
Содержание

Предисловие.....	15
Благодарности	21
Об авторе.....	25
Предисловие автора.....	27
Глава 1. <i>Почему все не так просто</i>	39
Причинно-следственные связи	41
Сложность.....	42
Наше линейное мышление.....	44
Редукционизм	46
Идея целостности	48
Иерархический менеджмент	49
Гибкий менеджмент	52
Моя теория всего	53
Модель, предлагаемая в этой книге.....	54
Резюме	56
Подумать и сделать.....	56
Глава 2. <i>Гибкие методологии разработки ПО</i>	59
Прелюдия к гибким методологиям	60
Евангелие от Agile	62
Фундаментальные принципы Agile-методологий	65
Методологии, конкурирующие с Agile	69
Препятствие на пути Agile-методологий	73

	Линейный и проектный менеджмент	74
	Резюме	76
	Подумать и сделать	77
Глава 3.	<i>Теория сложности</i>	79
	Важность междисциплинарного подхода	80
	Общая теория систем	82
	Кибернетика	83
	Теория динамических систем	84
	Теория игр	84
	Эволюционная теория	85
	Теория хаоса	86
	Общая картина наших знаний о поведении систем	87
	Простота: новая модель	89
	Еще раз об упрощении	93
	Адаптивные и неадаптивные системы	95
	Не злоупотребляем ли мы научным подходом?	96
	Новая эра: мышление в категориях теории сложности	98
	Резюме	101
	Подумать и сделать	101
Глава 4.	<i>Информационно-инновационная система.</i>	103
	Инновации — ключ к выживанию	105
	Знания	107
	Креативность	110
	Мотивация	112
	Разнообразие	115
	Личность	118
	Почему только люди способны управлять сложными системами	121
	От создания идей к их реализации на практике	122
	Резюме	123
	Подумать и сделать	124
Глава 5.	<i>Как добавить людям энергии</i>	125
	Фазы креативности	126
	Как создать рабочую среду, способствующую креативности	129
	Креативные методики	131

Внешняя мотивация	132
Внутренняя мотивация	136
Демотивация	137
Десять базовых потребностей членов команды	138
Что мотивирует людей: найдите баланс	142
Сделайте ваши награды внутренними	147
Разнообразие? Вы имеете в виду связи!	147
Оценка личности сотрудника	150
Четыре этапа при оценке личности членов команды	152
Создание командных ценностей: набор инструментов	154
Определитесь со своими личными ценностями	157
Политика отсутствия дверей	159
Резюме	161
Подумать и сделать	161
 Глава 6. <i>Базовые принципы самоорганизации</i>	163
Самоорганизация в контексте	164
Самоорганизация с целью создания ценности	165
Самоорганизация в сравнении с анархией	168
Самоорганизация и эмерджентность	169
Эмерджентность в командах	172
Самоорганизация, самонаправление и самоотбор	174
Принцип темноты	175
Теорема Конанта–Эшби	177
Распределенный контроль	178
Делегирование прав и полномочий как концепция	180
Передача прав и полномочий как необходимость	181
Менеджеров можно сравнить с садовниками	183
Резюме	187
Подумать и сделать	187
 Глава 7. <i>Расширение прав и полномочий команд</i>	189
Не создавайте мотивационные долги	190
Попробуйте стать волшебником	192
Выбирайте волшебников, а не политиков	193
Расширение полномочий в сравнении с делегированием	194
Уменьшайте свой страх, повышайте свой статус	196
Выбирайте правильный уровень зрелости	197

	Выбирайте правильный уровень полномочий.	200
	Кому передавать полномочия — командам или отдельным сотрудникам	205
	Чек-лист для делегирования	207
	Если хотите, чтобы что-то было сделано, наберитесь терпения	208
	Сопровождайте своего менеджера сопротивлением	210
	Не забывайте про десять внутренних потребностей людей	212
	Мягко влияйте на рабочую среду	213
	Доверие	214
	Уважение.	218
	Резюме	222
	Подумать и сделать.	223
Глава 8.	<i>Осмысленное лидерство и управление</i>	225
	Игра «Жизнь»	226
	Классы клеточных автоматов.	227
	Ложная метафора	229
	Вы не дизайнер игр.	230
	Но... одной самоорганизации недостаточно	232
	Управляйте системой, а не людьми	234
	Менеджеры или лидеры?	236
	Правильнее сравнивать лидеров с правителями	237
	Смысл жизни	239
	Предназначение команды	242
	Постановка внешних целей	246
	Резюме	247
	Подумать и сделать.	248
Глава 9.	<i>Настройка ограничений</i>	249
	У людей должна быть общая цель	250
	Чек-лист для Agile-целей	253
	Рассказывайте о своей цели	256
	Видение против миссии	258
	Примеры организационных целей	260
	Разрешите своей команде иметь автономную цель.	262
	Достигайте компромисса между вашей целью и целью вашей команды	263

	Создайте список границ передаваемых полномочий . . .	264
	Выберите правильный менеджерский угол зрения	265
	Защищайте людей.	267
	Защищайте ресурсы, находящиеся в общем пользовании.	269
	Требования к качеству	271
	Заключите общественный договор.	273
	Резюме	275
	Подумать и сделать.	276
Глава 10.	<i>Искусство создавать правила</i>	277
	Самообучающиеся системы	278
	Правила в сравнении с ограничениями.	280
	Слепая зона гибких методологий	284
	Важная тема: профессиональное мастерство	287
	Положительная обратная связь.	289
	Отрицательная обратная связь	291
	Дисциплина × умения = компетентность	294
	Разнообразие правил	297
	Принцип субсидиарности.	299
	Восприятие рисков и иллюзия безопасности	301
	Меметика	303
	Разбитые окна	307
	Резюме	309
	Подумать и сделать.	310
Глава 11.	<i>Как вырастить компетентность</i>	311
	Семь методов развития компетенций	313
	Оптимизируйте систему в целом — на разных уровнях.	316
	Оптимизируйте систему в целом — в разных измерениях	318
	Советы при выборе операционных показателей	321
	Четыре ингредиента саморазвития	324
	Менеджмент, кончинг, менторинг	327
	Подумайте о сертификации сотрудников	329
	Используйте силу социального давления	331
	Используйте адаптируемые инструменты	333
	Подумайте о супервайзере	335

	Проводите встречи с глазу на глаз	339
	Встречи для обсуждения оценок 360°	340
	Повышайте стандарты	344
	Управляйте системой, а не правилами или людьми	345
	Резюме	347
	Подумать и сделать	348
Глава 12.	<i>Коммуникация в организационных структурах</i> . . .	351
	Это баг или фича?	352
	Коммуникация и обратная связь	353
	Передача искаженной информации — это норма	356
	Возможности коммуникаторов	358
	Сетевые эффекты	363
	Настройка соединения	365
	Конкуренция и сотрудничество	368
	Группы и границы	370
	Гиперпродуктивность, или Автокатализ	372
	Формирование паттернов	375
	Симметрия в масштабе: большие и маленькие паттерны	378
	Как расти: вертикально или горизонтально?	380
	Резюме	382
	Подумать и сделать	383
Глава 13.	<i>Как выращивать структуру</i>	385
	О внешней среде, продуктах, размерах организаций и людях	386
	Сначала подумайте о специализации	389
	...И только потом о генерализации	391
	Расширьте названия должностей	392
	Культивируйте неформальное лидерство	395
	Поддерживайте границы между командами	397
	Оптимальный размер команды — пять человек (скорее всего)	399
	Функциональные и кросс-функциональные команды . . .	402
	Два принципа организационного дизайна	405
	Выберите свой организационный стиль	407
	Превратите каждую команду в небольшую бизнес-единицу, создающую ценность . . .	410

Создавайте отдельные команды для решения однотипных задач	412
Каждый иерархический уровень должен создавать дополнительную ценность	417
Сколько нужно менеджеров, чтобы изменить организацию?	420
Создавайте гибридные организации	421
Анархия мертва, да здравствует панархия	422
Не имейте секретов	425
Сделайте все заметным	427
Связывайте людей.	428
Стремитесь к адаптивности.	429
Резюме	430
Подумать и сделать.	431
 Глава 14. <i>Ландшафт изменений</i>	433
Внешняя среда находится вовсе не «где-то там».	434
Страх неопределенности	436
Законы изменений	438
Каждый продукт успешен... пока не потерпит неудачу ..	440
Успех и степень приспособленности: все относительно ...	442
Как адаптироваться к изменениям	443
Адаптация, исследование, прогнозирование	444
Гонка Черной Королевы	447
Можно ли измерить сложность?	450
Становятся ли программные продукты более сложными?	451
Форма вещей: фазовое пространство	454
Аттракторы и конвергенция	456
Устойчивость и возмущения	458
Адаптивный ландшафт	460
Формируем ландшафт	463
Направленная и ненаправленная адаптация.	465
Резюме	466
Подумать и сделать.	467
 Глава 15. <i>Как улучшить всё</i>	469
Линейные улучшения в сравнении с нелинейными	471
Определите свое местоположение.	474

Советы путешествующим по зыбкому ландшафту	476
Измените внешнюю среду: пусть гора придет к вам	479
Сделайте изменения желанными	482
Сделайте застой болезненным	484
О пользе ошибок	485
Стратегия шума	486
Стратегия секса	489
Стратегия горизонтального переноса опыта	490
Не копируйте улучшения	494
Несколько завершающих практических советов по теме непрерывного улучшения	496
Продолжайте движение	498
Резюме	500
Подумать и сделать	500
 Глава 16. <i>Все модели неверны, но некоторые из них полезны</i>	 503
Шесть компонентов Менеджмента 3.0.	503
Да, моя модель «неверна»	506
Другие модели точно так же «неверны».	509
Типичные заблуждения практиков гибких методологий	512
Инструкция по управлению сложностью	514
Резюме	518
Подумать и сделать	519
 Библиография.	 521

Предисловие автора

Это книга о **гибком, или Agile-менеджменте** — управленческом аналоге гибких методологий разработки ПО. Я считаю, что гибкий менеджмент недостаточно представлен в мире, который требует гибких подходов. Существуют десятки книг по Agile-методологиям для разработчиков, тестировщиков, коучей и проектных менеджеров, но практически нет книг по этой тематике для Agile-менеджеров и лидеров команд. Но, если организации хотят внедрить Agile-практики, абсолютно необходимо, чтобы лидеры команд и другие руководители знали лучший подход для управления и лидерства в их командах.

Исследования показывают, что при переходе к гибким методам основным препятствием оказывается традиционный менеджмент [VersionOne 2009]. Командам разработчиков ПО тяжело внедрять такие процессы, как Scrum, XP или канбан, если их «лидеров» заклинило на устаревших управленческих подходах. Менеджерам необходимо понять, в чем заключается их новая роль в XXI веке и как добиваться от команд разработчиков максимальных результатов. Данная книга предназначена для менеджеров, которые хотят перейти на гибкие методы управления в своих компаниях, и на разработчиков, которые уже используют эти методы при создании ПО, но хотят больше узнать о менеджменте в целом.

Эта книга по менеджменту уникальна, поскольку целиком основана на научном подходе и теории сложности. В отличие от других

книг по общему менеджменту, она не призывает вас открыть свое сердце, взяться за руки и повторять мантры. Многие менеджеры, особенно в высокотехнологичных компаниях, предпочитают пользоваться левым полушарием мозга, полагаясь на рациональное, аналитическое начало. Поэтому я написал книгу, апеллирующую к таким людям. Но и тем, кто предпочитает пользоваться правым полушарием, нечего опасаться. Научные идеи представлены достаточно неформально, с подробными объяснениями и обилием метафор и иллюстраций. Здесь даже можно найти как минимум пару действительно смешных шуток.

Одной из моих важнейших целей при написании этой книги было придерживаться *описательного* подхода, а ни в коем случае не *нормативного*. Цель — дать вам понять, как работают организация и Agile-команды для того, чтобы вы могли решить собственные проблемы. Мир слишком сложен, чтобы можно было отделаться списком практик, которым необходимо следовать. Что действительно необходимо менеджерам в XXI веке, так это понимание общих подходов, используя которые, они смогут создать свои собственные рецепты, соответствующие их конкретным потребностям [Mintzberg 2004: 252].

История этой книги

Мне потребовалось десять лет, чтобы написать эту книгу. В свое время я заинтересовался гибкими методологиями разработки ПО и теорией сложности (не помню, в какой последовательности), и в течение первых пяти лет авторы, пишущие об этих двух предметах, едва поспевали за моим интересом. При чтении разных книг у меня постепенно начала складываться общая картина. Я понял, что гибкие методы создания ПО — это практическое приложение теории сложности и команды разработчиков ПО и соответствующие проекты выступают в качестве примера таких систем. Также стало ясно, что практически никто не видит эту связь между теорией и практикой (заметными исключениями стали Джим Хайсмит и Кен Швабер). В результате примерно в 2005 году я попытался написать собственную книгу на эту тему. Но в тот момент ничего не получилось. У меня был в руках текст, но отсутствовали читатели. Были новые идеи, но не было обрат-

ной связи. Обилие теорий и минимум опыта. Я был преисполнен энтузиазма, но мне не хватило терпения.

Параллельно все эти десять лет я занимался управлением проектами по разработке ПО и приобрел обширный опыт, узнал о множестве способов неправильного управления проектами. Будучи руководителем и внедряя гибкие методологии разработки, я размышлял о роли менеджмента в этом процессе. Я был уверен, что менеджерам и лидерам команд должна отводиться важная роль. Но в книгах ничего не говорилось о том, в чем конкретно она должна состоять.

В январе 2008 года я запустил свой блог на <http://poor.nl> с целью получить обратную связь от читателей относительно моих идей в области разработки ПО, менеджмента и сложных систем, а также понять, интересна ли эта тематика вообще кому-нибудь. Через полтора года у меня было 4000 подписчиков. Я участвовал в интереснейших дискуссиях с экспертами со всего мира и удачно выступил на нескольких конференциях в Европе и США. Было похоже, что я нашел свою нишу.

В августе 2009 года, уже после глобального финансового кризиса, я подумал, что пришло время сделать вторую попытку написать книгу. На этот раз все было очень легко. У меня был архив моего блога, полезная обратная связь от читателей, десять лет менеджерского опыта (в основном отрицательного), полно времени (дела шли неважно) и достаточное количество подписчиков в блоге, чтобы мотивировать нескольких издателей предложить мне контракт на книгу. После подписания первого в моей жизни договора в качестве автора все, что мне оставалось сделать, — это удвоить свои усилия в части исследований, утроить интеллектуальные усилия и в четыре раза увеличить свою продуктивность как автора. (Все это звучит гораздо проще, чем было на самом деле.)

Вы, конечно, обратили внимание, что я не являюсь ни консультантом по Agile-методологиям, ни ученым, специализирующимся в области сложных систем. В этом моя сила — и моя слабость. Сила в том, что я редко страдаю туннельным зрением. Мое мышление не испорчено пристрастием к каким-либо конкретным научным подходам, методам или предпочитаемым по умолчанию решениям. Еще со школы мне удавалось видеть общие закономерности и аналогии между различными предметными областями,

и учителя в свое время советовали мне выбрать карьеру, которая имела бы отношение к анализу проблем. Моя слабость в том, что часто я воспринимаю проблему со слишком большой высоты. Мне не хватает детальных знаний, которые есть у ученых, и глубокого опыта, который имеется у консультантов, изучивших изнутри множество компаний. Зато мне повезло, что я сумел развить у себя способность писать простые, неожиданные, конкретные, убедительные и эмоциональные тексты. В конце концов, неидеальное, но хорошо написанное послание лучше, чем идеальное послание, которое никто не хочет читать.

Пока я писал эту книгу, я использовал свой блог для получения откликов от подписчиков, так что они не давали мне сбиться с пути, помогали точнее мыслить и отсеивать не слишком полезные идеи. В результате получилась именно та книга, которую я в течение десяти лет мечтал написать. Но в определенном смысле это и та книга, которую хотели увидеть мои читатели.

Структура книги

В этой книге вы не найдете конкретных кейсов или пространных списков «стандартных» подходов. Вместо этого здесь приводятся результаты исследований, метафоры, идеи и поводы для размышлений. Все это не делает книгу менее полезной. Напротив, существует точка зрения, что самые значительные достижения осуществляются путем копирования идей из одной области и адаптации их к другой области. Стратегии выживания в биологических экосистемах могут научить не хуже, чем кейсы из практики других компаний. Идеи редко идеально соответствуют вашей ситуации. Именно вы должны решать, могут ли эти идеи применяться конкретно в вашем случае и если могут, то как.

Пользоваться книгой просто. Начните с начала. Потом читайте и переворачивайте страницы. Закончив читать страницу, переверните ее и начните следующую. В один прекрасный момент вы уткнетесь в пустую страницу. Это будет конец книги.

Глава 1 — *введение*. В ней описано, как линейное мышление порой приводит нас к неправильным выводам. Здесь же впервые представлена центральная модель книги: *Менеджмент 3.0* — шесть

углов зрения, с которых в ней рассматриваются команды и организации.

В главах 2 и 3 содержится обзор *гибких методологий разработки программного обеспечения и теории сложности*. Они представляют собой две основы гибкого менеджмента и тех шести компонентов модели, о которых идет речь в последующих главах.

В главах 4 и 5 описывается первый компонент модели Менеджмента 3.0, а именно как заряжать энергией людей. В первой из этих глав излагаются теоретические аспекты проблемы, а во второй — практические. Здесь говорится, что люди — это важнейшая часть организации и что менеджеры должны делать все, что в их силах, чтобы сотрудники проявляли активность, были креативными и мотивированными.

Главы 6 и 7 посвящены второму компоненту модели — расширению полномочий команд и созданию условий для их самоорганизации. Для решения этой задачи необходимы полномочия и доверие. Опять же, в первой из двух глав речь идет в основном о теории, а во второй — о практике.

В главах 8 и 9 объясняется, что такое настройка ограничений. Самоорганизация команд может привести к любым результатам, поэтому людям необходимо дать четкое направление и цель, а также обеспечить их защиту и защиту ресурсов, находящихся в общем пользовании. Это третий компонент модели Менеджмента 3.0.

В главах 10 и 11 обсуждается проблема компетенций, недостаток которых может не позволить командам достичь поставленных целей. Поэтому в зоне ответственности менеджеров должна находиться задача развития необходимых компетенций и дисциплины. Развитие компетенций — это четвертый компонент модели Менеджмента 3.0.

Главы 12 и 13 посвящены пятому компоненту Менеджмента 3.0 — функционированию команд в контексте организации. Подчеркивается важность выбора правильной социально-сетевой структуры, обеспечивающей беспрепятственный обмен информацией.

Главы 14 и 15 рассматривают процесс «Улучшай все», который будет шестым и последним компонентом модели. Подчеркивается необхо-

димось непрерывного улучшения функционирования людей, команд и организации в целом как способа максимально отодвинуть провал. Как и прежде, материал первой из этих двух глав носит более теоретический, а второй — более прикладной характер.

И наконец, в главе 16 мы подводим итоги и проводим общий обзор Менеджмента 3.0, а также его сравнение с некоторыми другими моделями менеджмента.

Как видите, каждый из шести компонентов модели Менеджмента 3.0 описывается двумя главами, и в каждом случае первая из двух глав носит более теоретический, а вторая — более прикладной характер. Можно прочитать только практические главы, чтобы узнать о том, как именно применять гибкий менеджмент, но в этом случае вы не поймете, почему рекомендуются именно эти методы.

Содержание отдельных глав не слишком сильно зависит друг от друга. Так что *в теории* вы можете читать о шести компонентах модели в любом порядке. Однако *на практике*, вероятно, легче всего начинать с первой главы. Я лично не проверял, какое именно из 720 возможных сочетаний этих компонентов будет наиболее удобным с точки зрения восприятия.

Возможно, время от времени вы будете замечать, что темы, обсуждаемые в этой книге, не всегда тесно увязаны друг с другом. Это сделано намеренно. Мне представлялось важным, чтобы материал был организован вокруг шести компонентов моей модели и чтобы через всю книгу проходило четкое разделение между теорией и практикой. Иногда было нелегко организовать материал в рамках одной главы и четко обозначить связи между различными темами. Но мне кажется, что я справился с этим достаточно хорошо. Выражаю надежду, что восприятие моей книги читателями будет менее критичным, чем восприятие автора.

Содержание книги

Текст книги написан в бета-версии Microsoft Word 2010. Все иллюстрации нарисованы и отсканированы мной, а затем раскрашены в приложении Paint.NET. Иногда в книге попадаются серые вставки с вопросами или замечаниями и ответами на них. Большинство этих вопросов задавались читателями моего блога и рецензентами

первых версий книги. Я также использую много ссылок на сайт «Википедии». Некоторые считают, что ссылаться на «Википедию» — порочная практика, но я с этим не согласен. Я предпочитаю ссылки на живой ресурс, над которым постоянно ведется работа по улучшению, чем на ветки потенциально мертвого дерева.

Предвидя обвинения в излишнем теоретизировании, я сделал так, что в сумме объем практических глав превышает объем тех, что отданы теории. Более того, в конце каждой главы есть раздел «Подумать и сделать», что делает книгу еще более полезной с практической точки зрения.

Часто говорят, что применение метафор улучшает понимание абстрактных понятий — именно поэтому я так часто ими пользуюсь. В этой книге вы найдете сравнения менеджеров с садовниками, волшебниками, регулировщиками дорожного движения и другими интересными людьми. Вначале я подумывал назвать эту книгу «Абстрактный садовник». Но в конечном итоге решил использовать другое название, потому что любая метафора имеет свойство изнашиваться, если ей пользоваться часто и для описания разных явлений. Поэтому теперь при необходимости я стараюсь подобрать отдельную метафору для каждого случая.

У этой книги есть сопутствующий сайт <https://www.management30.com>. На нем вы найдете дополнительные материалы, не вошедшие в книгу, первоначальные версии иллюстраций (разрешается их похитить и использовать для своих целей), материалы, присланные читателями, и ссылки на другие ресурсы, посвященные гибким методологиям, разработке ПО и теории сложности.

О названии модели

«Менеджмент 3.0» — довольно странное название. Но, как мне кажется, указание на версию 3.0 дает верное представление о направлении развития менеджмента в XXI веке.

Менеджмент 1.0 = иерархии

Некоторые называют этот менеджмент научным, другие — командно-контролирующим. Но в основе одна и та же идея: организацию выстраивают и управляют ею сверху вниз и властные полно-

мочия в руках немногих. У тех, кто находится вверху иерархической организации, самые высокие зарплаты, самые большие эго и самые дорогие офисные кресла. У тех, кто внизу, денег существенно меньше, меньше ответственности и нет мотивации работать хорошо.

В качестве компенсации тех рисков, что несут с собой высокие менеджерские должности, топ-менеджеры наделены возможностью манипулировать бонусами, что во многих случаях позитивно влияет скорее на их личное благосостояние, чем на результаты возглавляемых ими компаний. В качестве побочного фактора извращенные бонусные системы внесли свой вклад в мировой финансовый кризис.

Можно сделать уверенный вывод, что менеджмент 1.0, пусть он до сих пор и наиболее распространенная версия во всем мире, имеет целый ряд серьезных недостатков. Он устарел и нуждается в обновлении.

Менеджмент 2.0 = дань преходящим увлечениям

Некоторые люди осознают, что вне совсем уж стандартных ситуаций менеджмент 1.0 работает плохо, поэтому были созданы разнообразные и не до конца научные модели и расширения типа системы сбалансированных показателей, шести сигм, теории ограничений и тотального управления качеством. Будучи надстройками менеджмента 1.0, эти модели исходят из того, что организации управляются сверху и призваны помочь топ-менеджерам улучшить «дизайн» своих организаций. Иногда это срабатывает, иногда — нет.

Параллельно возникают другие модели и сервисы, фокусирующиеся на искусстве и мастерстве менеджмента. Многие книги типа «Менеджер за одну минуту», «Двадцать один закон лидерства» или «От хорошего к великому» содержат базовые принципы и рекомендации, которым менеджерам полагается следовать, а также советы больше практиковаться и набирать опыт. Опять же, иногда такие советы и рекомендации работают, иногда — нет. Наборы рекомендаций меняются чаще, чем подгузники у младенца.

Менеджмент 2.0 — это все тот же знакомый нам менеджмент 1.0, к которому добавили некоторое количество надстроек, чтобы несколько снять остроту проблем, порожденных старомодной системой. Но в основе архитектуры менеджмента 2.0 лежат те же устаревшие иерархии.

Менеджмент 3.0 = сложные системы

В последние несколько десятилетий мы стали свидетелями зарождения и развития теории сложности, вначале в применении к математике и биологии, а затем к экономике и социологии. Это было крупным прорывом. Стивен Хокинг считал это направление в науке настолько важным, что называл XXI век веком сложности.

Одно из важнейших прозрений новой теории заключается в том, что все организации представляют собой сети. Люди могут сколько угодно изображать свои компании в виде иерархий, но это не отменяет того факта, что на практике они будут сетями. Во-вторых, теория сложности в применении к социальным системам показывает, что менеджмент в первую очередь должен заниматься людьми и их взаимоотношениями, а не структурой департаментов и получением прибыли.

Многие из нас уже в курсе, что термин «лидерство» — не более чем модное название ситуации, когда менеджеры делают правильные вещи и делают их правильно. Но мышление категориями сложных систем добавляет в наш словарь новое измерение. Оно заставляет нас воспринимать организации как живые системы, а не как машины.

Иногда стоит менять названия. От них многое зависит. Название «Менеджмент 3.0» подчеркивает, что менеджмент нуждается в изменениях. Компании Microsoft обычно требуется сделать три релиза продукта, чтобы он нормально заработал. Я считаю, что в своем третьем воплощении менеджмент наконец нашел надежную научную основу. Предлагавшиеся ранее надстройки и апгрейды все еще полезны. Но мы обязаны поменять свою исходную гипотезу с иерархий на сетевые структуры, потому что XXI век — это эпоха сложности.

О подзаголовке книги

В подзаголовке книги «Лидерство и управление командами» упомянута тема **лидерства** — это термин, который часто используется неправильно. Есть два типа людей, которые неверно его интерпретируют. Я называю их принцами и жрецами.

Принцы

Некоторые утверждают, что «лидерство — это не то же самое, что менеджмент» в том смысле, что лидерство предполагает вдохновение, в то время как менеджмент относится скорее к исполнению. Они утверждают, что лидерство находится на «более высоком уровне», чем менеджмент. Меня всякий раз коробит, когда компания называет своих топ-менеджеров «лидерами».

Каждый сотрудник, начиная от президента компании и вплоть до последнего разработчика, может вдохновлять коллег и указывать им направление. У лидеров по определению нет формальных рычагов власти над своими последователями. Но какой же акционер доверит деньги «лидеру», не располагающему формальными полномочиями? Это глупая затея.

К сожалению, в данный момент среди топ-менеджеров распространена мода называть себя лидерами независимо от того, есть у них последователи или нет. Топ-менеджеры используют «лидерство» как социальный миф для укрепления своих позиций в организационной иерархии [Назу 2007: 110]. Я называю таких топ-менеджеров *принцами* (и принцессами), поскольку они думают, что занимаемая должность дает им больше прав на роль лидера, чем всем остальным, а еще потому, что они предпочитают блестящие предметы здравому смыслу.

Жрецы

Еще одна категория людей утверждает, что «менеджмент не нужен». Они говорят о социальных сетях, «Википедии», Linux и других замечательных достижениях социальных групп, которым удалось сформировать общую цель и в результате многого добиться. Они полагают, что «самоорганизующиеся» группы вообще не нуждаются в менеджерах, только в лидерах, обладающих видением.

К сожалению, данная точка зрения игнорирует тот факт, что ни в одном из таких примеров речь не идет о бизнесе. Если никто не будет *владельцем* активов организации, то никто и не нужен, чтобы *управлять* ими. Акционеры вряд ли оценят, если в результате самоорганизации их биотехнологическая компания будет трансформирована в кейтеринговый бизнес. Не стоит принимать во внимание и мнение сотрудников о том, нужны им менеджеры или нет. Менеджеры необходимы акционерам для того, чтобы управлять их бизнесом. Самоорганизация сама по себе лишена ценности. Требуется кто-то заинтересованный в результате, кто и будет решать, «хороши» или «плохи» результаты самоорганизации.

Увы, некоторые считают, что иерархия — это всегда «плохо», а самоорганизация — всегда «хорошо». Я называю их жрецами (и жрицами), потому что они проповедуют свою веру в то, что считают «хорошим», хотя (как показано в этой книге) никаких научных оснований для этой веры нет.

Прагматики

Когда речь идет о менеджменте и лидерстве, реальность требует от нас оставаться прагматиками. Любой бизнес нуждается в менеджменте от лица акционеров. Да, у менеджеров *должны* быть лидерские качества. Но многие лидерские роли могут исполняться самоорганизующимися людьми (не занимающими менеджерских должностей), находящимися на самых разных позициях в компании. Эти неформальные лидеры должны понимать, что направление, в котором происходит самоорганизация, необходимо немного корректировать и что делается это акционерами через распределение полномочий среди менеджеров.

Если вы похожи на меня, то вы не принц и не жрец, вы — один из простолюдинов. Я буду называть нас прагматиками. Мы понимаем, что управленческая иерархия — это базовая необходимость (и нечем тут хвастаться) и что основная часть работы совершается внутри социально-сетевой структуры, состоящей из равных: лидеров и последователей. Коммуникация осуществляется через сети, а полномочия — через иерархию.

Я написал эту книгу для прагматиков.

Глава 1

Почему все не так просто

*Всякая сложная проблема имеет решение —
простое, удобное и ошибочное.*

Г.Л. Менкен, журналист и писатель (1880–1956)

Однажды мне удалось некоторое время побыть миллионером — по крайней мере на бумаге. Частные инвесторы оценили мой стартап в интернете в 10 миллионов евро, при этом мне принадлежало 70% финансовой фикции, которую им удалось создать вокруг моего проекта. Мне даже присудили титул *предпринимателя года*, поскольку я очень хорошо передал свое видение проекта. А созданные мной цветные графики, показывающие ожидаемые доходы и расходы, выглядели просто сказочно — опять же на бумаге.

Несмотря на все это, деньги, которые мы с инвесторами вложили в проект, не привели к росту прибыли. Создание дополнительного контента не привело к увеличению посещаемости нашего сайта. Мы наняли больше программистов, но это не сказалось каким-либо существенным образом на скорости разработки. Договоренности, которые у нас были с другими сайтами, не привели к росту доходов. Доходы оказались даже *ниже*, чем перед первым раундом инвестиций. Уверен, что вам незнакомо название нашего не слишком известного сайта. Все попытки раскрутить его позорно провалились. А когда лопнул пузырь доткомов, это

вообще поставило жирный крест на всем проекте — так же, как и на множестве других, которыми были заняты все вокруг нас.

И тем не менее сам процесс был весьма увлекательным. Мы многому научились. О, как многому мы научились! Если правда, что лучше всего учишься на собственных ошибках, то к настоящему моменту я должен был быть просто всеведущим божеством. В качестве менеджера, руководившего процессом разработки, лидера команды, менеджера проекта и просто разработчика программного обеспечения я совершил столько ошибок, что мне до сих пор кажется странным, что вместе со своим проектом я не обрушил весь интернет. Но мы действительно в результате очень многому научились.

При написании этой книги меня поддерживала надежда, что и вы многому научитесь на моих ошибках и ошибках тех, кто совершал их до меня. За последние десять лет я понял, что при разработке программного обеспечения оптимальные результаты дают именно **Agile-методологии** (см. главу 2 «Гибкие методологии разработки ПО»). Я также убедился, что серьезнейшее препятствие на пути принятия Agile-методологий во всем мире — это традиционный менеджмент. Я исхожу из представления, что вы либо руководитель, либо просто интересуетесь менеджментом. Возможно, вы разработчик, технический директор, глава проектной группы или тестировщик. На данный момент это не имеет особого значения. Важно то, что вы хотите больше узнать о менеджменте — о так называемых Agile-методологиях менеджмента и разработки. Обещаю, что вы действительно узнаете много нового. Задача этой книги — научить вас хорошо разбираться в *гибких* методологиях и помочь в создании *гибкой* организации. Мы скоро перейдем к конкретному обсуждению гибких методологий, но сначала необходимо уделить внимание основам, которые заключаются в знании того, как ведут себя люди и системы. Вы спросите: «Зачем это нужно?» По той же причине, по которой врачи сначала изучают, как устроен человеческий организм, пилоты постигают функционирование самолета, а программисты знакомятся с устройством компьютера. Ну а менеджеры должны знать, как функционируют социальные системы.

На своем горьком опыте я убедился, что, как бы детально мы ни планировали тот или иной проект, в реальности события почти

наверняка будут развиваться по-другому. Системе, в которой вы функционируете, безразлично, какие у вас планы. Возможно, вы полагаете, что из точки А можно попасть в точку В, и при этом не исключено, что вы правы — но только в теории. Но теории редко срабатывают на практике, а у *предсказуемости* есть коварная сестра, которую зовут *сложность*.

Но я забегаю вперед. Как будет показано далее, люди предпочитают *воспринимать* происходящее линейно, а следовательно, скорее всего, я поступаю правильно, линейно выстраивая изложение в книге. Эта история берет свое начало в причинно-следственных связях. В данной главе мы исследуем эти связи и нелинейность, а ближе к концу главы познакомимся с моделью Менеджмента 3.0.

Причинно-следственные связи

Представлением о том, что обычно все происходит в соответствии с нашими планами (как казалось и мне, когда я был миллионером на бумаге), мы обязаны своей врожденной склонности к **детерминизму**. Он утверждает, что «будущие события неизбежно вытекают из предыдущих в соответствии с законами природы». Детерминизм говорит нам, что причиной любого события является другое событие, произошедшее ранее. С логической точки зрения это значит, что если нам известно все о текущем состоянии дел и мы знаем все варианты перехода из одного состояния в другое, то мы должны быть способны предсказывать будущие события, рассчитав их на основе предшествующих событий и законов природы. Если вам кинуть мяч, вы можете поймать его, поскольку в состоянии определить его траекторию. Вы вполне способны оценить, сколько у вас останется денег до конца месяца после того, как вы хорошенько погуляете с друзьями; или как лучше вывести из себя брата либо сестру и при этом не получить по шее.

В мире науки детерминизм оказался чрезвычайно успешным, позволив ученым предсказывать огромное количество разнообразных событий и явлений. Например, используя механику Ньютона, ученые уверенно предсказывают возвращение кометы Галлея в Солнечную систему в 2061 году. Научный метод предсказания будущих событий на основе событий, им предшество-

вавших, а также законов природы оказался настолько успешным, что философ Иммануил Кант провозгласил всеобщий детерминизм в качестве необходимого условия любого научного знания [Prigogine, Stengers 1997: 4].

Детерминизм позволяет разработчикам программного обеспечения проектировать, планировать и предсказывать поведение своего программного продукта в реальных условиях использования. Они создают или вносят изменения в программный код, чтобы задать или изменить поведение программного продукта после компиляции и развертывания у пользователя. Если на мгновение отвлечься от ошибок программирования, сбоев операционных систем, аварийных отключений электричества, неквалифицированных пользователей и других рисков, то можно сказать, что предсказания разработчиков очень часто оказываются весьма точными. Тот же детерминизм позволил мне в свое время сделать вполне верный прогноз, что мой проект обанкротится, если не удастся найти больше клиентов.

Но, как это ни было бы странно, одного детерминизма недостаточно. Хотя мы умеем предвидеть очередное появление кометы Галлея и можем еще на стадии разработки предсказать, как будет функционировать программное обеспечение, мы не в состоянии определить погоду на месяц вперед. Мы также не в состоянии точно предвидеть результат сложной комбинации желаемых параметров программного обеспечения, время, имеющееся на разработку, требуемые для проекта ресурсы или (что, к сожалению, случилось с моим проектом) наступление момента, когда появятся новые клиенты.

Так в чем же проблема?

Сложность

Если вежливый и послушный сын ваших соседей — олицетворение предсказуемости, то его своенравная и взбалмошная младшая сестра может служить символом **сложности**. Предсказуемость позволяет вам ходить на работу, назначать встречи, заниматься спортом и смотреть телевизор. В то же самое время сложность зачастую превращает взаимодействия между вами и внешним миром в непредсказуемый хаос, полный неожиданных проблем и сюрпризов.

Многие иногда путают создаваемые сложностью проблемы с проблемой больших чисел (когда одновременно происходит огромное количество событий), но сложные явления не всегда предполагают наличие большого количества элементов. Возьмем, например, молекулу воды (фигурально выражаясь, естественно, на практике это сделать очень непросто). Эта молекула состоит всего из двух атомов водорода и одного атома кислорода. Ничего сложного, не так ли? И тем не менее даже такая простая структура из трех атомов проявляется неожиданным образом в сложных явлениях текучести, эффектах, связанных с плотностью воды, и других физических и химических явлениях [Solé 2000: 13], которые не поддаются легкому объяснению с точки зрения поведения отдельных атомов. Таким образом, сложность необязательно будет проявлением больших чисел. Достаточно трех молекул воды, чтобы состоящая из них система характеризовалась сложным поведением — примером будет знаменитая **задача трех тел**.

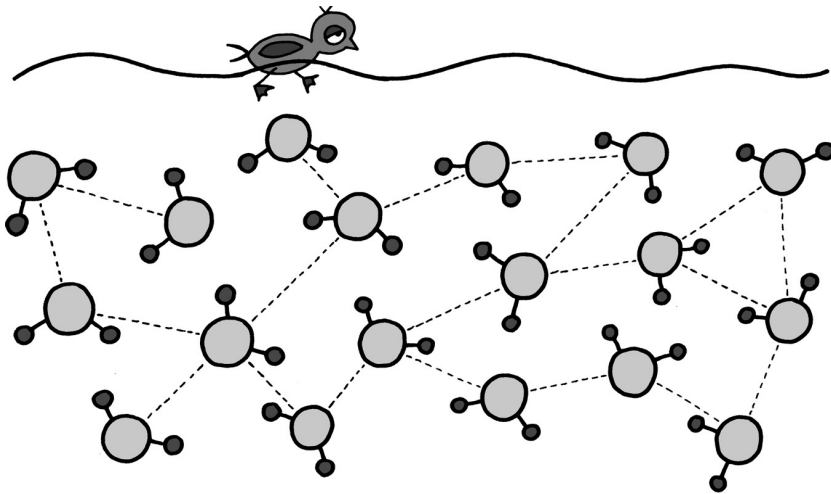


Рис. 1.1. Что же на самом деле происходит в воде?

К счастью, с того момента, когда Кант с энтузиазмом объявил причинность основой научного знания, наука не стояла на месте. Теория динамических систем, теория хаоса, теория сетей, теория игр и ряд других научных дисциплин добились значительного

прогресса, объяснив, почему некоторые явления невозможно предсказывать и почему некоторые события невозможно планировать или рассчитать заранее — их можно только испытывать или наблюдать. Часто весь комплекс исследований в области сложных систем собирательно именуют **теорией сложности** (см. главу 3 «Теория сложности»).

Если развитие науки начиная с XVII века проходило под знаком детерминизма, то сложность как предмет исследования возникла в XX веке; соответствующие исследования значительно ускорились с того момента, когда в конце XX века **теория сложности** выделилась в отдельную научную дисциплину. Физик-теоретик Стивен Хокинг утверждал, что XXI век будет веком сложности [Chui 2000].

Развитие теории сложности — хорошая новость для руководителей, лидеров команд и менеджеров проектов (а также всех прочих «лидеров» и «менеджеров»), работающих в компаниях, создающих ПО. Это означает, что возник *научный* подход к исследованию сложных систем, включая проблемы разработки программного обеспечения и управления организациями в целом. И хотя для меня момент истины опоздал ровно на 10 миллионов евро, я согласен со Стивеном Хокингом, что представление о сложности — ключевая парадигма XXI века.

Наше линейное мышление

К сожалению, применяя теорию сложности к решению конкретных проблем, мы постоянно сталкиваемся с определенным неудобством: наш мозг предпочитает видеть простые причинно-следственные связи и игнорировать сложность. В своей статье «Рожденные верить: Как наш мозг создает богов» [Brooks 2009] автор показывает, что человеческий мозг чрезмерно ориентирован на установление причинно-следственных связей, что заставляет нас видеть их даже там, где их нет. Как отмечается в статье, дети думают, что острые скалы созданы для того, чтобы о них могли чесаться животные, а реки — чтобы по ним можно было плавать на лодках. По всей видимости, устройство человеческого мозга заставляет нас повсюду усматривать целеполагание и причинно-следственные связи, даже если для этого нет никаких оснований.

«Вы слышите шорох в кустах и делаете вывод, что там кто-то притаился». <...> Вероятно, чрезмерная склонность повсюду усматривать причины и следствия дает преимущество с точки зрения выживания. Если вокруг полно хищников, недостаточно обнаруживать их в девяти случаях из десяти. Лучше перестраховаться и убежать, даже если в некоторых случаях опасность окажется иллюзорной. В конце концов, цена такой перестраховки невелика¹.

Наш мозг жестко запрограммирован отдавать предпочтение «линейному мышлению» (представлению о предсказуемости следствий, если известны причины) перед «нелинейным» (гипотезой, что в реальности все обстоит гораздо сложнее). Мы привыкли считать, что события от начала и до конца разворачиваются линейно. В школе нас учат решать линейные уравнения, а более часто встречающиеся на практике нелинейные игнорируются просто потому, что справиться с ними гораздо труднее. Нам легче принять утверждение «это сделал он», чем утверждение «некоторые вещи просто случаются». Если в наличии проблема В, то мы предполагаем, что ее причиной стало событие А. Причиной финансового кризиса стали банкиры; в сокращении числа рабочих мест виноваты иммигранты; в плохой атмосфере в компании виноват менеджер; таяние полярных льдов вызвано выбросами CO₂; проектной группе не удалось уложиться в отведенные сроки из-за того, что кто-то плохо работал. Линейное мышление воспринимает мир как пространство, наполненное легкообъяснимыми событиями, вызванными простыми причинами и имеющими простые следствия. Джеральд Вайнберг называет это **ошибкой причинности** [Weinberg 1992: 90].

Наша мыслительная зависимость от детерминизма заставляет людей искать способы *контроля*, которые позволили бы обеспечить наступление *желательных* событий и ненаступление *нежелательных*. В конце концов, если известно, что ситуация А имеет своим результатом событие В, а ситуация А' — событие С, при этом С лучше, чем В, то всего-навсего надо заставить А пре-

¹ Michael Brooks, “Born believers: How your brain creates God,” New Scientist, February 4, 2009 [Brooks 2009: 32].

вернуться в A' , и все будет хорошо. Так по крайней мере часто кажется.

Инженеры и другие люди с техническим складом ума особенно восприимчивы к идеям, базирующимся на идее управляемости. Именно инженеры создали **научный менеджмент**, основанный на отдаче распоряжений и контроле их исполнения, который всецело господствовал с начала XX века. И именно они придумали системы контроля, которые до сих пор существуют во многих организациях [Stacey 2000a: 7]. Сейчас уже всем известно, что системы контроля эффективны, только если речь идет о повторяющихся операциях, не требующих особых размышлений. Но они *не работают* в ситуациях, когда необходим творческий подход при разработке новых продуктов! Поэтому было бы только справедливо, если бы инженеры и вытащили нас из того управленческого болота, в которое они нас в свое время затянули.

Детерминизм в управлении побуждает менеджеров искать причины, которые привели бы к получению точно такого результата, который им необходим, путем предварительного проектирования этого результата и детального планирования сверху вниз. Чем крупнее организация, тем больше усилий затрачивается на анализ поведения ее составных частей с целью заставить их взаимодействовать так, чтобы в результате поставленные цели были достигнуты.

Прежде и я охотно создавал себе мир иллюзий, основанный на предварительном проектировании и планировании сверху вниз. Мой бизнес-план (который, кстати, был даже отмечен профессиональными премиями) состоял из 30 страниц тщательно выверенной чепухи. В нем в деталях было расписано, как мы разбогатеем. В тот момент мы верили в этот бизнес-план. В конце концов, поскольку его разработал я, как он мог быть неправильным?

Редукционизм

Подход, в рамках которого систему разбирают на части, а затем изучают взаимодействие этих частей, чтобы понять, как работает целое, называется **редукционизмом**. Суть подхода в том, что «явления могут быть исчерпывающе объяснены в терминах других, более фундаментальных явлений». Мы можем разобрать самолет

на детали и увидеть, как он функционирует, изучив каждый винтик; мы можем понять, как работает компьютерная программа, проанализировав ее код; в настоящее время ученые пытаются изучать болезни и врожденные дефекты, анализируя геном человека в надежде идентифицировать отдельные гены, ответственные за те или иные проблемы.

Редукционистский подход хорош только до определенного предела (рис. 1.2). После многолетних исследований ученые все еще не понимают, как работает сознание. Несмотря на созданные за последние сто с лишним лет многочисленные теории, экономисты так и не смогли предложить модель, которая позволяла бы достоверно предсказывать экономические кризисы. Многочисленные теории изменения климата дают совершенно разные прогнозы последствий глобального потепления. И хотя нет недостатка в методиках, моделирующих процесс разработки программного обеспечения, тем не менее во всем мире множество проектов все еще наталкиваются на непредвиденные проблемы. Живые организмы, сознание человека, экономика, изменение климата и проекты по разработке ПО — все эти системы устроены таким образом, что их поведение невозможно предсказать путем деконструкции и изучения компонентов по отдельности.

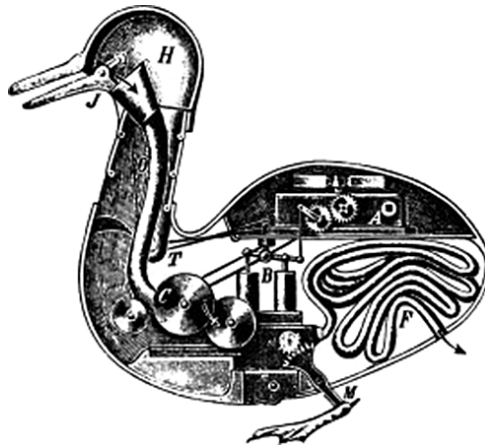


Рис. 1.2. Пример редукционизма, зашедшего слишком далеко

Примечание: Рисунок из «Википедии», находится в свободном доступе.

Способность людей интерпретировать окружающую действительность весьма ограничена

В процессе подготовки этой книги некоторые рецензенты указали мне, что отличительная черта людей — тенденция неточно интерпретировать информацию, поступающую из окружающей действительности. Мы склонны игнорировать любые факты, в которые мы не верим и которые противоречат уже сложившимся в наших головах моделям. Такая избирательность действительно мешает нам более или менее точно предвидеть будущие события.

Идея целостности

Холизм как теория предполагает, что поведение системы несводимо к сумме поведений ее отдельных частей, а напротив, решающим образом определяется ее свойствами как единого целого. Этот подход часто воспринимают как противоположность редукционизму, хотя ученые, исследующие такие системы, полагают, что сложность будет связующим звеном между обоими подходами и каждый из них необходим, но недостаточен [Corning 2002: 69].

Даже некоторые из наиболее рьяных редукционистов отказались от представления, что все явления могут быть сведены к поведению их составных частей. Философ Дэниел Деннет предложил термин «**Жадный редукционизм**» [Dennet 1995] для обозначения форм редукционизма, которые стремятся *полностью* вывести поведение системы из взаимодействия между ее составными частями. Например, утверждение, что «гиперссылки — это не более чем электроны и на самом деле они не существуют» будет примером такого жадного редукционизма. В качестве контраргумента я сказал бы, что если жадные редукционисты правы, то значит, они тоже не существуют, и это полностью снимает все их смехотворные аргументы. Но я отвлекся.

В качестве компромисса биолог-эволюционист Ричард Докинз предложил понятие **иерархического редукционизма** [Dawkins 1996], смысл которого в том, что сложная система может быть представлена в виде иерархии, в которой события на каждом

уровне могут быть объяснены поведением компонентов, находящихся в данной иерархии одним уровнем ниже, но только одним уровнем. Если следовать этой логике, вы не сможете объяснить провал своего проекта тем, что вам помешали кварки и лептоны.

Многие ошибочно полагают, будто бы из редукционизма следует, что мы в состоянии реконструировать любую систему, если понимаем, как функционируют ее составные части. В этом и состоит заблуждение: даже если мы отлично понимаем, как ведут себя все компоненты системы, это не значит, что система сводится к сумме своих составных частей [Miller, Page 2007: 41]. Знание компонентов, находящихся на нижних уровнях системы, вовсе не означает, что мы сможем воссоздать всю систему как единое целое. Интересно, что, если исходить из редукционистского подхода и отследить изначальную причину проблемы (например, воспользовавшись *методикой анализа основной причины*), мы все равно не сможем создать систему, в которой данная проблема отсутствовала бы. Например, мы можем установить причину конкретного случая сердечной недостаточности (редукционизм), но нам никогда не удастся создать сердце, которое принципиально не будет подвержено сердечной недостаточности (конструкционизм).

Значит ли все это, что методика анализа основной причины бесполезна?

Нет, она *чрезвычайно* полезна. Говоря о ее недостатках, мы лишь имеем в виду, что она обращена исключительно в прошлое. С ее помощью можно решить проблемы, которые уже случились, и исключить их возникновение в будущем. Но эта методика никак не помогает предсказывать, *что именно в будущем может пойти не так*.

Иерархический менеджмент

Взгляд на систему как на единое целое (холизм) и иерархический редукционизм сходятся на том, что не все в поведении сложной системы может быть объяснено событиями, происходящими на ее более низких уровнях. Обе гипотезы допускают, что каждому

уровню присущи свои особенные и не сводимые к более элементарным уровням свойства. Например, как бы тщательно вы ни вглядывались, у вас не получится без определенных затруднений идентифицировать внутри деконструированной утки рычажки, подшипники и шестеренки, предназначенные для ходьбы, плавания и крикания (см. рис. 1.2). И тем не менее, увидев этот объект в парке, вы сразу понимаете, что это утка.

Все вышесказанное имеет далекоидущие последствия для менеджеров сложных систем вроде нас с вами, а также для менеджеров, управляющих разработками, проектными менеджеров и лидеров команд. Это означает, что тот, кто знает все о функционировании определенного уровня в иерархической системе, может оказаться недостаточно квалифицированным, чтобы работать на других уровнях в той же системе, потому что для этого требуются иные знания. Молекулярный биолог может оказаться недостаточно «квалифицированным», чтобы выполнять обязанности садовника, потому что знание того, как функционируют живые организмы на уровне клеток-эукариотов, генов и РНК, не подразумевает умения ухаживать за садом; а хорошему садовнику совсем необязательно знать о хромосомах и геноме. Точно так же генеральный директор должен иметь обширные знания о том, как управлять компанией, но при этом он может быть полным профаном в том, что касается коучинга и других навыков управления людьми (я уверен, что многие читатели лично сталкивались с такими ситуациями).

Управление организацией требует совершенно иных знаний и опыта, чем управление людьми, хотя *некоторое* представление о том, как система функционирует на более низких уровнях, *может* оказаться полезным. Инженер-программист Джозел Сполски предложил **закон дырявых абстракций** [Spolsky 2002] в качестве объяснения, почему в системах компоненты, находящиеся на более высоких уровнях, могут проявлять себя неожиданным образом в результате воздействия на них событий, происходящих на более низких уровнях, хотя более высокие уровни, по идее, должны быть изолированы от такого воздействия. Более высокие программные уровни, которые подвергаются воздействию событий, происходящих на более низких программных уровнях, считаются дырявыми. Типичным свидетельством такого рода дырявых