

## ПРЕДИСЛОВИЕ

По божьей милости человечество жаждет духовной гармонии, эстетического совершенства, безопасности для своих близких, справедливости и свободы. Высокая производительность труда не в состоянии непосредственно обеспечить эти потребности. Тем не менее она избавляет человека от нужды, высвобождая время для духовного развития, эстетического наслаждения, семейных радостей, а также позволяет обществу уделять особое внимание распространению религии, укреплению справедливости и защите свобод.

Харлан Милс,  
“DPMA и производительность труда”

Программисты стремятся создавать работоспособные и полезные системы, но сталкиваются с необходимостью решать сложные задачи в условиях ограниченных вычислительных и трудовых ресурсов. Объектно-ориентированная технология (object-oriented (OO) technology) возникла и развивалась именно как средство решения проблем, связанных со сложностью, присущей многим программным системам. Время доказало мощь и универсальность объектного подхода.

### **Изменения, внесенные в третье издание**

Со времени выхода в свет второго издания книги *Object-Oriented Analysis and Design with Applications* технология достигла больших успехов. Некоторые из них перечислены ниже.

- Получило повсеместное распространение широкополосное беспроводное соединение в сети Интернет.
- Появилась нанотехнология, на основе которой создаются ценные изобретения.
- Роботы уже путешествуют по Марсу.
- Специальные эффекты, созданные с помощью компьютеров, позволили кинематографу воссоздавать любую фантазию с абсолютным реализмом.

- Стали доступными персональные транспортные средства на воздушной подушке.
- Повсеместное распространение получили мобильные телефоны.
- Расшифрован геном человека.
- Объектно-ориентированная технология стала основным методом промышленной разработки программного обеспечения.

Использование объектно-ориентированной парадигмы широко используется во всем мире. Однако, как выясняется, многие люди еще не освоили этот метод разработки проектов. Переработанное издание книги станет полезным как для читателей, уже применяющих объектно-ориентированный подход, так и для тех, кто не очень хорошо его понимает.

Люди, впервые изучающие объектно-ориентированный анализ и проектирование (object-oriented analysis and design (OOAD)), найдут в книге следующую информацию.

- Концептуальные основы и перспективы развития объектно-ориентированного подхода.
- Примеры использования методы OOAD на протяжении всего жизненного цикла системы.
- Введение в систему стандартных обозначений, используемых при проектировании систем и программного обеспечения, — Unified Modeling Language (UML 2.0)

Для опытных проектировщиков, использующих метод OOAD, книга будет интересной по следующим причинам

- Язык UML 2.0 по-прежнему неизвестен многим опытным проектировщикам. Они найдут в книге новую систему обозначений.
- С учетом критических замечаний, полученных после выхода в свет предыдущих изданий, в новом варианте книги сделан более сильный акцент на моделировании.
- Прочтя часть *Концепции*, читатели смогут понять, “почему вещи таковы, каковы они есть” в объектно-ориентированном мире. Многие люди до сих пор не имеют представления об эволюции концепций объектно-ориентированного проектирования. Впрочем, даже если читателю уже известны некоторые факты на эту тему, он не может отрицать значимости этой информации при первом изучении предмета.

Третье издание имеет четыре основных отличий от предыдущих.

1. За прошедшее время язык UML 2.0 получил официальное признание. Эта система обозначений изложена в главе 5. Для того чтобы облегчить читате-

лю изучение этого языка, мы отделили его основные элементы от элементов повышенной сложности.

2. В новом издании рассмотрены новые предметные области (см. часть III). В частности, рассмотрены абстракции разного уровня, начиная системами с высокоуровневой архитектурой и заканчивая системами, основанными на использовании Web-технологии.
3. После выхода в свет последнего по времени издания язык C++ претерпел новые изменения, связанные с концепциями объектно-ориентированного программирования. По мнению читателей подчеркивать эту особенность языка C++ уже нет необходимости. Рынок переполнен книгами, посвященными объектно-ориентированному программированию, а также учебниками по языкам программирования, основанным на этой парадигме. По этой причине мы решили изъять из книги большинство фрагментов программ.
4. В заключение, в ответ на пожелания читателей, мы сосредоточили основное внимание на аспектах моделирования в рамках парадигмы OOAD. В третьей части книги, посвященной приложениям, продемонстрированы примеры использования языка UML, причем в каждой главе рассмотрен один из этапов жизненного цикла проекта.

## Цели

Эта книга призвана служить практическим руководством по анализу и созданию объектно-ориентированных систем. Среди прочего следует выделить следующие конкретные задачи.

- Помочь читателям понять основные концепции объектной модели и описать ее эволюцию.
- Помочь читателям освоить систему обозначений и овладеть методами объектно-ориентированного анализа и проектирования.
- Научить читателя практическому применению методов объектно-ориентированного анализа в различных предметных областях.

Изложенные понятия имеют строгое теоретическое обоснование, но книга прежде всего призвана удовлетворить практические потребности и интересы разработчиков программного обеспечения: от архитекторов до программистов.

## Аудитория

Книга предназначена как для профессионалов, так и для студентов.

- Разработчикам практических систем и программного обеспечения мы покажем, как эффективно применять объектно-ориентированную технологию для решения реальных задач.

- Аналитикам или архитекторам систем мы покажем, как использовать методы объектно-ориентированного анализа и проектирования на всем пути от постановки задачи до реализации системы. Мы научим читателей отличать “хорошую” объектно-ориентированную архитектуру от “плохой” и находить компромиссы, обеспечивающие живучесть системы в реальных условиях. И что еще более важно, мы предлагаем новые подходы к изучению сложных систем.
- Менеджеру программного проекта мы подскажем, как распределить ресурсы в команде разработчиков, обеспечить качество программного обеспечения, оценить его и снизить риски, присущие сложному программному обеспечению.
- Студенты найдут в книге начальные знания, позволяющие им приобрести навыки в науке и искусстве создания сложных систем.

Книга может быть использована при чтении лекций на младших и старших курсах, а также для проведения профессиональных семинаров и самостоятельного изучения. Книга посвящена, в основном, методам разработки программного обеспечения, поэтому изложенный в ней материал идеально подойдет для курсов проектирования программ и для занятий по объектно-ориентированному языку.

## Структура

Книга состоит из трех больших частей — “Концепции”, “Метод” и “Приложения” — и содержит значительное количество вспомогательного материала.

## Концепции

Первая часть посвящена анализу сложности программного обеспечения, в частности, описанию путей, которыми эта сложность проявляется. В этой части описывается объектная модель, представляющая собой инструмент, позволяющий справиться со сложностью. В ней подробно рассматриваются основные элементы объектной модели: абстракция, инкапсуляция, модульность и иерархия, а также основные понятия, такие как класс и объект. Поскольку выявление осмысленных классов и объектов представляет собой главную задачу объектно-ориентированного проектирования, значительное место уделяется проблемам классификации. В частности, в книге рассматриваются подходы к классификации, принятые в других научных дисциплинах: биологии, лингвистике и психологии, а полученные выводы применяются для выявления классов и объектов, образующих программную систему.

## Метод

Вторая часть описывает метод построения сложных систем, основанный на объектной модели. Сначала вводится система графических обозначений (т.е. язык UML) для объектно-ориентированного анализа и проектирования, а затем рассматриваются основы обобщенного процесса разработки. Кроме того, в книге изучаются практические вопросы объектно-ориентированного проектирования, в частности, роль этого процесса в жизненном цикле программного обеспечения и его значение для управления проектами.

## Приложения

Третья часть книги содержит пять нетривиальных примеров, охватывающих разные предметные области: архитектуру системы, системы управления, криптоанализ, сбор данных и разработку Web-приложений. Выбор этих приложений объясняется тем, что они являются типичными примерами сложных задач, с которыми может столкнуться разработчик программного обеспечения. Выбранные принципы можно было бы легко продемонстрировать на примере решения простых задач, но, поскольку мы нацелены на создание реальных систем, намного интереснее показать, как объектная модель позволяет разрабатывать сложные приложения. Разработку программных систем невозможно свести к набору рецептов, поэтому мы настаиваем на необходимости постепенного развития приложений на основе четких принципов и продуманных моделей.

## Вспомогательные материалы

В текст книги вплетен значительный объем дополнительных сведений. В большинстве глав имеются специальные врезки, в которых содержится информация по отдельным важным темам. Кроме того, книга содержит приложение, посвященное основным свойствам объектно-ориентированных языков, глоссарий основных терминов и обширную библиографию, сгруппированную по темам, связанным с объектной моделью.

## Замечания об инструментах

Читатели часто задают вопросы об инструментах, использованных для построения диаграмм, помещенных в книге. Мы использовали два инструмента для построения диаграмм: IBM Rational Software Architect и Sparx Systems Enterprise Architect. Почему мы не ограничились одной программой? Реальность такова, что на рынке программных средств нет ни одного инструмента, который решал бы абсолютно все задачи. По мере углубления в темы, связанные с применением принципов ООАД, читатели непременно столкнутся с задачей, которую не решает ни одна программа. (В таком случае вы можете обратиться к распространенным

графическим редакторам, чтобы добиться желаемого результата.) Но не стоит придавать этим исключениям из правил слишком большое значение и отказываться от использования надежных средств OOAD, упомянутых выше.

### **Как читать книгу?**

Книгу можно читать последовательно или выборочно. Если вы стремитесь добиться глубокого понимания концепций объектной модели и принципов объектно-ориентированного проектирования, начните с главы 1 и читайте книгу последовательно. Если же вас в основном интересует система обозначений и процесс объектно-ориентированного анализа и проектирования, чтение следует начинать с глав 5 и 6. Менеджерам проектов, использующим объектно-ориентированный метод проектирования, будет особенно полезна глава 7. Если же читателей больше всего интересуют практические приложения объектно-ориентированной технологии в конкретных предметных областях, можно выбрать любую из глав 8–12.

## БЛАГОДАРНОСТИ

Книга посвящается моей жене Джен (Jan) в благодарность за ее любовь и поддержку.

На протяжении всей работы над первым и вторым изданиями мои представления об объектно-ориентированном проектировании уточнялись под влиянием многих людей, которым я хотел бы выразить особую благодарность. В их число входят Сэм Адамс (Sam Adams), Майк Акроид (Mike Akroid), Гленн Андерт (Glenn Andert), Сид Байлин (Sid Bailin), Кент Бек (Kent Beck), Дейв Бернштейн (Dave Bernstein), Дэниел Боброу (Daniel Bobrow), Дик Больц (Dick Bolz), Дэйв Балман (Dave Bulman), Кэйван Кэран (Kayvan Carun), Дэйв Коллинз (Dave Collins), Дамиан Конвэй (Damian Conway), Стив Кук (Steve Cook), Джим Коплиен (Jim Coplien), Брэд Кокс (Brad Cox), Уард Канингэм (Ward Cunningham), Том ДеМарко (Tom DeMarco), Майк Делвин (Mike Delvin), Ричард Гэбриел (Richard Gabriel), Вильям Генемерас (William Genemeras), Адель Голдберг (Adele Goldberg), Ян Грэхем (Ian Graham), Тони Хоар (Tony Hoare), Джон Хопкинс (Jon Hopkins), Майкл Джэксон (Michael Jackson), Ральф Джонсон (Ralph Johnson), Джеймс Кемпф (James Kempf), Норм Керт (Norm Kerth), Джордан Крейндрер (Jordan Kreindler), Дуг Ли (Doug Lea), Фил Леви (Phil Levy), Барбара Лисков (Barbara Liskov), Клифф Лонгмэн (Cliff Longman), Джеймс Макфарлэйн (James MacFarlane), Масуд Милани (Masoud Milani), Харлан Миллс (Harlan Mills), Роберт Мюррей (Robert Murray), Стив Нейс (Steve Neis), Джин Уйе (Gene Ouye), Дэйв Парнас (Dave Parnas), Билл Риддел (Bill Riddel), Мэри Бет Россон (Mary Beth Rosson), Кенни Рубин (Kenney Rubin), Джим Рамбо (Jim Rumbaugh), Курт Шмукер (Kurt Schmucker), Эд Зайдевиц (Ed Seidewitz), Дэн Шифман (Dan Shiftman), Дэйв Стивенсон (Dave Stevenson), Бьерн Страуструп (Bjarne Stroustrup), Дэйв Томас (Dave Thomas), Майк Вило (Mike Vilot), Тони Вассерман (Tony Wasserman), Питер Вегнер (Peter Wegner), Айсел Байт (Iseult White), Джон Уильямс (John Williams), Ллойд Уильямс (Lloyd Williams), Марио Волчко (Mario Wolczko), Никлаус Вирт (Niklaus Wirth) и Эд Йордан (Ed Yourdon).

Практические главы этой книги воплотили результаты моего участия в разработке сложных программных систем по всему миру для компаний Alcatel, Andersen Consulting, Apple, AT&T, Autotrol, Bell Northern Research, Boeing, Borland, Computer Sciences Corporation, Contel, Ericsson, Ferranti, General Electric, GTE, Holland Signaal, Hughes Aircraft Company, IBM, Lockheed, Martin Marietta, Motorola, NTT, Philips, Rockwell International, Shell Oil, Symantec, Taligent и TRW. Я имел возможность общаться с сотнями профессиональных программистов и менеджеров и благодарю их всех за помощь, позволившую сделать эту книгу более близкой к проблемам реальной жизни.

Особую благодарность я выражаю компании Rational за поддержку моей работы. Спасибо также Тони Холлу (Tony Hall), рисунки которого оживили довольно сухое техническое содержание книги. Наконец, благодарю трех моих кошек, Кэми (Samy), Энни (Annie) и Шэдоу (Shadow), составлявших мне

компанию в долгие часы ночной работы.

Гради Буч (Grady Booch)

Во-первых, я возношу хвалу Господу, без которого ничто не существовало бы. Я также благодарен своей семье, стойко переносившей часы разлуки, пока я работал на книгой. Благодарю своих родителей, приучивших меня честно работать. Спасибо Мэри Т. О'Брайен, предложившей мне принять участие в данной работе, а также Крису Гузиковскому (Chris Guzikowski), помогавшему мне довести дело до конца. Кроме того, я признателен моим соавторам, согласившимся принять меня в команду и упорно работавшими над этим проектом. В заключение, я хотел бы сказать, что сердечно благодарен Гради за его многолетнюю работу над созданием оригинальных и фундаментальных книг по объектно-ориентированному анализу и проектированию.

Боб Максимчук (Bob Maksimchuk)

Я хочу выразить мою признательность своей семье за их любовь и поддержку во всех моих начинаниях. Кроме того, я благодарен Гради за возможность внести свой вклад в третье издание его классической книги. В заключение, хочу сказать спасибо Бобу Максимчуку за помощь в работе над книгой.

Майк Энгл (Mike Engle)

Я посвящаю мою работу памяти своей матери, Джин Смит (Jean Smith), вдохновившей меня принять участие в этом проекте. Кроме того, я хотела бы выразить мою любовь и признательность своей семье, Руссель (Russell), Алисе (Alyssa) и Логан (Logan), за их поддержку. Благодарю Боба Максимчука и Майка Энгле за возможность разделить вместе с ними радость работы над книгой.

Бобби Дж. Янг (Bobbi J. Young)

Я хотела бы выразить особую благодарность моему мужу Бобу (Bob) и двум моим детям Кэтрин (Katherine) и Райан (Ryan), чья любовь и поддержка вдохновляли меня.

Кэлли А. Хьюстон (Kelli A. Houston)

Благодарю рецензентов, особенно Дэвида Норриса (Davyd Norris) и Брайна Лионса (Brian Lyons), а также многих других сотрудников издательства Addison-Wesley, работавших над книгой, особенно Криса Зана (Chris Zahn), который воплотил в жизнь этот огромный проект.

## ОБ АВТОРАХ

**Гради Буч** получил всемирное признание за его новаторскую работу в области архитектуры, проектирования и моделирования программных систем. С момента создания компании IBM Rational в 1981 году, он работал ее ведущим научным сотрудником. В марте 2003 года Гради получил звание почетного сотрудника IBM (IBM Fellow).

Гради — один из основоположников языка Unified Modeling Language (UML), а также автор нескольких оригинальных разработок компании Rational. Работая над многочисленными проектами, связанными с интенсивным программированием, Гради был не только архитектором программных систем, но и учил сотрудников принципам проектирования.

Гради является автором шести бестселлеров, включая справочное руководство *UML Users Guide* и знаменитую книгу *Object-Oriented Analysis with Applications*. Гради опубликовал несколько сотен технических статей по проектированию программного обеспечения. В статьях, напечатанных в начале 1980-х годов, он заложил основы теории и практики объектно-ориентированного проектирования. Гради читает лекции и дает консультации по всему миру.

Гради Буч является членом Ассоциации по вычислительной технике (Association for Computing Machinery (ACM)), Института инженеров по электротехнике и электронике (Institute of Electrical and Electronics Engineers (IEEE)), Американской ассоциации содействия развитию науки (American Association for the Advancement of Science (AAAS)), а также Общества специалистов по вычислительной технике, выступающих за социальную ответственность (Computer Professionals for Social Responsibility (CPSR)). Он является почетным сотрудником компании IBM, почетным членом ассоциации ACM Fellow, а также организаций World Technology Network Fellow и Software Development Forum Visionary.

Гради — один из основателей компаний Agile Alliance, Hillside Group и Worldwide Institute of Software Architects. Кроме того, в качестве консультанта он сотрудничает с университетом Норсфейс (Northface University).

Гради получил степень бакалавра наук в Академии военно-воздушных сил США (United States Air Force Academy) в 1977 году, а степень магистра по электротехнике — в университете Калифорнии в г. Санта-Барбара (University of California at Santa Barbara) в 1979 году.

Гради живет со своей женой и кошками в Колорадо. Он любит читать, путешествовать, петь и играть на губной гармошке.

**Роберт А. Максимчук** — руководитель научно-исследовательских работ (Research Director) в компании Unisys Chief Technology Office. В центре его научных интересов находятся новые технологии моделирования, разрабатываемые в среде Unisys 3D-Visual Enterprise. Боб сделал большой вклад в развитие методов проектирования систем, моделирования и анализа проектов в разных промышленных отраслях. Он соавтор книг *UML for Mere Mortals* и *UML for Database Design*, а также автор большого количества статей. Роберт путешествует по всему миру в качестве приглашенного лектора, выступая на многочисленных форумах и семинарах по языку UML и объектно-ориентированному проектированию. Боб является членом Института инженеров по электротехнике и электронике (Institute of Electrical and Electronics Engineers (IEEE)) и Международного совета по проектированию систем (International Council on Systems Engineering (INCOSE)).

**Майкл У. Энгл** — главный инженер компании Lockheed Martin Corporation. Его технический и управленческий опыт работы в области проектирования систем достигает 26 лет. Майкл принимал участие в разработке систем на самых разных этапах их жизненного цикла — от начала проекта до сопровождения. Используя свой опыт системного инженера и архитектора, Майк применяет объектно-ориентированные методы для выработки новаторских подходов к проектированию сложных систем.

**Бобби Янг**, доктор философии (Ph.D.) — руководитель научно-исследовательских работ (Director of Research) в компании Unisys Chief Technology Office. Она имеет многолетний опыт работы в области информационных технологий, в частности, в коммерческих компаниях и в Министерстве обороны США. Доктор Янг работала консультантом по подготовке менеджеров в области разработки программ, архитектуры предприятий, проектирования систем, а также объектно-ориентированного анализа и проектирования. В ходе своей карьеры она занималась процессами, протекающими на протяжении всего жизненного цикла программного обеспечения, вопросами методологии, а также архитектурой предприятий. Доктор Янг — обладательница научных степеней по биологии, компьютерным наукам и искусственному интеллекту. Степень доктора философии она получила в области управления информационными системами. Кроме того, Бобби Янг имеет воинское звание капитана 3-го ранга в отставке (Commander) Резерва военно-морского флота США (United States Naval Reserves).

**Джим Коналлен** — разработчик программного обеспечения в команде Model Driven Development Strategy компании IBM Rational, в которой он активно при-

меняет методы модельной архитектуры (Model Driven Architecture (MDA)), предложенные компанией Object Management Group (OMG), к инструментам моделирования компании IBM Rational. Кроме того, Джим много работает в области проектирования, основанного на активах (asset-based development), а также над спецификацией Reusable Asset Specification (RAS). Джим часто выступает на конференциях и публикует статьи. В круг его интересов входит проектирование Web-приложений.

Он разработал расширение языка UML Web Application Extension for UML (WAE), позволяющее разработчикам моделировать Web-архитектуры с помощью языка UML с достаточным уровнем абстракции и детализации. Эта работа стала основой методов IBM Rational Rose и Rational XDE Web Modeling.

Джим — автор двух изданий книги *Building Web Applications with UML*, в которой впервые были рассмотрены технологии Active Server Pages компании Microsoft и J2EE.

Свой жизненный опыт Джим получил, работая независимым консультантом в компании Rational, добровольцем Корпуса мира (Peace Corps), преподавателем в колледже и отцом трех мальчиков. Джим прослушал базовый курс и получил степень магистра в университете Виденер (Widener University) по проектированию компьютеров и программного обеспечения.

**Келли Хьюстон** — консультант по информационным технологиям (Consulting IT Specialist) в компании IBM Rational. Она является архитектором методов и отвечает за интеграцию методов, предложенных компанией IBM. Кроме того, Келли руководит работой группы Rational Method Composer (RMC) Special Interest Group (SIG) в компании IBM и проводит консультации и обучение пользователей и внутренних консультантов компании IBM эффективному использованию системы RMC.

## ОТ ИЗДАТЕЛЬСТВА

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо, либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг. Наши координаты:

E-mail: [info@williamspublishing.com](mailto:info@williamspublishing.com)

WWW: <http://www.williamspublishing.com>

Информация для писем

из России: 115419, Москва, ул. Лесная, д. 43, стр. 1

из Украины: 03150, Киев, а/я 152