

*Моей семье и всем моим друзьям, которые поддерживали
меня во время написания этой книги.*

—Грегор

Шэрон, моей любящей жене.

—Бобби

Предисловие

Джона Крупи

Как вы реагируете на появление новой технологии? Вы изучаете ее. Столкнувшись с платформой J2EE, я не нашел ничего лучшего, кроме как обратиться к ее спецификациям (книг по J2EE на то время еще не существовало). В частности, я сконцентрировал свое внимание на подмножестве J2EE — технологии EJB. Тем не менее изучение технологии является лишь первым шагом к ее эффективному применению. Преимущество платформенных технологий состоит в том, что они ограничивают разработчика определенным кругом задач. К сожалению, это не исключает неверного применения самой технологии.

По моим наблюдениям, за последние 15 лет в умах разработчиков программного обеспечения прочно укоренились идеи *эффективного* программирования и проектирования. В настоящее время на рынке существует огромное количество книг по программированию на Java и C#, однако гораздо меньше по проектированию приложений. Написав книгу *Core J2EE Patterns*, мы с Дипаком Алуrom (Deepak Alur) и Дэном Малксом (Dan Malks) хотели помочь J2EE-разработчикам в создании более эффективного кода. Основная идея нашей книги заключалась в использовании шаблонов проектирования. Как отметил Джеймс Бейти (James Baty), главный инженер компании Sun Microsystems, “шаблоны являются краеугольным камнем проектирования”. Я полностью согласен с этим высказыванием. К счастью, так же считают и авторы настоящей книги.

Книга, которую вы держите в руках, посвящена чрезвычайно популярной в последнее время теме: интеграции приложений с помощью обмена сообщениями. На мой взгляд, идея обмена сообщениями найдет свое применение не только при интеграции приложений, но и при создании Web-служб. Как это было на заре появления технологий J2EE и .NET, вопросам проектирования Web-служб уделяется пока еще очень мало внимания. Многие считают, что Web-службы являются всего лишь новым способом решения существующих задач интеграции, и я полностью с этим согласен. Тем не менее это не отменяет необходимость проектирования самих Web-служб. Изюминка данной книги состоит в наличии большого количества шаблонов, которые можно применять при проектировании Web-служб и других интеграционных систем. Бобби и Грегор не балуют читателя конкретными примерами спецификаций Web-служб, поскольку они все еще нуждаются в доработке и утверждении. Впрочем, это вполне простительно. Подлинная ценность книги проявится после объявления соответствующих спецификаций стандартами, что позволит использовать предложенные шаблоны проектирования для разработки соответствующих этим стандартам приложений. Вполне вероятно, что создание архитектур, ориентированных на использование служб, станет основным способом интеграции приложений в ближайшем будущем.

Прочтите эту книгу от начала до конца. Она способна кардинально изменить вашу карьеру.

Джон Крупи
Бетесда, Мэриленд
Август 2003 г.

Предисловие

Мартин Фаулер

В процессе написания книги *Patterns of Enterprise Application Architecture* (Фаулер М. *Архитектура корпоративных программных приложений*. — М.: Издательский дом “Вильямс”, 2004) мне посчастливилось получить подробную рецензию на включенный в нее материал от Кайла Брауна (Kyle Brown) и Рейчел Рейниц (Rachel Reinitz). В результате нескольких неофициальных встреч в офисе Кайла в Роли-Дарем мы пришли к выводу, что самым существенным пробелом моей книги стали системы асинхронного обмена сообщениями.

Безусловно, моя книга содержит множество пробелов и не претендует на звание исчерпывающего источника информации по шаблонам проектирования корпоративных приложений. Тем не менее отсутствие в ней описания систем асинхронного обмена сообщениями особенно болезненно ввиду того, что эти системы будут играть ключевую роль при решении задач интеграции. Приложения не могут существовать обособленно одно от другого, и нам нужны методики, позволяющие наладить связь между приложениями, изначально не предусматривавшими возможность взаимодействия.

На сегодняшний день существует множество различных технологий интеграции приложений. На мой взгляд, наиболее многообещающей из них является обмен сообщениями. Организация эффективного обмена сообщениями — нетривиальная задача, сложность которой обуславливается асинхронной природой сообщений, а также различиями в подходах к асинхронному программированию.

К сожалению, мне не хватило времени, сил и, что самое главное, знаний, чтобы достойно раскрыть тему обмена сообщениями в *Patterns of Enterprise Application Architecture*. Тем не менее я все-таки нашел решение этой проблемы в виде Грегора и Бобби, взявшихся за написание книги, которую вы сейчас держите в руках.

Следует отметить, что авторы блестяще справились с поставленной перед ними задачей. Если вы уже работали с системами обмена сообщениями, настоящая книга поможет вам систематизировать накопленный опыт и знания. Если же вы только приступаете к изучению данной темы, книга поспособствует закладке прочного фундамента, который пригодится вам при использовании любой технологии обмена сообщениями.

Мартин Фаулер
Мелроуз, Массачусетс
Август 2003 г.

Вступление

Эта книга посвящена интеграции корпоративных приложений с помощью обмена сообщениями. В ней не рассматриваются конкретные технологии или продукты. Вместо этого она ориентирована на разработчиков и интеграторов, использующих различные технологии и продукты для обмена сообщениями, такие как:

- MOM- и EAI-решения IBM WebSphere MQ, Microsoft BizTalk, TIBCO, WebMethods, SeeBeyond, Vitria и др.;
- реализации спецификации Java Message Service (JMS), внедренные в серверы приложений J2EE, а также в отдельные продукты;
- служба Microsoft Message Queuing (MSMQ), доступная через различные API, например через классы пространства имен Microsoft .NET System.Messaging;
- новые стандарты Web-служб, поддерживающие асинхронный обмен сообщениями (например, WS-ReliableMessaging), а также соответствующие API, такие как Java API for XML Messaging (JAXM) от Sun Microsystems или Web Services Extensions (WSE) от Microsoft.

Интеграция корпоративных приложений означает нечто большее, чем создание распределенного приложения с *n*-уровневой архитектурой. Тогда как отдельный уровень распределенного приложения не может выполняться самостоятельно, интегрированные приложения представляют собой независимые сущности, способные к взаимной координации по принципу слабой связи. Обмен сообщениями позволяет нескольким приложениям передавать данные и команды по сети в соответствии с подходом “отправил и забыл”. Последнее позволяет вызывающему приложению отправить информацию и вернуться к выполнению текущей задачи, переложив всю ответственность за доставку информации на систему обмена сообщениями. При необходимости вызывающее приложение может быть уведомлено о результате доставки информации с помощью функции обратного вызова. Несмотря на то что по сравнению с синхронным обменом сообщениями асинхронный подход способен усложнить проектирование приложения, возможность неограниченного числа повторов асинхронного вызова способна существенно повысить надежность коммуникаций. Асинхронный обмен сообщениями обладает и другими преимуществами, такими как регулирование числа запросов и балансировка нагрузки.

Для кого предназначена эта книга

Эта книга призвана помочь разработчикам приложений и системным интеграторам наладить взаимодействие между приложениями с помощью средств, ориентированных на обмен сообщениями. В первую очередь, она предназначена для следующих специалистов.

- **Архитекторы и разработчики, проектирующие и создающие сложные корпоративные приложения**, которые нуждаются в интеграции с другими приложениями. Мы ориентируемся на разработчиков приложений, использующих современные платформы, такие как Java 2 Enterprise Edition (J2EE) и Microsoft .NET. С помощью этой книги вы научитесь соединять приложение с уровнем обмена сообщениями для взаимодействия с другими приложениями. Следует отметить, что предметом рассмотрения этой книги является исключительно интеграция приложений, а не их создание. Шаблоны проектирования, применяющиеся при создании корпоративных приложений, рассматриваются в книге Мартина Фаулера *Patterns of Enterprise Application Architecture* (Фаулер М. *Архитектура корпоративных программных приложений*. — М.: Издательский дом “Вильямс”, 2004).
- **Архитекторы и разработчики, проектирующие и создающие интеграционные решения**, предназначенные для налаживания взаимодействия между пользовательскими приложениями. Вероятно, большинство читателей этой книги уже сталкивались с коммерческими средствами интеграции приложений, такими как IBM WebSphere MQ, TIBCO, WebMethods, SeeBeyond, Vitria и др. Каждое из этих средств содержит реализацию многих шаблонов проектирования, представленных в книге. Помимо фундаментальных концепций, стоящих за интеграцией приложений, принимать правильные решения при проектировании вам поможет универсальный язык шаблонов, не зависящих от конкретного вендора.
- **Корпоративные архитекторы**, в обязанности которых входит мониторинг программных и аппаратных активов предприятия. Язык и графическое представление шаблонов проектирования помогут описать крупномасштабные решения интеграции, охватывающие множество различных технологий. Кроме того, язык шаблонов поможет наладить эффективное взаимодействие между корпоративным архитектором и архитекторами/разработчиками корпоративных приложений и решений интеграции.

Чему учит эта книга

Основное назначение этой книги — познакомить читателя с принципами проектирования успешных решений интеграции корпоративных приложений. В процессе ее чтения вы узнаете:

- преимущества и ограничения асинхронного обмена сообщениями в сравнении с другими технологиями интеграции;
- как определить каналы обмена сообщениями, в которых нуждается приложение; регламентировать процедуру получения одного и того же сообщения несколькими потребителями; обрабатывать сообщения неправильного формата;

- когда следует отправлять сообщение, какую информацию оно должно содержать и как использовать специальные свойства сообщений;
- как доставить сообщение в конечную точку его назначения (в том числе и при отсутствии информации о получателе сообщения);
- как преобразовать сообщение при несовпадении формата отправителя и получателя;
- как спроектировать код соединения приложения с системой обмена сообщениями;
- как проводить управление и мониторинг системы обмена сообщениями.

Что не рассматривается в этой книге

На наш взгляд, все книги, в названиях которых встречается слово “корпоративный”, делятся на три категории. К первой категории относятся книги, авторы которых стараются всецело охватить предмет обсуждения, однако в конечном итоге останавливаются на подробном рассмотрении конкретных решений. Книги второй категории содержат практические советы по разработке конкретных решений, ограничивая при этом диапазон рассматриваемого материала. Наконец, можно попытаться написать книгу, объединяющую в себе лучшие характеристики первых двух категорий, однако в этом случае она либо не будет закончена, либо будет опубликована настолько поздно, что потеряет свою актуальность. Настоящая книга относится ко второй категории. Мы попытались написать книгу о создании интеграционных решений, пожертвовав при этом некоторыми смежными темами. В частности, мы не рассматривали вопросы безопасности, сложного отображения данных, рабочих потоков, механизма поддержки правил, масштабируемости, устойчивости, а также распределенной обработки транзакций (продукты XA, Tuxedo и др.). Асинхронный обмен сообщениями был выбран основной темой книги по нескольким причинам. Во-первых, это одна из наиболее перспективных технологий интеграции приложений, которой свойственно множество интереснейших вопросов проектирования. Во-вторых, рассмотрение асинхронного обмена сообщениями позволяет абстрагироваться от реализации конкретных решений интеграции, предлагаемых различными вендорами.

Эта книга не является руководством по конкретной технологии обмена сообщениями или межплатформенному решению. Чтобы подчеркнуть многообразие концепций, представленных в книге, мы включили в нее примеры использования целого ряда различных технологий, таких как JMS, MSMQ, TIBCO, BizTalk и XSL. Каждая из перечисленных технологий рассматривалась исключительно в аспекте принятия решений проектирования и возникающих при этом компромиссов. Для подробного изучения какой-либо из технологий интеграции приложений обратитесь к источникам, перечисленным в конце книги, или к ресурсам Интернета.

Структура книги

Как следует из названия книги, большая ее часть посвящена *шаблонам*. Шаблоны — это проверенный способ представления накопленного опыта и знаний в таких областях, как архитектура программных приложений, объектно-ориентированное проектирование и создание интеграционных решений с помощью технологий обмена сообщениями.

Каждый шаблон состоит из описания некоторой проблемы проектирования, обсуждения исходных условий и представления элегантного, сбалансированного решения. В большинстве случаев предлагаемое решение является результатом длительного процесса поиска. Каждый шаблон “вбирает” в себя опыт, накопленный старшими разработчиками и архитекторами в попытке дать оптимальный ответ на конкретную задачу интеграции. Хотим подчеркнуть, что в этой книге нет “придуманных” нами шаблонов. Шаблон нельзя “придумать”; к нему можно прийти в результате длительных “полевых” испытаний, извлекая уроки из собственных ошибок.

Если вы уже имеете опыт работы со средствами интеграции корпоративных приложений и с архитектурой асинхронного обмена сообщениями, то большинство из представленных в этой книге шаблонов наверняка покажутся вам до боли знакомыми. Однако даже в этом случае она может принести немалую пользу. Чтение книги поможет закрепить знания, накопленные в результате использования технологий обмена сообщениями, а также поспособствует развитию привычки документирования создаваемых решений и взаимоотношений между ними. Благодаря универсальному языку шаблонов вы сможете наладить эффективное взаимодействие со своими коллегами.

Представленные в книге шаблоны можно реализовать с помощью различных платформ и языков программирования. Чтобы облегчить процесс встраивания шаблона в конкретную среду, в книгу были включены примеры реализации шаблонов с привлечением нескольких популярных технологий, таких как JMS, MSMQ, TIBCO, BizTalk, XSL и др. Кроме того, мы продемонстрировали примеры создания целостных решений, основанных на сочетании шаблонов проектирования.

Интеграция корпоративных приложений с помощью архитектуры асинхронного обмена сообщениями — сложная и увлекательная задача. Искренне надеемся, что наша книга поможет вам блестяще решить ее.

О фотографии на обложке книги

На обложках всех книг серии *A Martin Fowler Signature Book* размещается фотография какого-либо моста. В некотором смысле нам повезло, так как мост великолепно иллюстрирует идею интеграции. На протяжении тысячелетий мосты помогали человеку преодолевать препятствия в виде рек, горных ущелий и оживленных автострад.

Мост Тайко-баси (“Барабанный мост”) перекинут через пруд перед храмом Сумиёси-тайся, расположенным в Осаке, Япония. Мы выбрали этот мост за его красоту и элегантность. Синтоистский храм Сумиёси-тайся был построен на берегу бухты Нанива в III веке н.э. и посвящен божествам, которые оберегали моряков, отправлявшихся в плавание. За много лет вода отступила, и сейчас храм удален от воды более чем на 5 км. В начале каждого года Сумиёси-тайся посещают более трех миллионов человек.

Грегор Хоп
Сан-Франциско, Калифорния

Бобби Вульф
Роли, Северная Каролина

Сентябрь 2003 г.
www.enterpriseintegrationpatterns.com

Ждем ваших отзывов!

Вы, читатель этой книги, и есть главный ее критик. 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 интересны любые ваши замечания 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 ли вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Отправляя письмо или сообщение, не забудьте указать название книги и ее авторов, а также свой обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию новых книг.

Наши электронные адреса:

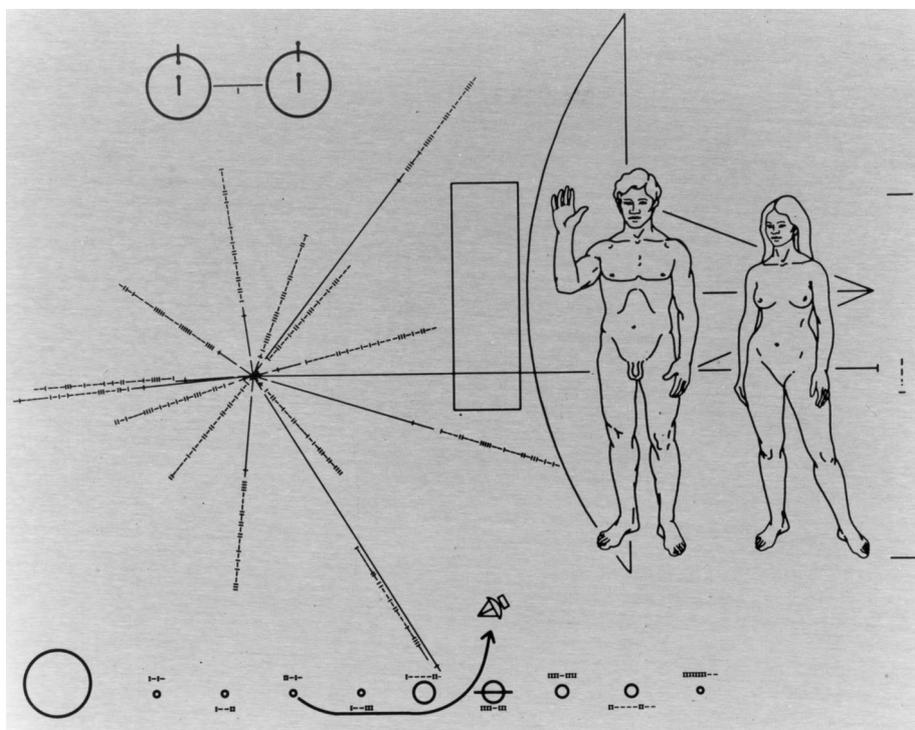
E-mail: info@williamspublishing.com

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

Наши почтовые адреса:

в России: 115419, Москва, а/я 783

в Украине: 03150, Киев, а/я 152



Д-р Карл Саган, "Послание внеземным формам жизни" (табличка, отправленная в космос на американском аппарате серии "Пионер")

Благодарности

Работа над этой книгой отняла у ее авторов немало времени и сил. Идея создания книги о шаблонах интеграции, основанных на обмене сообщениями, впервые возникла в 2001 году в процессе написания Мартином Фаулером своего бестселлера *Patterns of Enterprise Application Architecture* (Фаулер М. *Архитектура корпоративных программных приложений*. — М. : Издательский дом “Вильямс”, 2004). Кайл Браун (Kyle Brown), один из рецензентов *Patterns of Enterprise Application Architecture*, заметил, что в книге Мартина практически не освещается тема интеграции приложений. Этот пробел стал основным предметом обсуждения на нескольких встречах между Мартином и Кайлом, на которых присутствовали также Рейчел Рейниц (Rachel Reinitz), Джон Крупи (John Crupi) и Марк Вайцель (Mark Weitzel). Осенью 2001 года к дискуссии присоединился Бобби, а зимой 2002 года — Грегор. Летом 2002 года группа единомышленников представила на конференции по языкам шаблонов программ (Pattern Languages of Programs — PLoP) две работы, одна из которых была написана Бобби и Кайлом, а другая — Грегором. После конференции Кайл и Мартин сосредоточились на собственных проектах, а Грегор и Бобби решили объединить свои статьи для создания фундамента будущей книги. Примерно в это же время был открыт сайт www.enterpriseintegrationpatterns.com, посредством которого архитекторы и разработчики интеграционных решений всего мира могли принять участие в совместной работе над содержимым книги. Впоследствии к написанию книги были привлечены несколько соавторов. Спустя два года, прошедших с момента критического замечания Кайла относительно содержимого *Patterns of Enterprise Application Architecture*, рукопись настоящей книги была передана издателю.

Эта книга является плодом совместной работы огромного числа людей. Коллеги и друзья (многих из которых мы приобрели в процессе написания книги) подавали идеи для размещаемых в книге примеров, проверяли корректность материала с технической точки зрения, а также снабжали нас бесценными советами и критикой.

Отдельной благодарности заслуживают Кайл Браун и Мартин Фаулер — инициаторы создания этой книги. Если бы Мартин не принялся писать *Patterns of Enterprise Application Architecture*, а Кайл не сформировал группу для обсуждения шаблонов интеграции с помощью обмена сообщениями, наша книга могла бы никогда не увидеть свет.

Огромный вклад в создание книги внесли ее соавторы: Конрад Ф. Д’Круз (Conrad F. D’Cruz), Шон Невилл (Sean Neville), Майкл Дж. Реттиг (Michael J. Rettig) и Джонатан Саймон (Jonathan Simon). Написанные ими главы помогли пролить свет на практическую сторону применения шаблонов проектирования.

Одними из первых критиков книги стали участники конференции по языкам шаблонов программ PLoP 2002: Али Арсанджани (Ali Arsanjani), Кайл Браун, Джон Крупи, Эрик Эванс (Eric Evans), Мартин Фаулер, Брайан Мэрик (Brian Marick), Тоби Сарвер (Toby Sarver), Джонатан Саймон, Билл Транделл (Bill Trundell) и Марек Вокач (Marek Vokac).

Мы хотели бы поблагодарить рецензентов, множество раз перечитавших черновики книги и снабдивших нас сотней полезных советов:

- Ричарда Хелма (Richard Helm),
- Люка Хохманна (Luke Hohmann),
- Драгоша Манолеску (Dragos Manolescu),
- Дэвида Райса (David Rice),
- Русс Руфер (Russ Rufer) и всех членов Silicon Valley Patterns Group,
- Мэтью Шорта (Matthew Short).

Отдельное спасибо Русс за привлечение к работе над книгой сообщества Silicon Valley Patterns Group. Вклад этих людей был поистине неоценим: Роберт Бенсон (Robert Benson), Трейси Бялик (Tracy Bialik), Джеффри Блейк (Jeffrey Blake), Азад Болур (Azad Bolour), Джон Брюэр (John Brewer), Боб Эванс (Bob Evans), Энди Фарли (Andy Farlie), Джефф Глаза (Jeff Glaza), Фил Гудвин (Phil Goodwin), Алан Харриман (Alan Harriman), Кен Хеджмановски (Ken Hejmanowski), Дебора Кадда (Deborah Kaddah), Ритурадж Кирти (Rituraj Kirti), Ян Луни (Jan Looney), Крис Лопес (Chris Lopez), Джерри Луис (Jerry Louis), Тао-хунг Ма (Tao-hung Ma), Джефф Миллер (Jeff Miller), Стилян Пандев (Stilian Pandev), Джон Парелло (John Parello), Хема Пиллей (Hema Pillay), Русс Руфер, Рич Смит (Rich Smith), Кэрол Тислтуэйт (Carol Thistlethwaite), Дебби Утли (Debbie Utley), Уолтер Ваннини (Walter Vannini), Дэвид Выдра (David Vydra) и Тед Янг (Ted Young).

Список рассылки Web-сайта www.enterpriseintegrationpatterns.com позволил принять участие в обсуждении материала книги всем желающим. Наибольший вклад в дискуссию внес Билл Гранделл. Среди самых активных участников списка рассылки оказались также Венкатешвар Бомминени (Venkateshwar Bommineni), Данкан Крэгг (Duncan Cragg), Джон Крупи, Фокко Дегенаар (Fokko Degenaar), Шаилеш Госави (Shailesh Gosavi), Кристиан Холл (Christian Hall), Ральф Джонсон (Ralph Johnson), Пол Джулиус (Paul Julius), Орьян Люндберг (Orjan Lundberg), Драгош Манолеску, Роб Ми (Rob Mee), Шрикант Нарасимхан (Srikanth Narasimhan), Майкл Реттиг, Фрэнк Зауэр (Frank Sauer), Джонатан Саймон, Федерико Спинацци (Federico Spinazzi), Рэнди Стаффорд (Randy Stafford), Марек Вокач, Джо Уолнз (Joe Walnes) и Марк Вайцель.

Спасибо Мартину Фаулеру за то, что он позволил нам опубликовать книгу в серии *A Martin Fowler Signature Book*. Поддержка Мартина придала нам дополнительные силы, необходимые для завершения работы над книгой.

Мы благодарны Джону Крупи за написанное им предисловие. На протяжении всего времени работы над книгой Джон был нашим советчиком и вдохновителем, проявляя при этом незаурядное терпение и прекрасное чувство юмора.

Наконец мы хотели бы выразить огромную признательность замечательным сотрудникам издательства Addison-Wesley: главному редактору Майку Хендриксону (Mike Hendrickson), руководителю производства Эми Флейшер (Amy Fleischer), руководителю проекта Ким Арни Малкахи (Kim Arney Mulcahy), литературному редактору Кэрол Дж. Лолль (Carol J. Lallier), корректору Ребекке Райдер (Rebecca Rider), составителю предметного указателя Шэрон Хильгенберг (Sharon Hilgenberg), а также Жаклин Дюсетт (Jacquelyn Doucette), Джону Фуллеру (John Fuller) и Бернарду Гаффни (Bernard Gaffney).

Мы наверняка забыли упомянуть несколько имен, имевших непосредственное отношение к появлению *Enterprise Integration Patterns* на свет. Простите нас за это, и еще раз огромное спасибо за помощь. Надеемся, что все вы так же гордитесь этой книгой, как и мы.