

**Дэвис, А.** Асинхронное программирование в C# 5.0 / Алекс Дэвис ; пер. с англ. А. А. Слинкин. — Москва : ДМК Пресс, 2015. — 119 с. — (O'REILLY).

УДК 004.438С#:004.42

ББК 32

Ч/З №1 — 1 экз.

Из этого краткого руководства вы узнаете, как механизм `async` в C# 5.0 позволяет упростить написание асинхронного кода. Помимо ясного введения в асинхронное программирование вообще, вы найдете углубленное описание работы этого конкретного механизма и ответ на вопрос, когда и зачем использовать его в собственных приложениях.

В книге рассматриваются следующие вопросы:

- Как писать асинхронный код вручную и как механизм `async` скрывает неприглядные детали.
- Новые способы повышения производительности серверного кода в приложениях ASP.NET.
- Совместная работа `async` и WinRT в приложениях для Windows 8.
- Смысл ключевого слова `await` в `async`-методах.
- В каком потоке .NET выполняется асинхронный код в каждой точке программы.
- Написание асинхронных API, согласованных с паттерном Task-based Asynchronous Pattern (TAP).
- Распараллеливание программ для задействования возможностей современных компьютеров.
- Измерение производительности `async`-кода и сравнение с альтернативными подходами.

Книга рассчитана на опытных программистов на C#, но будет понятна и начинающим. Она изобилует примерами кода, который можно использовать в своих программах.

---

# Асинхронное программирование в C# 5.0

*Алекс Дэвис*



Москва, 2015

# ОГЛАВЛЕНИЕ

<b>Предисловие .....</b>	<b>9</b>
Предполагаемая аудитория.....	9
Как читать эту книгу .....	10
Принятые соглашения .....	10
О примерах кода.....	11
Как с нами связаться .....	11
Благодарности .....	12
Об авторе .....	12
<b>Глава 1. Введение.....</b>	<b>13</b>
Асинхронное программирование .....	13
Чем так хорош асинхронный код? .....	14
Что такое async? .....	15
Что делает async? .....	15
Async не решает все проблемы.....	17
<b>Глава 2. Зачем делать программу асинхронной ...</b>	<b>18</b>
Приложения с графическим интерфейсом пользователя для настольных компьютеров .....	18
Аналогия с кафе .....	19
Серверный код веб-приложения .....	20
Еще одна аналогия: кухня в ресторане.....	21
Silverlight, Windows Phone и Windows 8 .....	22
Параллельный код .....	23
Пример.....	24
<b>Глава 3. Написание асинхронного кода вручную ...</b>	<b>26</b>
О некоторых асинхронных паттернах в .NET .....	26
Простейший асинхронный паттерн .....	28
Введение в класс Task .....	29
Чем плоха реализация асинхронности вручную? .....	30
Переработка примера с использованием написанного вручную асинхронного кода.....	31

## Глава 4. Написание асинхронных методов ..... 33

Преобразование программы скачивания значков к виду, использующему <code>async</code> .....	33
<code>Task</code> и <code>await</code> .....	34
Тип значения, возвращаемого асинхронным методом .....	36
<code>Async</code> , сигнатуры методов и интерфейсы .....	37
Предложение <code>return</code> в асинхронных методах .....	38
Асинхронные методы «заразны» .....	39
Асинхронные анонимные делегаты и лямбда-выражения .....	40

## Глава 5. Что в действительности делает `await` ... 41

Приостановка и возобновление метода .....	41
Состояние метода .....	42
Контекст .....	43
Когда нельзя использовать <code>await</code> .....	44
Блоки <code>catch</code> и <code>finally</code> .....	44
Блоки <code>lock</code> .....	45
Выражения LINQ-запросов .....	46
Небезопасный код .....	47
Запоминание исключений .....	47
Асинхронные методы до поры исполняются синхронно .....	48

## Глава 6. Паттерн TAP ..... 50

Что специфицировано в TAP? .....	50
Использование <code>Task</code> для операций, требующих большого объема вычислений .....	52
Создание задачи-марионетки .....	53
Взаимодействие с прежними асинхронными паттернами .....	55
Холодные и горячие задачи .....	56
Предварительная работа .....	56

## Глава 7. Вспомогательные средства для асинхронного кода ..... 58

Задержка на указанное время .....	58
Ожидание завершения нескольких задач .....	59
Ожидание завершения любой задачи из нескольких .....	61
Создание собственных комбинаторов .....	62
Отмена асинхронных операций .....	64
Информирование о ходе выполнения асинхронной операции .....	65

## Глава 8. В каком потоке выполняется мой код? ... 67

До первого <code>await</code> .....	67
-------------------------------------	----

Во время асинхронной операции.....	68
Подробнее о классе SynchronizationContext .....	69
await и SynchronizationContext.....	69
Жизненный цикл асинхронной операции .....	70
Когда не следует использовать SynchronizationContext .....	73
Взаимодействие с синхронным кодом.....	74

## **Глава 9. Исключения в асинхронном коде ..... 76**

Исключения в аsync-методах, возвращающих Task.....	76
Незамеченные исключения .....	78
Исключения в методах типа аsync void.....	79
Выстрелил и забыл.....	79
AggregateException и WhenAll.....	80
Синхронное возбуждение исключений .....	81
Блок finally в аsync-методах .....	82

## **Глава 10. Организация параллелизма с помощью механизма аsync ..... 83**

await и блокировки.....	83
Акторы.....	85
Использование акторов в C# .....	86
Библиотека Task Parallel Library Dataflow .....	87

## **Глава 11. Автономное тестирование асинхронного кода..... 90**

Проблема автономного тестирования в асинхронном окружении .....	90
Написание работающих асинхронных тестов вручную .....	91
Поддержка со стороны каркаса автономного тестирования ....	91

## **Глава 12. Механизм аsync в приложениях ASP.NET ..... 93**

Преимущества асинхронного веб-серверного кода.....	93
Использование аsync в ASP.NET MVC 4 .....	94
Использование аsync в предыдущих версиях ASP.NET MVC .....	94
Использование аsync в ASP.NET Web Forms .....	95

## **Глава 13. Механизм аsync в приложениях WinRT ... 97**

Что такое WinRT? .....	97
Интерфейсы IAsyncAction и IAsyncOperation<T> .....	98
Отмена .....	99

Информирование о ходе выполнения .....	99
Реализация асинхронных методов в компоненте WinRT .....	100

## **Глава 14. Подробно о преобразовании асинхронного кода, осуществляемом компилятором ..... 102**

Метод-заглушка .....	102
Структура конечного автомата.....	103
Метод MoveNext .....	105
Наш код.....	106
Преобразование предложений return в код завершения.....	106
Переход в нужное место метода.....	106
Приостановка метода в месте встречи await.....	107
Возобновление после await .....	107
Синхронное завершение.....	107
Перехват исключений.....	108
Более сложный код .....	108
Разработка типов, допускающих ожидание .....	109
Взаимодействие с отладчиком .....	110

## **Глава 15. Производительность асинхронного кода ..... 112**

Измерение накладных расходов механизма async.....	112
Async и блокирующая длительная операция .....	113
Оптимизация асинхронного кода для длительной операции ....	116
Async-методы и написанный вручную асинхронный код.....	116
Async и блокирование без длительной операции .....	117
Оптимизация асинхронного кода без длительной операции ....	118
Резюме .....	119