

Содержание

Предисловие	22
Введение	24
От издателя перевода	26

ЧАСТЬ I. ОСНОВЫ CLR

Глава 1. Модель выполнения кода в среде CLR	28
Компиляция исходного кода в управляемые модули	28
Объединение управляемых модулей в сборку	32
Загрузка CLR	34
Исполнение кода сборки	37
IL-код и верификация	44
Небезопасный код	45
IL и защита интеллектуальной собственности	46
NGen.exe	47
Библиотека FCL	47
CTS	49
CLS	52
Взаимодействие с неуправляемым кодом	57
Глава 2. Компоновка, упаковка, развертывание и администрирование приложений и типов	58
Задачи развертывания в .NET Framework	58
Компоновка типов в модуль	60
Файл параметров	61
Несколько слов о метаданных	64

Объединение модулей для создания сборки.	71
Добавление сборок в проект в среде Visual Studio	78
Использование утилиты Assembly Linker	79
Включение в сборку файлов ресурсов.	81
Ресурсы со сведениями о версии сборки	82
Номера версии.	86
Региональные стандарты	87
Развертывание простых приложений (закрытое развертывание сборок)	88
Простое средство администрирования (конфигурационный файл)	90
Алгоритм поиска файлов сборки.	92

Глава 3. Совместно используемые сборки и сборки со строгим именем.	94
Два вида сборок — два вида развертывания.	95
Назначение сборке строгого имени.	96
Глобальный кэш сборки	102
Построение сборки, ссылающейся на сборку со строгим именем.	104
Устойчивость сборок со строгими именами к несанкционированной модификации	106
Отложенное подписание.	107
Закрытое развертывание сборок со строгими именами	110
Как исполняющая среда разрешает ссылки на типы	111
Дополнительные административные средства (конфигурационные файлы)	115
Управление версиями при помощи политики издателя.	117

ЧАСТЬ II. ПРОЕКТИРОВАНИЕ ТИПОВ

Глава 4. Основы типов.	122
Все типы — производные от System.Object	122
Приведение типов	124
Приведение типов в C# с помощью операторов is и as	126
Пространства имен и сборки	128
Связь между сборками и пространством имен	132
Как разные компоненты взаимодействуют во время выполнения	133

Глава 5. Примитивные, ссылочные и значимые типы	142
Примитивные типы в языках программирования	142
Проверяемые и непроверяемые операции для примитивных типов	146
Ссылочные и значимые типы	150
Как CLR управляет размещением полей для типа	155
Упаковка и распаковка значимых типов	156
Изменение полей в упакованных значимых типах посредством интерфейсов (и почему этого лучше не делать)	169
Равенство и тождество объектов	172
Хеш-коды объектов	175
Примитивный тип данных <code>dynamic</code>	177
 Глава 6. Основные сведения о членах и типах	186
Члены типа	186
Видимость типа	189
Дружественные сборки	189
Доступ к членам типов	191
Статические классы	193
Частичные классы, структуры и интерфейсы	194
Компоненты, полиморфизм и версии	196
Вызов виртуальных методов, свойств и событий в CLR	198
Разумное использование видимости типов и модификаторов доступа к членам	202
Работа с виртуальными методами при управлении версиями типов	205
 Глава 7. Константы и поля	210
Константы	210
Поля	212
 Глава 8. Методы	215
Конструкторы экземпляров и классы (ссылочные типы)	215
Конструкторы экземпляров и структуры (значимые типы)	219
Конструкторы типов	222
Методы перегруженных операторов	226
Операторы и взаимодействие языков программирования	229

Особое мнение автора о правилах Microsoft, связанных с именами методов операторов	229
Методы операторов преобразования	230
Методы расширения	234
Правила и рекомендации	237
Расширение разных типов методами расширения	238
Атрибут расширения	240
Частичные методы	241
Правила и рекомендации	244
Глава 9. Параметры	245
Необязательные и именованные параметры	245
Правила использования параметров	246
Атрибут <code>DefaultParameterValue</code> и необязательные атрибуты	248
Неявно типизированные локальные переменные	248
Передача параметров в метод по ссылке	251
Передача переменного количества аргументов	257
Типы параметров и возвращаемых значений	259
Константность	261
Глава 10. Свойства	263
Свойства без параметров	263
Автоматически реализуемые свойства	267
Осторожный подход к определению свойств	268
Свойства и отладчик Visual Studio	270
Инициализаторы объектов и коллекций	271
Анонимные типы	273
Тип <code>System.Tuple</code>	276
Свойства с параметрами	279
Выбор главного свойства с параметрами	283
Производительность при вызове методов доступа	284
Доступность методов доступа свойств	285
Обобщенные методы доступа свойств	285
Глава 11. События	286
Разработка типа, поддерживающего событие	287

Этап 1. Определение типа для хранения всей дополнительной информации, передаваемой получателем уведомления о событии	288
Этап 2. Определение члена-события.	289
Этап 3. Определение метода, ответственного за уведомление зарегистрированных объектов о событии.. . . .	290
Этап 4. Определение метода, преобразующего входную информацию в желаемое событие	292
Реализация событий компилятором	293
Создание типа, отслеживающего событие	295
Явное управление регистрацией событий	298
Глава 12. Обобщения	302
Обобщения в библиотеке FCL	307
Инфраструктура обобщений	308
Открытые и закрытые типы.	309
Обобщенные типы и наследование.	311
Идентификация обобщенных типов	313
Разрастание кода	314
Обобщенные интерфейсы	315
Обобщенные делегаты.	316
Контравариантные и ковариантные аргументы-типы в делегатах и интерфейсах	317
Обобщенные методы.	319
Обобщенные методы и выводение типов	320
Обобщения и другие члены	322
Верификация и ограничения	322
Основные ограничения	325
Дополнительные ограничения	327
Ограничения конструктора	328
Другие проблемы верификации	329
Глава 13. Интерфейсы	333
Наследование в классах и интерфейсах	333
Определение интерфейсов	334
Наследование интерфейсов	335

Подробнее о вызовах интерфейсных методов 338

Явные и неявные реализации интерфейсных методов
(что происходит за кулисами) 339

Обобщенные интерфейсы 341

Обобщения и ограничения интерфейса 344

Реализация нескольких интерфейсов с одинаковыми сигнатурами
и именами методов 345

Совершенствование безопасности типов за счет явной
реализации интерфейсных методов 346

Опасности явной реализации интерфейсных методов 348

Дилемма разработчика: базовый класс или интерфейс? 351

ЧАСТЬ III. ОСНОВНЫЕ ТИПЫ ДАННЫХ

Глава 14. Символы, строки и обработка текста 356

Символы 356

Тип System.String 359

Создание строк 359

Неизменяемые строки 362

Сравнение строк. 362

Интернирование строк 369

Создание пулов строк 372

Работа с символами и текстовыми элементами в строке. 372

Прочие операции со строками 375

Эффективное создание строк 375

Создание объекта StringBuilder 376

Члены типа StringBuilder 377

Получение строкового представления объекта 379

Форматы и региональные стандарты 380

Форматирование нескольких объектов в одну строку 384

Создание собственного средства форматирования 386

Получение объекта посредством разбора строки. 389

Кодировки: преобразования между символами и байтами. 391

Кодирование и декодирование потоков символов и байтов 397

Кодирование и декодирование строк в кодировке Base-64 398

Защищенные строки 399

Глава 15. Перечислимые типы и битовые флаги	403
Перечислимые типы	403
Битовые флаги	409
Добавление методов к перечислимым типам	413
Глава 16. Массивы	416
Инициализация элементов массива	418
Приведение типов в массивах	421
Базовый класс System.Array	423
Реализация интерфейсов IEnumerable, ICollection и IList	424
Передача и возврат массивов	425
Массивы с ненулевой нижней границей	426
Внутренняя реализация массивов	427
Небезопасный доступ к массивам и массивы фиксированного размера	432
Глава 17. Делегаты	434
Знакомство с делегатами	434
Обратный вызов статических методов	437
Обратный вызов экземплярных методов	438
Тонкости использования делегатов	439
Обратный вызов нескольких методов (цепочки делегатов)	443
Поддержка цепочек делегатов в C#	448
Дополнительные средства управления цепочками делегатов	448
Обобщенные делегаты	451
Упрощенный синтаксис работы с делегатами	452
Упрощение 1: не создаем объект делегата	452
Упрощение 2: не определяем метод обратного вызова	453
Упрощение 3: не создаем обертку для локальных переменных для передачи их методу обратного вызова	457
Делегаты и отражение	460
Глава 18. Настраиваемые атрибуты	464
Сфера применения настраиваемых атрибутов	464
Определение класса атрибутов	468

Конструктор атрибута и типы данных полей и свойств	471
Выявление настраиваемых атрибутов	473
Сравнение экземпляров атрибута	477
Выявление настраиваемых атрибутов без создания объектов, производных от Attribute	480
Условные атрибуты.	484
Глава 19. Null-совместимые значимые типы	485
Поддержка в C# null-совместимых значимых типов	487
Оператор объединения null-совместимых значений	490
Поддержка в CLR null-совместимых значимых типов	491
Упаковка null-совместимых значимых типов.	491
Распаковка null-совместимых значимых типов	492
Вызов метода GetType через null-совместимый значимый тип.	492
Вызов интерфейсных методов через null-совместимый значимый тип	493

ЧАСТЬ IV. КЛЮЧЕВЫЕ МЕХАНИЗМЫ

Глава 20. Исключения и управление состоянием	496
Определение «исключения»	496
Механика обработки исключений	498
Блок try	499
Блок catch	499
Блок finally	501
CLS-совместимые и CLS-несовместимые исключения	503
Класс System.Exception	505
Классы исключений, определенные в FCL	509
Генерирование исключений	511
Создание классов исключений	513
Продуктивность вместо надежности	515
Приемы работы с исключениями	524
Активно используйте блоки finally	525
Не надо перехватывать все исключения	526
Корректное восстановление после исключения.	528

Отмена незавершенных операций при невозможности восстановления исключений..	529
Скрытие деталей реализации для сохранения контракта..	530
Необработанные исключения..	533
Отладка исключений	537
Скорость обработки исключений..	540
Области ограниченного выполнения..	543
Контракты кода	546

Глава 21. Автоматическое управление памятью

(уборка мусора)	554
Управляемая куча	554
Выделение ресурсов из управляемой кучи	555
Алгоритм уборки мусора	557
Уборка мусора и отладка	560
Поколения	562
Запуск уборки мусора	568
Большие объекты	569
Режимы уборки мусора	570
Программное управление уборщиком мусора	573
Мониторинг использования памяти приложением	574
Освобождение ресурсов при помощи механизма финализации.. . . .	576
Типы, использующие системные ресурсы	583
Интересные аспекты зависимостей	588
Другие возможности уборщика мусора для работы с системными ресурсами	590
Внутренняя реализация финализации	594
Мониторинг и контроль времени жизни объектов..	597

Глава 22. Хостинг CLR и домены приложений 606

Хостинг CLR.	606
Домены приложений	609
Доступ к объектам из других доменов..	612
Выгрузка доменов	624
Мониторинг доменов	626

- Уведомление о первом управляемом исключении домена 627
- Использование хостами доменов приложений. 628
 - Исполняемые приложения 628
 - Полнофункциональные интернет-приложения Silverlight 629
 - Microsoft ASP.NET и веб-службы XML 629
 - Microsoft SQL Server 630
 - Будущее и мечты 630
- Нетривиальное управление хостингом 631
 - Применение управляемого кода. 631
 - Разработка надежных хост-приложений 631
 - Возвращение потока в хост 633
- Глава 23. Загрузка сборок и отражение 636**
 - Загрузка сборок 637
 - Использование отражения для создания динамически расширяемых приложений 641
 - Производительность отражения 642
 - Нахождение типов, определенных в сборке 644
 - Объект Type 644
 - Создание иерархии типов, производных от Exception 646
 - Создание экземпляра типа. 648
 - Создание приложений с поддержкой подключаемых компонентов . . 650
 - Нахождение членов типа путем отражения. 653
 - Нахождение членов типа 654
 - Обращение к членам типов. 658
 - Использование дескрипторов привязки для снижения потребления памяти процессом 663
- Глава 24. Сериализация 666**
 - Практический пример сериализации/десериализации. 667
 - Сериализуемые типы 672
 - Управление сериализацией и десериализацией. 673
 - Сериализация экземпляров типа. 677
 - Управление сериализованными и десериализованными данными. . . 679
 - Определение типа, реализующего интерфейс ISerializable, не реализуемый базовым классом 684

Контексты потока ввода-вывода	686
Сериализация в другой тип и десериализация в другой объект	688
Суррогаты сериализации	691
Цепочка селекторов суррогатов	694
Переопределение сборки и/или типа при десериализации объекта ..	695

Глава 25. Взаимодействие с компонентами WinRT..... 698

Проекция уровня CLR и правила системы типов компонентов WinRT.	700
Основные концепции системы типов WinRT.....	700
Проекция уровня .NET Framework	705
Асинхронные вызовы WinRT API из кода .NET	705
Взаимодействия между потоками WinRT и потоками .NET	710
Передача блоков данных между CLR и WinRT	712
Определение компонентов WinRT в коде C#.....	715

ЧАСТЬ V. МНОГОПОТОЧНОСТЬ

Глава 26. Потоки исполнения..... 724

Для чего Windows поддерживает потоки?	724
Ресурсоемкость потоков.	725
Так дальше не пойдет!	729
Тенденции развития процессоров	732
CLR- и Windows-потоки	733
Потоки для асинхронных вычислительных операций	734
Причины использования потоков.....	736
Планирование и приоритеты потоков.	739
Фоновые и активные потоки	744
Что дальше?	746

Глава 27. Асинхронные вычислительные операции..... 747

Пул потоков в CLR	747
Простые вычислительные операции	748
Контексты исполнения	750
Скоординированная отмена	752

- Задания 757
 - Завершение задания и получение результата 758
 - Отмена задания 760
 - Автоматический запуск задания по завершении предыдущего 762
 - Дочерние задания 764
 - Структура задания. 765
 - Фабрики заданий 767
 - Планировщики заданий 769
- Методы For, ForEach и Invoke класса Parallel 771
- Встроенный язык параллельных запросов. 775
- Периодические вычислительные операции 779
 - Разновидности таймеров. 782
- Как пул управляет потоками 783
 - Ограничение количества потоков в пуле 783
 - Управление рабочими потоками 784

- Глава 28. Асинхронные операции ввода-вывода 787**
 - Операции ввода-вывода в Windows 787
 - Асинхронные функции C# 792
 - Преобразование асинхронной функции в конечный автомат. 795
 - Расширяемость асинхронных функций 799
 - Асинхронные функции и обработчики событий 803
 - Асинхронные функции в FCL. 804
 - Асинхронные функции и исключения. 806
 - Другие возможности асинхронных функций 807
 - Потоковые модели приложений. 810
 - Асинхронная реализация сервера 813
 - Отмена операций ввода-вывода 814
 - Некоторые операции ввода-вывода
должны выполняться синхронно 815
 - Проблемы FileStream 816
 - Приоритеты запросов ввода-вывода. 817

Глава 29. Прimitivesкие конструкции синхронизации потоков	820
Библиотеки классов и безопасность потоков	822
Прimitivesкие конструкции пользовательского режима и режима ядра	824
Конструкции пользовательского режима	825
Volatile-конструкции	826
Interlocked-конструкции	832
Реализация простой циклической блокировки	837
Универсальный Interlocked-паттерн	841
Конструкции режима ядра	843
События	847
Семафоры	850
Мьютексы	851
Глава 30. Гибридные конструкции синхронизации потоков ..	854
Простая гибридная блокировка	854
Зацикливание, владение потоком и рекурсия	857
Гибридные конструкции в FCL	859
Классы ManualResetEventSlim и SemaphoreSlim	859
Класс Monitor и блоки синхронизации	860
Класс ReaderWriterLockSlim	866
Класс OneManyLock	868
Класс CountdownEvent	871
Класс Barrier	872
Выводы по гибридным конструкциям	873
Блокировка с двойной проверкой	875
Паттерн условной переменной	880
Асинхронная синхронизация	882
Классы коллекций для параллельного доступа	888
Словарь соответствия русскоязычных и англоязычных терминов	893