайтесь придерживаться этого принципа, создавая свою игру. Фильмы могут дать множество идей для отдель-

ных сцен и сюжетной линии. Просто помните, что многие фильмы послужили основой для ряда коммерческих видеоигр. По этой причине вы должны воспринимать фильмы, как кладезь идей и концепций, а не как сценарий для игры.

В копилку Игрока

В настоящее время есть несколько случаев, когда компьютерная игра послужила основой для создания фильма. Как пример можно привести фильмы Mortal Combat («Смертельная схватка»), Final Fantasy («Последняя фантазия») и Resident Evil («Обитель зла»).

Вне зависимости от ваших предпочтений, помните, что игра должна развлекать. На самом деле я думаю, что именно это делает компьютерные игры столь привлекательными для программистов. Главная цель игры – максимум развлечения. Кто не хотел бы проводить целые дни в мыслях о том, как развлечься? Если ваша игра не доставляет радости, то ей не помогут ни великолепная графика, ни потрясающий звук. Я стараюсь показать, что при разработке игры главное – это доставить максимальное удовольствие пользователю. После того как вы сформировали основную идею игры и решили во что бы то ни стало сделать ее максимально захватывающей, пора переходить к проработке сюжетной линии.

Разработка сюжетной линии

Даже если вы создаете игру в стиле «экшн», разрабатывая сюжетную линию, вы сможете более четко представить ландшафт и существ, населяющих ваш мир. Помещая игру в контекст истории, вы переносите игрока в свой мир. Для игр, в которых история является неотъемлемой частью, часто полезно дополнять сюжетную линию историческими справками, которые описывают историю от сцены к сцене. Исторические справки помогают создать визуальный план всей игры, основанный на истории. Такие справки помогут избежать отклонения от основной сюжетной линии при разработке игры.

Режимы игры

Последнее, что вы должны сделать на начальной стадии разработки игры, – это понять, какие режимы игр вы будете поддерживать. Вы хотите создать игру для одного, двух игроков, сетевую игру или некую комбинацию? Вероятно, такое решение может показаться чересчур простым, однако именно оно может оказать существенное влияние на логику игры в дальнейшем. Несмотря на то что Java предлагает значительную поддержку сетей, разработка сетевых игр обычно очень сложное занятие.

С другой стороны, многие игры, рассчитанные на одного игрока, требуют более сильного искусственного интел-

лекта, чтобы компьютер мог оказывать мощное сопротивление игроку. Создание искусственного интеллекта – задача не из легких, поэтому вам необходимо оценить свои силы и, прежде чем приступить к работе, выбрать режим игры. В главе 13 вы познакомитесь с принципами создания искусственного интеллекта.

Пример разработки игры на J2ME

Как и создание любой другой игры, разработка игр на J2ME требует разнообразных средств компиляции и тестирования. Откомпилированные на Java классы должны пройти процесс предварительной верификации, который необходим для проверки кода на соответствие ограничениям безопасности J2ME. Также игры, написанные на J2ME, требуют среду для тестирования, аналогичную реальному мобильному телефону. Конечно, всегда необходимо тестировать игры и на реальных телефонах, но прежде целесообразно делать это в эмуляторе, а уж затем переходить к реальным устройствам. Стандартный компилятор Java используется для построения игр, а затем на передний план выходят другие инструменты разработки.

В копилку Игрока

Стандартный компилятор Java входит в состав пакета Java Software Development Kit (SDK), который можно бесплатно загрузить с Web-сайта Sum Microsystems (http://javasun.com/). Также там вы найдете и J2ME

Wireless Toolkit, необходимый для разработки игр на Java.

Приложения, написанные с использованием J2ME, в соответствии со спецификацией MIDP называются мидлеты (MIDlet). Поэтому любая игра, созданная на J2ME, является мидлетами. Классы мидлетов хранятся в файлах байт-кода с расширением. class. Однако перед распространением классы должны быть проверены, чтобы гарантировать невыполнение запрещенных операций. Дело в том, что эта предварительная проверка необходима вследствие ограничений виртуальной машины, используемой в мобильных устройствах. Эта виртуальная машина называется K Virtual Machine, или KVM. Чтобы KVM была как можно меньше и эффективнее, необходимо минимизировать число верификаций, выполняемых во время выполнения приложения. Поэтому некоторые из этих верификаций выполняются еще на стадии разработки в процессе предварительной верификации.

В копилку Игрока

Название KVM также имеет отношение к требованиям виртуальной машины к ресурсам: KVM необходимы килобайты памяти, а не мегабайты. Иначе говоря, KVM разработана так, чтобы она помещалась в килобайтах памяти, в отличие от виртуальной машины J2SE, которой могут потребоваться мегабайты.

Предварительная верификация выполняется непосредственно после компиляции, ее результатом является новый

файл класса, который уже проверен и готов к распространению. Мидлеты (MIDlet) должны быть упакованы в специальные архивы JAR (Java Archive – архив Java), они очень похожи на ZIP-архивы, которые вы, вероятно, использовали для сжатия больших файлов. Мидлеты (MIDlet) также требуют дополнительного описания, которое включается в JAR-файл. Ниже перечислена основная информация, обычно включаемая в архив JAR:

- ► классы мидлета;
- ► вспомогательные классы;
- ► ресурсы (изображения, звуки и т. п.);
- ► файлы манифестов (.mf);
- ► дескриптор приложения (.jad).

В копилку Игрока

Файлы JAR используются для упаковки Java-классов и последующего более эффективного распространения. Файл манифеста – это текстовый файл с описанием классов, включенных в JAR-архив.

Дескриптор приложения, файл JAD, – это файл, в котором содержится описание мидлетов, хранящихся в файле JAR. Обратите внимание, что я сказал «мидлеты» (множественное число). Да, зачастую в файле JAR хранится несколько мидлетов. Такой набор мидлетов называется пакетом мидлетов (MIDlet suit). В случае мобильных игр вам, вероятно, захочется предложить несколько игр в качестве единого продукта (например, набор простейших игр), в этом

случае вам придется поместить все игры в один файл JAR.

В основе пакетов мидлетов лежит идея одновременного использования несколькими приложениями доступных ресурсов мобильного устройства. KVM-устройства необходимы для обеспечения навигации и выбора конкретного мидлета из пакета. Файл JAD, включенный в JAR-файл, очень полезен, поскольку в нем содержится информация, необходимая для установки и доступа к конкретному мидлету.

Я знаю, вы, вероятно, подумали, что этот раздел будет посвящен разработке игр на J2ME, а не устройству файлов JAR или пакетов мидлетов. На самом деле, чтобы понять процесс разработки приложений, вам необходимо знать некоторые основы сборки мидлетов для распространения. Процесс разработки мидлетов можно разбить на следующие стадии:

1. редактирование;
2. компиляция;
3. предварительная верификация;
4. эмуляция;
5. тестирование на устройстве;
6. использование.

Шаги 1 и 2 должны быть вам хорошо знакомы, поскольку эти стадии обязательны при программировании на любом из языков. Этапы 3 и 4 немного отличаются. Вы уже знаете, что шаг предварительной верификации необходим для подтверждения того, что приложение не выполняет недопустимых действий (например, засорение памяти устройства). На

шаге 4 вы проверяете свой мидлет с помощью специального инструмента – эмулятора.

Эмулятор J2ME представляет собой не то, о чем говорит название. Эмулятор имитирует реальный мобильный телефон в вашем настольном компьютере. Это позволяет тестировать мидлеты, не загружая код в мобильное устройство. Отладку также проще осуществлять, если приложение запущено в эмуляторе. Шаг 5 в процессе разработки мидлета – это его тестирование на реальном устройстве. В реальности, вам, вероятно, придется проверять работоспособность созданного приложения на нескольких различных устройствах, чтобы убедиться, что оно работает правильно. Такая проверка следует сразу после того, как вы устранили все ошибки в игре и близки к распространению приложения как готового продукта. Другими словами, перед тестированием на реальных устройствах приложение проходит тестирование на эмуляторе.

Последний этап разработки игры – это ее распространение. Распространение игры может быть очень простым, например, вы можете отправить ее по электронной почте друзьям, и они установят игру на своих мобильных телефонах, или весьма сложным, если вы продаете игру и распространяете ее через беспроводные сети. В последнем случае вам придется решить ряд технических проблем, например, убедиться, что ваша игра соответствует стандартам безопасности, а загружаемый файл имеет цифровую подпись с действующим

сертификатом безопасности. О распространении мобильных игр вы узнаете из главы 16.

В копилку Игрока

Поскольку мобильные игры обычно сильно нагружают процессор мобильного устройства, вам, вероятно, потребуется разрабатывать несколько отличающиеся друг от друга версии игры для различных моделей телефонов. Например, вам придется удалить некоторые элементы игры для менее мощных устройств. Также вам придется предусмотреть отличие экранов на различных моделях. Просто помните, что при тестировании игры на различных моделях телефона вам, вероятно, придется изменять код, чтобы приспособить ее для работы на некоторых из моделей.

Теперь, когда вы имеете представление о том, как разрабатываются мидлеты, я хотел бы рассказать, какие инструменты необходимы для их сборки:

- Java 2 SDK;
- J2ME Wireless Toolkit.

Java 2 SDK – это стандартный инструмент разработки Java. Инструмент J2ME Wireless Toolkit служит дополнением к среде разработки и работает в составе Java 2 SDK и включает верификатор байт-кода и несколько эмуляторов J2ME, необходимых для верификации и проверки мидлетов. Помимо стандартного J2ME Wireless Toolkit, некоторые производители мобильных телефонов предлагают свои

инструменты для разработки мидлетов. Например, компания Nokia предлагает различные MIDlet SDK, направленные на каждую из линеек телефонов, поддерживающих Java. Motorola также предлагает пакет SDK для J2ME, ориентированный на телефоны Motorola.

Обычно набор инструментов, предлагаемый производителями, содержит дополнительные профили устройств, которые помогут вам при эмуляции мидлетов, а также расширения API, поддерживаемые выпускаемыми устройствами. Хотя зачастую эти API очень мощны, я советую использовать стандартные MIDP API, чтобы создаваемые вами игры можно было без проблем распространять между мобильными телефонами.

В копилку Игрока

Дополнительные API, доступные на нескольких телефонах, предоставляют большие возможности по сравнению с ограниченными требованиями MIDP. Например, Location API для J2ME содержит классы и интерфейсы для определения физического местоположения мобильного телефона с помощью GPS или близости телефона к передающей станции беспроводной сети.

Знакомство с J2ME Wireless Toolkit

J2ME Wireless Toolkit – это набор инструментов для разработки, созданный компанией Sun Microsystems, который

при использовании с J2ME SDK позволяет разрабатывать MIDP-приложения. Пакет J2ME Wireless Toolkit вы найдете на прилагаемом CD, а на сайте http://java.sun.com/products/j2mewtoolkit/ вы можете проверить последние обновления. Пакет состоит из следующих инструментов:

- верификатор байт-кода;
- эмулятор J2ME;
- KToolbar;
- инициализирующий сервер.

Вы уже узнали, что верификатор байт-кода проверяет классы, входящие в состав игры, перед тем как они попадут в распространяемый пакет. Вы также знаете, что эмулятор J2ME используется для тестирования приложений на настольном компьютере, не загружая и не запуская их на реальном мобильном телефоне. Полезные функции эмулятора позволят вам имитировать различные мобильные телефоны. Например, вам, вероятно, понадобится эмулировать мобильный телефон с определенным размером экрана, отличным от стандартных устройств, в этом случае вы просто создаете новую конфигурацию и эмулируете мобильный телефон с новыми параметрами. Эмулятор J2ME также очень полезен при имитации настроек безопасности мобильных телефонов, что позволит создать более реалистичную среду выполнения.

KToolbar – это среда визуальной разработки, в которой можно собирать, компилировать, упаковывать и тестировать

приложения J2ME с графическим интерфейсом. Это контрастирует с другими инструментами J2ME, которые запускаются из командной строки. Далее вы будете часто использовать KToolbar для сборки и тестирования приложений.

В платформе MIDP 2.0 новинкой является поддержка инициализации Over The Air (по сети), или OTA, которая реализует механизм загрузки приложений в мобильные телефоны через беспроводную сеть. Пакет J2ME Wireless Toolkit включает инициализирующий сервер, который позволяет загружать и устанавливать приложение на эмулируемое устройство точно так же, как это будет делать пользователь, загружая приложение на мобильное устройство.

В копилку Игрока

Инициализация (provisioning) – это процесс верификации и установки мидлета на телефон, поддерживающий Java К сожалению, до MIDP 2.0 не было стандарта инициализации мидлетов, поэтому процесс загрузки и инициализации определялся изготовителем телефона.

Использование KToolbar

В этом уроке я часто буду обращаться к среде разработки и показывать, как она улучшает и ускоряет процесс разработки и сборки мидлетов. KToolbar – это самая простая среда визуальной разработки, поддерживающая J2ME.

Она настолько проста, что даже не содержит редактора кода. KToolbar сфокусирован на управлении файлами кода и автоматизации процесса сборки и тестирования. Используя приложение KToolbar, вы можете преодолеть необходимость использования командной строки инструментов J2ME и выполнить компиляцию, верификацию и эмуляцию в одной среде. На рис. 2.1 показано приложение KToolbar, в котором открыт проект J2ME.

Рис. 2.1. KToolbar предоставляет минимальные средства для разработки игр на J2ME

Хотя инструмент KToolbar – это минимальная визуальная среда, его достоинство заключается в том, что он бесплатно поставляется вместе с J2ME Wireless Toolkit. Просто помните, что вам придется найти подходящий текстовый редактор (например, Блокнот (Notepad) в операционной системе Windows) для редактирования файлов кода. С другой стороны, если у вас уже есть визуальная среда разработки Java, даже если она не поддерживает J2ME, ее полезно использовать для редактирования файлов кода J2ME.

Управление проектами J2ME

KToolbar предлагает простой способ управления проектами мидлетов и настройками сборки. Когда вы создаете новый проект в KToolbar, он автоматически появляется в папке apps, расположенной в папке установки J2ME Wireless Toolkit. Так, например, если Wireless Toolkit установлен в папке WTK21, то все приложения будут создаваться в папке WTK21\apps. Чтобы создать новый проект, щелкните по кнопке New Project (Новый проект), расположенной на панели инструментов. На рис. 2.2 показано диалоговое окно, в котором запрашивается название проекта и имя класса мидлета.

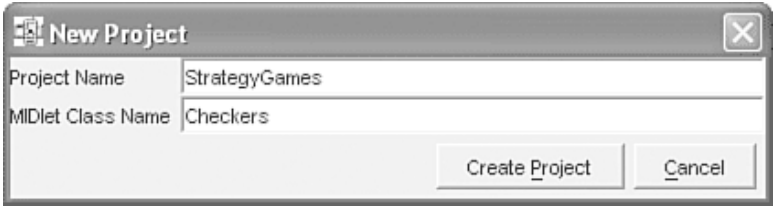

Рис. 2.2. Чтобы создать новый проект в KToolbar, просто введите имя проекта и название класса мидлета

Имя проекта будет использовано для названия JAR-файла, который устанавливается на мобильный телефон. Помните, что имя проекта может применяться ко всему пакету мидлетов, в то время как имя класса идентифицирует отдельный мидлет внутри пакета. В большинстве случаев в проекте содержится лишь один мидлет, поэтому вы можете использовать одно и то же имя как для класса мидлета, так и для приложения.

Чтобы открыть существующий проект в KToolbar, на панели инструментов щелкните по кнопке Open Project (Открыть проект). Будут отображены проекты, созданные в папке apps, расположенной в папке установки J2ME Wireless Toolkit. На рис. 2.3 показано диалоговое окно Open Project (Открыть проект), в котором вы можете выбрать проект, хранящийся в папке apps.

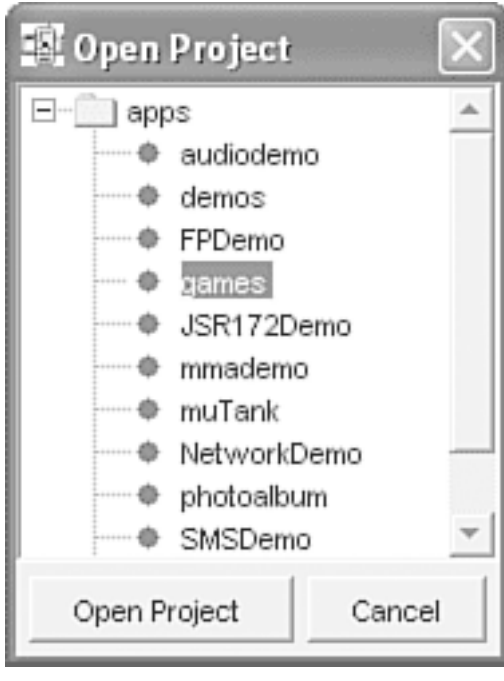

Рис. 2.3. В KToolbar можно открывать только те проекты, которые находятся в папке apps пакета J2ME Wireless Toolkita

После того как проект открыт в KToolbar, вы можете изменить его настройки, для чего щелкните по кнопке Settings (Настройки), расположенной на панели инструментов. Откроется диалоговое окно Settings (Настройки), показанное на рис. 2.4.

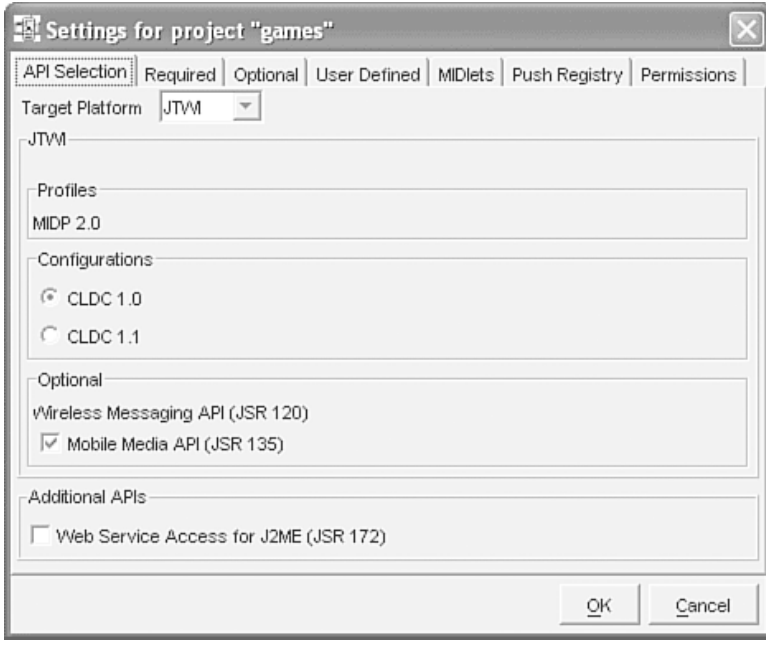

Рис. 2.4. Диалоговое окно Settings (Настройки) инструмента KToolbar предоставляет вам доступ к большому числу настроек проекта

Пока настройки проекта не очень важны, поскольку в большинстве случаев подходят и настройки по умолчанию. Возможно, единственное, чему стоит уделить внимание, так это закладке MIDlet, на которой показаны мидлеты, входящие в состав проекта. На рис. 2.5 показаны три различных мидлета, содержащихся внутри одного проекта.

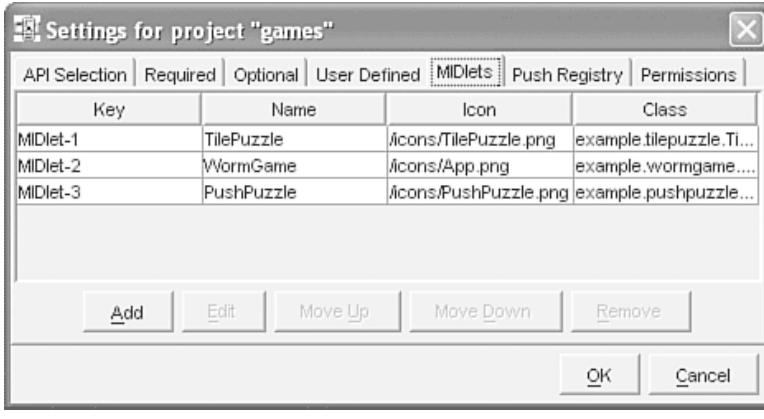

Рис. 2.5. Проект в KToolbar может состоять из нескольких мидлетов, при этом проект представляет собой пакет мидлетов

Игровой проект, показанный на рис. 2.5, содержит две игры-головоломки и игру Worm. Давайте, используя KToolbar, откомпилируем и упакуем эти игры, а потом протестируем их.

Сборка мидлета

Теперь вы подошли, вероятно, к самой трудной части главы: компиляции примера мидлета игры. На самом деле мы будем компилировать все три мидлета игр, входящие в проект J2ME Wireless Toolkit. Я пошутил, говоря, что это будет очень сложно. Достаточно щелкнуть по кнопке Build (Со-

брать), расположенной на панели инструментов, и проект будет собран. На рис. 2.6 показан процесс сборки в KToolbar.

Рис. 2.6. Использовать KToolbar для сборки мидлетов или их пакетов очень просто, для этого щелкните по кнопке Build (Собрать), расположенной на инструментальной панели

Хотя процесс сборки включает в себя несколько отдельных шагов, они обычно выполняются так быстро, что их очень трудно различить. На рис. 2.6 показана завершающая стадия построения проекта. Теперь у вас есть верифицированный, откомпилированный и собранный пакет мидлетов,

который можно установить и запустить на мобильном телефоне или протестировать в эмуляторе J2ME.

Тестирование игрового мидлета

Эмулятор J2ME бесценен при тестировании мидлетов игр в процессе разработки. Эмулятор целесообразно использовать ввиду трудностей, возникающих при загрузке кода на реальное устройство снова и снова. Намного эффективнее тестировать мидлеты на настольном компьютере, а к проверке работы на реальных устройствах переходить на поздних этапах отладки.

Чтобы запустить эмулятор J2ME в KToolbar, на инструментальной панели щелкните по кнопке Run (Запустить). На рис. 2.7 показано, как игры из пакета мидлетов отображаются в эмуляторе.

+5550000 - DefaultColor...
MIDlet
Help
Select one to launch:
TilePuzzle
WormGame
PushPuzzle
Launch

Рис. 2.7. По умолчанию в эмуляторе J2ME имитируется телефон с разноцветной лицевой панелью и экраном с размерами 180x210

Как показано на рис. 2.7, эмулятор J2ME отображает изображение телефона, на экране которого выводится состав тестируемого пакета мидлетов. Чтобы запустить одно из приложений мидлета, выберите нужный, щелкая мышью по кнопкам телефона, после чего нажмите кнопку Launch (Запустить). Вы можете также использовать клавиши со стрелками на клавиатуре компьютера, после чего нажать клавишу Enter (Ввод). На рис. 2.8 показана игра Worm при ее выполнении в эмуляторе с настройками телефона по умолчанию.

+5550000 - DefaultColor...
MIDlet
Help
Java
L: 5
S: 0
H: 15
Exit
Menu

Рис. 2.8. Игра Worm – хороший пример простого игрового мидлета

Пример игры Worm – это вариация известной игры Snake, о которой вы узнали в предыдущей главе. Поиграйте немного в игру и поэкспериментируйте с эмулятором. Когда закончите и закроете окно эмулятора, попробуйте другую конфигурацию устройства, которую можно выбрать из выпадающего меню инструмента KToolbar. На рис. 2.9 показана игра PushPuzzle, эмулированная на устройстве с конфигурацией Qwerty.

Рис. 2.9. Конфигурация устройства Qwerty позволяет эмулировать выполнение игр на устройстве, похожем на мобильный телефон с полной клавиатурой

Как видно, эмулятор весьма гибок при имитации мобильных устройств, его можно эффективно использовать как базис для разработки мидлетов игр.

В копилку Игрока

Как и в случае любых инструментов из пакета

J2ME, эмулятор можно запустить непосредственно из командной строки. Однако KToolbar делает использование эмулятора много проще и избавляет вас от необходимости использования командной строки.

Хотя при выборе приложения вы можете попытаться щелкать мышью непосредственно по экрану эмулируемого телефона, помните, что вы должны управлять телефоном, используя только щелчки мыши по кнопкам на корпусе телефона и клавиши клавиатуры.

В копилку Игрока

Если вы хотите протестировать примеры игр на реальном мобильном телефоне, вам необходимо прочитать документацию к своему телефону. Не загружая игры через сеть (речь об этом пойдет чуть позже), для загрузки вы, вероятно, можете использовать USB-кабель или беспроводное Bluetooth-соединение. Но это зависит от вашей модели телефона.

Эмулятор J2ME и реальные устройства

J2ME Wireless Toolkit создан как многоцелевой пакет инструментов разработки J2ME, предназначенный для программирования приложений для большого числа различных моделей телефонов. По этой причине вы не найдете ни одного названия компании производителя и ни одной модели телефона в J2ME Wireless Toolkit. Если говорить подробнее, то

J2ME Wireless Toolkit поддерживает эмуляцию следующих типов устройств:

- ► телефон с черно-белым дисплеем;
- ► устройство Qwerty;
- ► телефон с цветным дисплеем;
- ► Media-обложка.

Первые два типа устройств наиболее важны для эмуляции игр на мобильных телефонах, хотя, вероятно, вам понадобится протестировать игры на других типах устройств. В таблице 2.1 приведены характеристики каждого из типов устройств.

Таблица 2.1. Мобильные устройства, поддерживаемые J2ME Wireless Kit

Наименование устройства	Размер экрана	Цвета	Клавиатура
Мобильный телефон с черно-белым экраном	180 208	256 оттенков серого	ITU-T
Мобильный телефон с цветным экраном	180 208	256 цветов	ITU-T
Media-обложка	180 208	256 цветов	ITU-T
Устройство Qwerty	640 240	256 цветов	QWERTY

Таблица отражает возможности J2ME по эмуляции различных устройств. Обратите внимание, что все устройства, кроме поддерживающих Qwerty, имеют вертикально-ориентированный экран. Телефоны с черно-белыми экранами поддерживают отображение 256 оттенков серого цвета,

а остальные устройства поддерживают 256 цветов. Также устройства отличаются и клавиатурами. Устройства типа Qwerty имеют полную клавиатуру Qwerty, похожую на клавиатуру обычного компьютера. Такие устройства имеют больший размер, по сравнению с обычным мобильным телефоном, подобные модели выпускаются фирмой Research In Motion (RIM). Клавиатура типа ITU-T – это типичная клавиатура мобильных телефонов.

Хотя основные типы устройств, представленные в J2ME Wireless Toolkit, полезны для тестирования мидлетов игр, при этом не указываются конкретные модели и изготовители, но, вероятно, вам потребуется использовать телефон с более конкретными параметрами. Самый простой способ сделать это – запустить J2ME Wireless Toolkit, поставляемый конкретным производителем. Например, инструменты J2ME, предлагаемые Motorola или Nokia, содержат эмуляторы всех основных производимых устройств с поддержкой Java. Тестируя мидлеты игр в эмуляторах этих устройств, вы можете с большой степенью точности увидеть, как работает мидлет на реальном устройстве, как он смотрится на экране.

Резюме

Перед тем как более детально погрузиться в J2ME Wireless Toolkit, вы познакомились с основами разработки компьютерных игр. Хотя они и не высечены на камне, ими

полезно руководствоваться при начале работы над своим шедевром. В этой главе было уделено внимание пакету J2ME Wireless Toolkit, а также его специальным инструментам, дающим возможность собирать мобильные Java-игры.

Вы узнали, что эмулятор J2ME – это особая часть пакета разработки J2ME, он позволяет тестировать мидлеты на настольном компьютере без необходимости загрузки в реальное устройство. Вам, конечно, придется проверять разработанное приложение и на настоящем устройстве, однако эмулятор позволяет ускорить процесс разработки и тестировать приложения на реальных устройствах в случаях крайней необходимости. Вы узнали о стандартном эмуляторе J2ME, который входит в состав пакета J2ME Wireless Toolkit, а также – как его можно усовершенствовать для эмуляции особых моделей мобильных телефонов.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, купив полную легальную версию на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.