Моему отцу, который обещал прочитать эту книгу, даже если не сможет понять в ней ни слова. Возможно, твое сердце переполняется гордостью каждый раз, когда ты говоришь своим друзьям, что твой сын известный автор книг

Моей матери, которая верила в меня, когда никто не верил, и упорно трудилась для того, чтобы я получил любую возможность добиться успеха в жизни. Твой тяжелый труд, вера и стойкость всегда были источником вдохновения для меня. Спасибо, что передала мне эти черты характера, поскольку они были основой всех моих достижений, включая эту книгу

Краткое оглавление

ЧАСТЬ I. Знакомство с Apache Pulsar 29

1 ■ *Введение в Apache Pulsar* 32
2 ■ *Концепции и архитектура Pulsar* 77
3 ■ *Взаимодействие с Pulsar* 115

ЧАСТЬ II. Основы разработки с использованием Apache Pulsar 151

4 ■ *Функции Pulsar* 153
5 ■ *Коннекторы ввода-вывода Pulsar* 194
6 ■ *Обеспечение безопасности Pulsar* 232
7 ■ *Реестр схем* 270

ЧАСТЬ III. Практическая разработка приложений с использованием Apache Pulsar 303

8 ■ *Паттерны применения Pulsar Functions* 305
9 ■ *Паттерны устойчивости* 329
10 ■ *Доступ к данным* 368
11 ■ *Машинное обучение в Pulsar* 392
12 ■ *Периферийная аналитика* 415

Оглавление

Предисловие от издательства ... 13
Предисловие ... 14
Предисловие автора ... 16
Благодарности ... 19
Об этой книге ... 21
Об авторе ... 27
Об иллюстрации на обложке ... 28

ЧАСТЬ I. Знакомство с Apache Pulsar ... 29

1 ***Введение в Apache Pulsar*** ... **32**

1.1. Корпоративные системы обмена сообщениями ... 33
1.1.1. Основные функциональные возможности ... 36
1.2. Паттерны потребления сообщений ... 37
1.2.1. Обмен сообщениями по схеме публикация–подписка ... 37
1.2.2. Очередь сообщений ... 38
1.3. История развития систем обмена сообщениями ... 39
1.3.1. Системы обмена сообщениями общего назначения ... 39
1.3.2. Программное обеспечение среднего звена, ориентированное на сообщения ... 40
1.3.3. Сервисная шина предприятия ... 42
1.3.4. Распределенные системы обмена сообщениями ... 45
1.4. Сравнение с Apache Kafka ... 52
1.4.1. Многоуровневая архитектура ... 52
1.4.2. Потребление сообщений ... 55
1.4.3. Постоянство хранения данных ... 58
1.4.4. Подтверждение приема сообщений ... 61
1.4.5. Длительность хранения сообщений ... 64
1.5. Почему необходим именно Pulsar ... 65
1.5.1. Гарантированная доставка сообщений ... 66
1.5.2. Неограниченная масштабируемость ... 66
1.5.3. Устойчивость к критическим ошибкам ... 67
1.5.4. Поддержка миллионов тем ... 69
1.5.5. Георепликация и активная отказоустойчивость ... 70
1.6. Варианты использования из реальной практики ... 72
1.6.1. Универсальные системы обмена сообщениями ... 72
1.6.2. Платформы микросервисов ... 73
1.6.3. Автомобили с сетевыми функциями ... 74
1.6.4. Выявление случаев мошенничества ... 74

1.7. Дополнительные информационные ресурсы 75
1.8. Резюме 76

2 *Концепции и архитектура Pulsar* 77

2.1. Физическая архитектура Pulsar 78
2.1.1. Многоуровневая архитектура Pulsar 79
2.1.2. Уровень обслуживания без сохранения состояния 81
2.1.3. Уровень хранения потоковых данных 84
2.1.4. Хранилище метаданных 89
2.2. Логическая архитектура Pulsar 92
2.2.1. Абоненты, пространства имен и темы 92
2.2.2. Адресация тем в Pulsar 97
2.2.3. Производители, потребители и подписки 98
2.2.4. Типы подписки 99
2.3. Долговременное хранение сообщений и сроки их хранения 104
2.3.1. Долговременное хранение данных 104
2.3.2. Квоты для журналов регистрации 106
2.3.3. Окончание срока хранения сообщений 108
2.3.4. Сравнение журнала регистрации сообщений и определения срока хранения сообщений 109
2.4. Многоуровневое хранилище 110
2.5. Резюме 114

3 *Взаимодействие с Pulsar* 115

3.1. Начинаем работать с Pulsar 116
3.2. Администрирование Pulsar 117
3.2.1. Создание абонента, пространства имен и темы 118
3.2.2. API администрирования на языке Java 119
3.3. Клиенты Pulsar 120
3.3.1. Клиент Pulsar на языке Java 123
3.3.2. Клиент Pulsar на языке Python 134
3.3.3. Клиент Pulsar на языке Go 138
3.4. Дополнительное администрирование 144
3.4.1. Метрики постоянно хранимой темы 144
3.4.2. Инспекция сообщений 147
3.5. Резюме 148

ЧАСТЬ II. Основы разработки с использованием Apache Pulsar 151

4 *Функции Pulsar* 153

4.1. Потоковая обработка 153
4.1.1. Обычная пакетная обработка 154

4.1.2. Микропакетная обработка данных 155
4.1.3. Потоковая нативная обработка 155
4.2. Что такое Pulsar Functions 156
4.2.1. Модель программирования 158
4.3. Разработка функций Pulsar 159
4.3.1. Ориентированные на язык функции 159
4.3.2. Pulsar SDK 160
4.3.3. Функции с сохранением состояния 166
4.4. Тестирование функций Pulsar 170
4.4.1. Модульное тестирование 171
4.4.2. Комплексное тестирование 173
4.5. Развертывание функций Pulsar 178
4.5.1. Генерация артефакта развертывания 179
4.5.2. Конфигурация функции 182
4.5.3. Развертывание функции 186
4.5.4. Жизненный цикл развертывания функции 189
4.5.5. Режимы развертывания 190
4.5.6. Поток данных в функции Pulsar 192
4.6. Резюме 193

5 *Коннекторы ввода-вывода Pulsar* 194

5.1. Что такое коннекторы ввода-вывода Pulsar 195
5.1.1. Коннекторы-приемники 196
5.1.2. Коннекторы-источники 199
5.1.3. Коннекторы типа PushSource 201
5.2. Разработка коннекторов ввода-вывода Pulsar 203
5.2.1. Разработка коннектора-приемника 203
5.2.2. Разработка коннектора PushSource 205
5.3. Тестирование коннекторов ввода-вывода Pulsar 209
5.3.1. Модульное тестирование 209
5.3.2. Комплексное тестирование 211
5.3.3. Упаковка коннекторов ввода-вывода Pulsar 214
5.4. Развертывание коннекторов ввода-вывода Pulsar 215
5.4.1. Создание и удаление коннекторов 215
5.4.2. Отладка развернутых коннекторов 218
5.5. Встроенные коннекторы Pulsar 221
5.5.1. Запуск кластера MongoDB 221
5.5.2. Связывание контейнеров Pulsar и MongoDB 223
5.5.3. Конфигурирование и создание приемника для MongoDB 224
5.6. Администрирование коннекторов ввода-вывода Pulsar 226
5.6.1. Вывод списка коннекторов 226
5.6.2. Мониторинг коннекторов 228
5.7. Резюме 231

6 *Обеспечение безопасности Pulsar* *232*

6.1. Шифрование на транспортном уровне 233
6.2. Аутентификация 242
6.2.1. Аутентификация TLS 243
6.2.2. Аутентификация JSON Web Token (JWT) 249
6.3. Авторизация 255
6.3.1. Роли 256
6.3.2. Пример сценария 258
6.4. Шифрование сообщений 264
6.5. Резюме 269

7 *Реестр схем* *270*

7.1. Обмен информацией между микросервисами 271
7.1.1. API микросервисов 272
7.1.2. Обоснование необходимости реестра схем 274
7.2. Реестр схем Pulsar 275
7.2.1. Архитектура 275
7.2.2. Управление версиями схем 279
7.2.3. Совместимость схемы 279
7.2.4. Стратегии проверки совместимости схем 282
7.3. Использование реестра схем 287
7.3.1. Моделирование события заказа продуктов в Avro 289
7.3.2. События производства заказов продуктов 292
7.3.3. События потребления заказов продуктов 295
7.3.4. Полный пример 296
7.4. Развитие схемы 299
7.5. Резюме 302

ЧАСТЬ III. Практическая разработка приложений с использованием Apache Pulsar 303

8 *Паттерны применения Pulsar Functions* **305**

8.1. Конвейеры данных 306
8.1.1. Процедурное программирование 306
8.1.2. Программирование с использованием потока данных 307
8.2. Паттерны маршрутизации сообщений 310
8.2.1. Паттерн «разделитель» 310
8.2.2. Паттерн «динамический маршрутизатор» 315
8.2.3. Паттерн «маршрутизатор на основе содержимого» 319
8.3. Паттерны преобразования сообщений 322
8.3.1. Паттерн «транслятор сообщений» 322
8.3.2. Паттерн «улучшение содержимого» 325
8.3.3. Паттерн «фильтр содержимого» 327
8.4. Резюме 328

9 *Паттерны устойчивости* 329

9.1. Устойчивость Pulsar Functions 331
9.1.1. Неблагоприятные события 331
9.1.2. Обнаружение ошибок 337
9.2. Паттерны проектных решений по обеспечению устойчивости 339
9.2.1. Паттерн повтора 340
9.2.2. Паттерн «прерыватель замкнутого цикла» 344
9.2.3. Паттерн «ограничитель скорости» 350
9.2.4. Паттерн «ограничитель времени» 354
9.2.5. Паттерн «кеш» 358
9.2.6. Паттерн «откат» 360
9.2.7. Паттерн «обновление идентификационных данных» 362
9.3. Несколько уровней устойчивости 365
9.4. Резюме 367

10 *Доступ к данным* 368

10.1. Источники данных 369
10.2. Варианты использования доступа к данным 371
10.2.1. Проверка устройства 371
10.2.2. Данные о локации водителя 383
10.3. Резюме 391

11 *Машинное обучение в Pulsar* 392

11.1. Развертывание моделей машинного обучения 393
11.1.1. Режим пакетной обработки 393
11.1.2. Режим обработки почти в реальном времени 394
11.2. Развертывание модели в режиме почти реального времени 395
11.3. Векторы признаков 397
11.3.1. Хранилища признаков 398
11.3.2. Вычисление признаков 400
11.4. Оценка времени доставки 401
11.4.1. Экспорт модели машинного обучения 401
11.4.2. Отображение вектора признаков 404
11.4.3. Развертывание модели 407
11.5. Нейронные сети 409
11.5.1. Тренировка нейронной сети 410
11.5.2. Развертывание нейронной сети средствами языка Java 412
11.6. Резюме 414

12 *Периферийная аналитика 415*

12.1. Архитектура промышленного интернета вещей 419
12.1.1. Уровень восприятия и реакции 419
12.1.2. Уровень передачи данных 421
12.1.3. Уровень обработки данных 421
12.2. Уровень обработки данных на основе Pulsar 422
12.3. Периферийная аналитика 425
12.3.1. Телеметрические данные 426
12.3.2. Одномерные и многомерные наборы данных 428
12.4. Одномерный анализ 429
12.4.1. Снижение шума 430
12.4.2. Статистический анализ 432
12.4.3. Аппроксимация 437
12.5. Многомерный анализ 440
12.5.1. Создание двунаправленной сетки обмена сообщениями 441
12.5.2. Создание многомерного набора данных 445
12.6. Послесловие 451
12.7. Резюме 453

Приложение A. Запуск Pulsar в Kubernetes 454

A.1. Создание кластера Kubernetes 454
A.1.1. Предварительные условия для установки 456
A.1.2. Minikube 456
A.2. Pulsar Helm chart 458
A.2.1. Что такое Helm 459
A.2.2. Pulsar Helm chart 461
A.3. Использование Pulsar Helm chart 465
A.3.1. Администрирование Pulsar в среде Kubernetes 467
A.3.2. Конфигурирование клиентов 468

Приложение B. Георепликация 469

B.1. Синхронная георепликация 469
B.2. Асинхронная георепликация 472
B.2.1. Конфигурирование асинхронной георепликации 473
B.3. Паттерны асинхронной георепликации 477
B.3.1. Георепликация по схеме ведущий–ведущий 477
B.3.2. Георепликация по схеме активный–резервный 479
B.3.3. Агрегатная георепликация 480

Предметный указатель 482

Предисловие от издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте *www.dmkpress.com*, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу *dmkpress@gmail.com*; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу *http://dmkpress.com/authors/publish_book/* или напишите в издательство по адресу *dmkpress@gmail.com*.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в основном тексте или программном коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу *dmkpress@gmail.com*, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательство «ДМК Пресс» очень серьезно относится к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу *dmkpress@gmail.com*.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Предисловие

Книга «Apache Pulsar в действии» – это недостающее руководство, которое проведет вас через все этапы работы с Apache Pulsar. Эту книгу я порекомендовал бы прочесть всем: от разработчиков, начинающих изучать механизм обмена сообщениями по схеме публикация–подписка (pub-sub), до тех, у кого есть опыт обмена сообщениями, и до опытных пользователей Pulsar.

Работа над проектом Apache Pulsar началась в компании Yahoo! приблизительно с 2012 г. во время экспериментов с новой архитектурой, которая должна была решить эксплуатационные задачи на существующих платформах обмена сообщениями. К тому же это было время, когда некоторые значительные сдвиги в мире инфраструктуры данных становились все более заметными. Разработчики приложений начали обращать больше внимания на масштабируемый и надежный механизм обмена сообщениями как на основной компонент для создания следующего поколения программных продуктов. В то же время компании определили крупномасштабные системы анализа потоковых данных в реальном времени как важнейший компонент и преимущество для бизнеса.

Pulsar был спроектирован с нуля с целью установления прочной связи между этими двумя сферами – механизма обмена по схеме публикация–подписка (pub-sub) и анализа потоковых данных, которые весьма часто были изолированы друг от друга. Мы работали над созданием инфраструктуры, представляющей следующее поколение платформ обработки данных в реальном времени, где единственная система могла бы поддерживать все варианты использования на протяжении полного жизненного цикла событий, связанных с данными.

Со временем это представление расширялось, как можно ясно видеть по широкому диапазону компонентов, описанных в этой книге. В проект была добавлена поддержка упрощенной обработки с помощью Pulsar Functions, фреймворка коннекторов Pulsar IO, поддержка схем данных и многие другие функциональные возможности. Не изменилась только конечная цель – создание в высшей степени масштабируемой, гибкой и надежной платформы для обработки данных в реальном времени, позволяющей любому пользователю работать с данными, хранящимися в Pulsar, в наиболее удобной форме.

Я знаком с автором этой книги Дэвидом Хьеррумгором и работал вместе с ним в течение нескольких лет. За это время я обратил вни-

мание на его глубокую увлеченность работой в сообществе Pulsar. Он всегда готов помочь пользователям решить технические проблемы и затруднения, а также продемонстрировать им все возможности Pulsar для решения конкретной задачи обработки данных.

Особенно важным я считаю тот факт, что в книге органично сочетается теория и абстрактные концепции с ясностью практических пошаговых примеров и что эти примеры основаны на широко известных вариантах использования и паттернах проектирования обмена сообщениями, которые, несомненно, заинтересуют многих читателей. Здесь действительно каждый найдет что-то полезное для себя, и каждый сможет ознакомиться со всеми аспектами и возможностями, которые предлагает Pulsar.

— *Маттео Мерли (Matteo Merli),*
технический директор (CTO) Stream Native,
один из создателей и председатель комитета управления
проектом (PMC Chair) Apache Pulsar

Предисловие автора

В 2012 г. команда Yahoo! искала глобальную платформу с георепликацией, которая могла бы обеспечить потоковую обработку всех данных Yahoo! при обмене сообщениями между различными приложениями, например Yahoo Mail и Yahoo Finance. В то время существовали два основных типа систем обработки динамических данных: очереди сообщений, обрабатывающие особо важные бизнес-события в реальном времени, и потоковые системы, которые обрабатывали динамически масштабируемые конвейеры данных. Но при этом не было платформы, предоставляющей одновременно обе функциональные возможности, требуемые для компании Yahoo.

После тщательного исследования положения дел в области обмена сообщениями и потоковой обработки данных стало ясно, что существующие технологии не способны удовлетворить эти потребности, поэтому команда Yahoo! начала работу по созданию объединенной платформы обмена сообщениями и потоковой обработки динамических данных под названием Pulsar. После четырех лет работы в 10 центрах данных, обрабатывающих миллиарды сообщений в день, компания Yahoo! решила открыть исходный код своей платформы обмена сообщениями под защитой лицензии Apache в 2016 г.

Впервые я встретился с Pulsar осенью 2017 г. Тогда я возглавлял группу поддержки профессиональных сервисов в Hortonworks, сосредоточенной на работе с платформой потоковой обработки данных под названием Hortonworks Data Flow (HDF), в которую входили компоненты Apache NiFi, Kafka и Storm. В мои обязанности входил контроль за развертыванием этих технологий в инфраструктуре пользователя и помощь на начальной стадии разработки приложений потоковой обработки данных.

Самым большим затруднением, с которым мы столкнулись при работе с Kafka, была помощь нашим клиентам в правильном администрировании этого средства и специфическом определении необходимого количества разделов для конкретной темы, чтобы достичь надлежащего баланса скорости и эффективности при допущении дальнейшего роста объема данных. Тем из вас, кто знаком с Kafka, печально известен тот факт, что это кажущееся простым решение оказывает серьезное воздействие на масштабируемость тем, а для выполнения процедуры изменения этого значения (даже с 3 на 4) необходим весьма медленный процесс ребалансировки, на

протяжении которого все балансируемые темы становятся недоступными для чтения или записи.

Это требование ребалансировки неизменно вызывало негативную реакцию всех клиентов, использовавших HDF, и вполне справедливо, потому что они воспринимали его как препятствие для масштабирования кластера Kafka при непрерывном росте объемов данных. Только из собственного опыта клиенты узнавали, как трудно изменять в обе стороны масштаб платформы обмена сообщениями. Еще более худшим являлся тот факт, что невозможно было просто «прикрутить» еще несколько узлов для наращивания вычислительной мощности существующего клиентского кластера без соответствующего изменения конфигурации тем, чтобы использовать их для выделения большего количества разделов для существующих тем, чтобы перераспределить данные по только что добавленным узлам. Такая невозможность горизонтального масштабирования мощности потоковой обработки без ручного вмешательства (или необходимости написания огромного объема скриптов) приводила к прямому конфликту с желанием большинства наших клиентов переместить свои платформы обмена сообщениями в облако и воспользоваться всеми преимуществами гибких вычислительных возможностей, предоставляемых облачной средой.

Именно тогда я открыл для себя платформу Apache Pulsar и обнаружил, что ее заявление о том, что она «естественна для облака», особенно привлекательно, потому что она устраняет обе болевые точки масштабируемости. Хотя HDF позволял клиентам быстро приступить к работе, возникали трудности в управлении, да и сама платформа не была предназначена для функционирования в облаке. Я понял, что Apache Pulsar был намного лучшим решением, чем предлагаемое в тот момент нашим клиентам, и попытался убедить свою группу разработчиков рассмотреть возможность замены Kafka на Pulsar в нашем продукте HDF. Я даже зашел так далеко, что написал коннекторы, которые позволили Pulsar работать с компонентом Apache NiFi нашего стека, чтобы облегчить внедрение, но безрезультатно.

Когда в январе 2018 г. ко мне обратились первоначальные разработчики Apache Pulsar и предложили присоединиться к небольшому стартапу под названием Streamlio, я сразу же воспользовался возможностью поработать с ними. В то время Pulsar был пока еще молодым проектом, его только что включили в программу инкубации Apache, и следующие 15 месяцев мы провели, работая над тем, чтобы наш неоперившийся «птенчик» прошел через процесс инкубации и получил статус проекта высшего уровня.

Это было в самый разгар ажиотажа вокруг потоковой передачи данных, и Кафка был доминирующим игроком на этом поле, поэтому естественно, что все считали эти термины взаимозаменяемыми.

По общему мнению, Kafka был единственной доступной платформой для потоковой передачи данных. На основе своего предыдущего опыта я взял на себя смелость неустанно проповедовать то, что знал как технологически превосходное решение, – одинокий голос вопиющего в пресловутой пустыне.

К весне 2019 г. количество участников и пользователей в сообществе Apache Pulsar существенно увеличилось, но надежной документации по этой технологии катастрофически не хватало. Поэтому, когда мне впервые предложили написать «Apache Pulsar в действии», я сразу же воспринял это как возможность удовлетворить острую потребность для сообщества Pulsar. Хотя мне так и не удалось убедить своих коллег присоединиться ко мне в этом начинании, они были бесценным источником рекомендаций и информации на протяжении всего процесса и использовали эту книгу как средство передачи вам части своих знаний.

Эта книга предназначена для абсолютных новичков в Pulsar и является сочетанием информации, собранной мною во время непосредственного сотрудничества с основателями Pulsar в процессе активной разработки этой платформы, и опыта, накопленного во время работы с организациями, включившими Apache Pulsar в производственный процесс.

Книга предназначена для того, чтобы помочь вам преодолеть препятствия и ловушки, с которыми другие столкнулись во время своих исследований Pulsar. Прежде всего эта книга придаст вам уверенности при разработке приложений потоковой обработки и микросервисов с использованием Pulsar и языка программирования Java. Несмотря на то что я решил использовать Java для большинства примеров кода в книге из-за личного знакомства с этим языком, я также создал аналогичный комплект исходного кода с использованием Python и загрузил его в свою учетную запись GitHub для тех, кто предпочитает писать код на этом языке.

Благодарности

По своей природе мы не можем возвращать блага тем, от кого мы их получаем, или делаем это редко. Но благо, которое мы получаем, должно быть воздано кому-то снова, строка за строкой, дело за делом, цент за центом.

— Ральф Уолдо Эмерсон (Ralph Waldo Emerson)

Я хочу воспользоваться предоставленной возможностью, чтобы поблагодарить всех, кто так или иначе внес свой вклад в создание этой книги, и признать тот факт, что я никогда не смог бы взяться за такой грандиозный проект без тех, кто помог заложить для него основы. В духе Эмерсона, пожалуйста, считайте эту книгу моим способом отплатить за все знания и поддержку, которую вы предоставили мне.

Было бы большим упущением, если бы я не начал этот список с самого первого человека, который познакомил меня с удивительным миром программирования в очень раннем возрасте шести лет: директора начальной школы, мистера Роджерса, который решил посадить меня перед компьютером, а не под арест за невнимательность во время урока математики в первом классе. Вы познакомили меня с чистой творческой радостью программирования и направили меня на путь обучения на протяжении всей жизни.

Я также хочу поблагодарить группу разработчиков Yahoo!, которая создала Pulsar: вы написали потрясающую программу и открыли ее исходный код сообществу, чтобы все мы могли ею наслаждаться. Без вас невозможно было бы написать эту книгу.

Хочу поблагодарить всех своих бывших коллег по Streamlio, особенно Джерри Пенга (Jerry Peng), Айвена Келли (Ivan Kelly), Маттео Мерли (Matteo Merli), Сиджи Гуо (Sijie Guo) и Санжива Кулкарни (Sanjeev Kulkarni), за участие в качестве членов комиссии по управлению проектами Apache PMC – Apache Pulsar, Apache BookKeeper или в обоих проектах. Без вашего руководства и выполнения обязанностей Pulsar не стал бы тем, чем он является сегодня. Также хочу поблагодарить бывшего гендиректора (CEO) Картика Рамасами (Karthik Ramasamy) за помощь в расширении сообщества Apache Pulsar во время совместной работы в Streamlio: я действительно ценю ваше наставничество.

Благодарю всех моих бывших коллег из Splunk за усилия по интеграции Apache Pulsar в столь крупную организацию и за помощь

в продвижении его внедрения внутри организации. После предоставления новой технологии вы сразу взялись за дело и сделали все возможное, чтобы ваши усилия увенчались успехом. Особую благодарность хочу выразить группе создания коннекторов, особенно Аламуси (Alamusi), Гими (Gimi), Алексу (Alex) и Спайку (Spike).

Я также хотел бы поблагодарить рецензентов, которые нашли время в своей весьма насыщенной жизни, чтобы прочитать мою рукопись на разных этапах ее разработки. Ваши положительные отзывы были приятным подтверждением того, что я на правильном пути, и улучшали настроение, когда процесс написания становился слишком утомительным. Ваши отрицательные отзывы всегда были конструктивными и формировали новую точку зрения на материал, которую может создать только свежий взгляд. Эта обратная связь была чрезвычайно ценна для меня и в итоге привела к тому, что книга стала намного лучше, чем она была бы без вашего участия. Спасибо всем вам: Алессандро Кампеис (Alessandro Campeis), Александер Шварц (Alexander Schwartz), Андрес Сакко (Andres Sacco), Энди Кеффалас (Andy Keffalas), Анжело Симоне Скотто (Angelo Simone Scotto), Крис Винер (Chris Viner), Эмануэле Пиччинелли (Emanuele Piccinelli), Эрик Платон (Eric Platon), Джампьеро Гранателла (Giampiero Granatella), Джанлука Ригетто (Gianluca Righetto), Джилберто Таккари (Gilberto Taccari), Хенри Сапутра (Henry Saputra), Игорь Савин (Igor Savin), Джейсон Рендел (Jason Rendel), Джереми Чен (Jeremy Chen), Кабир Ахмед (Kabeer Ahmed), Кент Спиллнер (Kent Spillner), Ричард Тобиас (Richard Tobias), Санкет Наик (Sanket Naik), Сатей Кумар Сах (Satej Kumar Sah), Симоне Сгуацца (Simone Sguazza) и Торстен Вебер (Thorsten Weber).

Спасибо всем онлайн-рецензентам за то, что нашли время, чтобы предоставить мне ценные отзывы через онлайн-форум издательства Manning, особенно Крису Латимеру (Chris Latimer) и его сверхъестественному умению находить все орфографические и грамматические ошибки, которые не смог обнаружить Microsoft Word. Все будущие читатели в долгу перед вами.

Наконец, что не менее важно, я хочу поблагодарить редакторов издательства Manning, особенно Карен Миллер (Karen Miller), Ивана Мартиновича (Ivan Martinović), Адриану Сабо (Adriana Sabo), Алена Куньо (Alain Couniot) и Нинослава Черкеза (Ninoslav Čerkez). Спасибо за сотрудничество и за терпение, когда дела шли плохо. Это был длительный процесс, и я бы не справился без вашей поддержки. Ваш вклад в обеспечение качества этой книги улучшил ее для всех читателей. Также благодарю всех остальных сотрудников Manning, которые работали со мной над изданием и продвижением книги. Это была действительно командная работа.

Об этой книге

Книга «Apache Pulsar в действии» была написана как введение в технологию потоковой обработки данных, чтобы помочь читателям познакомиться с терминологией, семантикой и особенностями, которые необходимо знать для восприятия парадигмы потоковой обработки при переходе от технологии пакетной обработки данных. Она начинается с исторического обзора развития систем передачи сообщений за последние 40 лет и показывает, как Pulsar оказался на вершине этого эволюционного цикла.

После краткого описания основной терминологии в сфере передачи сообщений и обсуждения двух основных паттернов потребления сообщений рассматривается архитектура Pulsar с физической точки зрения, где основное внимание уделяется его проектному решению, наиболее подходящему для облачной среды, а также с точки зрения его логического структурирования данных и поддержки мультиарендности (множественной аренды, многоарендной архитектуры).

В остальной части книги все внимание сосредоточено на том, как можно использовать встроенную вычислительную платформу Pulsar под названием Pulsar Functions для разработки приложений с применением простого API. Этот подход демонстрируется реализацией варианта использования в обработке заказов: на воображаемом предприятии по доставке продуктов питания создается приложение с применением микросервисов исключительно на основе Pulsar Functions с дополнительным развертыванием модели машинного обучения для оценки времени доставки.

Для кого предназначена эта книга

Книга «Apache Pulsar в действии» предназначена главным образом для разработчиков на языке Java, интересующихся технологией обработки потоковых данных, или для разработчиков микросервисов, которые ищут альтернативный фреймворк для порождения (генерации) событий. Группы DevOps, заинтересованные в развертывании и функционировании Pulsar в своих организациях, также найдут эту книгу полезной. Одним из основных критических замечаний по Apache Pulsar является общее отсутствие документации и сообщений в блогах, доступных в интернете, и, хотя без сомнений ожидается, что это изменится в ближайшем будущем, я наде-

юсь, что книга поможет заполнить этот пробел в промежуточный период и принесет пользу всем, кто хочет узнать больше о потоковой обработке в целом и об Apache Pulsar в частности.

Как организована эта книга: дорожная карта

Книга состоит из 12 глав, разделенных на три части. Часть I начинается с введения в Apache Pulsar и определения его места в 40-летней истории развития систем передачи сообщений, а также сравнения Pulsar с разнообразными платформами обработки сообщений, существующими до него:

- в главе 1 представлена история систем передачи сообщений и определено место Pulsar в 40-летней истории развития технологии передачи сообщений. Также приведен краткий обзор некоторых архитектурных преимуществ Pulsar над другими системами и обоснование выбора Pulsar как единственной платформы обработки сообщений;
- в главе 2 подробно рассматривается многозвенная архитектура Pulsar, обеспечивающая независимое динамическое масштабирование хранилища данных или сервисных уровней. Здесь также описаны некоторые широко применяемые паттерны потребления сообщений, их отличия друг от друга и методы их поддержки в Pulsar;
- в главе 3 демонстрируется взаимодействие с Apache Pulsar из командной строки, а также с использованием его программного интерфейса (API). После изучения этой главы вы будете вполне уверенно запускать локальный экземпляр Apache Pulsar и взаимодействовать с ним.

В части II более глубоко рассматриваются основные варианты использования и функциональные возможности Pulsar, в том числе способы выполнения основных операций обработки сообщений и методы защиты кластера Pulsar, а также более продвинутые функциональные средства, например реестр схем. Кроме того, здесь представлен фреймворк Pulsar Functions с описанием того, как создавать, развертывать и тестировать функции:

- глава 4 представляет собственный потоковый вычислительный фреймворк Pulsar под названием Pulsar Functions с описанием некоторых основных принципов его проектирования и конфигурации, а также показывает, как разрабатывать, тестировать и развертывать функции;
- в главе 5 представлен фреймворк коннекторов Pulsar, предназначенный для перемещения (данных) между Apache Pulsar и внешними системами хранения, такими как реляционные базы данных, хранилища ключ–значение и хранилища blob-

объектов, например S3. Здесь вы узнаете, как разрабатывается коннектор – приводится пошаговое описание процесса разработки;

- глава 6 содержит подробные пошаговые инструкции по обеспечению безопасности кластера Pulsar для гарантии того, что ваши данные защищены как в режиме передачи, так и при хранении;
- в главе 7 рассматривается встроенный реестр схем Pulsar, обосновывается его необходимость, объясняется, как он может помочь в упрощении разработки микросервиса. Также описан процесс эволюции схемы и способ обновления схем, используемых внутри конкретного экземпляра Pulsar Functions.

В части III все внимание сосредоточено на использовании Pulsar Functions для реализации микросервисов и показано, как реализуются разнообразные паттерны проектирования широко применяемых микросервисов с помощью Pulsar Functions. В этой части демонстрируется процесс разработки приложения для предприятия по доставке пищи, чтобы сделать все примеры более реалистичными и применить более сложные варианты использования, включая обеспечение отказоустойчивости, доступ к данным и методику использования Pulsar Functions для развертывания моделей машинного обучения, которые могут работать с данными в реальном времени:

- в главе 8 показано, как реализовать часто применяемые паттерны маршрутизации сообщений, такие как разделение сообщений, маршрутизация на основе содержимого и фильтрация. Также демонстрируется реализация разнообразных паттернов преобразования сообщений, например извлечение значений и перевод сообщения;
- глава 9 подчеркивает важность механизма отказоустойчивости, встроенного в микросервисы, и демонстрирует его реализацию внутри экземпляра Pulsar Functions на основе Java с помощью библиотеки resiliency4j. Здесь рассматриваются различные события, которые могут произойти в программе, управляемой событиями, а также разнообразные паттерны, которые можно использовать для защиты сервисов от аварийных сценариев такого рода, чтобы максимизировать время работы приложения;
- в главе 10 рассматривается, как можно обеспечить доступ к данным из различных внешних систем при внутреннем использовании специально созданных функций Pulsar. Показаны разнообразные способы получения информации внутри микросервисов и условия, которые вы должны учитывать с точки зрения вероятных задержек;

- глава 11 представляет полный процесс развертывания разнообразных типов моделей машинного обучения внутри функции Pulsar с использованием различных фреймворков машинного обучения. Также рассматривается весьма важная тема: как передать необходимую информацию в модель, чтобы получить точный прогноз;
- в главе 12 описано использование Pulsar Functions в периферийной вычислительной среде для выполнения анализа в реальном времени данных интернета вещей (IoT). Глава начинается с подробного описания периферийной вычислительной среды и различных уровней ее архитектуры, и только после этого рассматривается, как применить Pulsar Functions для обработки информации на периферии, чтобы отправлять только краткие сводки, а не полный набор данных.

Завершают книгу два приложения, демонстрирующие более продвинутые рабочие сценарии, включающие развертывание в среде Kubernetes и георепликацию:

- в приложении A содержатся пошаговые инструкции, необходимые для развертывания Pulsar в среде Kubernetes с использованием схем Helm, предоставляемых как часть проекта с открытым исходным кодом. Также рассматривается изменение этих схем для соответствия среде, которую вы используете;
- в приложении B описан встроенный в Pulsar механизм георепликации и некоторые часто используемые паттерны репликации, применяемые в современном производстве. Далее подробно описан процесс реализации одного из таких паттернов георепликации в Pulsar.

Примеры исходного кода

Книга содержит множество примеров исходного кода как в форме пронумерованных листингов, так и в виде отдельной строки или нескольких строк в обычном тексте. В обоих случаях исходный код отформатирован с использованием `такого шрифта постоянной ширины` для отделения его от обычного текста. Иногда код также дополнительно выделяется **`полужирным шрифтом`**, чтобы специально выделить фрагмент кода, который изменился по сравнению с предыдущими этапами (примерами) в текущей главе, например при добавлении нового функционального средства (свойства) в существующую строку кода.

Во многих случаях изначальный исходный код был переформатирован: добавлены символы перехода на новую строку и изменено выравнивание для адаптации к доступному пространству на странице этой книги. В редких случаях, когда даже такие меры