



Содержание

Предисловие к русскому изданию	16
Предисловие	18
Благодарности	20
Об этой книге	22
Об иллюстрации на обложке	28
Часть I. Ядро Spring	29
Глава 1. Введение в Spring	30
1.1. Упрощение разработки на языке Java	32
1.1.1. Свобода использования POJO	33
1.1.2. Внедрение зависимостей	35
1.1.3. Применение аспектно-ориентированного программирования	40
1.1.4. Устранение шаблонного кода с помощью шаблонов ...	47
1.2. Контейнер компонентов	50
1.2.1. Работа с контекстом приложения	51
1.2.2. Жизненный цикл компонента	53
1.3. Обзор возможностей Spring	54
1.3.1. Модули Spring	55
1.3.2. Дополнительные возможности Spring	59
1.4. Что нового в Spring	64
1.4.1. Что нового в Spring 2.5?	64
1.4.2. Что нового в Spring 3.0?	65
1.4.3. Что нового в экосистеме Spring?	66
1.5. В заключение	67
Глава 2. Связывание компонентов	69
2.1. Объявление компонентов	70

2.1.1. Подготовка конфигурации Spring	71
2.1.2. Объявление простого компонента	73
2.1.3. Внедрение через конструкторы	74
2.1.4. Область действия компонента	80
2.1.5. Инициализация и уничтожение компонентов	82
2.2. Внедрение в свойства компонентов	84
2.2.1. Внедрение простых значений	86
2.2.2. Внедрение ссылок на другие компоненты	87
2.2.3. Связывание свойств с помощью пространства имен p	91
2.2.4. Внедрение коллекций	92
2.2.5. Внедрение пустого значения	99
2.3. Внедрение выражений	99
2.3.1. Основы языка выражений SpEL	100
2.3.2. Выполнение операций со значениями SpEL	105
2.3.3. Обработка коллекций на языке SpEL	111
2.4. В заключение	116

Глава 3. Дополнительные способы связывания компонентов

компонентов	117
3.1. Объявление родителей и потомков компонентов	118
3.1.1. Абстрактные компоненты	119
3.1.2. Общие абстрактные свойства	122
3.2. Внедрение методов	124
3.2.1. Основы замещения методов	125
3.2.2. Использование внедрения методов чтения	130
3.3. Внедрение не-Spring компонентов	132
3.4. Пользовательские редакторы свойств	135
3.5. Специальные компоненты Spring	140
3.5.1. Компоненты постобработки	140
3.5.2. Постобработка контейнера	144
3.5.3. Внешние файлы с настройками свойств	145
3.5.4. Извлечение текстовых сообщений	147
3.5.5. Уменьшение связности с использованием событий ...	150
3.5.6. Создание «осведомленных» компонентов	153
3.6. Компоненты, управляемые сценариями	156
3.6.1. Добавляем лайм в кокос	158

3.6.2. Компонент, управляемый сценарием.....	159
3.6.3. Внедрение в свойства компонентов, управляемых сценариями	162
3.6.4. Обновление компонентов, управляемых сценариями	164
3.6.5. Создание компонентов, управляемых сценариями, непосредственно в конфигурации.....	165
3.7. В заключение.....	166

Глава 4. Сокращение размера XML-конфигурации Spring

168

4.1. Автоматическое связывание свойств компонентов	169
4.1.1. Четыре типа автоматического связывания	169
4.1.2. Смешивание автоматического и явного связывания.....	175
4.2. Связывание посредством аннотаций	177
4.2.1. Использование аннотации @Autowired	178
4.2.2. Автоматическое связывание с применением стандартной аннотации @Inject.....	183
4.2.3. Использование выражений в аннотациях внедрения зависимостей.....	186
4.3. Автоматическое определение компонентов.....	188
4.3.1. Аннотирование компонентов для автоматического определения	189
4.3.2. Включение фильтров в элемент component-scans ...	190
4.4. Конфигурирование Spring в программном коде на Java....	192
4.4.1. Подготовка к конфигурированию на языке Java	192
4.4.2. Определение конфигурационных классов	193
4.4.3. Объявление простого компонента	194
4.4.4. Внедрение зависимостей в конфигурации на языке Java	195
4.5. В заключение.....	196

Глава 5. Аспектно-ориентированный Spring

198

5.1. Знакомство с AOP	200
5.1.1. Определение терминологии AOP	201
5.1.2. Поддержка AOP в Spring	204

5.2. Выбор точек сопряжения в описаниях срезов	207
5.2.1. Определение срезов множества точек сопряжения	209
5.2.2. Использование указателя bean()	210
5.3. Объявление аспектов в XML	211
5.3.1. Объявление советов, выполняемых до или после	213
5.3.2. Объявление советов, выполняемых до, и после	216
5.3.3. Передача параметров советам	218
5.3.4. Внедрение новых возможностей с помощью аспектов	221
5.4. Аннотирование аспектов	224
5.4.1. Создание советов, выполняемых и до, и после	227
5.4.2. Передача аргументов аннотированным советам	228
5.4.3. Внедрение с помощью аннотаций	230
5.5. Внедрение аспектов AspectJ	231
5.6. В заключение	235
Часть II. Основы приложений Spring	237
Глава 6. Работа с базами данных	238
6.1. Философия доступа к данным в Spring	239
6.1.1. Знакомство с иерархией исключений доступа к данным в Spring	241
6.1.2. Шаблоны доступа к данным	244
6.1.3. Использование классов поддержки DAO	247
6.2. Настройка источника данных	249
6.2.1. Использование источников данных из JNDI	249
6.2.2. Использование пулов соединений	250
6.2.3. Источник данных JDBC	252
6.3. Использование JDBC совместно со Spring	253
6.3.1. Борьба с разбуханием JDBC-кода	254
6.3.2. Работа с шаблонами JDBC	258
6.4. Интеграция Hibernate и Spring	265
6.4.1. Обзор Hibernate	267
6.4.2. Объявление фабрики сеансов Hibernate	268
6.4.3. Создание классов для работы с Hibernate, независимых от Spring	271

6.5. Spring и Java Persistence API	273
6.5.1. Настройка фабрики диспетчера сущностей.....	274
6.5.2. Объект DAO на основе JPA.....	280
6.6. Кеширование.....	282
6.6.1. Настройка кеширования.....	285
6.6.2. Настройка компонентов для кеширования.....	289
6.6.3. Декларативное кеширование с помощью аннотаций	292
6.7. В заключение.....	294
Глава 7. Управление транзакциями.....	296
7.1. Знакомство с транзакциями	297
7.1.1. Описание транзакций в четырех словах.....	299
7.1.2. Знакомство с поддержкой транзакций в Spring	300
7.2. Выбор диспетчера транзакций.....	301
7.2.1. Транзакции JDBC	303
7.2.2. Транзакции Hibernate	303
7.2.3. Транзакции Java Persistence API	304
7.2.4. Транзакции Java Transaction API	305
7.3. Программное управление транзакциями в Spring	306
7.4. Декларативное управление транзакциями.....	309
7.4.1. Определение атрибутов транзакций.....	309
7.4.2. Объявление транзакций в XML.....	315
7.4.3. Определение транзакций с помощью аннотаций.....	318
7.5. В заключение.....	320
Глава 8. Создание веб-приложений с помощью Spring MVC.....	321
8.1. Обзор Spring MVC	322
8.1.1. Путь одного запроса через Spring MVC.....	322
8.1.2. Настройка Spring MVC	324
8.2. Создание простого контроллера.....	327
8.2.1. Настройка поддержки аннотаций в Spring MVC.....	328
8.2.2. Контроллер главной страницы.....	329
8.2.3. Поиск представлений	334
8.2.4. Объявление представления главной страницы	340
8.2.5. Завершение определения контекста приложения Spring	342

8.3. Контроллер обработки входных данных	344
8.3.1. Создание контроллера, обрабатывающего входные данные.....	345
8.3.2. Представление, отображающее список сообщений	348
8.4. Обработка форм	349
8.4.1. Отображение формы регистрации	350
8.4.2. Обработка данных формы	353
8.4.3. Проверка входных данных	355
8.5. Выгрузка файлов	361
8.5.1. Добавление в форму поля выгрузки файла	361
8.5.2. Прием выгружаемых файлов	363
8.5.3. Настройка Spring для выгрузки файлов	367
8.6. Альтернативы JSP1	368
8.6.1. Использование шаблонов Velocity	368
8.6.2. Использование шаблонов FreeMarker.....	377
8.7. Генерирование вывода, отличного от HTML.....	383
8.7.1. Создание электронных таблиц Excel	384
8.7.2. Создание документов PDF	388
8.8. В заключение.....	391
Глава 9. Использование Spring Web Flow	392
9.1. Установка Spring Web Flow	393
9.1.1. Настройка расширения Web Flow в Spring.....	394
9.2. Элементы последовательности операций.....	397
9.2.1. Состояния	398
9.2.2. Переходы	402
9.2.3. Данные в последовательностях.....	404
9.3. Соединяем все вместе: последовательность pizza.....	407
9.3.1. Определение основной последовательности	407
9.3.2. Сбор информации о клиенте	412
9.3.3. Оформление заказа	420
9.3.4. Прием оплаты	424
9.4. Безопасность веб-последовательностей	427
9.5. Интеграция Spring Web Flow с другими фреймворками.....	427
9.5.1. JavaServer Faces.....	428
9.6. В заключение.....	430

Глава 10. Безопасность в Spring	432
10.1. Введение в Spring Security	433
10.1.1. Обзор Spring Security.....	434
10.1.2. Использование конфигурационного пространства имен Spring Security	435
10.2. Безопасность веб-запросов.....	436
10.2.1. Сервлет-фильтры	437
10.2.2. Минимальная настройка безопасности	438
10.2.3. перехват запросов	443
10.3. Безопасность на уровне представлений	447
10.3.1. Доступ к информации об аутентификации	448
10.3.2. Отображение с учетом привилегий.....	449
10.4. Аутентификация пользователей.....	452
10.4.1. Настройка репозитория в памяти	453
10.4.2. Аутентификация с использованием базы данных... ..	455
10.4.3. Аутентификация с использованием LDAP	457
10.4.4. Включение функции «запомнить меня»	462
10.5. Защита методов	463
10.5.1. Защита методов с помощью аннотации @Secured	464
10.5.2. Использование аннотации JSR-250 @RolesAllowed	465
10.5.3. Защита с помощью аннотаций, выполняемых до и после вызова.....	466
10.5.4. Объявление точек внедрения для защиты методов.....	472
10.6. В заключение.....	472
Часть III. Интеграция Spring	474
Глава 11. Взаимодействие с удаленными службами	476
11.1. Обзор механизмов удаленных взаимодействий в Spring	477
11.2. Использование RMI.....	481
11.2.1. Экспортирование службы RMI	481

11.2.2. Внедрение службы RMI	484
11.3. Экспортирование удаленных служб с помощью Hessian и Burlap	488
11.3.1. Экспортирование службы с помощью Hessian/Burlap	489
11.3.2. Доступ к службам Hessian/Burlap	492
11.4. Использование Spring Http Invoker	494
11.4.1. Экспортирование компонентов в виде служб HTTP Invoker	495
11.4.2. Доступ к службам HTTP Invoker	497
11.5. Экспортирование и использование веб-служб	498
11.5.1. Создание конечных точек JAX-WS с поддержкой Spring	500
11.5.2. Проксирование служб JAX-WS на стороне клиента	505
11.6. В заключение	507

Глава 12. Поддержка архитектуры REST в Spring 509

12.1. Обзор архитектуры REST	510
12.1.1. Основы REST	510
12.1.2. Поддержка REST в Spring	511
12.2. Создание контроллеров, ориентированных на ресурсы	512
12.2.1. Устройство контроллера, противоречащего архитектуре REST	512
12.2.2. Обработка адресов URL в архитектуре RESTful	514
12.2.3. Выполнение операций в стиле REST	519
12.3. Представление ресурсов	523
12.3.1. Договоренность о представлении ресурса	524
12.3.2. Преобразование HTTP-сообщений	528
12.4. Клиенты REST	532
12.4.1. Операции класса RestTemplate	534
12.4.2. Чтение ресурсов	536
12.4.3. Изменение ресурсов	540
12.4.4. Удаление ресурсов	542
12.4.5. Создание новых ресурсов	543
12.4.6. Обмен ресурсами	546

12.5. Отправка форм в стиле RESTful	549
12.5.1. Отображение скрытого поля с именем метода	550
12.5.2. Преобразование типа запроса	552
12.6. В заключение	553
Глава 13. Обмен сообщениями в Spring	555
13.1. Краткое введение в JMS	556
13.1.1. Архитектура JMS	557
13.1.2. Преимущества JMS	561
13.2. Настройка брокера сообщений в Spring	563
13.2.1. Создание фабрики соединений	563
13.2.2. Объявление приемников ActiveMQ	565
13.3. Работа с шаблонами JMS в Spring	566
13.3.1. Борьба с разбуханием JMS-кода	566
13.3.2. Работа с шаблонами JMS	568
13.4. Создание POJO, управляемых сообщениями	575
13.4.1. Создание объекта для чтения сообщений	576
13.4.2. Настройка обработчиков сообщений	578
13.5. Механизмы RPC, основанные на сообщениях	579
13.5.1. Механизм RPC, основанный на сообщениях, в фреймворке Spring	580
13.5.2. Механизм RPC, основанный на сообщениях, в Lingo	583
13.6. В заключение	586
Глава 14. Управление компонентами Spring с помощью JMX	588
14.1. Экспортирование компонентов Spring как управляемых компонентов	589
14.1.1. Экспортирование методов по их именам	593
14.1.2. Определение экспортируемых операций и атрибутов с помощью интерфейсов	596
14.1.3. Объявление управляемых компонентов с помощью аннотаций	597
14.1.4. Разрешение конфликтов между управляемыми компонентами	599
14.2. Удаленные компоненты MBean	601

14.2.1. Экспортирование удаленного компонента MBean	602
14.2.2. Доступ к удаленным компонентам MBean.....	603
14.2.3. Проксирование управляемых компонентов	605
14.3. Обработка извещений	606
14.3.1. Прием извещений	609
14.4. В заключение	610

Глава 15. Создание веб-служб на основе модели contract-first

15.1. Введение в Spring-WS	613
15.2. Определение API службы (в первую очередь!).....	616
15.2.1. Создание примеров XML-сообщений	616
15.3. Обработка сообщений в веб-службе	623
15.3.1. Создание конечной точки на основе модели JDOM	625
15.3.2. Маршалинг содержимого сообщений.....	628
15.4. Связываем все вместе	632
15.4.1. Spring-WS: общая картина	633
15.4.2. Отображение сообщений в конечные точки	634
15.4.3. Настройка конечной точки службы.....	635
15.4.4. Настройка маршалера сообщений	636
15.4.5. Обработка исключений в конечной точке	639
15.4.6. Создание WSDL-файлов	641
15.4.7. Развертывание службы.....	646
15.5. Использование веб-служб Spring-WS	647
15.5.1. Работа с шаблонами веб-служб.....	648
15.5.2. Использование поддержки шлюза веб-служб	656
15.6. В заключение	658

Глава 16. Spring и Enterprise JavaBeans

16.1. Внедрение компонентов EJB в Spring.....	661
16.1.1. Проксирование сеансовых компонентов (EJB 2.x).....	663
16.1.2. Внедрение компонентов EJB в компоненты Spring	668
16.2. Разработка компонентов с поддержкой Spring (EJB 2.x)	670

16.3. Spring и EJB3	673
16.3.1. Pitchfork.....	674
16.3.2. Введение в Pitchfork	676
16.3.3. Внедрение ресурсов с помощью аннотации	676
16.3.4. Объявление перехватчиков с помощью аннотаций.....	678
16.4. В заключение	680
Глава 17. Прочее	682
17.1. Импортирование внешних настроек.....	683
17.1.1. Подстановка переменных-заполнителей	684
17.1.2. Переопределение свойств	687
17.1.3. Шифрование внешних определений свойств.....	689
17.2. Внедрение объектов из JNDI	692
17.2.1. Работа с обычным JNDI API.....	692
17.2.2. Внедрение объектов из JNDI	695
17.2.3. Внедрение компонентов EJB в Spring	699
17.3. Отправка электронной почты	701
17.3.1. Настройка отправки электронной почты	701
17.3.2. Создание электронных писем.....	704
17.4. Выполнение заданий по расписанию и в фоновом режиме	711
17.4.1. Объявление методов, вызываемых по расписанию	712
17.4.2. Объявление асинхронных методов	714
17.5. В заключение	716
17.6. Конец?	717
Предметный указатель.....	719



Предисловие к русскому изданию

Русское издание книги «Spring в действии» представляет собой объединение 2-го и 3-го изданий оригинальной книги «Spring in Action». Автор, пытаясь максимально сократить объем нового издания, убрал из него обсуждение тем, достаточно важных на наш взгляд, и постоянно отсылает интересующихся к предыдущему изданию. По этой причине было принято решение добавить в перевод нового издания главы и разделы из предыдущего оригинального издания (перечисленные ниже), где обсуждаются темы, остающиеся актуальными и поныне.

Глава 3 «Дополнительные способы связывания компонентов» (глава 3 во 2-м издании) рассматривает дополнительные приемы связывания компонентов. Эти приемы не имеют такого же широкого применения, как приемы связывания, представленные в главе 2, но не теряют от этого своей практической ценности.

Раздел 6.6 «Кеширование» (раздел 5.7 во втором издании) демонстрирует приемы кеширования, играющие важную роль в обеспечении высокой производительности приложений.

Раздел 9.5 «Интеграция Spring Web Flow с другими фреймворками» (раздел 15.4 во втором издании). В 3-м издании демонстрируются приемы использования фреймворка Spring Web Flow, однако автор убрал из третьего издания разделы, демонстрирующие возможность интеграции Spring Web Flow с другими фреймворками, такими как Jakarta Struts и JavaServer Faces. Если интеграция с фреймворком Jakarta Struts действительно потеряла свою актуальность, то проблема интеграции с JavaServer Faces все еще представляет интерес для разработчиков.

Глава 15 «Создание веб-служб на основе модели contract-first» демонстрирует применение фреймворка Spring-WS для создания веб-служб с применением модели «contract-first», когда сначала создается WSDL-определение веб-службы, а затем на его основе разрабатывается реализация, за счет чего достигается отделение внешнего и внутреннего API службы.

Глава 16 «Spring и Enterprise JavaBeans» описывает приемы внедрения компонентов EJB в контекст Spring, способы создания компонентов EJB с поддержкой Spring и возможность применения аннотаций EJB 3 к компонентам Spring.

К сожалению, во 2-м и 3-м изданиях обсуждение основывается на разных примерах, поэтому в перечисленных выше главах и разделах (кроме главы 3) читатель столкнется с упоминанием примеров, не обсуждавшихся ранее. Мы искренне надеемся, что это не мешает читателю ухватить основной смысл и заранее приносим свои извинения за несогласованность.



Предисловие

Ого! Прошло почти семь лет с того момента, как была выпущена версия Spring 1.0 и мы с Райаном Брейденбахом (Ryan Breidenbach) приступили к работе над первым изданием книги «Spring в действии». Кто бы мог предположить тогда, что Spring окажет такое сильное влияние на разработку приложений на языке Java?

В первом издании Райан и я пытались охватить все аспекты фреймворка Spring. И мы во многом преуспели. В то время все, что касалось Spring, легко можно было уместить в 11 глав, вместе с описанием приемов внедрения зависимостей, АОР, хранения данных, транзакций, Spring MVC и Acegi Security в главных ролях. Конечно, тогда все это необходимо было сдобрить большим количеством XML. (Помнит ли кто-нибудь, как выглядело объявление транзакций с применением `TransactionProxyFactoryBean`?)

К тому времени, когда я взялся за второе издание книги, фреймворк Spring значительно вырос. Я снова попытался уместить все в единственную книгу и обнаружил, что это невозможно. Фреймворк Spring вырос настолько, что его невозможно стало описать в 700–800-страничной книге. Из второго издания фактически пришлось вырезать целые главы из-за нехватки места.

С момента выхода второго издания прошло три года, и сменились две основные версии Spring. Фреймворк Spring разросся еще больше, и теперь пришлось бы написать несколько томов, чтобы полностью охватить все его возможности. Уместить все необходимые знания о Spring в одну книгу стало просто нереально.

Поэтому я даже не пытался сделать это.

Часто с каждым последующим изданием книги становятся все толще. Но вы наверняка обратили внимание, что это, третье издание «Spring в действии», содержит меньше страниц, чем второе издание. Это стало возможным по двум причинам.

Так как я не смог уместить все в один том, мне пришлось выбирать, какие темы должны быть включены. Я решил сосредоточиться на самых основных, которые, на мой взгляд, должно знать

большинство разработчиков, использующих фреймворк Spring. Это не говорит о том, что другие темы неважны, но эти составляют самую суть разработки с применением Spring.

Другая причина, почему это издание стало меньше, в том, что одновременно с ростом фреймворк Spring становился проще с каждым выпуском. Богатый набор пространств имен, конфигураций Spring, заимствование моделей программирования на основе аннотаций, а также внедрение рациональных соглашений и умолчаний уменьшили конфигурацию Spring, которая превратилась из последовательности страниц разметки XML в горстку элементов.

Но будьте уверены: несмотря на меньшее число страниц, мне удалось уместить в них много новых сведений о Spring. Наряду с описанием приемов внедрения зависимостей, АОР и декларативных транзакций, существующих уже относительно давно, книга содержит новые сведения, появившиеся или изменившиеся после выхода второго издания, часть которых перечислена ниже:

- ❑ создание компонентов на основе аннотаций существенно уменьшает размер XML-конфигурации Spring;
- ❑ новый язык выражений для динамического вычисления значений свойств компонентов во время выполнения;
- ❑ совершенно новый фреймворк Spring MVC, действующий на основе аннотаций, который стал намного гибче прежнего иерархического фреймворка контроллеров;
- ❑ обеспечение безопасности приложений с помощью фреймворка Spring Security стало намного проще благодаря новому конфигурационному пространству имен, удобным значениям по умолчанию и поддержке правил безопасности, ориентированных на применение выражений;
- ❑ отличная поддержка конструирования и обработки ресурсов REST, основанная на фреймворке Spring MVC.

Надеюсь, что эта книга станет исчерпывающим руководством и для опытных, и для начинающих разработчиков, использующих фреймворк Spring в своих проектах.



Об этой книге

Фреймворк Spring Framework создавался с весьма конкретной целью – чтобы упростить разработку приложений Java EE. Аналогично и книга «Spring в действии» была написана, чтобы упростить обучение применению фреймворка Spring. Моя цель не в том, чтобы дать вам подробный перечень Spring API. Вместо этого я надеюсь описать Spring Framework как разработчик приложений Java EE, представив практические примеры программного кода. Поскольку фреймворк Spring имеет модульную организацию, эта книга организована таким же образом. Я понимаю, что разные разработчики имеют разные потребности. Некоторые могут желать изучать Spring Framework, начиная с самых основ, другие – проявлять разборчивость в выборе тем и двигаться в собственном темпе. В этом смысле начинающие могут воспринимать книгу как инструмент для первоначального знакомства с фреймворком Spring, а желающие глубже вникнуть в конкретные особенности – как руководство и справочник.

Кому адресована эта книга

Третье издание книги «Spring в действии» адресовано всем Java-разработчикам, но особенно полезной она будет для разработчиков корпоративных приложений на Java. Я проведу вас через примеры программного кода, сложность которых будет расти на протяжении каждой главы, но вы должны понимать, что истинная мощь фреймворка Spring проявляется в простоте разработки крупных приложений. Поэтому разработчики корпоративных приложений смогут в полной мере оценить примеры, представленные в книге.

Большая часть фреймворка Spring обеспечивает реализацию корпоративных служб, между Spring и EJB можно провести множество параллелей. Поэтому в сравнении этих двух фреймворков будет полезен любой опыт. Данной теме будет посвящена часть этой книги. Фактически последние пять глав демонстрируют способности фреймворка Spring поддерживать интеграцию веб-приложений.

Особенно ценной эту часть книги найдут разработчики корпоративных приложений.

Структура книги

Книга «Spring в действии» делится на три части. Первая часть является введением в основы фреймворка Spring Framework. Во второй части будут представлены общие элементы приложений на основе фреймворка Spring. Заключительная часть покажет, как можно использовать фреймворк Spring для интеграции с другими приложениями и службами.

В первой части будет проведено исследование двух основных особенностей фреймворка Spring Framework – внедрения зависимостей (Dependency Injection, DI) и аспектно-ориентированного программирования (Aspect-Oriented Programming, AOP). Это даст возможность хорошо усвоить основы Spring, которые будут использоваться на протяжении всей книги.

В главе 1 будут представлены DI и AOP и рассказано, как их использовать в разработке слабосвязанных Java-приложений.

В главе 2 подробно будет рассказываться, как конфигурировать и связывать прикладные объекты, используя прием внедрения зависимостей. Она покажет, как писать слабосвязанные компоненты и устанавливать их зависимости и свойства в контейнере Spring с использованием XML.

После овладения основами связывания компонентов можно приступить к знакомству с дополнительными особенностями контейнера Spring в главе 3. Среди прочего здесь рассказывается, как управлять жизненным циклом компонентов приложения, как определять отношения родитель/потомок в конфигурациях компонентов и как внедрять компоненты сценариев, написанных на языках Ruby и Groovy.

После знакомства в главе 4 с основами конфигурирования Spring с помощью XML будут представлены аспектно-ориентированные альтернативы.

Глава 5 рассматривает возможность использования Spring AOP для отделения сквозных задач от объектов, которые они обслуживают. Эта глава также готовит почву для последующих глав, где Spring AOP будет использоваться для предоставления декларативных услуг, таких как транзакции, безопасность и кеширование.

Вторая часть основывается на особенностях DI и AOP, представленных в первой части, и показывает, как использовать эти понятия для создания общих элементов приложения.

Глава 6 охватывает механизм хранения данных в Spring. Здесь будет представлена поддержка JDBC в Spring, позволяющая убрать из приложения массу шаблонного программного кода, связанного с использованием JDBC. Здесь также будет показано, как Spring взаимодействует с фреймворками хранения данных, такими как Hibernate и Java Persistence API (JPA).

Глава 7 дополняет главу 6, демонстрируя возможность обеспечения целостности баз данных с использованием поддержки транзакций в Spring. Она покажет, как Spring с помощью AOP позволяет наделить простые прикладные объекты мощностью декларативных транзакций.

В главе 8 будет представлен веб-фреймворк MVC, входящий в состав Spring. Здесь будет показано, как Spring может прозрачно связывать веб-параметры бизнес-объектов и одновременно обеспечивать контроль допустимости значений и обработку ошибок. Здесь также будет показано, насколько просто расширять функциональность веб-приложений с помощью контроллеров Spring MVC.

В главе 9 исследуется Spring Web Flow – расширение Spring MVC, позволяющее создавать диалоговые веб-приложения. В этой главе рассказывается, как создавать веб-приложения, обеспечивающие выполнение действий пользователями в определенной последовательности.

В главе 10 рассказывается, как обеспечить защиту приложений с помощью Spring Security. Здесь будет показано, как Spring Security защищает приложение на уровне веб-запросов, с использованием сервлетов-фильтров, и на уровне метода, с использованием Spring AOP.

После создания приложения на основе тех знаний, что были получены во второй части, может появиться желание организовать его взаимодействие с другими приложениями или службами. Как это сделать, вы узнаете в третьей части книги.

Глава 11 рассматривает возможность представления прикладных объектов в виде удаленных служб. Здесь также рассказывается, насколько просто можно получить доступ к удаленным службам, как если бы они были обычными объектами в приложении. В число рассматриваемых здесь технологий распределенных вычислений входят RMI, Hessian/Burlap, веб-службы, действующие на основе протокола SOAP, и собственная технология Spring HTTP Invoker.

Глава 12 вновь возвращается к фреймворку Spring MVC и демонстрирует возможность его использования для обеспечения доступа

к данным в приложении как к ресурсам RESTful. Кроме того, здесь рассказывается, как создаются REST-клиенты с помощью Spring RestTemplate.

Глава 13 рассказывает о возможности использования Spring для передачи и получения асинхронных сообщений с помощью JMS. Помимо основных операций, поддерживаемых JMS API в Spring, здесь также будет показано, как использовать открытый проект Lingo для реализации и использования удаленных служб на базе JMS.

Глава 14 покажет, как управлять прикладными объектами с помощью JMX.

В главе 15 будет представлен иной подход к реализации веб-служб, с применением фреймворка Spring-WS. Здесь вы узнаете, как с помощью Spring-WS создавать веб-службы с применением модели «contract-first», когда определение API службы отделено от его реализации.

Несмотря на то, что фреймворк Spring в значительной степени устраняет необходимость использовать компоненты EJB, на практике все же может возникнуть потребность совместного использования Spring и EJB. Возможности интеграции Spring и EJB исследуются в главе 16. Здесь вы узнаете, как писать компоненты EJB с поддержкой Spring, как внедрять ссылки на компоненты EJB в контекст приложения Spring и даже как использовать EJB-подобные аннотации для настройки компонентов Spring.

В заключение исследования фреймворка Spring, в главе 17, будет показано, как задействовать Spring для выполнения заданий по расписанию, отправлять электронную почту, обращаться к ресурсам, настроенным с использованием JNDI.

Соглашения об оформлении программного кода

Книга содержит множество примеров программного кода. Эти примеры всегда будут напечатаны моноширинным шрифтом. Если мне потребуется привлечь ваше внимание к какой-то части примера, он будет выделен жирным шрифтом. Все имена классов, методов и фрагменты XML в тексте также будут напечатаны моноширинным шрифтом.

Многие классы и пакеты в Spring имеют весьма длинные (но выразительные) имена. Вследствие этого везде, где необходимо, будут добавляться стрелки (↵), обозначающие продолжение строки.

Не все примеры программного кода в этой книге будут полными. Часто я буду показывать только один-два метода класса, о котором

идет речь, чтобы не уходить в сторону от обсуждаемой темы. Полные исходные тексты приложений, представленных в книге, можно загрузить с веб-сайта издательства www.manning.com/SpringActionThirdEdition.

Об авторе

Крейг Уоллс – программист с 13-летним опытом и соавтор книг «XDoclet in Action» (Manning, 2003) и двух предыдущих изданий «Spring in Action» (Manning, 2005 и 2007). Он горячо пропагандирует фреймворк Spring Framework, часто выступая перед местными группами пользователей и на конференциях, а также ведет обсуждение Spring в своем блоге. Когда Крейг не пишет программный код, он проводит все свое свободное время с супругой и двумя дочерьми, шестью птицами, четырьмя собаками, двумя котами и постоянно меняющимся числом тропических рыб. Крейг живет в Плано, штат Техас.

Автор в сети

Одновременно с покупкой третьего издания книги «Spring в действии» вы получаете бесплатный доступ к частному веб-форуму, организованному издательством Manning Publications, где можно оставлять комментарии о книге, задавать технические вопросы, а также получать помощь от автора и других пользователей. Чтобы получить доступ к форуму и зарегистрироваться на нем, откройте в веб-браузере страницу www.manning.com/SpringinActionThirdEdition. Здесь описывается, как попасть на форум после регистрации, какие виды помощи доступны и правила поведения на форуме.

Издательство Manning обязуется предоставить своим читателям место встречи, где может состояться содержательный диалог между отдельными читателями и между читателями и автором. Но со стороны автора отсутствуют какие-либо обязательства уделять форуму какое-то определенное внимание – его присутствие на форуме остается добровольным (и неоплачиваемым). Мы предлагаем задавать автору стимулирующие вопросы, чтобы его интерес не угасал!

Форум и архивы предыдущих обсуждений будут доступны на сайте издательства, пока книга находится в печати.

О названии

Сочетая в себе введение, краткий обзор и примеры использования, книги из серии «в действии» предназначены для изучения и

запоминания. Согласно исследованиям в когнитивистике, люди запоминают вещи, которые они узнают в процессе самостоятельного изучения.

В издательстве Manning нет ученых-когнитивистов, тем не менее мы уверены, что для надежного усваивания необходимо пройти через стадии исследования, игры и, что интересно, пересказа всего, что было изучено. Люди усваивают новые знания до уровня овладения ими, только после их активного исследования. Люди учатся в действии. Особенность учебников из серии «в действии» – в том, что они основываются на примерах. Это побуждает читателя пробовать, играть с новым кодом, исследовать новые идеи.

Есть и другая, более прозаическая причина выбора такого названия книги: наши читатели – занятые люди. Они используют книги для выполнения работы или для решения проблем. Им нужны книги, позволяющие легко перепрыгивать с места на место и изучать только то, что они хотят, и только когда они этого хотят. Им нужны книги, помогающие в действии. Книги этой серии создаются для таких читателей.