

ВТОРОЕ ИЗДАНИЕ

Распределенные системы

Паттерны и парадигмы

*для масштабируемых и надежных систем
на основе Kubernetes*

Брендан Бёрнс

Краткое содержание

ЧАСТЬ I. БАЗОВЫЕ ПОНЯТИЯ

Глава 1. Введение	21
Глава 2. Важные концепции распределенных систем.....	30

ЧАСТЬ II. ОДНОУЗЛОВЫЕ ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ

Глава 3. Паттерн Sidecar.....	45
Глава 4. Паттерн Ambassador	58
Глава 5. Адаптеры.....	70

ЧАСТЬ III. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ ОБСЛУЖИВАЮЩИХ СИСТЕМ

Глава 6. Реплицированные сервисы с распределением нагрузки	86
Глава 7. Шардированные сервисы	103
Глава 8. Паттерн Scatter/Gather	121
Глава 9. Функции и событийно-ориентированная обработка	131
Глава 10. Выбор владельца.....	145

ЧАСТЬ IV. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ СИСТЕМ ПАКЕТНЫХ ВЫЧИСЛЕНИЙ

Глава 11. Системы на основе очередей задач	163
Глава 12. Событийно-ориентированная пакетная обработка	177
Глава 13. Координированная пакетная обработка	196

ЧАСТЬ V. УНИВЕРСАЛЬНЫЕ КОНЦЕПЦИИ

Глава 14. Паттерны мониторинга и наблюдаемости	209
Глава 15. Использование и обслуживание ИИ	225
Глава 16. Распространенные паттерны отказов	235

Оглавление

Предисловие	13
Кому стоит прочесть эту книгу	13
Зачем я написал эту книгу	13
Современный мир распределенных систем.....	14
Как ориентироваться в книге.....	14
Условные обозначения	16
Онлайн-ресурсы.....	16
Использование примеров кода	17
Благодарности	17
От издательства.....	18

ЧАСТЬ I БАЗОВЫЕ ПОНЯТИЯ

Глава 1. Введение	21
Краткая история разработки систем	22
Краткая история паттернов проектирования в разработке ПО	23
Формализация алгоритмического программирования	23
Паттерны в объектно-ориентированном программировании	24
Расцвет программного обеспечения с открытым исходным кодом.....	25
Ценность паттернов, практик и компонентов	25
Стоя на плечах гигантов.....	26
Общий язык обсуждения подходов к разработке.....	26
Общие повторно используемые компоненты	27
Резюме	29
Глава 2. Важные концепции распределенных систем	30
API и RPC.....	30
Задержка.....	31
Надежность	32
Процентили	33
Идемпотентность.....	34
Семантика доставки	35
Реляционная целостность.....	35

Согласованность данных	37
Оркестрация и Kubernetes	39
Проверки работоспособности	39
Резюме	40

ЧАСТЬ II

ОДНОУЗЛОВЫЕ ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ

Глава 3. Паттерн Sidcar	45
Пример реализации паттерна Sidcar. Добавление возможности	
HTTPS-соединения к унаследованному сервису	46
Динамическая конфигурация с помощью паттерна Sidcar	47
Модульные контейнеры приложений	49
Практикум. Развертывание контейнера topz	50
Создание простейшего PaaS-сервиса с помощью паттерна Sidcar	51
Разработка модульных и повторно используемых реализаций	
паттерна Sidcar	53
Параметризованные контейнеры	53
Определение API всех контейнеров	54
Документирование контейнеров	56
Резюме	57
Глава 4. Паттерн Ambassador	58
Использование паттерна Ambassador для шардирования сервиса	59
Практикум. Шардируем Redis-хранилище	61
Использование паттерна Ambassador для реализации	
сервиса-посредника	64
Использование паттерна Ambassador для проведения экспериментов	
и разделения запросов	66
Практикум. Реализация 10%-ных экспериментов	66
Резюме	69
Глава 5. Адаптеры	70
Мониторинг	71
Практикум. Мониторинг с помощью Prometheus	72
Журналирование	74
Практикум. Нормализация форматов журналов	
с помощью fluentd	75
Мониторинг работоспособности сервисов	77
Практикум. Комплексный мониторинг работоспособности MySQL	78
Резюме	80

ЧАСТЬ III

ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ ОБСЛУЖИВАЮЩИХ СИСТЕМ

Глава 6. Реплицированные сервисы с распределением нагрузки.....	86
Сервисы без внутреннего состояния	86
Датчики готовности для балансировщика нагрузки	88
Практикум. Создание реплицированного сервиса с помощью Kubernetes.....	89
Сервисы с закреплением сессий.....	90
Сервисы с репликацией на уровне приложения.....	92
Добавляем кэширующий слой.....	92
Развертывание кэширующего сервера	93
Практикум. Развертывание кэширующей прослойки	95
Расширение возможностей кэширующего слоя.....	97
Ограничение частоты запросов и защита от атак типа «отказ в обслуживании» (DoS).....	97
SSL-мост	99
Практикум. Развертывание nginx и SSL-моста.....	100
Резюме	102
Глава 7. Шардированные сервисы	103
Шардирование кэша	104
Зачем нужен шардированный кэш	105
Роль кэша в производительности системы	106
Реплицированный и шардированный кэш	108
Практикум. Развертывание реализации паттерна Ambassador и сервиса memcache для организации шардированного кэша	109
Шардирующие функции	113
Выбор ключа	115
Согласованные хеш-функции	116
Практикум. Построение согласованного шардированного прокси-сервера	117
Шардирование реплицированных сервисов.....	118
Системы с «горячим» шардированием.....	118
Резюме	120

Глава 8. Паттерн Scatter/Gather.....	121
Scatter/Gather с распределением нагрузки корневым узлом.....	122
Практикум. Распределенный поиск в документах	123
Scatter/Gather с шардированием листовых узлов.....	125
Практикум. Шардированный поиск в документах	126
Выбор подходящего количества листовых узлов.....	127
Масштабирование Scatter/Gather-систем с учетом надежности и производительности	129
Резюме	130
Глава 9. Функции и событийно-ориентированная обработка	131
Как определить, когда полезен подход FaaS	132
Преимущества FaaS.....	132
Проблемы разработки FaaS-систем.....	133
Потребность в фоновой обработке.....	134
Необходимость хранения данных в памяти	134
Стоимость постоянного использования запросно-ориентированных вычислений	135
Паттерны FaaS	136
Паттерн Decorator. Преобразование запроса или ответа	136
Практикум. Подстановка значений по умолчанию до обработки запроса	138
Обработка событий.....	139
Практикум. Реализация двухфакторной аутентификации.....	140
Событийные конвейеры	141
Практикум. Реализация конвейера для регистрации нового пользователя.....	142
Резюме	144
Глава 10. Выбор владельца.....	145
Как определить, нужен ли выбор владельца.....	147
Основы процесса выбора владельца.....	149
Практикум. Развертывание etcd.....	150
Реализация блокировок	151
Практикум. Реализация блокировок в etcd	154
Реализация владения	155
Практикум. Реализация аренды в etcd.....	156
Параллельный доступ к данным	157
Резюме	160

ЧАСТЬ IV

ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ СИСТЕМ ПАКЕТНЫХ ВЫЧИСЛЕНИЙ

Глава 11. Системы на основе очередей задач	163
Система на основе обобщенной очереди задач	163
Интерфейс контейнера-источника	165
API очереди задач	166
Интерфейс контейнера-исполнителя	167
Общая инфраструктура очередей задач	169
Практикум. Реализация генератора миниатюр видеофайлов	171
Динамическое масштабирование исполнителей	173
Паттерн Multi-Worker	175
Резюме	176
Глава 12. Событийно-ориентированная пакетная обработка	177
Паттерны событийно-ориентированной обработки	178
Паттерн Copier	179
Паттерн Filter	180
Паттерн Splitter	181
Паттерн Sharder	182
Паттерн Merger	184
Практикум. Создание событийно-ориентированного потока задач для регистрации нового пользователя	186
Инфраструктура publish/subscribe	188
Практикум. Развертывание Kafka	188
Устойчивость и производительность в очередях задач	191
Перехват заданий	191
Ошибки, приоритеты и повторы	192
Резюме	195
Глава 13. Координированная пакетная обработка	196
Паттерн Join (барьерная синхронизация)	197
Паттерн Reduce	199
Практикум. Подсчет	200
Суммирование	201
Гистограмма	202
Практикум. Конвейерная маркировка и обработка изображений	203
Резюме	206

ЧАСТЬ V

УНИВЕРСАЛЬНЫЕ КОНЦЕПЦИИ

Глава 14. Паттерны мониторинга и наблюдаемости	209
Основы мониторинга и наблюдаемости	209
Журналирование.....	211
Метрики.....	214
Базовый мониторинг запросов	216
Расширенный мониторинг запросов.....	218
Оповещение	219
Трассировка.....	221
Агрегирование информации	223
Резюме	224
Глава 15. Использование и обслуживание ИИ	225
Основы систем ИИ.....	225
Размещение модели	227
Распространение модели	228
Разработка с использованием моделей	230
Генерация ответа с дополнением результатами поиска	231
Тестирование и развертывание.....	232
Резюме	234
Глава 16. Распространенные паттерны отказов.....	235
Грохочущее стадо	235
Отсутствие ошибок — это ошибка	237
Клиентские и ожидаемые ошибки.....	238
Ошибки управления версиями	239
Миф о необязательных компонентах.....	240
Ой, мы все стерли	241
Проблемы с широтой входных данных	244
Обработка устаревших заданий.....	246
Проблема второй системы.....	248
Резюме	250
Заключение	251
Об авторе	253
Иллюстрация на обложке.....	254