

## Сложность

Врач, строитель и программистка спорили о том, чья профессия древнее. Врач заметил: "В Библии сказано, что Бог сотворил Еву из ребра Адама. Такая операция может быть проведена только хирургом, поэтому я по праву могу утверждать, что моя профессия самая древняя в мире". Тут вмешался строитель и сказал: "Но еще раньше в Книге Бытия сказано, что Бог сотворил из хаоса небо и землю. Это было первое и, несомненно, наиболее выдающееся строительство. Поэтому, дорогой доктор, вы не правы. Моя профессия самая древняя в мире". Программистка при этих словах откинулась в кресле и с улыбкой произнесла: "А кто же по-вашему сотворил хаос?"

## 1 Сложность, присущая программному обеспечению

### 1.1 Простые и сложные программные системы

Звезда в преддверии коллапса; ребенок, который учится читать; клетки крови, атакующие вирус, - это только некоторые из потрясающе сложных объектов физического мира. Компьютерные программы тоже бывают сложными, однако их сложность совершенно другого рода. Брукс пишет: "Эйнштейн утверждал, что должны существовать простые объяснения природных процессов, так как Бог не действует из каприза или по произволу. У программиста нет такого утешения: сложность, с которой он должен справиться, лежит в самой природе системы" [?].

Мы знаем, что не все программные системы сложны. Существует множество программ, которые задумываются, разрабатываются, сопровождаются и используются одним и тем же человеком. Обычно это начинающий программист или профессионал, работающий изолированно. Мы не хотим сказать, что все такие системы плохо сделаны или, тем более, усомниться в квалификации их создателей. Но такие системы, как правило, имеют очень ограниченную область применения и короткое время жизни. Обычно их лучше заменить новыми, чем пытаться повторно использовать, переделывать или расширять. Разработка подобных программ скорее утомительна, чем сложна, так что изучение этого процесса нас не интересует.

Нас интересует разработка того, что мы будем называть *промышленными программными продуктами*. Они применяются для решения самых разных задач, таких, например, как системы с обратной связью, ко-

которые управляют или сами управляются событиями физического мира и для которых ресурсы времени и памяти ограничены; задачи поддержания целостности информации объемом в сотни тысяч записей при параллельном доступе к ней с обновлениями и запросами; системы управления и контроля за реальными процессами (например, диспетчеризация воздушного или железнодорожного транспорта). Системы подобного типа обычно имеют большое время жизни, и большое количество пользователей оказывается в зависимости от их нормального функционирования. В мире промышленных программ мы также встречаем среды разработки, которые упрощают создание приложений в конкретных областях, и программы, которые имитируют определенные стороны человеческого интеллекта.

Существенная черта промышленной программы - уровень сложности: один разработчик практически не в состоянии охватить все аспекты такой системы. Грубо говоря, сложность промышленных программ превышает возможности человеческого интеллекта. Увы, но сложность, о которой мы говорим, по-видимому, присуща всем большим программным системам. Говоря *"присуща"*, мы имеем в виду, что эта сложность здесь неизбежна: с ней можно справиться, но избавиться от нее нельзя.

Конечно, среди нас всегда есть гении, которые в одиночку могут выполнить работу группы обычных людей-разработчиков и добиться в своей области успеха, сравнимого с достижениями Франка Ллойда Райта или Леонардо да Винчи. Такие люди нам нужны как архитекторы, которые изобретают новые идиомы, механизмы и основные идеи, используемые затем при разработке других систем. Однако, как замечает Петерс: "В мире очень мало гениев, и не надо думать, будто в среде программистов их доля выше средней"[2]. Несмотря на то, что все мы чуточку гениальны, в промышленном программировании нельзя постоянно полагаться на божественное вдохновение, которое обязательно поможет нам. Поэтому мы должны рассмотреть более надежные способы конструирования сложных систем. Для лучшего понимания того, чем мы собираемся управлять, сначала ответим на вопрос: почему сложность присуща всем большим программным системам?