

Содержание

| | |
|---|----|
| Предисловие | 8 |
| Предполагаемая аудитория | 8 |
| Организация материала | 8 |
| Типографские соглашения | 9 |
| О примерах кода | 10 |
| Доступность на Safari | 10 |
| Вопросы и замечания | 10 |
| Благодарности | 11 |
| | |
| Глава 1. Приступая к работе | 13 |
| 1.0. Введение | 14 |
| 1.1. Установка JRuby | 17 |
| 1.2. Управление пакетами с помощью RubyGems | 21 |
| 1.3. Одновременное использование Ruby и JRuby | 22 |
| 1.4. Совместное использование gem-пакетов | 24 |
| 1.5. Обращение к Java-классам из Ruby | 25 |
| 1.6. Преобразование Ruby-массива в Java-массив | 29 |
| 1.7. Включение JAR-файлов в путь поиска классов | 30 |
| 1.8. Расширение Java-класса в Ruby | 31 |
| 1.9. Реализация Java-интерфейса на Ruby | 32 |
| 1.10. Открытие Java-классов в JRuby | 37 |
| 1.11. Конфигурирование Eclipse для разработки на языке JRuby | 39 |
| 1.12. Конфигурирование NetBeans для разработки на JRuby | 43 |
| 1.13. Распознавание платформы в приложении JRuby | 47 |
| | |
| Глава 2. JRuby on Rails | 49 |
| 2.0. Введение | 50 |
| 2.1. Установка и настройка Rails | 51 |
| 2.2. Пакетирование приложения Rails для работы в среде Java EE | 54 |
| 2.3. Веб-приложения с внешним репозиторием gem-пакетов | 56 |
| 2.4. Конфигурирование сервлета JRuby-Rack | 57 |
| 2.5. Пакетирование приложения Rails с источником данных JNDI | 58 |
| 2.6. Развертывание Rails на сервере Tomcat | 59 |
| 2.7. Развертывание Rails на сервере JBoss | 62 |
| 2.8. Развертывание Rails на сервере Jetty | 64 |

| | |
|--|------------|
| 2.9. Развертывание Rails с помощью jetty_rails | 65 |
| 2.10. Развертывание Rails на сервере Mongrel | 68 |
| 2.11. Развертывание Rails на сервере GlassFish v2 | 69 |
| 2.12. Gem-пакет GlassFish v3 | 71 |
| 2.13. Использование библиотеки ActiveRecord вне Rails | 72 |
| 2.14. Получение информации о Java-сервлете | 74 |
| 2.15. Конфигурирование хранилища сеансов | 75 |
| 2.16. Управление классами, каталогами и прочими файлами, упакованными в WAR-файл | 75 |
| 2.17. Изменение имени WAR-файла и местоположения рабочей области | 77 |
| 2.18. Развертывание приложения Rails в корневом контексте | 77 |
| 2.19. Создание приложения Rails в интегрированной среде Aptana Studio | 79 |
| 2.20. Доступ к статическим файлам из приложения Rails, работающего в контейнере Java EE | 82 |
| Глава 3. Интеграция с Java | 85 |
| 3.0. Введение | 86 |
| 3.1. Выполнение Ruby-кода из Java-приложения | 87 |
| 3.2. Вызов JRuby с помощью каркаса Bean Scripting Framework | 91 |
| 3.3. Вызов JRuby с помощью технологии Java Scripting | 93 |
| 3.4. Протоколирование из Ruby с помощью библиотеки Jakarta Commons Logging | 95 |
| 3.5. Использование встроенных в Java средств параллельной обработки | 97 |
| 3.6. Создание методов-аксессоров в духе JavaBean | 100 |
| 3.7. Написание единообразного кода | 101 |
| 3.8. Преобразование XML-документов с помощью библиотеки TrAX | 102 |
| 3.9. Создание пула сред исполнения JRuby | 104 |
| 3.10. Удаленное управление с помощью технологии JMX | 106 |
| 3.11. Доступ к платформенно-зависимым библиотекам из JRuby | 108 |
| Глава 4. Java в системах масштаба предприятия | 111 |
| 4.0. Введение | 112 |
| 4.1. Создание контекста JNDI | 113 |

| | |
|--|-----|
| 4.2. Отправка JMS-сообщений | 115 |
| 4.3. Получение JMS-сообщений | 118 |
| 4.4. Реализация компонента Enterprise JavaBean на JRuby | 120 |
| 4.5. Определение Spring-компонентов на JRuby | 123 |
| 4.6. Создание самообновляемых Spring-компонентов на JRuby | 127 |
| 4.7. Встраивание Spring-компонентов, написанных на JRuby | 130 |
| 4.8. Реализация Aware-интерфейсов Spring в JRuby-объектах | 131 |
| 4.9. Создание MVC-контроллеров Spring с помощью JRuby | 134 |
| 4.10. Hibernate и JRuby | 137 |
| 4.11. Java Persistence API и JRuby..... | 140 |
| 4.12. Выполнение вызовов по протоколу SOAP | 141 |
| 4.13. Упрощение доступа к LDAP-каталогу | 143 |

Глава 5. Пользовательский интерфейс и графические приложения

| | |
|--|-----|
| 5.0. Введение | 148 |
| 5.1. Создание приложений Swing | 148 |
| 5.2. Обработка событий Swing | 150 |
| 5.3. Долго работающие задачи в приложениях Swing | 151 |
| 5.4. Пакетирование автономных приложений | 153 |
| 5.5. Пакетирование JRuby-приложений, запускаемых по технологии Web Start | 155 |
| 5.6. Написание апплетов на JRuby | 157 |
| 5.7. Манипулирование изображениями | 161 |
| 5.8. Создание приложений SWT | 164 |
| 5.9. Доступ к рабочему столу | 166 |
| 5.10. Доступ к системному лотку | 167 |
| 5.11. Разработка приложений Swing на предметно-ориентированных языках на базе JRuby | 169 |
| 5.12. Использование библиотеки Monkeybars для разработки приложений Swing | 173 |
| 5.13. Создание приложений Qt с помощью JRuby | 177 |

Глава 6. Инструменты сборки

| | |
|--|-----|
| 6.0. Введение | 182 |
| 6.1. Включение Ruby-сценариев в процесс сборки системой Ant | 182 |

| | |
|---|------------|
| 6.2. Применение Ruby в условных конструкциях Ant | 185 |
| 6.3. Написание задания Ant на Ruby | 187 |
| 6.4. Включение Ruby-сценариев в процесс сборки системой Maven | 188 |
| 6.5. Написание подключаемого к Maven модуля на JRuby | 190 |
| 6.6. Сборка Java-проектов с помощью Raven | 193 |
| 6.7. Ссылка на библиотеки в Raven | 195 |
| 6.8. Организация частного репозитория Raven | 196 |
| 6.9. Прогон тестов JUnit с помощью Raven | 197 |
| 6.10. Сборка Java-проектов с помощью Buildr | 198 |
| 6.11. Ссылка на библиотеки в Buildr | 201 |
| 6.12. Сборка с помощью Rake в контексте сервера Hudson | 202 |
| 6.13. Добавление Ruby-сценария в качестве задачи сервера Hudson | 203 |
| Глава 7. Тестирование | 205 |
| 7.0. Введение | 206 |
| 7.1. Автономное тестирование Java-кода с помощью Test/Unit | 206 |
| 7.2. Автономное тестирование Java-кода с помощью библиотеки dust | 209 |
| 7.3. Автономное тестирование Java-кода с помощью библиотеки Expectations | 210 |
| 7.4. Тестирование Java-кода с помощьюRSpec | 212 |
| 7.5. Создание mock-объектов с помощью библиотеки Mocha | 217 |
| 7.6. Модификация пути поиска классов для JtestR | 219 |
| 7.7. Группировка тестов для JtestR | 219 |
| 7.8. Аргументы командной строки при запуске JtestR | 220 |
| 7.9. Совместное использование JtestR и Ant | 222 |
| 7.10. Совместное использование JtestR и Maven | 223 |
| 7.11. Повышение производительности JtestR | 224 |
| Глава 8. Сообщество пользователей JRuby | 227 |
| 8.0. Введение | 228 |
| 8.1. Сборка JRuby из исходных кодов | 228 |
| 8.2. Отправка извещения о недоработке в JRuby | 229 |
| 8.3. Списки рассылки JRuby | 231 |
| Предметный указатель | 232 |

Предисловие

Язык JRuby – это просто Ruby, воспользовавшийся преимуществами виртуальной машины Java. Он взял все лучшее от Java и дополнил Ruby целым рядом фантастических возможностей.

– *Чарльз Наттер,*
руководитель проекта JRuby,
Twitter, 7 августа 2008

В этом высказывании Чарльз Наттер кратко выразил оба аспекта, которые в последнее время привлекли внимание к проекту JRuby: во-первых, тот факт, что Java – это отличная платформа, пригодная для самых разных языков, а не только собственно для Java, а, во-вторых, рост интереса к языку программирования Ruby. В этой книге мы рассмотрим множество возможных вариантов использования JRuby. Если говорить словами Чарльза, то в одних рецептах речь пойдет о том лучшем, что взято от Java, в других – о фантастических возможностях, привнесенных в Ruby, а в третьих – о том и другом вместе.

Предполагаемая аудитория

Если вы хотите получить от JRuby максимум возможного, то должны свободно переходить от Java к Ruby и обратно. Работая над этой книгой, мы ориентировались на читателя, который в той или иной мере владеет обоими языками, хотя, быть может, одним лучше, чем другим. Поэтому обширного введения вы здесь не найдете, разве что в первой главе, где мы описываем, в каких вопросах Ruby и Java похожи, а в каких – различаются.

Вообще говоря, мы ставили себе целью не рассказать о том или ином уже имеющемся в Ruby или Java средстве, а объяснить, как JRuby позволяет воспользоваться имеющимися средствами или расширить их. Так, рецепты в главе, посвященной JRuby on Rails, рассчитаны на читателя, который уже создал хотя бы одно (работающее) приложение Rails.

Организация материала

Глава 1. *Пристапная к работе*

Эта глава представляет собой краткое введение в JRuby, за которым следует описание ряда простейших приемов работы с этим языком, в том числе применение системы управления пакетами RubyGems и техника обращения к Java-коду из программы на Ruby. В конце главы приводятся рецепты, касающиеся установки и настройки различных интегрированных сред разработки (IDE) для работы с JRuby.

Глава 2. *JRuby on Rails*

Эта глава посвящена разнообразным сценариям развертывания приложений Ruby on Rails с использованием JRuby.

Глава 3. *Интеграция с Java*

Эта глава начинается несколькими рецептами о вызове Ruby-кода из Java-программы. Затем следуют рецепты, описывающие обращение из Ruby к таким популярным Java-библиотекам, как Java Native Access (JNA) и Jakarta Commons Logging.

Глава 4. *Java в системах масштаба предприятия*

Все рецепты, собранные в этой главе, относятся к использованию JRuby в таких предназначенных для создания крупномасштабных систем каркасах, как JMS, JNDI, EJB, Spring и Hibernate.

Глава 5. *Пользовательский интерфейс и графические приложения*

В этой главе описываются некоторые основанные на JRuby каркасы для создания графических пользовательских интерфейсов. Сюда же включены рецепты, касающиеся операций над графическими изображениями, апплетов и интеграции с рабочим столом.

Глава 6. *Инструменты сборки*

В этой главе речь пойдет об использовании JRuby для усовершенствования процедуры сборки Java-проектов. Оба наиболее популярных в мире Java инструмента сборки – Ant и Maven – допускают различные способы подключения JRuby. Здесь же вы найдете рецепты, посвященные программам Raven и Buildr, которые ориентированы исключительно на JRuby.

Глава 7. *Тестирование*

Эта глава посвящена главным образом пакету JtestR, который включает в себя JRuby и ряд популярных инструментов тестирования для языка Ruby. Содержащиеся здесь рецепты научат вас писать на Ruby тесты для Java-программ.

Глава 8. *Сообщество пользователей JRuby*

Это последняя глава. В нее включены сведения о том, как можно принять участие в жизни сообщества пользователей JRuby.

Типографские соглашения

В книге применяются следующие соглашения:

Курсив

Таким начертанием выделяются URL, имена каталогов и файлов, параметры. Иногда оно применяется для выделения важной информации.

Моноширинный шрифт

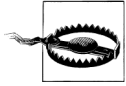
Применяется для листингов программ, а также внутри текста для обозначения элементов программы, как то: имена пространств имен, классов и методов.

Моноширинный курсив

Текст, вместо которого надо подставить значения, вводимые пользователем.



Этот значок обозначает совет, рекомендацию или замечание общего характера.



Этот значок обозначает предупреждение или предостережение.

О примерах кода

Эта книга призвана помогать вам в работе. Поэтому вы можете использовать приведенный в ней код в собственных программах и в документации. Спрашивать у нас разрешение необязательно, если только вы не собираетесь воспроизводить значительную часть кода. Например, не требуется разрешение, чтобы включить в свою программу несколько фрагментов кода из книги. Однако для продажи или распространения примеров на компакт-диске нужно получить разрешение. Можно без ограничений цитировать книгу и примеры в ответах на вопросы. Но чтобы включить значительные объемы кода в документацию по собственному продукту, нужно получить разрешение.

Мы высоко ценим, хотя и не требуем, ссылки на наши издания. В ссылке обычно указываются название книги, имя автора, издательство и ISBN, например: «*JRuby Cookbook*, by Justin Edelson and Henry Liu. Copyright 2009 Justin Edelson and Henry Liu, 978-0-596-51980-3».

Если вы полагаете, что планируемое использование кода выходит за рамки изложенной выше лицензии, пожалуйста, обратитесь к нам по адресу permissions@oreilly.com.

Доступность на Safari

Если на обложке вашей любимой книги присутствует значок Safari® Enabled, это означает, что книга доступна он-лайн в сетевой библиотеке Safari издательства O'Reilly. У Safari есть преимущество перед обычными электронными книгами. Это виртуальная библиотека, которая позволяет легко находить тысячи технических книг, копировать примеры программ, загружать отдельные главы и быстро получать точную и актуальную информацию. Бесплатный доступ по адресу <http://safari.oreilly.com>.



Вопросы и замечания

Мы тщательно проверили информацию, приведенную в этой книге, и протестировали код. Однако полностью исключить ошибки и опечатки невозможно. Если вы

обнаружите какую-нибудь ошибку или захотите высказать пожелание для будущих изданий, пожалуйста, обращайтесь по адресу:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (в США или Канаде)
707-829-0515 (международный или местный)
707-829-0104 (факс)

Замечания и вопросы технического характера следует отправлять по адресу:
bookquestions@oreilly.com

Для этой книги есть веб-страница, на которой выкладываются списки замеченных ошибок, примеры и планы для будущих изданий. Адрес страницы:

<http://www.oreilly.com/catalog/9780596519803>

Дополнительную информацию о наших книгах и прочих предприятиях можно найти на сайте:

<http://www.oreilly.com>

Благодарности

Мы благодарны коллективу издательства O'Reilly, особенно нашему редактору Майку Лукидесу (Mike Loukides) и литературному редактору Коллин Горман (Colleen Gorman). Спасибо также Стивену Шинглеру (Steven Shingler) за его вклад в главу 4. И всем рецензентам этой книги: Хуану Пабло Таркино (Juan Pablo Tarquino), Джону Пэрселлу (John Purcell) и Дэвиду Кунтцу (David Koontz).

Эта книга никогда не появилась бы на свет без неустанных трудов всей команды, работающей над проектом JRuby, а именно: Чарльза Наттера (Charles Nutter), Томаса Энебо (Nick Sieger) и Ола Бини (Ola Bini). Благодарим также корпорацию Sun и компанию ThoughtWorks за постоянную поддержку JRuby. Проект JRuby размещен на сайте The Codehaus; мы благодарны его создателю Бобу Маквиртеру (Bob McWhirter).

Мы также признательны Нику Рокуэллу (Nick Rockwell), который не перестает подбадривать и воодушевлять нас.

Justin Edelson

Эта книга не состоялась бы, если бы не любовь и поддержка моей чудесной жены Элизабет. Отдельное спасибо моим сыновьям: Оуэну, который сам напечатал свое имя, и Бенджамину, который пока еще этого не умеет.

Благодарю своих коллег в компании MTV Networks: Майкла Бенуа (Michael Benoit), Кита Гриффина (Keith Griffin), Рамеша Нуталapati (Ramesh Nuthalapati), Илью Резникова (Ilya Reznikov), Криса Синдела (Chris Sindel), Джеффа Йемина (Jeff Yemin) и Чжун Чжоу (Jun Zhou) за достойную работу.

Спасибо также Уоррену Хабибу (Warren Habib) за поддержку.

Генри Лю

Спасибо моему другу Йону Баэру (Jon Baer), который впервые привел меня на мероприятие, посвященное языку Ruby, и на протяжении многих лет составляет мне компанию в работе. Я благодарен Фрэнсису Хуангу (Francis Hwang), Мэтту Пеллетье (Matt Pelletier), Себастьяну Дельмонту (Sebastian Delmont), Троттеру Кэшину (Trotter Cashion) и всем остальным участникам группы пользователей Ruby в Нью-Йорке. Они учили меня Ruby и Rails, отвечали на мои наивные вопросы. Именно их увлеченность этой технологией и побудила меня копать глубже. Благодарю всех своих коллег по компании MTV Networks и в особенности Марка Эйка (Mark Ache), Люка Мэрфи (Luke Murphy) и Стива Азуэту (Steve Aзуета) за непрестанную поддержку. А больше всех благодарю свою семью и подругу Наоми; без нее ничего с этой книгой не получилось бы.