

УДК 004.4
ББК 32.973.202-018.2
П12

Пай П., Абрахам П.

П12 Реактивное программирование на C++ / пер. с англ. В. Ю. Винника. – М.: ДМК Пресс, 2019. – 324 с.: ил.

ISBN 978-5-97060-778-7

В книге изложены понятия и принципы функционального реактивного программирования, помогающие строить параллельные, асинхронные приложения с наименьшими усилиями и минимумом ошибок. Реактивное программирование – парадигма программирования, ориентированная на потоки данных и распространение изменений, это путь для лёгкого создания пользовательских интерфейсов, анимации или моделирования систем, изменяющихся во времени.

Всесторонне рассмотрена библиотека RxCpp, описана разработка реактивных микросервисов на C++, а также использование библиотеки Qt/C++ в реактивном стиле. Изучив эту книгу, вы будете хорошо разбираться в тонкостях реактивной модели программирования и методах её реализации на новейшей версии стандарта C++17.

Издание предназначено для разработчиков C++, желающих получить максимум эффективности от своих приложений.

УДК 004.4
ББК 32.973.202-018.2

Authorized Russian translation of the English edition of C++ Reactive Programming ISBN 9781788629775 © 2018 Packt Publishing.

This translation is published and sold by permission of Packt Publishing, which owns or controls all rights to publish and sell the same.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN 978-1-78862-977-5 (англ.)
ISBN 978-5-97060-778-7 (рус.)

© 2018 Packt Publishing
© Оформление, издание, перевод, ДМК Пресс, 2019

Содержание

Над книгой работали	11
Предисловие	12

Глава 1. Модель реактивного программирования – обзор и история

Событийно-ориентированная модель программирования	18
Событийно-ориентированное программирование в системе X Window.....	19
Событийно-ориентированное программирование в среде Microsoft Windows	20
Событийно-ориентированное программирование в каркасе Qt	22
Событийно-ориентированное программирование средствами библиотеки MFC	23
Прочие модели событийно-управляемого программирования	24
Ограничения классических моделей обработки событий.....	24
Реактивная модель программирования	25
Ключевые интерфейсы реактивной программы.....	26
Методы вталкивания и втягивания данных	28
Дуальность интерфейсов IEnumerable и IObservable	28
Превращение событий в наблюдаемый источник	31
Методологические замечания	36
Итоги	37

Глава 2. Современный язык C++ и его ключевые идиомы

Принципы проектирования языка C++.....	40
Абстракция нулевой стоимости	40
Выразительность	40
Взаимозаменяемость	43
Усовершенствования языка, повышающие качество кода.....	44
Автоматический вывод типов	44
Единообразный синтаксис инициализации.....	46
Вариадические шаблоны	46
Ссылки rvalue	48
Семантика перемещения.....	50
Умные указатели	52
Лямбда-функции	54
Функциональные объекты и лямбда-функции	55

Композиция, карринг и частичное применение функций.....	57
Обёртки над функциями.....	60
Операция композиции функций.....	61
Прочие возможности языка.....	63
Выражения-свёртки	63
Сумма типов: тип variant	64
Прочее	65
Циклы по диапазонам и наблюдатели	65
Итоги	69

Глава 3. Параллельное и многопоточное

программирование на языке C++	70
Что такое параллельное программирование.....	71
Здравствуй, мир потоков!	72
Управление потоками	74
Запуск потока.....	74
Присоединение к потоку.....	75
Передача аргументов в поток	77
Использование лямбда-функций	79
Управление владением	80
Совместный доступ потоков к данным.....	82
Двоичные семафоры	84
Предотвращение тупиков	87
Условные переменные	91
Потокобезопасный стек	93
Итоги	96

Глава 4. Асинхронное программирование

и неблокирующая синхронизация в языке C++	98
Асинхронные задачи в языке C++	99
Фьючерсы и обещания	100
Класс std::packaged_task	102
Функция std::async	104
Модель памяти в языке C++	106
Параллельный доступ к памяти	106
Соглашение о порядке модификации памяти	107
Атомарные операции и типы в языке C++	108
Атомарные типы.....	108
Тип std::atomic_flag.....	111
Тип std::atomic<bool>	113
Тип std::atomic<T*> и арифметика указателей	116
Общий случай шаблона std::atomic<>	117
Порядок доступа к памяти	118
Последовательно согласованный порядок доступа	119

Результат : последовательная согласованность.....	120
Семантика захвата и освобождения	120
Ослабленный порядок доступа к памяти	122
Неблокирующая очередь.....	124
Итоги	126

Глава 5. Знакомство с наблюдаемыми источниками 127

Шаблон «Наблюдатель»	128
Ограниченность классического шаблона «Наблюдатель»	131
Обобщённый взгляд на шаблоны проектирования	133
Объектно-ориентированная модель программирования и иерархии	135
Обработка выражений с помощью шаблонов «Композит» и «Посетитель»	136
Разглаживание многоуровневых композитов для итеративного доступа	142
Операции отображения и фильтрации списков.....	146
От наблюдателей к наблюдаемым источникам.....	149
Итоги	153

Глава 6. Введение в программирование потоков

событий на языке C++ 155

Что такое программирование потоков данных.....	156
Преимущества модели программирования потоков данных	157
Прикладное программирование с использованием библиотеки Streams	157
Ленивые вычисления	158
Пример программы для обработки потока данных.....	159
Агрегирование значений в парадигме потоков данных	160
Погружение стандартных контейнеров в парадигму потоков данных	160
Несколько слов о библиотеке Streams	161
Программирование потоков событий	162
Преимущества программирования на основе потоков событий.....	162
Библиотека Streamulus и её программная модель	162
Библиотека Spreadsheet для оповещения об изменениях данных	168
Библиотека RaftLib – ещё один инструмент обработки потоков данных	170
Потоки данных и реактивное программирование	172
Итоги	173

Глава 7. Знакомство с моделью маршрутов данных

и библиотекой RxCpp 174

Парадигма маршрутов данных.....	175
Знакомство с библиотекой RxCpp	176
Библиотека RxCpp и её модель программирования	177
Простой пример взаимодействия источника с наблюдателем	178
Фильтрация и преобразование потоков данных.....	178

Создание потока из контейнера.....	179
Создание собственных наблюдаемых источников	179
Конкатенация потоков	180
Отписка от потока данных.....	180
Визуальное представление потоков данных	181
Операции над потоками данных.....	181
Операция average.....	182
Операция scan.....	182
Соединение операций в конвейер	183
Работа с планировщиками.....	183
Сага о двух операциях: как разглаживать потоки потоков	186
Прочие важные операции	191
Беглый взгляд на ещё не изученное.....	192
Итоги	193

Глава 8. Ключевые элементы библиотеки RxCpp

Наблюдаемые источники данных	194
Что такое объект-производитель	195
Горячие и холодные источники данных	195
Горячие источники данных	196
Горячие источники данных и механизм повтора	198
Наблюдатели и подписчики.....	199
Единство наблюдаемого и наблюдателя.....	200
Планировщики.....	203
Методы observe_on и subscribe_on.....	206
Планировщик с циклом выполнения run_loop.....	208
Операции над потоками данных.....	209
Операции создания потоков.....	210
Операции преобразования данных	210
Операции фильтрации	211
Операции комбинирования данных.....	212
Операции обработки ошибок	212
Вспомогательные операции	212
Логические операции.....	213
Математические операции и агрегирование потоков.....	213
Операции для управления подключениями	213
Итоги	214

Глава 9. Реактивное программирование графических интерфейсов на основе каркаса Qt

Введение в программирование интерфейсов пользователя на основе каркаса Qt.....	216
--	-----

Объектная модель библиотеки Qt	217
Сигналы и слоты	218
Подсистема событий	220
Обработчики событий	221
Отправка событий	221
Система метаобъектов	222
Программа «Здравствуй, мир» на основе библиотеки Qt	222
События, сигналы и слоты на примере	225
Создание собственного визуального объекта	225
Создание главного диалогового окна приложения	227
Запуск приложения	231
Интеграция библиотек RxCpp и Qt	232
Реактивная фильтрация событий из каркаса Qt	233
Создание окна и размещение его элементов	235
Наблюдатели для различных типов событий	236
Знакомство с библиотекой RxQt	238
Итоги	241

Глава 10. Шаблоны и идиомы реактивного программирования на языке C++

Объектно-ориентированное программирование и шаблоны проектирования	242
Основные каталоги шаблонов	244
Шаблоны «Банды четырёх»	244
Каталог POSA	245
Ещё раз о шаблонах проектирования	246
От шаблонов проектирования к реактивному программированию	248
Разглаживание иерархии и линейный проход	254
От итераторов к наблюдаемым источникам	256
Шаблон «Ячейка»	257
Шаблон «Активный объект»	260
Шаблон «Ресурс взаимны»	262
Шаблон «Шина событий»	263
Итоги	267

Глава 11. Реактивные микросервисы на языке C++

Язык C++ и веб-программирование	269
Модель программирования REST	269
Библиотека REST SDK для языка C++	270
Программирование HTTP-клиента с использованием библиотеки C++ REST SDK	270
Программирование HTTP-сервера	272
Тестирование HTTP-сервера с помощью утилит curl и postman	275

Создание HTTP-клиента с помощью библиотеки libcurl.....	276
Реактивная библиотека-обёртка RxCurl	277
Использование формата JSON с протоколом HTTP	278
Использование библиотеки C++ REST SDK для создания сервера	282
Обращение к REST-сервисам с помощью библиотеки RxCurl.....	290
Несколько слов об архитектуре реактивных микросервисов.....	292
Мелкоблочные сервисы.....	293
Разнородное хранение данных	294
Независимое развёртывание сервисов	294
Оркестровка и хореография сервисов.....	295
Реактивный стиль запросов к веб-сервисам	295
Итоги	295

Глава 12. Особые возможности потоков и обработка

ошибок.....	297
Средства обработки ошибок в библиотеке RxCpp.....	300
Выполнение действия в ответ на ошибку	300
Восстановление после ошибки	302
Обработка ошибки путём перезапуска источника данных.....	305
Автоматическое выполнение завершающих действий в случае ошибки	307
Обработка ошибок и планировщики	308
Примеры обработки потоков событий	313
Агрегирование потоков данных	313
Событийно-управляемое приложение	315
Итоги	319