

Оглавление

Предисловие	15
Введение	17
Благодарности	19
Об этой книге	21
Для кого предназначена эта книга	21
Наш маршрут	22
Условные обозначения и возможности для скачивания	23
Об авторе	25
Иллюстрация на обложке	26
От издательства	26

Часть 1. Последовательность изучения внедрения зависимостей

Глава 1. Дегустационное меню внедрения зависимостей.	30
1.1. Написание удобного в сопровождении кода	32
1.1.1. Отучиться от внедрения зависимостей	32
1.1.2. Назначение внедрения зависимостей	35
1.2. Hello DI (Привет, B3)	41
1.2.1. Код для Hello DI	42
1.2.2. Достоинства внедрения зависимостей	44
1.3. Что следует и что не следует внедрять	51
1.3.1. Швы	52
1.3.2. Стабильные зависимости	53
1.3.3. Нестабильные зависимости	53
1.4. Область применения внедрения зависимостей	55
1.4.1. Композиция объектов	56
1.4.2. Жизненный цикл объектов	56
1.4.3. Перехват	58
1.4.4. Внедрение зависимостей в трех измерениях	59
1.5. Резюме	59

Глава 2. Детальный пример	61
2.1. Неверная реализация	62
2.1.1. Разработка сильно связанного приложения	63
2.1.2. Тест «На дым» (Smoke test)	69
2.1.3. Оценка	70
2.1.4. Анализ	74
2.2. Правильная реализация	76
2.2.1. Новая разработка приложения	78
2.2.2. Анализ слабо связанной реализации	87
2.3. Расширение приложения-примера	90
2.3.1. Архитектура	91
2.3.2. Корзина покупок	92
2.4. Резюме	95
Глава 3. Контейнеры внедрения зависимостей	96
3.1. Введение в контейнеры внедрения зависимостей.	99
3.1.1. Привет, контейнер	100
3.1.2. Автоподключение	103
3.2. Конфигурирование контейнеров внедрения	106
3.2.1. Конфигурирование контейнеров с использованием XML.	108
3.2.2. Конфигурирование контейнеров при помощи кода.	110
3.2.3. Конфигурирование контейнеров по соглашению	112
3.3. Паттерны контейнеров внедрения	116
3.3.1. Корень компоновки	116
3.3.2. «Регистрация, преобразование, высвобождение»	122
3.4. Панорама контейнеров внедрения	129
3.4.1. Выбор контейнера внедрения зависимостей.	129
3.4.2. Microsoft и внедрение зависимостей.	131
3.5. Резюме	134

Часть 2. Каталог паттернов внедрения зависимостей

Глава 4. Паттерны внедрения зависимостей	140
4.1. «Внедрение конструктора»	142
4.1.1. Как это работает	142

4.1.2. Когда должно использоваться внедрение конструктора . . .	144
4.1.3. Известные способы применения	145
4.1.4. Пример: добавление сервиса конвертации валют в покупательскую корзину	145
4.1.5. Родственные паттерны	148
4.2. «Внедрение свойства»	148
4.2.1. Как это работает	149
4.2.2. Когда следует применять внедрение свойства	150
4.2.3. Известные способы применения	152
4.2.4. Пример: реализация сервиса профиля валюты для BasketController	153
4.2.5. Родственные паттерны	155
4.3. Внедрение метода	156
4.3.1. Как это работает	156
4.3.2. Когда следует использовать внедрение метода	157
4.3.3. Известные способы применения	158
4.3.4. Пример: конвертация валют в корзине	159
4.3.5. Родственные паттерны	162
4.4. «Окружающий контекст»	162
4.4.1. Как это работает	163
4.4.2. Когда следует использовать окружающий контекст	165
4.4.3. Известные способы применения	169
4.4.4. Пример: кэширование валюты	169
4.4.5. Родственные паттерны	177
4.5. Резюме	177
Глава 5. Антипаттерны внедрения зависимостей	179
5.1. «Диктатор»	182
5.1.1. Пример: создание новых экземпляров зависимостей	182
5.1.2. Пример: фабрика	184
5.1.3. Анализ	189
5.2. «Гибридное внедрение»	191
5.2.1. Пример: ProductService с внешним умолчанием	192
5.2.2. Анализ	194
5.3. «Ограниченное конструирование»	197
5.3.1. Пример: динамическое связывание ProductRepository	197
5.3.2. Анализ	199

5.4. «Локатор сервисов»	202
5.4.1. Пример: ProductService, использующий «Локатор сервисов»	204
5.4.2. Анализ	206
5.5. Резюме	210
Глава 6. Рефакторинг внедрения зависимостей	212
6.1. Соотнесение значения времени исполнения с абстракциями	213
6.1.1. Абстракции с зависимостями времени исполнения	214
6.1.2. Пример: выбор алгоритма расчета пути	217
6.1.3. Пример: использование CurrencyProvider	218
6.2. Использование краткосрочных зависимостей	220
6.2.1. Закрытие соединений с помощью абстракций	221
6.2.2. Пример: обращение к сервису управления продуктами	224
6.3. Устранение циклических зависимостей	226
6.3.1. Устранение циклов зависимостей	227
6.3.2. Пример: компоновка окна	230
6.4. Использование свехвнедрения конструктора	234
6.4.1. Распознавание и разрешение свехвнедрения конструктора	234
6.4.2. Пример: рефакторинг получения заказов	237
6.5. Мониторинг связанности	241
6.5.1. Модульное тестирование связанности	242
6.5.2. Интеграционное тестирование связывания	245
6.5.3. Использование NDepend для отслеживания связей	247
6.6. Резюме	249

Часть 3. Самостоятельное создание внедрения зависимостей

Глава 7. Компоновка объектов	256
7.1. Компоновка в консольных приложениях	259
7.1.1. Пример: обновление валют	259
7.2. Компоновка приложений ASP.NET MVC	262
7.2.1. Расширяемость ASP.NET MVC	263
7.2.2. Пример: реализация CommerceControllerFactory	266

7.3. Компоновка WCF-приложений	269
7.3.1. Расширяемость WCF	269
7.3.2. Пример: подключение сервиса управления продуктами . . .	271
7.4. Компоновка WPF-приложений	279
7.4.1. WPF-компоновка	279
7.4.2. Пример: подключение насыщенного клиента для управления продуктами	280
7.5. Компоновка приложений ASP.NET	285
7.5.1. Компоновка в ASP.NET	285
7.5.2. Пример: подключение CampaignPresenter	287
7.6. Компоновка PowerShell cmdlets	292
7.6.1. Пример: компоновка cmdlets для управления корзиной . . .	293
7.7. Резюме	298
Глава 8. Время жизни объектов	299
8.1. Управление временем жизни зависимостей	301
8.1.1. Введение в управление временем жизни	302
8.1.2. Управление временем жизни с помощью контейнера	305
8.2. Использование одноразовых зависимостей	311
8.2.1. Использование одноразовых зависимостей	311
8.2.2. Управление одноразовыми зависимостями	315
8.3. Каталог жизненных стилей	320
8.3.1. Singleton.	320
8.3.2. Transient	323
8.3.3. Per Graph.	325
8.3.4. Web Request Context (Контекст веб-запроса)	327
8.3.5. Pooled (Пулированный, реализованный как пул)	332
8.3.6. Прочие жизненные стили	337
8.4. Резюме	341
Глава 9. Перехват	342
9.1. Вводная информация о перехватах	344
9.1.1. Пример: реализация аудита	344
9.1.2. Паттерны и принципы перехвата	348
9.2. Реализация сквозных аспектов приложения	352
9.2.1. Перехват с «Прерывателем цепи» (Circuit Breaker)	354
9.2.2. Обработка исключительных состояний	360
9.2.3. Добавление функционала безопасности	362

9.3. Объявление аспектов	364
9.3.1. Использование атрибутов для объявления аспектов	364
9.3.2. Применение динамического перехвата	369
9.3.3. Пример: перехват с Windsor	372
9.4. Резюме	378

Часть 4. Контейнеры внедрения зависимостей

Глава 10. Castle Windsor.	384
10.1. Знакомство с Castle Windsor	385
10.1.1. Разрешение объектов	386
10.1.2. Конфигурирование контейнера	388
10.1.3. Размещение конфигурации	394
10.2. Управление жизненным циклом	395
10.2.1. Конфигурирование стилей жизненных циклов	396
10.2.2. Использование продвинутых стилей жизненных циклов	397
10.2.3. Разработка специальных стилей жизненных циклов	400
10.3. Работа с несколькими компонентами	406
10.3.1. Выбираем одного кандидата из нескольких	407
10.3.2. Подключение последовательностей	410
10.3.3. Подключение декораторов	413
10.4. Конфигурирование сложных API	415
10.4.1. Конфигурирование примитивных зависимостей	416
10.4.2. Регистрирование компонентов с помощью блоков кода	417
10.4.3. Подключение с внедрением свойств	419
10.5. Резюме	420
Глава 11. StructureMap.	422
11.1. Знакомство со StructureMap	423
11.1.1. Разрешение объектов	425
11.1.2. Конфигурирование контейнера	427
11.1.3. Размещение конфигурации	434
11.2. Управление жизненным циклом	436
11.2.1. Конфигурирование стилей жизненных циклов	437
11.2.2. Разработка собственного стиля жизненных циклов	441

11.3. Работа с несколькими компонентами	447
11.3.1. Выбираем одного кандидата из нескольких	447
11.3.2. Подключение последовательностей	451
11.3.3. Подключение декораторов.	454
11.4. Конфигурирование сложных API	458
11.4.1. Конфигурирование примитивных зависимостей	458
11.4.2. Создание компонентов с помощью блоков кода	459
11.4.3. Подключение с внедрением свойств	460
11.5. Резюме	462