

«В этой замечательной книге описываются системы, при разработке которых приходится иметь дело с такими сложными вопросами, как параллелизм и ограничения реального времени. Автору удалось организовать материал таким образом, что книга будет полезна всем проектировщикам программ. От других изданий, посвященных объектно-ориентированному проектированию, ее отличает наличие весьма полезных и подробных методических рекомендаций».

Роберт Дж. Петит IV. The Aerospace Corporation

В книге «**UML. Проектирование систем реального времени, распределенных и параллельных приложений**» приводится детальный обзор концепций объектно-ориентированного проектирования, технологий, применяемых при создании параллельных и распределенных систем, методики использования прецедентов и нотации языка UML. Книга посвящена важной области разработки программного обеспечения – проектированию распределенных систем и систем реального времени с высокой степенью параллелизма. Пользуясь нотацией языка UML, автор убедительно демонстрирует, как можно эффективно применять для создания таких систем методы объектно-ориентированного анализа и проектирования.

В книге очень подробно описывается COMET, метод анализа и проектирования, специально адаптированный для разработки распределенных приложений и систем реального времени. В его основу положены итеративный жизненный цикл ПО, использование прецедентов и языка UML. Рассматривается весь цикл разработки программ – от моделирования требований и анализа до проектирования. На многочисленных примерах демонстрируется применение COMET в самых разных ситуациях.

К числу представленных тем относятся:

- объектно-ориентированный жизненный цикл разработки программного обеспечения;
- моделирование прецедентов с акцентом на встраиваемые подсистемы;
- статический и динамический анализ системы;
- разбиение системы на классы и объекты;
- конечные автоматы и диаграммы состояний;
- технологии распределенных объектов;
- проектирование архитектуры параллельных систем;
- разбиение на задачи и проектирование классов;
- анализ производительности систем реального времени.

Данное издание незаменимо для всех работающих в этой быстро развивающейся области. Описание различных технологий, систематический подход, подробные рекомендации и содержательные примеры – все это способно оказать существенную помощь при разработке высококачественных распределенных систем и приложений реального времени.

**Хассан Гома** работает профессором на кафедре программной техники в Университете Джорджа Мейсона, Фейрфакс, штат Вирджиния, и занимается как прикладными, так и академическими исследованиями. Он является признанным авторитетом в области проектирования распределенных приложений и систем реального времени.

Более подробную информацию о серии «**Объектно-ориентированные технологии в программировании**» можно найти по адресу: <http://www.awl.com/cseng/otseries>

Категория: Программирование/Системы моделирования/UML

Internet-магазин: [www.dmkpress.ru](http://www.dmkpress.ru)

Книга – почтой

Россия, 107014, Москва, а/я 468

тел.: (095) 962-1703, 369-7528

e-mail: [orders@dmkpress.ru](mailto:orders@dmkpress.ru)

Addison  
Wesley



Addison  
Wesley



UML  
Проектирование систем  
реального времени, распределенных  
и параллельных приложений



# ПРОЕКТИРОВАНИЕ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ, РАСПРЕДЕЛЕННЫХ И ПАРАЛЛЕЛЬНЫХ ПРИЛОЖЕНИЙ

ХАССАН ГОМА

Предисловие Питера Фримена и Брана Селика



Хассан Гома





УДК 004.415.2  
ББК 32.973.26-018.1  
Г64

Г64 Гома Х.

UML. Проектирование систем реального времени, параллельных и распределенных приложений: Пер. с англ. – М.: ДМК Пресс. – 704 с.: ил.  
(Серия «Объектно-ориентированные технологии в программировании»).

**ISBN 5-94074-101-0**

Данная книга описывает процесс создания распределенных, параллельных систем и систем реального времени с точки зрения проектирования архитектуры системы. Подробно рассмотрены ключевые вопросы, возникающие в процессе разработки: управление временем отклика, синхронизация, актуальность и непротиворечивость данных. На многочисленных примерах автор показывает, как с помощью одной и той же универсальной нотации UML описать такие, казалось бы, далекие области, как автоматизированная банковская система, бортовой компьютер автомобиля и система управления лифтами в многоэтажном здании – без привязки к какой-либо программной или аппаратной платформе и языку программирования.

Издание будет чрезвычайно полезно аналитикам, менеджерам предприятий и информационных систем, руководителям и архитекторам проектов, а также программистам, которые имеют дело с приложениями UML: книга поможет изложить свои идеи так, чтобы можно было реализовать проект, практически не зная той предметной области, для которой пишется система.

Translation copyright – by DMC Press

(Designing Concurrent, Distributed, and Real-time Applications with UML, First Edition by Hassan Goma, Copyright, All Rights Reserved)

Публикуется по согласованию с издательством, выпустившим оригинал: ADDISON-WESLEY LONGMAN, Pearson Education Inc.

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.

Материал, изложенный в данной книге, многократно проверен. Но, поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 0-201-65793-7 (англ.)  
ISBN 5-94074-101-0 (рус.)

Copyright © by Hassan Goma  
© Перевод на русский язык, оформление  
ДМК Пресс



# Содержание

|                   |    |
|-------------------|----|
| Предисловие ..... | 25 |
|-------------------|----|

|   |           |
|---|-----------|
| <b>ЧАСТЬ I. Нотация UML, концепции проектирования, технологии, жизненные циклы и методы .....</b> | <b>35</b> |
|---|-----------|

|                         |    |
|-------------------------|----|
| Глава 1. Введение ..... | 36 |
|-------------------------|----|

|   |    |
|---|----|
| 1.1. Объектно-ориентированные методы и UML .....        | 37 |
| 1.2. Метод и нотация .....                              | 38 |
| 1.3. Параллельные приложения .....                      | 38 |
| 1.3.1. Последовательные и параллельные программы .....  | 39 |
| 1.3.2. Последовательные и параллельные приложения ..... | 39 |
| 1.3.3. Параллельные задачи .....                        | 40 |
| 1.4. Системы и приложения реального времени .....       | 40 |
| 1.5. Распределенные системы и приложения .....          | 42 |
| 1.6. Резюме .....                                       | 43 |

|                                  |    |
|----------------------------------|----|
| Глава 2. Обзор нотации UML ..... | 44 |
|----------------------------------|----|

|   |    |
|---|----|
| 2.1. Диаграммы UML .....  | 44 |
| 2.2. Диаграммы прецедентов .....                                    | 45 |
| 2.3. Нотация UML для классов и объектов .....                       | 45 |
| 2.4. Диаграммы классов .....  | 46 |
| 2.5. Диаграммы взаимодействия .....                                 | 47 |
| 2.5.1. Диаграммы кооперации .....                                   | 47 |
| 2.5.2. Диаграммы последовательности .....                           | 48 |
| 2.6. Диаграммы состояний .....                                      | 48 |
| 2.7. Пакеты .....   | 50 |
| 2.8. Диаграммы параллельной кооперации .....                        | 51 |
| 2.8.1. Обмен сообщениями на диаграммах параллельной кооперации .... | 51 |
| 2.9. Диаграммы развертывания .....                                  | 51 |
| 2.10. Механизмы расширения UML .....                                | 53 |

|                              |    |
|------------------------------|----|
| 2.11. UML как стандарт ..... | 54 |
| 2.12. Резюме .....           | 55 |

## Глава 3. Концепции проектирования

|                        |    |
|------------------------|----|
| ПО и архитектуры ..... | 56 |
|------------------------|----|

|  |    |
|--|----|
| 3.1. Объектно-ориентированные концепции .....  | 56 |
| 3.1.1. Основные концепции .....  | 56 |
| 3.1.2. Объекты и классы .....  | 57 |
| 3.2. Соккрытие информации .....  | 58 |
| 3.2.1. Соккрытие информации<br>в объектно-ориентированном проектировании .....                 | 59 |
| 3.2.2. Соккрытие информации<br>в применении к внутренним структурам данных .....               | 59 |
| 3.2.3. Соккрытие информации при проектировании интерфейса<br>с устройствами ввода/вывода ..... | 62 |
| 3.2.4. Проектирование объектов, скрывающих информацию .....                                    | 63 |
| 3.3. Наследование .....  | 64 |
| 3.4. Активные и пассивные объекты .....  | 65 |
| 3.5. Параллельная обработка .....  | 66 |
| 3.5.1. Преимущества параллельного выполнения задач .....                                       | 66 |
| 3.5.2. Тяжеловесные и облегченные процессы .....   | 67 |
| 3.6. Кооперация между параллельными задачами .....   | 68 |
| 3.6.1. Проблема взаимного исключения .....   | 68 |
| 3.6.2. Пример взаимного исключения .....   | 69 |
| 3.6.3. Проблема синхронизации задач .....  | 70 |
| 3.6.4. Пример синхронизации задач .....  | 70 |
| 3.6.5. Проблема производителя/потребителя .....  | 72 |
| 3.6.6. Слабо связанный обмен сообщениями .....   | 73 |
| 3.6.7. Сильно связанный обмен сообщениями с ответом .....                                      | 74 |
| 3.6.8. Сильно связанный обмен сообщениями без ответа .....                                     | 74 |
| 3.6.9. Пример обмена сообщениями<br>между производителем и потребителем .....                  | 75 |
| 3.7. Соккрытие информации<br>в применении к синхронизации доступа .....                        | 76 |
| 3.7.1. Классы и объекты, скрывающие информацию .....   | 76 |
| 3.8. Мониторы .....  | 77 |
| 3.8.1. Пример монитора .....   | 77 |
| 3.8.2. Условная синхронизация .....  | 78 |
| 3.9. Шаблоны проектирования .....  | 79 |

|  |    |
|--|----|
| 3.10. Программные архитектуры и компонентные системы ..... | 80 |
| 3.10.1. Компоненты и разъемы .....                         | 81 |
| 3.10.2. Компонентные системы .....                         | 81 |
| 3.11. Резюме .....   | 81 |

## Глава 4. Технологии

|  |     |
|--|-----|
| параллельных и распределенных систем .....                         | 83  |
| 4.1. Среды для параллельной обработки .....                        | 83  |
| 4.1.1. Мультипрограммная среда .....                               | 83  |
| 4.1.2. Симметричная мультипроцессорная среда .....                 | 83  |
| 4.1.3. Распределенная среда .....                                  | 84  |
| 4.2. Поддержка исполнения  |     |
| в мультипрограммной и мультипроцессорной средах .....              | 85  |
| 4.2.1. Сервисы операционной системы .....                          | 85  |
| 4.2.2. Стандарт POSIX .....  | 86  |
| 4.2.3. Операционные системы реального времени .....                | 87  |
| 4.3. Планирование задач .....                                      | 88  |
| 4.3.1. Алгоритмы планирования задач .....                          | 88  |
| 4.3.2. Состояния задач .....                                       | 89  |
| 4.3.3. Контекстное переключение задач .....                        | 90  |
| 4.4. Вопросы ввода/вывода в операционной системе .....             | 90  |
| 4.4.1. Контроллеры устройств .....                                 | 90  |
| 4.4.2. Обработка прерываний .....                                  | 91  |
| 4.4.3. Ввод/вывод с опросом .....                                  | 92  |
| 4.5. Технологии клиент-серверных и распределенных систем ...       | 93  |
| 4.5.1. Конфигурации клиент-серверных и распределенных систем ..... | 93  |
| 4.5.2. Коммуникационные сетевые протоколы .....                    | 95  |
| 4.6. Технология World Wide Web .....                               | 97  |
| 4.6.1. Язык Java и World Wide Web .....                            | 98  |
| 4.7. Сервисы распределенных операционных систем .....              | 98  |
| 4.7.1. Служба имен .....   | 99  |
| 4.7.2. Связывание клиентов и серверов .....                        | 99  |
| 4.7.3. Сервисы распределенного обмена сообщениями .....            | 100 |
| 4.7.4. Сервисы сокетов .....                                       | 101 |
| 4.7.5. Обмен сообщениями через порты .....                         | 101 |
| 4.7.6. Восстановление после ошибок .....                           | 101 |
| 4.8. ПО промежуточного слоя .....                                  | 102 |
| 4.8.1. Платформы для распределенных вычислений .....               | 102 |
| 4.8.2. Вызовы удаленных процедур .....                             | 102 |
| 4.8.3. Вызов удаленных методов в языке Java .....                  | 104 |



|  |            |
|--|------------|
| 4.9. Стандарт CORBA .....  | 104        |
| 4.9.1. Брокер объектных запросов .....   | 104        |
| 4.9.2. Язык определения интерфейса в CORBA .....   | 105        |
| 4.9.3. Статическое и динамическое связывание .....   | 106        |
| 4.9.4. Сервисы CORBA .....   | 107        |
| 4.9.5. Интеграция унаследованных приложений<br>в каркас распределенных объектов .....      | 107        |
| 4.10. Другие компонентные технологии .....   | 108        |
| 4.10.1. Технология COM .....   | 108        |
| 4.10.2. Технология JavaBeans .....   | 108        |
| 4.10.3. Технология Jini .....  | 108        |
| 4.11. Системы обработки транзакций .....   | 109        |
| 4.11.1. Характеристики транзакций .....  | 109        |
| 4.11.2. Мониторы обработки транзакций .....  | 110        |
| 4.12. Резюме .....   | 111        |
| <b>Глава 5. Жизненные циклы и методы разработки<br/>    программного обеспечения .....</b> | <b>112</b> |
| 5.1. Определение жизненного цикла ПО .....   | 112        |
| 5.1.1. Модель водопада .....   | 112        |
| 5.1.2. Недостатки модели водопада .....  | 113        |
| 5.1.3. Временные прототипы .....   | 114        |
| 5.1.4. Создание эволюционирующих прототипов<br>в ходе инкрементной разработки .....        | 115        |
| 5.1.5. Комбинирование временных прототипов<br>и инкрементной разработки .....              | 116        |
| 5.1.6. Спиральная модель .....   | 117        |
| 5.1.7. Унифицированный процесс разработки ПО .....   | 118        |
| 5.2. Верификация и утверждение проекта .....   | 118        |
| 5.2.1. Контроль качества ПО .....  | 119        |
| 5.2.2. Анализ производительности .....   | 119        |
| 5.3. Тестирование программного обеспечения .....   | 119        |
| 5.3.1. Автономное тестирование .....   | 120        |
| 5.3.2. Тестирование сопряжений .....   | 120        |
| 5.3.3. Комплексное тестирование .....  | 120        |
| 5.3.4. Приемо-сдаточные испытания .....  | 121        |
| 5.4. Эволюция методов проектирования ПО .....  | 121        |
| 5.5. Эволюция методов<br>объектно-ориентированного анализа и проектирования .....          | 123        |

|   |     |
|---|-----|
| 5.6. Обзор современных методов проектирования<br>параллельных систем и систем реального времени ..... | 125 |
| 5.7. Резюме .....   | 126 |

|  |            |
|--|------------|
| <b>ЧАСТЬ II. COMET – метод архитектурного<br/>проектирования и моделирования<br/>параллельных объектов<br/>с применением UML .....</b> | <b>127</b> |
|--|------------|

|  |            |
|--|------------|
| <b>Глава 6. Введение в метод COMET .....</b> | <b>128</b> |
|--|------------|

|   |     |
|---|-----|
| 6.1. Жизненный цикл разработки<br>объектно-ориентированного ПО в методе COMET ..... | 128 |
| 6.1.1. Моделирование требований .....   | 128 |
| 6.1.2. Аналитическое моделирование .....  | 128 |
| 6.1.3. Проектное моделирование .....  | 129 |
| 6.1.4. Инкрементное конструирование ПО .....  | 130 |
| 6.1.5. Инкрементная интеграция ПО .....   | 130 |
| 6.1.6. Комплексное тестирование .....   | 130 |
| 6.2. Сравнение жизненного цикла COMET<br>с другими процессами разработки ПО .....   | 131 |
| 6.2.1. Сравнение жизненного цикла COMET с USDP .....                                | 131 |
| 6.2.2. Сравнение жизненного цикла COMET со спиральной моделью ....                  | 131 |
| 6.3. Модель требований, аналитическая и проектная модели ...                        | 131 |
| 6.3.1. Виды деятельности при моделировании требований .....                         | 132 |
| 6.3.2. Виды деятельности при аналитическом моделировании .....                      | 132 |
| 6.3.3. Виды деятельности при проектном моделировании .....                          | 133 |
| 6.4. Основы COMET .....   | 134 |
| 6.4.1. Разработка модели требований .....   | 134 |
| 6.4.2. Разработка аналитической модели .....  | 134 |
| 6.4.3. Разработка проектной модели .....  | 135 |
| 6.5. Резюме .....   | 137 |

|   |            |
|---|------------|
| <b>Глава 7. Моделирование прецедентов .....</b> | <b>138</b> |
|---|------------|

|  |     |
|--|-----|
| 7.1. Прецеденты .....                  | 138 |
| 7.2. Актеры .....                      | 139 |
| 7.3. Актеры, роли и пользователи ..... | 141 |
| 7.4. Выявление прецедентов .....       | 141 |



|  |            |
|--|------------|
| 7.5. Документирование прецедентов в модели прецедентов ....  | 142        |
| 7.6. Примеры прецедентов .....   | 143        |
| 7.6.1. Прецедент «Снять Деньги» .....  | 143        |
| 7.6.2. Прецедент «Получить Справку» .....  | 145        |
| 7.6.3. Прецедент «Перевести Деньги» .....  | 146        |
| 7.7. Отношения прецедентов .....   | 147        |
| 7.7.1. Отношение расширения .....  | 147        |
| 7.7.2. Отношение включения .....   | 149        |
| 7.7.3. Некоторые рекомендации .....  | 150        |
| 7.8. Пакеты прецедентов .....  | 150        |
| 7.9. Резюме .....  | 151        |
| <b>Глава 8. Статическое моделирование .....</b>  | <b>152</b> |
| 8.1. Ассоциации между классами .....   | 152        |
| 8.1.1. Изображение ассоциаций на диаграммах классов .....  | 153        |
| 8.1.2. Кратность ассоциаций .....  | 153        |
| 8.1.3. Другие ассоциации .....   | 155        |
| 8.1.4. Атрибуты связи .....  | 155        |
| 8.1.5. Классы-ассоциации .....   | 156        |
| 8.2. Иерархии композиции и агрегирования .....   | 157        |
| 8.3. Иерархия обобщения/специализации .....  | 159        |
| 8.4. Ограничения .....   | 160        |
| 8.5. Статическое моделирование и язык UML .....  | 160        |
| 8.5.1. Статическое моделирование предметной области .....  | 161        |
| 8.6. Статическое моделирование контекста системы .....   | 162        |
| 8.6.1. Внешние классы .....  | 163        |
| 8.6.2. Пример разработки диаграммы классов контекста системы<br>с внешними классами .....            | 164        |
| 8.6.3. Актеры и внешние классы .....   | 165        |
| 8.6.4. Пример разработки диаграммы классов контекста системы<br>на основе рассмотрения актеров ..... | 165        |
| 8.7. Статическое моделирование сущностных классов .....  | 166        |
| 8.8. Резюме .....  | 167        |
| <b>Глава 9. Разбиение на классы и объекты .....</b>  | <b>168</b> |
| 9.1. Критерии разбиения на объекты .....   | 168        |
| 9.2. Категории классов приложения .....  | 169        |
| 9.3. Структурирование категорий объектов .....   | 170        |
| 9.4. Внешние и интерфейсные классы .....   | 171        |
| 9.4.1. Категории внешних классов .....   | 171        |
| 9.4.2. Идентификация интерфейсных классов .....  | 172        |