

Лекция 9: Антипаттерны

Юрий Литвинов

yurii.litvinov@gmail.com

1. Введение

В этой лекции будет рассказано про то, как делать не надо. В литературе есть термин “Антипаттерны”, имеющий тот же смысл, что и паттерны, но наоборот: если паттерны — это часто встречающиеся решения, приводящие к известным преимуществам, то антипаттерны — это часто встречающиеся решения, приводящие к известным проблемам. Оказывается, что знание антипаттернов для архитектора даже полезнее, чем знание паттернов, поскольку позволяет избежать очень частых и очень дорогостоящих ошибок.

Точно так же, как и паттерны, антипаттерны документируют, чтобы иметь общий словарь (например, не пускаться в пространные рассказы про то, что есть принцип единственности ответственности, что класс, который делает вообще всё — это плохо, и почему, а просто сказать, что это антипаттерн “God Object”). Кроме того, описание антипаттерна, помимо описания проблемы и пояснения, почему это плохо, должно по-хорошему содержать и рекомендации по тому, как это исправить.

Хочется заметить, что антипаттерны часто описывают решения, которые сами по себе неплохи и вполне применяются на практике без всяких проблем, типичный пример — это “Busy waiting”, успешно применяющийся в микроконтроллерах. Источником проблем часто является не само решение, а контекст его применения — например, “Busy waiting” может быть очень плохой идеей, если минутами грузит процессор на ноутбуке.

Антипаттерны, так же, как и паттерны, имеют некую классификацию, но не по назначению, а по “сфере применения” — антипаттерны реализации (в том числе, специфичные для конкретного языка или технологии, которые мы в этой лекции затрагивать не будем), архитектурные антипаттерны и организационные антипаттерны (которые мы тоже затрагивать практически не будем, потому что это больше про управление проектами, чем про архитектуру). Мы начнём с антипаттернов реализации и продолжим архитектурными антипаттернами.

Дальнейшее изложение будет вестись по книжке AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis by William J. Brown, Raphael C. Malveau, Skip McCormick, Thomas J. Mowbray, Wiley, 1998, 336pp. Книга столь же старая, что и книга про паттерны, и с момента её издания появилась куча специфичных для языков и технологий антипаттернов, но они нам как раз и не очень интересны. Антипаттерны, касающиеся разработки в целом, за это время несколько не поменялись. Книга не входит в рекомендованную по этому курсу литературу, потому что там много воды и наиболее содержательные части всё равно будут в этой лекции рассказаны, так что читайте только если особо интересно.

